Generative Adversarial Fusion for Supercell Thunderstorm Forecasting

Yash Jain

Abstract

Supercell thunderstorms have long evaded forecasters, leading to inadequate warning and preventable deaths. Current prediction methods use direct weather observation and algorithmic computer models, while they are not very computationally intensive, they lack in their ability to maintain accurate predictions. Machine learning has been showing some promising results with generative networks proving to show the most promise. This paper proposes a novel machine learning model that attempts to address this problem by taking advantage of two different types of crucial data. My proposed architecture combines three distinct Machine Learning models: a Recurrent Deep Neural Network, a Wasserstein Generative Adversarial Network, and a Transformer Generative Adversarial Network to leverage numerical and visual data. While my research is purely theoretical at this stage, I believe that my proposed machine learning architecture has the potential to pave the way for future studies and practical implementations in the field of supercell thunderstorm prediction, providing significant improvements in forecast accuracy and lead time.

Background

Supercell thunderstorms are a type of rotating thunderstorm that can spawn tornadoes. Though supercell thunderstorms do not always produce tornadoes, straight line winds, torrential rain, and hail leave lasting devastation in the wake of a storm. Storms can happen at any time of the year and range in location, specifically in North America, from the southern tip of Texas (United States) to central Manitoba (Canada), with the frequency and intensity increasing with each successive year[1]. Some of the deadliest events include the 2011 Super Outbreak, Daulatpur-Saturia outbreak of 1989, and Tri-State tornado outbreak of 1925. In each case, a tornado of at least EF/F4 strength touched down and the damage ranged from 2.38 Billion USD to 10.2 Billion USD[234]. The deadliest storm on record, the 1989 Daulatpur-Saturia outbreak killed almost 1300 people. Despite major technological breakthroughs, predicting supercell storms is incredibly difficult. Weather predictions have improved over the last 75 years; the first unofficial tornado warning in 1948 gave people ten minutes to prepare for the pending storm. However, in 2023, meteorologists are still only able to issue a warning a mere hour before the storm is due to hit. The rise of Machine Learning is proving to be a more accurate way to predict supercell storms and can give communities greater accuracy to better prepare for potential devastation and save lives. The Shenzhen Key Laboratory of Severe Weather in South China has developed several generative models that outperform recent high performing applications of neural networks as well as current prediction methods, including the leading optical flow algorithm[5]. While many of these models are extremely good at analyzing imagery, they lack the ability to use numerical observation data, which diminishes the long term precision and accuracy of these models. I propose a multi stage Machine learning model for Supercell thunderstorm forecasting that could improve long term forecasting. This model follows a 3 stage process: one model will serve as the initial predictor for the image data and another will serve as the initial predictor for the numerical data, then a final model will serve as the final predictor using both the predicted numerical/image and observed numerical/image data.

Meteorological Overview

Supercell Thunderstorms, powerful storms with deep, persistently rotating updrafts, form in the presence of four main factors: wind shear, instability, lift, and moisture. Wind shear, a change in wind speed or direction with height, instigates the rotating updraft, a strong upward movement of air, which is a defining feature of a supercell thunderstorm. Additionally, the variation in temperature triggers a rotation in the form of a pressure gradient,

---

[1] National Weather Service. (n.d.). What is a Supercell? Retrieved April 27, 2023, from https://www.weather.gov/ama/supercell

[2] Encyclopædia Britannica. (n.d.). Saturia–Manikganj Sadar tornado. Retrieved April 27, 2023, from https://www.britannica.com/event/Saturia-Manikganj-Sadar-tornado

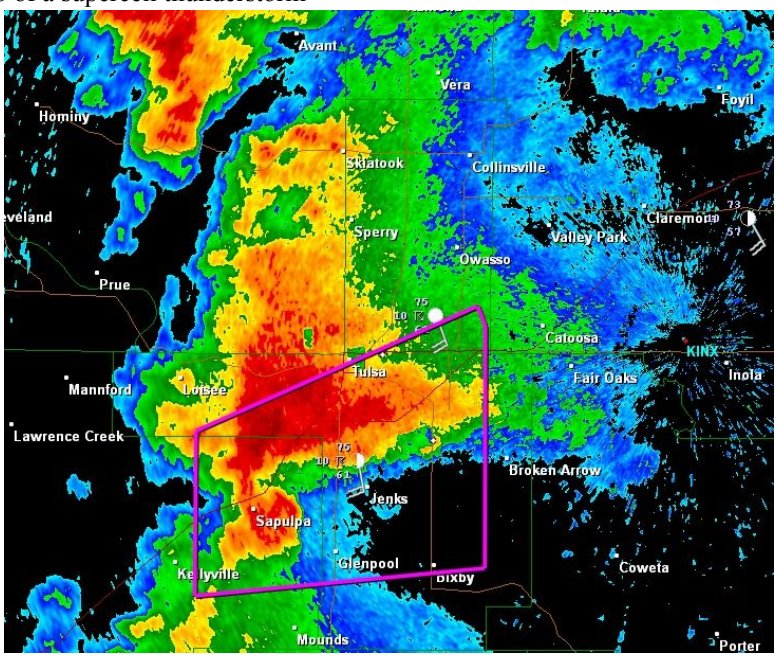
[3] TornadoFacts.net. (n.d.). Tri-State Tornado Facts, History and Information. Retrieved April 27, 2023, from https://www.tornadofacts.net/tri-state-tornado-facts.html

[4] Rafferty, J. P. (n.d.). Super Outbreak of 2011. Encyclopædia Britannica. Retrieved April 27, 2023, from https://www.britannica.com/event/Super-Outbreak-of-2011

[5] Storm Prediction Center. (n.d.). Supercells. Retrieved April 27, 2023, from https://www.spc.noaa.gov/misc/AbtDerechos/supercells.htm

resulting in a Mesocyclone, a vortex of air within a convective storm[6]. A way you can measure this instability is through the Convective Available Potential Energy (CAPE) index[7], which evaluates the buoyancy of an air parcel and determines the amount of energy for convection, which is the process of warm air rising and cool air sinking. An air parcel is an arbitrary blob of air at ambient temperature and zero humidity. During the ascent, the air parcel expands, cools, and rises due to the decrease in atmospheric pressure. If cooled to the dew point, this air parcel will condense due to an overload of moisture and cause cloud formation[8]. When buoyancy values are high, CAPE values are high, which is indicative of a supercell thunderstorm outbreak. CAPE is most commonly calculated using a model sounding—a vertical profile of the atmosphere containing dew point, temperature, wind shear, and much more. Predicting these could prove beneficial to understanding and predicting many more phenomena. Many of these phenomena have aspects relating to supercell thunderstorm formation and can serve to improve general forecasts. Supercells tend to form in bands along a cold front, the leading edge of a cooler mass of air that replaces warmer air; each supercell is no longer 10 to 20 miles wide but can clump together to create a line stretching thousands of miles. Supercells have unique features, including a dry hook as displayed in Figure 1, which indicates that a tornado is present[9]. In addition, they often create an indent in the cloud formation and produce significant hail.

Fig. 1 radar hook echo of a supercell thunderstorm



The first efforts of forecasting started in the 1870s when the US Army Signal Corps developed a centralized weather forecasting program. They sent 2000 spotters to document tornadoes and report the weather conditions during and after the storm. In 1882, US Army Signal Corps Sergeant John P. Finley analyzed several years of tornado outbreaks and established 15 rules that showed the possibility of tornadoes. However, for a period of 50 years, from 1888 to 1938, the word "Tornado" was banned from forecasting on the grounds that it would provoke fear amongst the public. Had this ban been lifted earlier, it may have lessened the death toll of the 1928 Tri-state tornado outbreak. Even after the ban was lifted, tornado research didn't garner much attention until the

[6] Storm Prediction Center. (n.d.). Supercells. Retrieved April 27, 2023, from https://www.spc.noaa.gov/misc/AbtDerechos/supercells.htm

[7] Colorado State University. (n.d.). Convective Available Potential Energy. Retrieved April 27, 2023, from https://rammb.cira.colostate.edu/wmovl/VRL/Tutorials/SatManu-eumetsat/SATMANU/Basic/Convection/Cape.htm

[8] Brune, W. (n.d.). How can supersaturation be achieved? In Fundamentals of Atmospheric Science. LibreTexts. Retrieved April 27, 2023, from https://geo.libretexts.org/Bookshelves/Meteorology_and_Climate_Science/Book%3A_Fundamentals_of_Atmospheric_Science_(Brune)/05%3A_Cloud_Physics/5.04%3A_How_can_supersaturation_be_achieved%3F

[9] US Safe Room. (n.d.). Protect Yourself from Supercell Thunderstorms. Retrieved April 27, 2023, from https://www.ussaferoom.com/protect-yourself-supercell-thunderstorms/

mid-1940s. This is when the Weather Bureau formed experimental tornado warning systems, which were reported through trained spotters. In 1948, a breakthrough occurred, when Maj. Ernest Fawbush and Capt. Robert Miller successfully issued the first tornado warning to Tinker Air Force Base in Oklahoma City using reports from a nearby airport's weather station. This event renewed passion for tornado forecasting[10]. Throughout the 50s and 60s, forecasting improved through the implementation of radar and improved formulas.

In the 70s, model guidance started to become mainstream. These were computer-generated forecasts that quickly began to outperform human forecasted methods[11]. In 1977, computer-generated forecasts had a model skill—the reliability of the model when forecasting storms 3 days in advance—of 0.49, where 0.4 is the human baseline. 10 years later, computer-generated forecasts had a model skill of 0.63 on a scale from 0 to 1. From the 80s to today, model guidance has gotten better through the use of more variables and better computing power[12][13]. There are many model guidances for short-term and long-term prediction. There are several short-term prediction models like HRRR and NAM 5km, while more precise than long-term models, start to lose accuracy even with small time frames. Precision is the resolution of the model while accuracy is how close, in location, time, and intensity, the forecasting is to the real event. Additionally, the HRRR model cannot see through heavy cloud cover, which makes temperature forecasting difficult. The long-term guidances like GFS and ECMWF are less precise for hyperlocal events and more accurate for longer timeframes.

The major flaw with these models is that they tend to struggle with forecasting hyperlocal and rare events for an extended period. In addition, they struggle to forecast embedded storms, which can cause even more damage than the main storm line itself. One modern example that encapsulates this issue is the winter/ice storm of February 2018. While accurately predicted, it was only about 18 hours before the storm when forecasters issued the watches and warnings. Despite that, the storm still overperformed and gave regions as far south as Gainesville Florida 0.25 inches of ice[14].

Another example is the March 14th, 2017 Blizzard. It crippled the Northeast with NYC receiving 7.8 inches of wet snow and the majority of New England receiving between 18 and 24 inches of snow, within the storm were embedded storms that produced whiteout conditions. A major point of contention was the HRRR and NAM model guidance as just the day prior, the HRRR and NAM model forecasted snow in southern Long Island and Northern coastal NJ. By the next day, the NAM model forecasted rain as far north as Plymouth Massachusetts while the HRRR model forecasted an icy mix for New York City[15]. As a result of forecasting errors, there were between 16 and 19 casualties from this storm[16]. These inaccuracies can prove deadly in any severe weather event as warnings determine a location's preparation for an oncoming storm. Recent developments in Machine Learning, however, have shown promising results. The earliest application of machine learning in weather forecasting was in 2016, when Yunjie Liu in the *Application of Deep Convolutional Neural Networks for Detecting Extreme Weather in Climate Datasets*[17] created a Convolutional Neural Network (CNN) to identify different extreme weather events. A

[10] Fox Weather. (n.d.). First Ever Tornado Forecast - March 1948 - Tinker Air Force Base, Oklahoma. Fox Weather. Retrieved April 26, 2023, from https://www.foxweather.com/learn/first-ever-tornado-forecast-march-1948-tinker-air-force-base-oklahoma

[11] NOAA. (n.d.). History of Tornado Forecasting. Celebrating200years.noaa.gov. Retrieved April 26, 2023, from https://celebrating200years.noaa.gov/magazine/tornado_forecasting/

[12] National Weather Service. (n.d.). About Models. Weather.gov. Retrieved April 26, 2023, from https://www.weather.gov/about/models

[13] Met Office. (n.d.). History of numerical weather prediction. Met Office. Retrieved April 26, 2023, from https://www.metoffice.gov.uk/weather/learn-about/how-forecasts-are-made/computer-models/history-of-numerical-weather-prediction

[14] National Weather Service. (n.d.). Improved High-Resolution Models Support Accurate Forecast of a Very Rare Winter Storm. Weather.gov. Retrieved April 26, 2023, from https://www.weather.gov/news/183101_winter-storm

[15] Cioffi, J. (n.d.). Snow Battles HRRR Model vs NAM Model. Meteorologist Joe Cioffi. Retrieved April 26, 2023, from https://www.meteorologistjoecioffi.com/snow-battles-hrrr-model-vs-nam-model/

[16] Wikipedia contributors. (2022, October 30). March 2017 North American blizzard. In Wikipedia, The Free Encyclopedia. Retrieved 23:41, April 26, 2023, from https://en.wikipedia.org/wiki/March_2017_North_American_blizzard

[17] Liu, Y., Racah, E., Prabhat, Correa, J., Khosrowshahi, A., Lavers, D., Kunkel, K., Wehner, M., & Collins, W. (2016). Application of Deep Convolutional Neural Networks for Detecting Extreme Weather in Climate Datasets [PDF file]. Retrieved from https://arxiv.org/abs/1605.01156

CNN is a type of neural network that is commonly used for image analysis. It employs the use of convolutions, a small matrix that performs operations to enhance certain features of an image, and other image processing methods. Since 2016, neural network applications have been expanded to use Generative Networks, Deep Neural Networks, and much more. In Liujia Liu's *Two Stage UA-GAN for Precipitation Nowcasting*[18], the researchers created a two stage Generative Adversarial Network, which combined a Deep Neural Network with an attention based Generative Adversarial Network (GAN).

Dataset

The dataset for this study was curated from the National Weather Service's Warn on Forecast (WOF)[19]. This resource was selected due to its comprehensive coverage of severe weather events, along with the granularity of its data, which includes both radar precipitation imagery and radar sounding data. The images provided by WOF cover an extensive 400 by 500 mile area of the United States, with a resolution of 800 by 900 pixels. These images represent the visual component of the Generative Adversarial Network (GAN). The numerical component for the Neural Network was sourced from the model sounding data provided by WOF. This data is unique in its temporal resolution, offering updates every 5 minutes, and its range of environmental variables, including Dew Point, Temperature, Convective Available Potential Energy (CAPE), Helicity, and Air Parcel Buoyancy over varying elevations. Data extraction was performed using the Powershell curl command to fetch image samples and a custom Python script to generate the respective curl commands, allowing for a streamlined and efficient data collection process. To prepare the data for use in the neural network, several preprocessing steps were carried out. The radar data, color-coded based on the dBZ (radar reflectivity) of the precipitation event, was normalized using a radar dependent scaling method. The numerical sounding data required conversion from image form to a usable numerical format, necessitating the use of an additional image-to-number neural network. The WOF dataset is an ongoing resource, with data available from 2018 and continually updated. Since the time of initial collection, over 2000 new image samples have been added, further enhancing the diversity and robustness of the dataset. This ensures that the trained model is capable of generalizing effectively to new data. Given the broad range of environmental variables contained within the WOF dataset, the proposed neural network architecture is versatile, facilitating easy interchange of variables to cater to specific analysis requirements.

Machine Learning

Machine learning has gained immense popularity in recent years and has shown its efficacy in many applications. At its core, a neural network is a cluster of neurons—a node that has weights and a bias—that takes specific sets of data, or the input, and performs mathematical operations on that data in an attempt to extract information. During the training step, the network makes predictions by having the neuron perform an activation function on a linear combination of weights. Essentially determining how active the neuron is in producing the final result. Next, the neural network calculates the "loss" or error in the model, eventually changing the weights based on the loss number. This loss can also be represented as a function, the minima of said function being the optimal model. The network continues this process for a specified amount of time.

Data is incredibly important when it comes to training machine learning models. In my application, data on many different storms is necessary, with enough variation to cover different storm types. The importance of data collection and curation cannot be overstated. Unfortunately, comprehensive documentation of supercell thunderstorms with high-resolution radar imagery during the storms' development has only recently become available.

As stated earlier, the current leading model for weather prediction is a GAN. A GAN is a type of Unsupervised learning model—one without explicit labels for images—as its goal is to understand the rules governing a set of images and attempt to create images similar to them.

A GAN has a generator, which generates images, and a discriminator, which determines if the images are similar to the input images. A generator starts with a random noisy image and through the discriminator's loss function, refines the image to eventually pass the discriminator's test. The discriminator assigns a binary 0 or 1 value for whether the image is reflective of the original dataset. When the generator "tricks" the discriminator, the discriminator cannot tell the difference between generated images and input images. *Attention* is best defined as the

[18] Xu, L., Niu, D., Zhang, T., Chen, P., Chen, X., & Li, Y. (2022). Two-Stage UA-GAN for Precipitation Nowcasting [PDF file]. Remote Sensing, 14(23), 5948. https://doi.org/10.3390/rs14235948

[19] National Severe Storms Laboratory. (n.d.). Warn on Forecast. Retrieved 5/31/2023, from https://wof.nssl.noaa.gov/

ability to take previous data or interactions into account and weigh the importance of that previous data with respect to the current data point[20]. A common metaphor is x pays attention to x. Metrics commonly used for evaluation of neural networks include sharpness, structural similarity index, and root mean squared error[21].

With the research that I have seen, there are gaps that I see and aim to fill. One of the largest gaps I see is that no study uses both numerical and visual data. The limitation of using only numerical data is that a holistic view of precipitation is missing. On the other hand, using visual data only leads to many missing important factors of atmospheric condition (alongside elevation based variables). This disparity creates a forecasting problem. Long term, precise, forecasting needs both radar and numerical data to create an accurate forecast. My proposed architecture capitalizes on both advantages. My model contains three separate neural networks, arranged in a two step model: A Multi-Head Attention Wasserstein Generative Adversarial Network (MHA WGAN), Recurrent Deep Neural Network (RDNN), and a Dual input Conditional Generative Adversarial Transformer network. The inputs consist of a radar image and a table of atmospheric conditions during that storm's lifetime. These include elevation along the columns and temperature, CAPE, wind shear, dew point, helicity, and barometric pressure along the rows. Each temporal observation corresponds to a new table. Additionally, the table of values correspond to the same storm as the radar data.

The MHA WGAN uses *Multi-Head Attention (MHA)*. MHA can run several *Attention* modules in parallel by dividing the image into sections to feed into each *Attention* model. In other words, it allows different attention modules to focus on specific nuances and features, developing a stronger understanding of the rules governing the dataset[22]. The radar data is normalized using a radar-specific scaling method, adjusting the radar reflectivity value (dBZ) to a standard range. These MHA modules act upon the noisy images by assigning different sections of the images contextualized values, which then allow for greater pattern recognition. However, a problem arises with having MHA modules in the generator of a GAN, a lot of varied training and testing data is required for accurate training of this model. Unfortunately, the sheer amount of data needed may be more than is available. With the current architecture, in excess of 50,000 samples may be necessary for acceptable training. To address this, we can leverage the power of the Wasserstein Architecture. I will elaborate on my specific implementation later. The generator architecture is the same as that of a regular GAN, in that it starts by generating images that get closer to tricking the discriminator; however, the discriminator uses the non-binary Wasserstein function to give a number in a range between zero and one instead of the binary zero or one. This Discriminator is called the critic and assigns higher scores to more "real" images and vice versa[23]. One flaw with this approach is exploding gradients. Gradients are the result of backpropagating through a neural network to assess loss. Backpropagating involves taking the expected result of the neural network and comparing it with the last set of neurons, then by using the chain rule, go to the second to last set of neurons and assess loss and continue to the front of the neural network. To assess the loss, the backpropagation calculates the gradient of the loss function with respect to the weights in the neural network for every input and output example in the testing set by using the chain rule. It then takes the partial derivative of the loss function with respect to the weight variable. Refer to figure 2. In other words, the gradient represents the value by which the weights of a neuron should be altered which causes a downward trajectory in the loss function to eventually reach a global minimum. The losses are the values in which the weights in a neural network should be altered. These values are stored in a matrix called the Jacobian matrix. The size of the Jacobian would be representative of the outputs; in my case, it would be $x * y * z$ where x and y represent the length and width of an image and z represents the amount of images outputted.

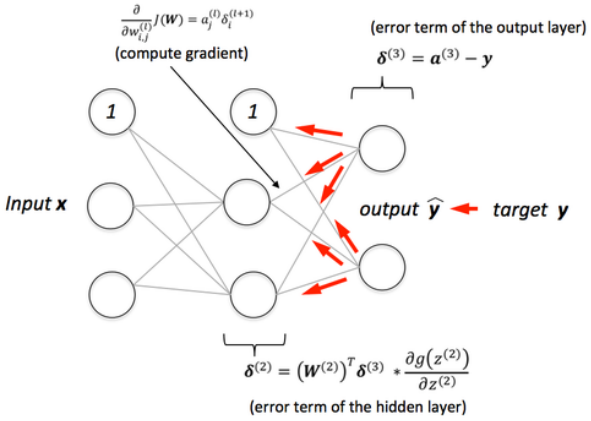
Fig. 2 visual representation of backpropagation

[20] Papers with Code. (n.d.). An Overview of Attention Mechanisms. Retrieved from https://paperswithcode.com/methods/category/attention-mechanisms-1

[21] Xu, L., Niu, D., Zhang, T., Chen, P., Chen, X., & Li, Y. (2022). Two-Stage UA-GAN for Precipitation Nowcasting [PDF file]. Remote Sensing, 14(23), 5948. https://doi.org/10.3390/rs14235948

[22] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention Is All You Need [PDF file]. Retrieved from https://arxiv.org/abs/1706.03762

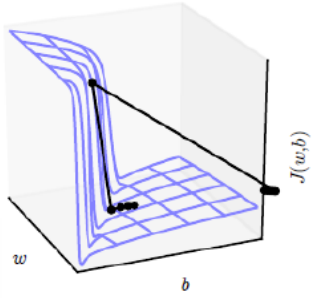
[23] Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein GAN [PDF file]. Retrieved from https://arxiv.org/abs/1701.07875



In complex loss functions, slopes can be vertically asymptotic or near vertically asymptotic. These will cause one of the gradients in the Jacobian matrix to skyrocket in value leading to unstable training. To compensate for this, we implement a gradient clipping function known as the Lipschitz Constraint (Refer to Figure 3). The Matrix R^{w*b*z} represents the Jacobian matrix and by what value the loss should change. In an exploding gradients problem, one dimension of the matrix would be near infinity. Figure 3 shows the graph of the loss function. In this case, the z-th dimension of the matrix has a value that throws off the rest. This causes the gradients to overshoot the minimum of the function and destabilize training as the weights in the neurons cannot "improve" effectively.

Fig. 3 a visual representation of the Jacobian matrix

$$A_{w,b,z} = \begin{pmatrix} \nabla_{1,1,1} & \nabla_{1,2,1} & \cdots & \nabla_{1,b,1} \\ \nabla_{2,1,1} & \nabla_{2,2,1} & \cdots & \nabla_{2,b,1} \\ \vdots & \vdots & \ddots & \vdots \\ \nabla_{w,1,1} & \nabla_{w,2,1} & \cdots & \nabla_{w,b,1} \end{pmatrix}$$

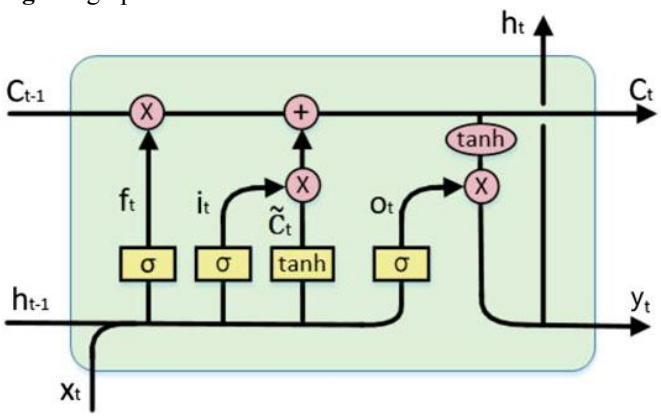


The Lipschitz constraint has shown its results in Ishaan Gurlajani's *Improved Training of Wasserstein GANs*[24] to drastically improve training performance. The goal of this model is to take in radar imagery a certain period prior to a certain storm event with short intervals between images and predict the next specified period. For a technical implementation, we would transform the initial noise vector through a series of leakyReLU, Batch norm, and Dense layers. These would then be passed through upsampling and convolutional layers to be fed into the MHA blocks. A weighted sum of the attention outputs would be taken in through a series of convolutional layers, would pass through a final tanh activation function in the range of -1 to 1. This would then be assessed by the critic and, with the Lipschitz constraint, will perform the backpropagation process. The Recurrent Deep Neural Network (RDNN) uses a specific type of recurrent block known as Long Short Term Memory (LSTM) as they perform better on more complex and varied data[25]. This manages to capture long term patterns with short term dependencies. An LSTM works by having three "gates": the forget gate, input gate, and output gate. The input gate determines which new information should be stored in the cell. The output gate determines what should be outputted from the current cell into the next cell. As a result, it creates a memory of previous data points. In the figure below, C_t is the new updated memory states and H_t is the next hidden state. X_t is the input state, C_{t-1}, and H_{t-1} are the previous cell and hidden states and y_t is the output. Refer to figure 4.

[24] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. (2017). Improved Training of Wasserstein GANs [PDF file]. Retrieved from https://arxiv.org/pdf/1704.00028v3.pdf

[25] Cahuantzi, R., Chen, X., & Güttel, S. (2023). A comparison of LSTM and GRU networks for learning symbolic sequences. arXiv preprint arXiv:2107.02248. Retrieved April 24, 2023, from https://arxiv.org/abs/2107.02248

Fig. 4 a graphic of the LSTM block



The forget gate is Y_t. The inputs would be the tabled values a specified time before a storm event (corresponding to the same event as the GAN from earlier) and the output would be tabled values a specified time after the same storm event. The final architecture that I am proposing is a dual-input Conditional Generative Adversarial Transformer Network (cGATN). This uses an altered version of the Generative Adversarial Transformer model proposed by Hudson[26] and Conditional Generative Adversarial Network proposed by Medhi[27]. To preprocess the data produced by the Deep Recurrent Neural Network (DRNN) and the Multi Head Attention Wasserstein Generative Adversarial Network (MHA WGAN), I will normalize the numerical weather data by converting each datapoint into a Z-score relative to the overall dataset. As stated earlier, radar data will be normalized using a radar-specific scaling method, adjusting the radar reflectivity value (dBZ) to a standard range. To merge the numerical weather data with the radar data for processing by the Bipartite Transformer, I employ a fully connected layer (FC). This layer processes a tensor of numerical data where the X-axis represents weather variables, the Y-axis signifies the timeline, and the Z-axis denotes different elevation levels. Each elevation level is sliced individually, and a 2D weight matrix is initialized to yield intermediate outputs for each elevation and time series. These outputs are then reshaped and aggregated into a 3D tensor. To enable fusion with the 2D radar data, I use a 3D convolution layer to reduce the tensor dimensionality to 2D. Subsequently, the model processes these inputs as grouped sequences, treating each group as a single input to capture temporal dependencies across different timesteps. I will use a sliding window approach to enhance temporal coherence within the model. The generator incorporates LSTM layers, processing the fused data to output a sequence of contextual encodings, represented as LSTM hidden states. These encodings are then fed into the Bipartite Transformer layer within the generator. The Bipartite Transformer layer refines these encodings, utilizing a blend of Simplex and Duplex Attention mechanisms. Simplex Attention captures patterns from the model's past and present states, from the first time step up to the current timestep t. Meanwhile, Duplex Attention extends the focus to future states, from the first time step up to a future timestep T where T > t. This allows the Transformer to consider potential predictive trends in the data. The Transformer block generates enhanced, contextualized encodings. These encodings, along with a random noise vector, are concatenated along the channel dimension, then passed to the upsampling and convolutional layers of the generator. As a Conditional GAN, the generator has already been conditioned with numerical data when generating radar images. For the discriminator, however, I fuse the numerical weather data with both the real and generated radar imagery, using the same fusion method employed in the generator. The real radar imagery refers to the original radar data prior to processing by the first WGAN, while the generated radar imagery is the output from the WGAN. The numerical weather data, which is fused with both real and generated radar imagery. These features are subsequently passed through a fully connected layer, producing the discriminator's output - a measure of its confidence in the reality of the given sample. This dual-input approach encourages the model to learn and adapt based on both original and generated data, thereby improving its overall performance and prediction capabilities.

[26] Hudson, D. A., & Zitnick, C. L. (2021). Generative Adversarial Transformers. arXiv preprint arXiv:2103.01209.
[27] Mirza, M., & Osindero, S. (2014). Conditional Generative Adversarial Nets. arXiv preprint arXiv:1411.1784.

Conclusion

This novel Machine Learning algorithm shows promise in complex supercell storm formation and has potential applications in forecasting snow squalls, blizzards, derechos, and occluded cold fronts. Occluded cold fronts can spawn bomb cyclones, causing significant damage in the form of snow, ice, and tornadoes, which have remained challenging for forecasters to predict. While this application addresses many key problems with current machine learning implementations, one major drawback is the requirement for extreme amounts of computing power for model training. However, this can be overcome by leveraging cloud computing. This model has the potential to create a significant impact in the field of weather prediction. Further improvements could be achieved by developing a three-step model that includes an additional input of model sounding image. This could also be modified to have a singular input with model sounding data, branching out into two neural networks, one with radar data and one with model sounding data, and then concatenating them into one output. Additionally, this model could be adapted to issue warning polygons for more hyperlocal regions, further in advance, enhancing its predictive capabilities.

Appendix

Vorticity: A measure of the rotation of a fluid or how it interacts with surrounding wind fields. This can be calculated by taking the CURL of a vector field.

Lift: The upper movement of air. In severe weather terminology, when air is traveling along an upward-sloping isentropic surface.

Isentropic Surface: a two-dimensional surface composed of points with equal potential temperature.

Potential Temperature: The temperature a parcel of dry air would have to be if heated or cooled adiabatically to standard pressure (100 kPa)

Adiabatic Heating/Cooling: Temperature changes that occur without any transfer of heat from itself to its surroundings. Temperature changes happen solely due to pressure.

Air Parcel: An imaginary unsaturated blob of air that can rise.

Unsaturated Air: Air with less than 100% relative humidity

Relative Humidity: the amount of moisture in the air compared to the maximum amount of moisture that air can hold at that specific temperature.

Specific Humidity: The ratio of moisture mass to the total mass of moisture and air.

Dew Point: The temperature at which air must be cooled to become saturated with water vapor. Condensation will occur and will cause cloud formation and precipitation.

Convection: The transfer of heat by the movement of a fluid through a medium, such as air or water. When a fluid is heated, it becomes less dense and rises.

Wind Shear: The change in wind speed over a short altitude, commonly associated with cold fronts and Supercell Thunderstorms.

Generative Adversarial Network(GAN): An unsupervised Machine Learning model that aims to produce images that closely resemble its input. It has two models: a generator and a discriminator. The Generator is given a random noisy matrix (usually Gaussian Noise). In a traditional GAN, the generator contains convolutional neural networks

in sequence. DownSampling2D layers, Conv2DTranspose layers for inverse convolutions, and then UpSampling2D layers to output into the next convolutional layer.

Attention: The standard attention module has an Encoder-Decoder framework. The encoder steps through each time sequence, performing Input embedding to create the contextualized input vector. The input vectors are represented by queries, keys, and values where the Queries represent the element in which we want to compute attention. The keys represent the elements we want to attend to and the Value represents the elements we want to aggregate based on the attention scores. The decoder is responsible for stepping through time and comparing the output (shifted) vector with the contextualized vector. Each query-key pair is stepped through in a full sequence. In each time step, the mechanism takes a hidden state of the output vector and uses either dot-product or cosine similarity to create normalized score values. These scores are then normalized through a softmax function to generate weights. These encoded vectors are then scaled using the weights through dot product attention to creating the contextualized vectors. These context vectors are fed through the decoder which performs a dot product of the Queries, Keys, and Values with the contextualized vectors. On images, the same process would occur with the first step being a concatenation of the image into a 1-D array to start.

Multi-Head Attention(MHA): This runs multiple *Attention* modules in parallel and allows for content mixing. The contextualized vectors from each head of the Multi-Head Attention model are concatenated to produce an output. Since the inputs are represented by Queries, keys, and values, the MHA mechanism can share and withhold these values from other attention mechanisms.

Wasserstein Function: This is a distance function that measures the "difference" between two independent probability distributions (a distance function is also interchangeable with a loss function in this case).

$$d(\widehat{F}_A, \widehat{F}_B) = \frac{1}{K}\sum_{k=1}^{K}(Q_A^{a_k} - Q_B^{a_k}) = (\widehat{\mu}_A - \widehat{\mu}_B)^2 + (\widehat{\sigma}_A - \widehat{\sigma}_A)^2 + 2\widehat{\sigma}_A\widehat{\sigma}_B(1 - \widehat{\rho}^{A,B})$$

$\widehat{F}_A$ represents one dataset and $\widehat{F}_B$ represents the other. μ represents the respective means of the datasets, σ represents the standard deviations and, ρ represents the Pearson Correlation. The Pearson correlation is a metric similar to the R squared value in which it measures the strength of a linear relationship between datasets.

Transformer: The Transformer takes in an input sequence and then converts it into embeddings and these serve as the positional representations of input elements. Then, performs Multi-Head attention on these inputs given the queries, keys, and values. Then it feeds the inputs through a Position wise feed-forward neural network which applies a nonlinearity to each positional vector with a ReLU or Sigmoid activation function in between. The output of the feed-forward neural network is added to the input embeddings and then normalized. Residual connections are also used in the encoder which allows input embeddings to skip certain steps and help mitigate overfitting. The output of the Encoder serves as the input features for the second layer of masked multi-head attention block in the decoder. The decoder takes embedded output vectors and puts them through 2 masked multi head attention blocks. The first Multi-head attention block computes the same as the encoder, just with the output vectors. The second Multi Head attention block computes how well the output and input sequences align and the Position-wise feed-forward neural network then produces the output.

References

1. Cristina, S. (n.d.). Implementing the Transformer Decoder from Scratch in TensorFlow and Keras. Machine Learning Mastery. Retrieved 4/19/23, from https://machinelearningmastery.com/implementing-the-transformer-decoder-from-scratch-in-tensorflow-and-keras/
2. Cristina, S. (n.d.). The Transformer Model. Machine Learning Mastery. Retrieved 4/19/23, from https://machinelearningmastery.com/the-transformer-model/
3. Kikaben. (n.d.). Transformers: Encoder-Decoder. Retrieved 4/19/23, from https://kikaben.com/transformers-encoder-decoder/
4. Storrs, A. (n.d.). Attention. Retrieved 4/19/23, from https://storrs.io/attention/
5. Brownlee, J. (n.d.). How Does Attention Work in Encoder-Decoder Recurrent Neural Networks? Machine Learning Mastery. Retrieved 4/19/23, from https://machinelearningmastery.com/how-does-attention-work-in-encoder-decoder-recurrent-neural-networks/
6. Google Developers. (n.d.). Discriminator. Retrieved 4/19/23, from https://developers.google.com/machine-learning/gan/discriminator
7. Towards Data Science. (n.d.). Attention in Computer Vision. Retrieved 4/19/23, from https://towardsdatascience.com/attention-in-computer-vision-fd289a5bd7ad
8. TensorFlow. (n.d.). Deep Convolutional Generative Adversarial Network. Retrieved 4/19/23, from https://www.tensorflow.org/tutorials/generative/dcgan
9. ThoughtCo. (n.d.). What Is Wind Shear? Retrieved 4/19/23, from https://www.thoughtco.com/what-is-wind-shear-3444340
10. Wikipedia. (n.d.). Wind shear. Retrieved 4/19/23, from https://en.wikipedia.org/wiki/Wind_shear
11. North Carolina State University Climate Office. (n.d.). Retrieved 4/19/23, from https://climate.ncsu.edu/
12. American Meteorological Society. (n.d.). Convection. Retrieved 4/19/23, from https://glossary.ametsoc.org/wiki/Convection
13. National Weather Service. (n.d.). Glossary - Convection. Retrieved April 23, 2023, from https://forecast.weather.gov/glossary.php?word=convection
14. Live Science. (n.d.). What is Dew Point? Retrieved April 23, 2023, from https://www.livescience.com/43269-what-is-dew-point.html
15. Weather Research Center. (n.d.). Specific Humidity vs Relative Humidity. Retrieved April 23, 2023, from https://wxresearch.org/specific-humidity-vs-relative-humidity/
16. Encyclopedia.com. (n.d.). Adiabatic Heating. Retrieved April 23, 2023, from https://www.encyclopedia.com/science/encyclopedias-almanacs-transcripts-and-maps/adiabatic-heating
17. Shodor Education Foundation. (n.d.). Potential Temperature Calculator. Retrieved April 23, 2023, from http://www.shodor.org/os411/courses/_master/tools/calculators/pottemp/pt1calc.html
18. National Weather Service. (n.d.). Glossary - Isentropic Surface. Retrieved April 23, 2023, from https://forecast.weather.gov/glossary.php?word=ISENTROPIC%20SURFACE
19. NASA Glenn Research Center. (n.d.). What is Lift? Retrieved April 23, 2023, from https://www1.grc.nasa.gov/beginners-guide-to-aeronautics/what-is-lift/
20. American Meteorological Society. (n.d.). Vorticity. Retrieved April 23, 2023, from https://glossary.ametsoc.org/wiki/Vorticity
21. NOAA National Weather Service JetStream. (n.d.). Constant Pressure Charts - 500 mb. Retrieved April 23, 2023, from https://www.noaa.gov/jetstream/upper-air-charts/constant-pressure-charts-500-mb
22. Machine Learning Mastery. (n.d.). The Transformer Model. Retrieved April 23, 2023, from https://machinelearningmastery.com/the-transformer-model/

23. Cahuantzi, R., Chen, X., & Güttel, S. (2023). A comparison of LSTM and GRU networks for learning symbolic sequences. arXiv preprint arXiv:2107.02248. Retrieved April 24, 2023, from https://arxiv.org/abs/2107.02248
24. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. (2017). Improved Training of Wasserstein GANs. arXiv preprint arXiv:1704.00028v3. Retrieved April 24, 2023, from https://arxiv.org/abs/1704.00028
25. Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein GAN. arXiv preprint arXiv:1701.07875v3. Retrieved April 24, 2023, from https://arxiv.org/abs/1701.07875
26. Liu, Y., Racah, E., Prabhat, Correa, J., Khosrowshahi, A., Lavers, D., Kunkel, K., Wehner, M., & Collins, W. (2016). Application of Deep Convolutional Neural Networks for Detecting Extreme Weather in Climate Datasets. arXiv preprint arXiv:1605.01156. Retrieved April 24, 2023, from https://arxiv.org/abs/1605.01156
27. Papers with Code. (n.d.). Multi-Head Attention. Retrieved April 23, 2023, from https://cs.paperswithcode.com/method/multi-head-attention
28. Papers with Code. (n.d.). Attention Mechanisms Methods. Retrieved April 23, 2023, from https://paperswithcode.com/methods/category/attention-mechanisms-1
29. Linzen, T., Dupoux, E., & Goldberg, Y. (2016). Assessing the ability of LSTMs to learn syntax-sensitive dependencies. Transactions of the Association for Computational Linguistics, 4(1), 521-535.
30. Wikipedia contributors. (n.d.). March 2017 North American blizzard - Wikipedia. Retrieved April 23, 2023 from https://en.wikipedia.org/wiki/March_2017_North_American_blizzard
31. Cioffi Jr., J. (n.d.). Snow Battles HRRR Model vs NAM Model - Meterologist Joe Cioffi Weather Forecast New York, New Jersey, and Connecticut. Retrieved April 23, 2023 from https://www.meteorologistjoecioffi.com/snow-battles-hrrr-model-vs-nam-model/
32. National Weather Service Newsroom (n.d.). Improved High-Resolution Models Support Accurate Forecast of a Very Rare Winter Storm from https://www.weather.gov/news/183101_winter-storm#:~:text=As%20the%20event%20drew%20closer%2C%20the%20HRRR%20%28and,of%20past%20model%20handling%20of%20shallow%20cold%20air.
33. - National Weather Service Newsroom [Press release]. Retrieved April 23rd 2023 from https://www.weather.gov/news/183101_winter-storm#:~:text=Despite%20the%20known%20NAM%20bias%20of%20over-predicting%20precipitation,hours%29%20precisely%20forecast%20both%20precipitation%20type%20and%20amount.
34. Met Office UK (n.d.). History of numerical weather prediction - Met Office UK. Retrieved April 23rd 2023 from https://www.metoffice.gov.uk/weather/learn-about/how-forecasts-are-made/computer-models/history-of-numerical-weather-prediction
35. National Weather Service (n.d.). Numerical Weather Prediction Models - National Weather Service. Retrieved April 23rd 2023 from https://www.weather.gov/about/models
36. NOAA Celebrating 200 Years (n.d.). Tornado Forecasting - NOAA Celebrating 200 Years. Retrieved April 23rd 2023 from https://celebrating200years.noaa.gov/magazine/tornado_forecasting/
37. Fox Weather LLC (n.d.). First Ever Tornado Forecast March 1948 Tinker Air Force Base Oklahoma - Fox Weather LLC. Retrieved April 23rd 2023 from https://www.foxweather.com/learn/first-ever-tornado-forecast-march-1948-tinker-air-force-base-oklahoma
38. US Safe Room Tornado Shelters (n.d.). Protect Yourself From Supercell Thunderstorms - US Safe Room Tornado Shelters. Retrieved April 23rd 2023 from https://www.ussaferoom.com/protect-yourself-supercell-thunderstorms/
39. LibreTexts Library (n.d.). How can supersaturation be achieved? - LibreTexts Library. Retrieved April 23rd 2023 from https://geo.libretexts.org/Bookshelves/Meteorology_and_Climate_Science/Book%3A_Fundamentals_of_Atmospheric_Science_(Brune)/05%3A_Cloud_Physics/5.04%3

40. RAMMB/CIRA (n.d.). CAPE - RAMMB/CIRA. Retrieved April 23rd 2023 from https://rammb.cira.colostate.edu/wmovl/VRL/Tutorials/SatManu-eumetsat/SATMANU/Basic/Convection/Cape.htm
41. NOAA Storm Prediction Center (n.d.). Supercells - NOAA Storm Prediction Center. Retrieved April 23rd 2023 from https://www.spc.noaa.gov/misc/AbtDerechos/supercells.htm
42. Encyclopædia Britannica (n.d.). Super Outbreak of 2011 - Encyclopædia Britannica. Retrieved April 23rd 2023 from https://www.britannica.com/event/Super-Outbreak-of-2011
43. Tornado Facts (n.d.). Tri-State Tornado Facts - Tornado Facts. Retrieved April 23rd 2023 from https://www.tornadofacts.net/tri-state-tornado-facts.html
44. Encyclopædia Britannica (n.d.). Saturia-Manikganj Sadar tornado - Encyclopædia Britannica. Retrieved April 23rd 2023 from https://www.britannica.com/event/Saturia-Manikganj-Sadar-tornado
45. National Weather Service Amarillo TX (n.d.). Supercell Thunderstorms - National Weather Service Amarillo TX. Retrieved April 23rd 2023 from https://www.weather.gov/ama/supercell
46. NOAA National Severe Storms Laboratory (n.d.). Warn-on-Forecast System - NOAA National Severe Storms Laboratory. Retrieved April 23rd 2023 from https://wof.nssl.noaa.gov/
47. Shi, X., Chen, Z., Wang, H., Yeung, D. Y., Wong, W. K., & Woo, W. C. (2015). Convolutional LSTM network: A machine learning approach for precipitation nowcasting. arXiv preprint arXiv:1506.04214v1. Retrieved April 23rd 2023 from https://github.com/sxjscience/HKO-7
48. Mastin, B. (2017, April 1). Hook Echo Tornado [Pinterest post]. Retrieved April 24, 2023, from https://www.pinterest.com/pin/314196511486756181/
49. Mirza, M., & Osindero, S. (2014). Conditional Generative Adversarial Nets. arXiv preprint arXiv:1411.1784.
50. Hudson, D. A., & Zitnick, C. L. (2021). Generative Adversarial Transformers. arXiv preprint arXiv:2103.01209.