HIGH SCHOOL EDITION
Journal of Student Research

Volume 11 Issue 3 (2022)

Python-Based Prediction of Rapid Intensification from MIMIC-TC Ensemble (PRIME)

Lorenzo Pulmano[1], Leya Joykutty[#] and Juliana Carvalho De Arruda Caulkins[#]

[1]American Heritage School, Rosenberg, TX, USA
[#]Advisor

ABSTRACT

Rapid intensification (RI), as defined by the National Hurricane Center (NHC), is an increase in the maximum sustained winds of a tropical cyclone (TC) of at least 30 knots (~34-35 mph) within a 24-hour period. Intensity forecasting is one of the most difficult aspects of TC analysis forecasting, with RI prediction being one of the most challenging issues. Predicting intensity and RI is critical for emergency responses, including evacuation and disaster prevention. Deep learning (DL) and its application in TC analysis holds much potential. Morphed Integrated Microwave Imagery at the Cooperative Institute for Meteorological Satellite Studies (MIMIC) is a product that synthesizes "morphed" images of TCs. MIMIC-TC is a product that uses 85-92 GHz microwave imagery to create the images. Using the Python programming language, a DL convolutional neural network (CNN) ensemble was developed as a proof-of-concept for prediction of RI, known as the Prediction of Rapid Intensification from MIMIC-TC Ensemble (PRIME). Six members comprise PRIME, split into three 10 and 20 epoch models. Each model has either 2, 3, or 4 convolutional layers. A MIMIC-TC dataset was created using available North Atlantic Basin (NATL) storms from 2019 and 2020, and a total of 1508 images were used for training the models. After running the Ensemble on all available storms from 2019 and 2020, it appeared all models were overfit, and subsequently gave inaccurate classifications. The average percentage of correct classifications of "No RI" (nRI) was 30%, and the average percentage of correct classifications of "Possible RI" (pRI) was 27%.

Introduction

The purpose of this study was to create a proof-of-concept deep learning (DL) convolutional neural network (CNN) that could classify, and hence predict, images of tropical cyclones (TCs) as undergoing rapid intensification (RI) or not.

Background

As defined by the National Hurricane Center (NHC), RI is an "increase in the maximum sustained winds of a tropical cyclone of at least 30 kt [or approximately 34-35 mph] in a 24-h period" ("Glossary of NHC terms," n.d.). It is dependent on a multitude of internal TC factors, such as internal convective processes, and environmental factors, such as vertical wind shear and sea surface temperatures (SSTs) (Judt & Chen, 2016; Fudeyasu et al., 2018). Because of this, RI presents a difficult challenge for forecasters and models. It is generally agreed that RI has low predictability, and high degrees of uncertainty exist in models (Judt & Chen, 2016).

Morphed Integrated Microwave Imagery at the Cooperative Institute of Meteorological Satellite Studies (MIMIC) is a "family of algorithms" that use satellite imagery to "create synthetic 'morphed' images" (Wimmers & Velden, 2007). MIMIC-TC is a product that uses 85-92 GHz microwave imagery to create the images. This band of microwave imagery provides a "superior representation of TC structures" and depicts "eyewall/banding structure[s]" (Wimmers et al., 2019).

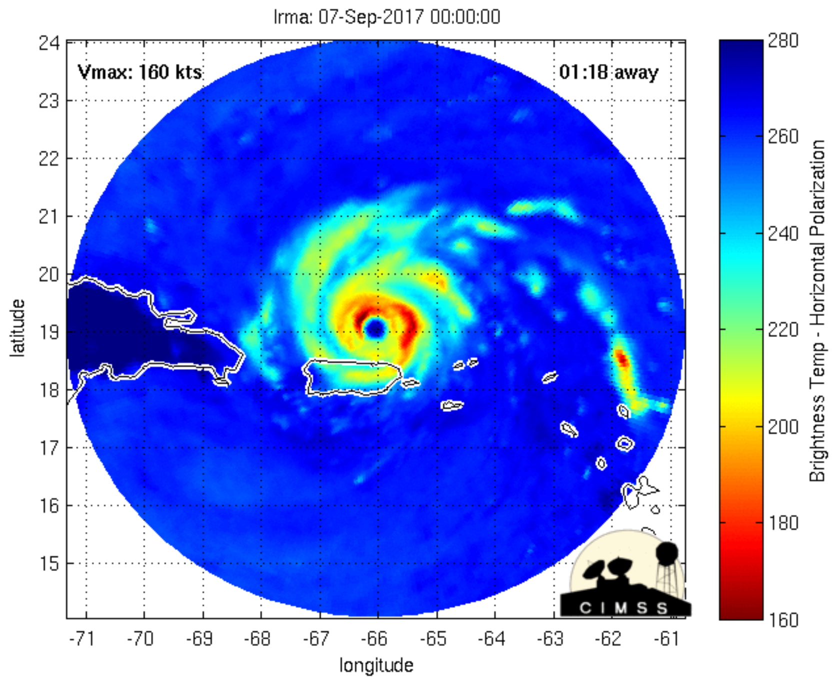


Figure 1. MIMIC-TC image from GIF of Hurricane Irma ("Irma," n.d.)

Deep learning (DL) is a new machine learning and artificial intelligence (AI) method that is growing in popularity. It is a "powerful… tool for developing predictive models in the sciences" and its use in "TC intensity analysis holds tremendous promise" (Wimmers et al., 2019). DL in meteorology is rapidly increasing, with applications in forecasting for renewable energy and improving outputs of precipitation models (Wimmers et al., 2019). The method is appearing to be incredibly valuable in nowcasting, defined by the World Meteorological Organization (WMO) to be "forecasting… over a period from the present to six hours ahead" (World Meteorological Organization, 2020). This TC nowcasting analysis is crucial and is especially important during RI.

A convolutional neural network (CNN) is commonly utilized in image classification tasks. Inspired by biological processes in the brain, it attempts to emulate how humans learn by using layers of "neurons." How CNNs operates and function is beyond the scope of this paper, but a summary will be provided (see Fig. 2 for a visualization). There are mainly three types of layers in a CNN:

1. A convolutional layer takes inputs in the form of 2 or 3 dimensional filters to identify key features within an image. The input then becomes a feature map, an intermediate product. Additional convolutional layers can take these feature maps, and again use filters to create another intermediate feature map (IBM Cloud Education, n.d.).
2. Pooling is used reduce the amount of information. CNNs are computationally heavy even with the use of graphics processing units (GPUs, graphics cards), and pooling can be used to reduce complexity and running time at the cost of losing information. However, losing information can also improve efficiency and limit the risk of overfitting, a situation in which a CNN model performs well on training data, but produces inaccurate results on testing data, or data the CNN has never seen before (IBM Cloud Education, n.d.).
3. Fully-connected layers, or dense layers, perform classification based on the features extracted (IBM Cloud Education, n.d.).

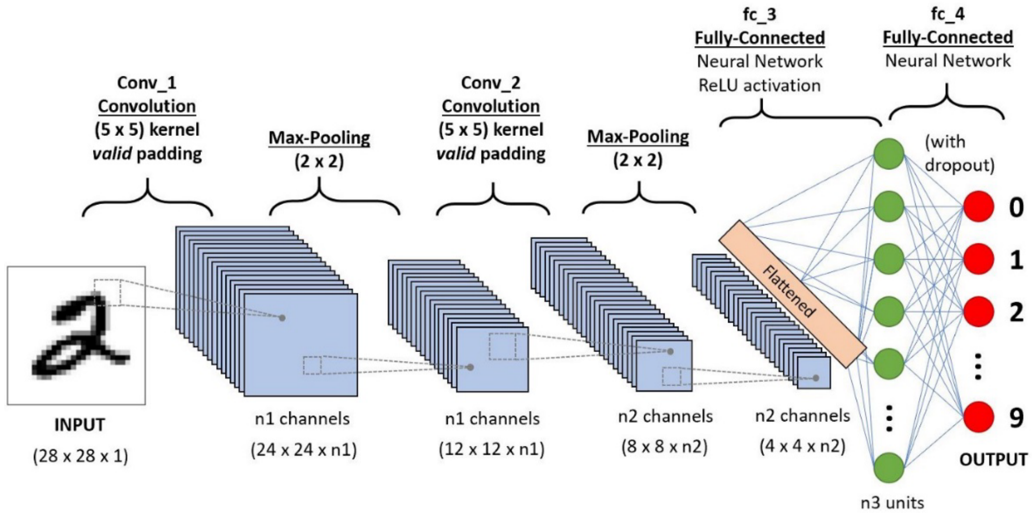


Figure 2. A CNN used to classify handwritten digits (Saha, 2018)

Model

Based on rationale detailed in the previous section, a proof-of-concept DL CNN was created to predict RI. This CNN model will use MIMIC-TC images for training and testing.

The Python programming language was used to create the model because of its wide use in DL applications. Various libraries exist for Python that make it manageable and feasible for models, especially for DL, to be created.

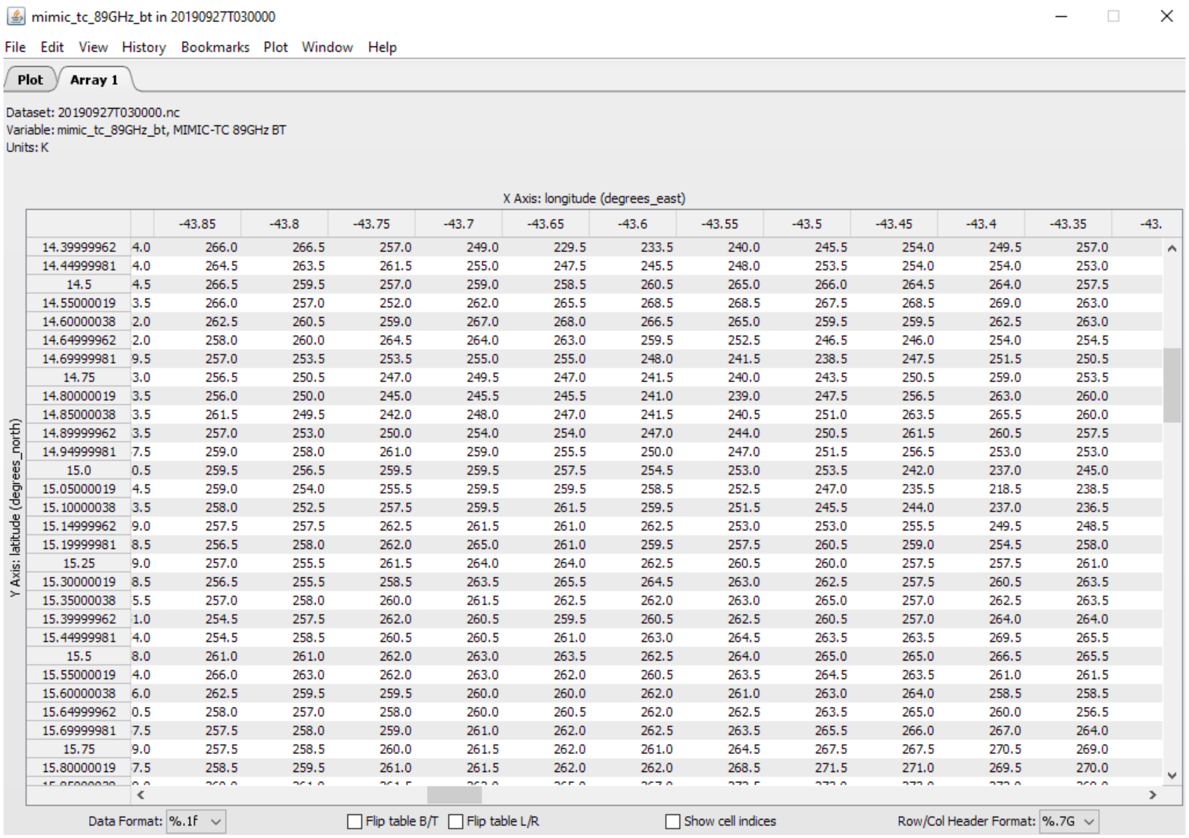
Data Collection and Organization

MIMIC-TC files form available Northern Atlantic Basin (NATL) storms from 2019 and 2020 were used for training and testing sets. Files older than 2019 are unavailable[1]. These files can be found at the Storm Archive at http://tropic.ssec.wisc.edu/real-time/mimic-tc/archive/archive.html. The files are stored as NetCDF4 (.nc) files[2]. These are downloaded from the archive and organized into their respective storm folders. The number of files range from a couple hundred to a couple thousand, as the files were created every 15 minutes during the duration of a storm.

Using National Aeronautics and Space Administration's (NASA's) Panoply Java application, these .nc files can be opened, and a .csv file containing image values can be extracted (NASA, n.d.). For creating the dataset, the following scheme was used:

1. Earliest available .nc file (even if it has a minute attribute)
2. 3-hour interval (00h, 03h, 06h, 09h, 12h, 15h, 18h, or 21h) immediately following the earliest available .nc file
3. Next 3-hour interval (Repeat)
4. Last available .nc file (even if it has a minute attribute)

[1] A. Wimmers (personal communication, October 21, 2021) stated that they "don't save the original data, only the imagery for the MIMIC-TC product."
[2] The files are named by year, month, day, hour UTC, and minute UTC. For example, "20201019T060000.nc" was created in 2020, on October 19 at 0600 UTC.



		-43.85	-43.8	-43.75	-43.7	-43.65	-43.6	-43.55	-43.5	-43.45	-43.4	-43.35	-43.
14.39999962	4.0	266.0	266.5	257.0	249.0	229.5	233.5	240.0	245.5	254.0	249.5	257.0	
14.44999981	4.0	264.5	263.5	261.5	255.0	247.5	245.5	248.0	253.5	254.0	254.0	253.0	
14.5	4.5	266.5	259.5	257.0	259.0	258.5	260.5	265.0	266.0	264.5	264.0	257.5	
14.55000019	3.5	266.0	257.0	252.0	262.0	265.5	268.5	268.5	267.5	268.5	269.0	263.0	
14.60000038	2.0	262.5	260.5	259.0	267.0	268.0	266.5	265.0	259.5	259.5	262.5	263.0	
14.64999962	2.0	258.0	260.0	264.5	264.0	263.0	259.5	252.5	246.5	246.0	254.0	254.5	
14.69999981	9.5	257.0	253.5	253.5	255.0	255.0	248.0	241.5	238.5	247.5	251.5	250.5	
14.75	3.0	256.5	250.5	247.0	249.5	247.0	241.5	240.0	243.5	250.5	259.0	253.5	
14.80000019	3.5	256.0	250.0	245.0	245.5	245.5	241.0	239.0	247.5	256.5	263.0	260.0	
14.85000038	3.5	261.5	249.5	242.0	248.0	247.0	241.5	240.5	251.0	263.5	265.5	260.0	
14.89999962	3.5	257.0	253.0	250.0	254.0	254.0	247.0	244.0	250.5	261.5	260.5	257.5	
14.94999981	7.5	259.0	258.0	261.0	259.0	255.5	250.0	247.0	251.5	256.5	253.0	253.0	
15.0	0.5	259.5	256.5	259.5	259.5	257.5	254.5	253.0	253.5	242.0	237.0	245.0	
15.05000019	4.5	259.0	254.0	255.5	259.5	259.5	258.5	252.5	247.0	235.5	218.5	238.5	
15.10000038	3.5	258.0	252.5	257.5	259.5	261.5	259.5	251.5	245.5	244.0	237.0	236.5	
15.14999962	9.0	257.5	257.5	262.5	261.5	261.0	262.5	253.0	253.0	255.5	249.5	248.5	
15.19999981	8.5	256.5	258.0	262.0	265.0	261.0	259.5	257.5	260.5	259.0	254.5	258.0	
15.25	9.0	257.0	255.5	261.5	264.0	264.0	262.5	260.5	260.0	257.5	257.5	261.0	
15.30000019	8.5	256.5	255.5	258.5	263.5	265.5	264.5	263.0	262.5	257.5	260.5	263.5	
15.35000038	5.5	257.0	258.0	260.0	261.5	262.5	262.0	263.0	265.0	257.0	262.5	263.5	
15.39999962	1.0	254.5	257.5	262.0	260.5	259.5	260.5	262.5	260.5	257.0	264.0	264.0	
15.44999981	4.0	254.5	258.5	260.5	260.5	261.0	263.0	264.5	263.5	263.5	269.5	265.5	
15.5	8.0	261.0	261.0	262.0	263.0	263.5	262.5	264.0	265.0	265.0	266.5	265.5	
15.55000019	4.0	266.0	263.0	262.0	263.0	262.0	260.5	263.5	264.5	263.5	261.0	261.5	
15.60000038	6.0	262.5	259.5	259.5	260.0	260.0	262.0	261.0	263.0	264.0	258.5	258.5	
15.64999962	0.5	258.0	257.0	258.0	260.0	260.5	262.0	262.5	263.5	265.0	260.0	256.5	
15.69999981	7.5	257.5	258.0	259.0	261.0	262.0	262.5	263.5	265.5	266.0	267.0	264.0	
15.75	9.0	257.5	258.5	260.0	261.5	262.0	261.0	264.5	267.5	267.5	270.5	269.0	
15.80000019	7.5	258.5	259.5	261.0	261.5	262.0	262.0	268.5	271.5	271.0	269.5	270.0	

Figure 3. Using Panoply, image values for "Lorenzo - 2019 09 27 03h" can be accessed and saved as .csv

The .csv is saved by "[storm name] – [year] [month] [day] [hour]h.csv"[3]. If there is a minute attribute, it would be saved as "[storm name] – [year] [month] [day] [hour]h [minute]m.csv"[4]. They are saved in the respective storm folder.

Creating the MIMIC-TC Dataset

For the training set, instances of RI were used for the "Possible RI" (hereafter pRI) classification. Using the NHC's Tropical Cyclone Reports (NHC Data Archive, n.d.), instances of RI were recorded for each storm from 2019 to 2020. The corresponding .csv files from that time frame were copied into a separate folder named "Possible RI", which itself is contained within another folder called "Training Set."

For "No RI" (hereafter nRI), storms Arthur, Isaias, and Paulette were chosen. All files from those storms were copied into the nRI folder.

All .csv files were then converted into .png files using a Python script (Fig. 4). They were saved into a similar scheme to that described earlier.

[3] For example, "EPSILON – 2020 10 19 06h.csv"

[4] For example, "EPSILON – 2020 10 26 20h 45m.csv"

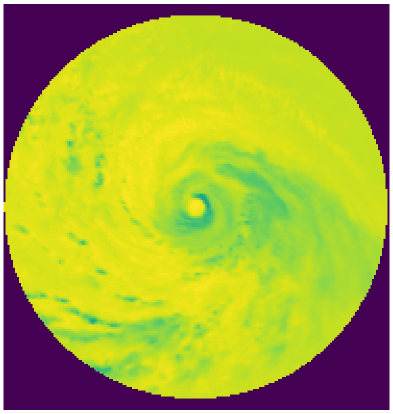


Figure 4. Lorenzo - 2019 09 27 03h converted into .png

Here is the structure and organization of the MIMIC-TC Dataset v6[5]:

- MIMIC-TC Dataset v6
    - Training Set (total of 1508 images)
        - Possible RI
        - No RI
    - Testing Set (total of 44 storms)
        - ALPHA
        - …
        - ZETA

The training images were copied and rotated 90, 180, and 270 degrees to create additional training data (Figs. 5, 6, 7). The number of images in the training set amount to 1508.

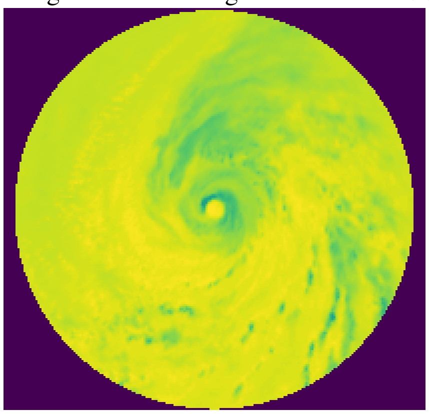


Figure 5. Lorenzo - 2019 09 27 03h ROT90.png[6]

[5] "v6" is an artifact stemming from previous iterations of the organization; adding a version number was easier to differentiate between current and deprecated versions of the dataset during development

[6] Images rotated by 90 degrees are the true orientation of the storm

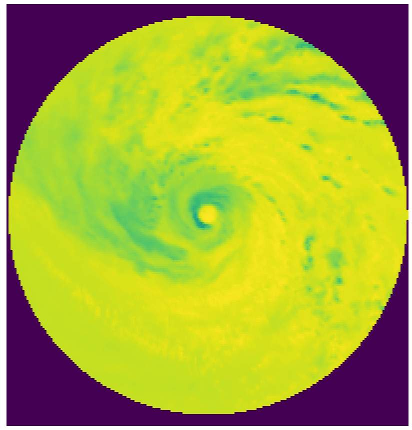


Figure 6. Lorenzo - 2019 09 27 03h ROT180.png

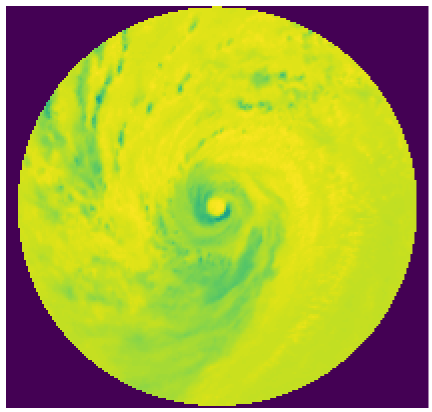


Figure 7. Lorenzo - 2019 09 27 03h ROT270.png

Model Architecture

The architecture of Prediction of Rapid Intensification from MIMIC-TC Ensemble (PRIME) is loosely based off "DeepMicroNet," a DL CNN created by Wimmers et al. (2019) to predict TC intensity. An ensemble of 6 models was created. They differed in the number of epochs, or number of times the model trained, and the number of convolutional layers. Layer attributes were provided for reproducibility of the model.

Rectified linear activation function (ReLU) was used for the activation for each convolutional layer. It is the most used activation function in DL models because it is considered "the best activation function for deep learning" (K, 2020).

Softmax for the dense layer was retained from DeepMicroNet.

Table 1. Model Architecture

Layer	Filters	Kernel Size	Strides	Activation
Conv2d	64	5	1	ReLU
Conv2d	64	3	1	ReLU
#Conv2d	64	3	1	ReLU
#Conv2d	64	3	1	ReLU
Layer	**Pool Size**	**-**	**Strides**	**-**
MaxPooling2D	2	-	2	-
Layer	**Units**	**-**	**-**	**Activation**
Flatten	-	-	-	-

Dense	2	-	-	Softmax

For 10 epochs, 2conv, 3conv, and 4conv models were created. For 20 epochs, 2conv, 3conv, and 4conv models were created. Hence, the ensemble members are:

- 10 epochs
    - 15-10e-2conv[7] (2 convolutions)
    - 15-10e-3conv (3 convolutions)
    - 15-10e-4conv (4 convolutions)
- 20 epochs
    - 15-20e-2conv (2 convolutions)
    - 15-20e-3conv (3 convolutions)
    - 15-20e-4conv (4 convolutions)

Training, Compiling, Fitting, and the Metrics

When initializing the model, it uses a data generator to randomly choose training images from the dataset and stores them as .pickle files.

For compiling, "sparse_categorical_crossentropy" was used for loss, "adam" for optimizer, and "accuracy" for metrics. Sparse categorical crossentropy is used for the mutually exclusive classifications and "saves time in memory as well as computation" ("Sparse_categorical_crossentropy vs categorical_crossentropy," n.d.; "Cross entropy vs. sparse cross entropy," n.d.).

When fitting for both 10 and 20 epoch models, the batch size was 32 and the validation split was 0.1.

Results

Each Ensemble member seems to be overfit. Looking at the model accuracy during training (Fig. 8), suspiciously high accuracy values are present for each model. When used on the testing data, it is clear the low accuracies indicate overfitting (Tables 2 and 3).

There are low percent averages for both correct prediction of nRI and pRI (Tables 2 and 3). The average Ensemble performance for nRI is 30%, and the average Ensemble performance for pRI is 27%.

[7] "15" is an artifact stemming from previous iterations of the model; adding a version number was easier to differentiate between current and deprecated versions of the model during development

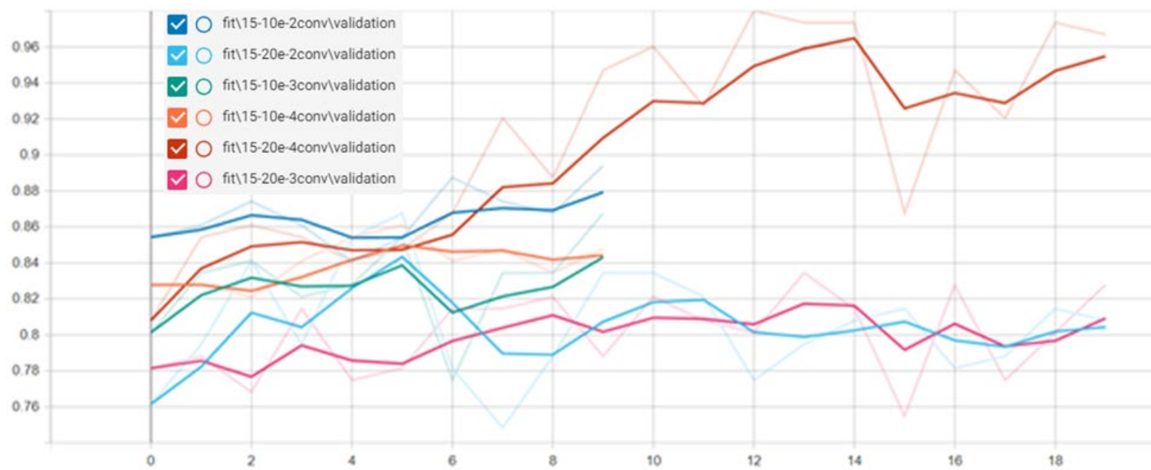


Figure 8. Model Accuracy (epoch_accuracy) for validation during training, 0.6 smoothing

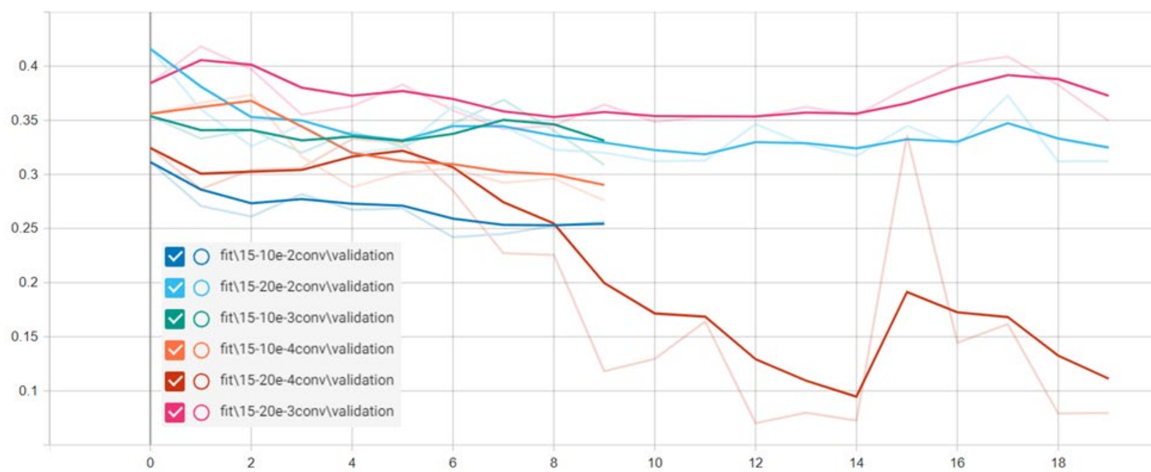


Figure 9. Model Loss (epoch_loss) for validation during training, 0.6 smoothing

Table 2. Percentage of Correct nRI Classifications for Each Ensemble Member

Storm	10e-2conv	10e-3conv	10e-4conv	20e-2conv	20e-3conv	20e-4conv
ALPHA	0%	0%	0%	0%	0%	0%
Arthur	0%	0%	0%	0%	100%	0%
Bertha	0%	0%	0%	0%	0%	0%
BETA	38%	31%	38%	31%	46%	0%
Cristobal	47%	15%	16%	35%	22%	0%
DELTA	48%	30%	33%	28%	35%	2%
Dolly	0%	0%	0%	0%	0%	0%
Dorian	53%	49%	49%	51%	54%	28%
Edouard	0%	0%	0%	0%	0%	0%
EPSILON	15%	15%	15%	15%	15%	0%
ETA	75%	58%	58%	70%	56%	15%
Fay	0%	0%	0%	0%	0%	0%

FIF-TEEN2019	100%	52%	52%	93%	59%	31%
GAMMA	97%	27%	33%	77%	27%	0%
Gonzalo	100%	100%	100%	100%	100%	100%
Hanna	73%	70%	77%	60%	77%	0%
Humberto	12%	12%	12%	16%	25%	0%
Imelda	9%	9%	13%	4%	26%	0%
IOTA	100%	100%	100%	100%	100%	100%
Isaias	37%	19%	31%	18%	21%	0%
Jerry	48%	32%	33%	41%	7%	4%
Josephine	98%	65%	68%	82%	68%	23%
Karen	71%	58%	58%	73%	76%	32%
Kyle	0%	0%	0%	0%	0%	0%
Laura	48%	18%	15%	35%	18%	15%
Lorenzo	49%	41%	41%	47%	44%	27%
Marco	69%	43%	43%	60%	74%	10%
Melissa	0%	0%	0%	0%	0%	0%
Nana	91%	81%	66%	97%	78%	38%
Nestor	3%	3%	3%	3%	3%	0%
Olga	16%	16%	16%	21%	21%	0%
Omar	0%	0%	0%	0%	0%	0%
Pablo	0%	0%	0%	0%	0%	0%
Paulette	31%	11%	11%	19%	13%	0%
Rebekah	0%	0%	0%	0%	0%	0%
Rene	99%	61%	66%	76%	64%	1%
Sally	32%	32%	31%	37%	51%	0%
Sebastien	53%	29%	31%	39%	31%	0%
Teddy	53%	51%	51%	49%	53%	29%
TEN2020	67%	8%	8%	75%	25%	0%
THETA	0%	0%	0%	0%	4%	0%
Vicky	89%	4%	7%	41%	11%	0%
Wilfred	100%	90%	88%	95%	90%	20%
ZETA	63%	3%	3%	55%	16%	3%
Averages	**43%**	**28%**	**29%**	**37%**	**34%**	**11%**

Table 3. Percentage of Correct pRI Classifications for Each Ensemble Member

Storm	10e-2conv	10e-3conv	10e-4conv	20e-2conv	20e-3conv	20e-4conv
DELTA	0%	29%	29%	7%	21%	14%
Dorian	0%	32%	32%	20%	36%	100%
EPSILON	56%	56%	50%	44%	19%	100%
ETA	0%	0%	0%	0%	0%	0%
GAMMA	0%	100%	100%	0%	43%	100%

Hanna	0%	0%	0%	0%	0%	100%
IOTA	0%	0%	0%	0%	0%	0%
Jerry	0%	31%	23%	8%	31%	100%
Laura	20%	20%	20%	0%	0%	87%
Lorenzo	0%	26%	26%	19%	33%	44%
Teddy	0%	43%	38%	0%	29%	14%
ZETA	30%	30%	30%	30%	10%	100%
Averages	**9%**	**31%**	**29%**	**11%**	**19%**	**63%**

Analysis of PRIME

The absence of more "symmetrical" characteristics in an image reduces accuracy. According to Fudeyasu et al. (2018), a "symmetric cloud structure is dominant… during RI." PRIME's accuracy on tropical depression (TD) and tropical storm (TS) imagery are low compared stronger storms as a result since higher intensity TCs exhibit more discernible features such as a symmetric eye and convection. A lack of any features in the image can also prompt a false positive, such as in TS Bertha's case (Fig. 10).

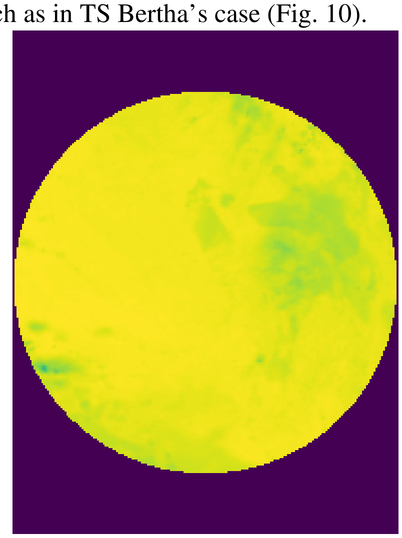


Figure 10. Bertha - 2020 05 28 15h.png, there is a lack of features

PRIME also tends to classify large timeframes of images into pRI, sometimes classifying the entirety of a storm's images as pRI. The Ensemble further tends to classify images during the later part a storm as pRI (Fig. 11).

10 epochs				20 epochs		
2conv				2conv		
Timestamp	nRI	pRI		Timestamp	nRI	pRI
2020 08 14 18h		1		2020 08 14 18h		1
2020 08 14 21h		1		2020 08 14 21h		1
2020 08 15 00h		1		2020 08 15 00h		1
2020 08 15 03h		1		2020 08 15 03h		1
2020 08 15 06h		1		2020 08 15 06h		1
2020 08 15 09h		1		2020 08 15 09h		1
2020 08 15 12h		1		2020 08 15 12h		1
2020 08 15 15h		1		2020 08 15 15h		1
2020 08 15 18h		1		2020 08 15 18h		1
2020 08 15 21h		1		2020 08 15 21h		1
2020 08 16 00h		1		2020 08 16 00h		1
2020 08 16 03h		1		2020 08 16 03h		1
2020 08 16 06h		1		2020 08 16 06h		1
2020 08 16 09h		1		2020 08 16 09h		1
2020 08 16 12h		1		2020 08 16 12h		1
2020 08 16 15h		1		2020 08 16 15h		1
2020 08 16 18h		1		2020 08 16 18h		1
2020 08 16 21h		1		2020 08 16 21h		1
2020 08 17 00h		1		2020 08 17 00h		1
2020 08 17 03h		1		2020 08 17 03h		1
2020 08 17 06h		1		2020 08 17 06h		1
2020 08 17 09h		1		2020 08 17 09h		1
2020 08 17 12h		1		2020 08 17 12h		1
2020 08 17 15h		1		2020 08 17 15h		1
2020 08 17 16h 45m		1		2020 08 17 16h 45m		1

Figure 11. Kyle PRIME results for 15-10e-2conv and 15-20e-2conv members, entirety of storm's lifetime is incorrectly classified as pRI

15-10e-3conv and 15-10e-4conv members of PRIME may be considered more "generalized" models, that is, they do not tend to predict a certain category. The former has an average of 28% of images correctly classified as nRI and 31% pRI, and the latter has 29% nRI and 29% pRI. However, it is clear from the low percentage of correct classifications for both nRI and pRI that PRIME is not accurate and cannot predict RI.

To increase accuracy, a different dense layer activation function could be used. The use of a larger dataset would allow for the model to train and validate on more images, increasing accuracy. The "use of dropout… for reducing overfitting" could also improve PRIME (Brownlee, 2019). Changing the learning rate and using leaky ReLU for convolutional layer activation may improve the model.

Conclusion

It is clear PRIME does not predict RI well.

Future Research and Other Applications

The model or a similar model may instead be used instead to predict eyewall replacement cycles (ERCs). ERCs are events that occur in mostly intense TCs, in which a secondary outer eyewall forms around the inner eyewall. TC intensity weakens during an event because the secondary eyewall saps strength from the primary eyewall.

Eventually, the outer eyewall replaces the inner eyewall, allowing the storm to re-intensify. The surface wind field often expands. The re-strengthening phase of an ERC may erupt as RI, greatly increasing the maximum wind speed and the threat to life and property (Sitkowski et al., 2011). Microwave imagery is already utilized to predict ERCs in the algorithm known as the Automated Rotational Center for Hurricane Eye Retrieval (ARCHER) Microwave-based Probability of Eyewall Replacement Cycle (MPERC), but because little is known about the genesis and process, ERCs are difficult to predict (Zhou & Wang, 2011). Additional diagnostic tools for intensity prediction would be valuable.

MIMIC-TC images may again be used for an ERC-predicting DL CNN. The images can effectively show the emergence of an outer eyewall, the strengthening of the outer eyewall, the weakening of the inner eyewall, and the replacement.

Acknowledgments

I would like to thank my research mentors, Leya M. Joykutty and Dr. Juliana Carvalho De Arruda Caulkins, and my advisor, Stefan Abi-Karam. I would like to thank my father for getting me a NVIDIA GeForce GTX 1080, a graphics card which had enough memory to handle the model.

References and Sources

API documentation : Tensorflow core v2.7.0. TensorFlow. (n.d.). Retrieved January 6, 2022, from https://www.tensorflow.org/api_docs

Brownlee, J. (2019, August 6). *A gentle introduction to dropout for Regularizing Deep Neural Networks*. Machine Learning Mastery. Retrieved January 6, 2022, from https://machinelearningmastery.com/dropout-for-regularizing-deep-neural-networks/

By: IBM Cloud Education. (n.d.). *What are convolutional neural networks?* IBM. Retrieved January 6, 2022, from https://www.ibm.com/cloud/learn/convolutional-neural-networks

Cross entropy vs. sparse cross entropy: When to use one over the other. Stackexchange. (n.d.). Retrieved January 6, 2022, from https://stats.stackexchange.com/questions/326065/cross-entropy-vs-sparse-cross-entropy-when-to-use-one-over-the-other

Documentation of scikit-learn 0.21.3¶. learn. (n.d.). Retrieved January 6, 2022, from https://scikit-learn.org/0.21/documentation.html

Fudeyasu, H., Ito, K., & Miyamoto, Y. (2018, November 1). *Characteristics of tropical cyclone rapid intensification over the western North Pacific*. AMETSOC. Retrieved January 6, 2022, from https://journals.ametsoc.org/view/journals/clim/31/21/jcli-d-17-0653.1.xml

Glossary of NHC terms. (n.d.). Retrieved January 6, 2022, from https://www.nhc.noaa.gov/aboutgloss.shtml

Image 0.19.0 docs¶. scikit. (n.d.). Retrieved January 6, 2022, from https://scikit-image.org/docs/stable/

IPython documentation¶. IPython Documentation - IPython 7.31.0 documentation. (n.d.). Retrieved January 6, 2022, from https://ipython.readthedocs.io/en/stable/

Irma. MIMIC. (n.d.). Retrieved January 6, 2022, from http://tropic.ssec.wisc.edu/archive/mimtc/2017_11L/web/displayGifsBy12hr_09.html

Judt, F., & Chen, S. S. (2016, November 1). *Predictability and dynamics of Tropical Cyclone Rapid intensification deduced from high-resolution stochastic ensembles*. AMETSOC. Retrieved January 6, 2022, from https://journals.ametsoc.org/view/journals/mwre/144/11/mwr-d-15-0413.1.xml

K, B. (2020, November 7). *Understanding relu: The most popular activation function in 5 minutes!* Medium. Retrieved January 6, 2022, from https://towardsdatascience.com/understanding-relu-the-most-popular-activation-function-in-5-minutes-459e3a2124f

Mimic. CIMSS Tropical Cyclones Page. (n.d.). Retrieved January 6, 2022, from http://tropic.ssec.wisc.edu/real-time/mimtc/tc.shtml

NASA. (n.d.). *NASA GISS: Panoply 5 netcdf, HDF and Grib Data Viewer*. NASA. Retrieved January 6, 2022, from https://www.giss.nasa.gov/tools/panoply/

NHC Data Archive. (n.d.). Retrieved January 6, 2022, from https://www.nhc.noaa.gov/data/

Nowcasting guidelines – a summary. World Meteorological Organization. (2020, June 10). Retrieved January 6, 2022, from https://public.wmo.int/en/resources/bulletin/nowcasting-guidelines-%E2%80%93-summary

NumPy documentation. (n.d.). Retrieved January 6, 2022, from https://numpy.org/doc/

Nvidia Cudnn. NVIDIA Developer. (2021, November 19). Retrieved January 6, 2022, from https://developer.nvidia.com/cudnn

OpenCV-python tutorials. OpenCV. (n.d.). Retrieved January 6, 2022, from https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html

Pandas documentation¶. pandas documentation - pandas 1.3.5 documentation. (n.d.). Retrieved January 6, 2022, from https://pandas.pydata.org/docs/

Pillow. (n.d.). Retrieved January 6, 2022, from https://pillow.readthedocs.io/en/stable/

Project jupyter. Project Jupyter. (n.d.). Retrieved January 6, 2022, from https://jupyter.org/

Saha, S. (2018, December 17). *A comprehensive guide to Convolutional Neural Networks-the eli5 way*. Medium. Retrieved January 6, 2022, from https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

Sitkowski, M., Kossin, J. P., & Rozoff, C. M. (2011, December 1). *Intensity and structure changes during hurricane eyewall replacement cycles*. AMETSOC. Retrieved January 6, 2022, from https://journals.ametsoc.org/view/journals/mwre/139/12/mwr-d-11-00034.1.xml?rskey=NmJ0xP

Sparse_categorical_crossentropy vs categorical_crossentropy (Keras, accuracy). Data Science Stack Exchange. (n.d.). Retrieved January 6, 2022, from https://datascience.stackexchange.com/questions/41921/sparse-categorical-crossentropy-vs-categorical-crossentropy-keras-accuracy

Statistical Data Visualization¶. seaborn. (n.d.). Retrieved January 6, 2022, from https://seaborn.pydata.org/

Visualization with python. Matplotlib. (n.d.). Retrieved January 6, 2022, from https://matplotlib.org/

Wimmers, A. J., & Velden, C. S. (2007, August 1). *Mimic: A new approach to visualizing satellite microwave imagery of tropical cyclones*. AMETSOC. Retrieved January 6, 2022, from https://journals.ametsoc.org/view/journals/bams/88/8/bams-88-8-1187.xml

Wimmers, A., Velden, C., & Cossuth, J. H. (2019, June 1). *Using deep learning to estimate tropical cyclone intensity from satellite passive microwave imagery*. AMETSOC. Retrieved January 6, 2022, from https://journals.ametsoc.org/view/journals/mwre/147/6/mwr-d-18-0391.1.xml?tab_body=fulltext-display

Zhou, X., & Wang, B. (2011, May 1). *Mechanism of concentric eyewall replacement cycles and associated intensity change*. AMETSOC. Retrieved January 6, 2022, from https://journals.ametsoc.org/view/journals/atsc/68/5/2011jas3575.1.xml?rskey=NmJ0xP