Machine learning for modelling of tipping points in climate variables

Final project Report

Joshua Curry

1 Abstract

Human-caused climate change is becoming an ever-increasing danger to our planet. We are close to, reaching a point upon which changes or tipping points will soon become irreversible. It is therefore essential that we are able to foresee when these tipping points arise. Machine Learning can be an efficient and capable tool in predicting these tipping points. This report outlines what other researchers have done in the use of Machine Learning to model tipping points, as well as demonstrating the development and results of my own example for modelling tipping points. Despite the failure of my model, the overall aim and essence of the project to develop a Machine Learning model that can assist climate researchers in their efforts to model climate tipping points remains credible.

I certify that all material in this dissertation which is not my own work has been identified.

Contents

2 Motivation, Aims, and Project Specification

2.1 Project Motivation

Around the world, we are seeing climate systems subjected to long-term shift in weather patterns, with some areas, such as Western Germany and Eastern Belgium, suffering, in July 2021, cataclysmic flood disasters. Furthermore, scientists have been recording for some time a signification reduction in the concentration of the Northern and Southern Hemispheres' sea ice cover [1] [2]. As a consequence of this, dealing with climate change increasingly grows in importance as it affects and threatens every aspect of life. As such, now more than ever, there is a need to handle and action potential irreversible change in climate systems. These irreversible changes, can be identified by key tipping points.

2.1.1 Definition and Identification of Tipping Points

For the Intergovernmental Panel on Climate Change [3], tipping points are "critical thresholds in a system that, when exceeded, can lead to a significant change in the state of the system, often with an understanding that the change is irreversible" [4] [5]. There are multiple variables in a climate system that can have a tipping point, and these variables are called climate elements. These climate elements can be quite immediate such as the Boreal Permafrost Collapse and the Arctic Winter Sea Ice Collapse, to longer term elements such as the West Antarctic Ice Sheet Collapse and the East Antarctic Subglacial Basin's Collapse [5].

2.1.2 Importance of Modelling Tipping Points

Due to human involvement, we are experiencing new tipping point thresholds [5]. Current research shows that there are up to 15 tipping elements currently active [5], the most notable being the internationally agreed Paris Agreement range of $1.5°C$ to $2°C$ increase in average global temperatures. This would trigger many tipping elements, and in the words of many scientists "Climate Tipping Points become likely (with a further four possible) [. . .], including collapse of the Greenland and West Antarctic ice sheets, die-off of low-latitude coral reefs, and widespread abrupt permafrost thaw" [5]. Right now, the world is heading towards an average global temperatures increase of $3.0°$ to $4.0°$ [5], already far exceeding the Paris Agreement range. Thus, there is strong need to understand when these tipping points are coming so that scientists can best prepare for them.

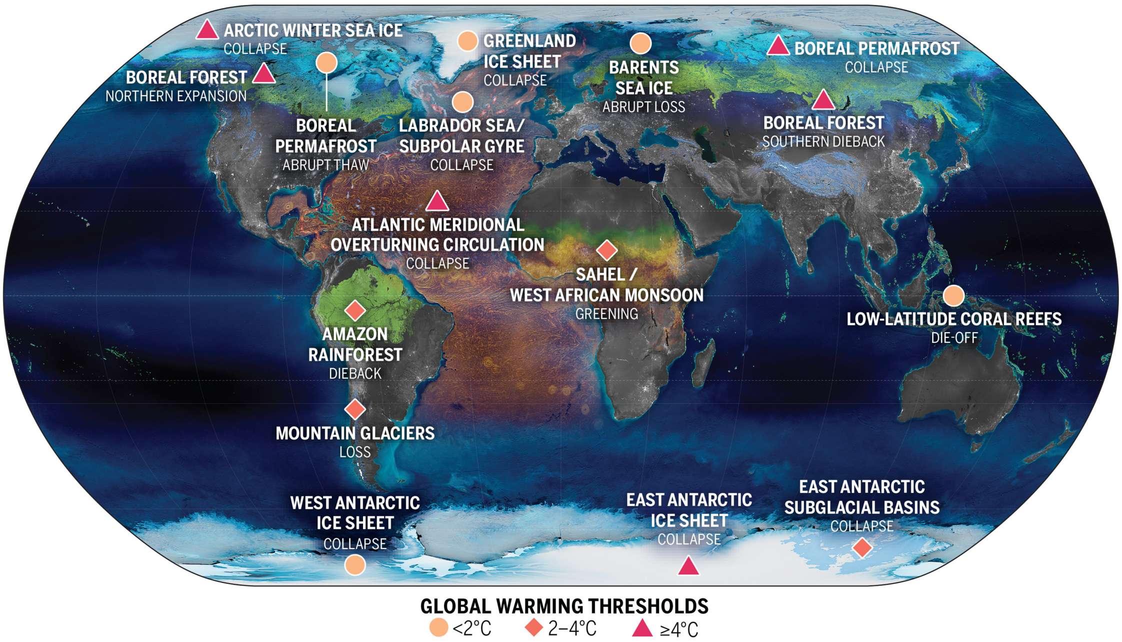

Figure 1: The location and temperature thresholds of climate tipping elements [5]

2.1.3 Why Use Machine Learning To Model Tipping Points

Due to the complexity of the natural world, climate researchers who wish to model this world as accurately as possible, have encountered many difficulties. Through the efforts of many skilled climate researchers, advanced and complicated models such as UKESM1 and HadGEM-GC3.1 have been developed. These are high performance, using complex equations in attempt to replicate the many features of earths climates, such as oceans, ice, land, and atmosphere in an effort to simulate the planet in its entirety. "The complexity of coupling between the ocean, land, and atmosphere physical climate and biogeochemical cycles in UKESM1 is unprecedented for an Earth system model" remarked Alistair A.Stellar *et al* [6] (Met Office Hadley Centre, Exeter) [7], [8].

Whilst variations from other components could cause some sensitivity to biases, it would be inaccurate to say that the models, such as UKESM1, are unproductive or unsuccessful. However, they are potentially not the most efficient in terms of resources due to their large scale and years of development time [8]. As such, Machine Learning could possibly provide a more resource efficient solution because a Machine Learning solution does not attempt to replicate the many minute factors in the real world. As in there will not be need to replicate each individual equation and system, it will only attempt to analyse the data provided and make predictions from that data, without regard to the physical real world. Whilst this may result in a less in depth model, it does mean that the task to be done will be far smaller in scale, thus not requiring the same high level of resources.

2.2 Existing Machine Learning Models For Modelling Tipping Points

There have been multiple pre-existing instances of academics using Machine Learning (ML) models for modelling tipping points. The first of these being T.M. Burry *et al* [9], where they used a Convolutional Neural Network—Long Short-Term Memory Network, which uses two different types of Artificial Neural Networks(ANN). The Convolutional Neural Network (CNN) layer analyses consecutive segments of the time series, identifying the unique characteristics within each segment, this is then fed as the input to the Long Short Term Memory (LSTM) layer which then analyses this input. LSTM is proficient at recognizing patterns and predicting sequences, which is why it was used here.

Using this architecture T.M. Burry *et al* [9] were able to successfully model tipping points using Deep Learning (DL), showing that DL can be used for modelling tipping points.

A second case of ML being used is P. Gutiérrez *et al* [10] who took advantage of Ordinal Regression. Ordinal regression (OR), also known as ranking learning, is a progression algorithm utilized for predicting an ordinal variable. Such a variable possesses an inherent order, yet the precise distances between its values are unknown. OR facilitates the prediction of whether a set of independent variables can effectively forecast an ordinal dependent variable, this was leveraged, to great effect, by P. Gutiérrez *et al* [10] in creating their model.

In a third case, researchers employed a Generative Adversarial Network (GAN) to support climate research efforts [7]. By providing a guiding direction, this approach aimed to streamline the modelling process for detecting tipping points, making it more computationally efficient. Their findings indicated that the ML model had potential in aiding climate researchers with Climate Tipping Point (CTP) predictions, serving as a guiding tool for domain experts.

Overall, these cases studies show the potential effectiveness ML has in assisting climate scientist in detecting tipping points. They also showcase the variety of different methods which can be used to do this.

2.3 Project Specification

This specification for this final project differs heavily from that of initial literature review. The initial proposal was to develop four separate models each using a different method of machine learning, these would then be compared to each other and non-machine learning models. However, this proved to be overly ambitious, as such the scope of the project was brought down. Instead, the project now focusses on developing a singular model and comparing that model to pre-existing machine learning and non-machine learning models.

2.3.1 The Aims of The Project

The aims of the project are:

- Using machine learning to develop an effective model to predict future changes in climate variables
- Using the model, identify future tipping points in the climate variables
- Analyse and evaluate the model, comparing to pre-existing machine learning and non-machine learning models

2.3.2 Success Criteria

Once the model is at a stage where it is possible to draw a conclusion on its effectiveness, it could be considered complete. For the model to truly succeed, however, it should produce results that align or are close to the advanced non-Machine Learning predictions such as the Intergovernmental Panel on Climate Change (IPCC) Sixth Assessment Report [11]. The overall success criteria is as follows:

1. The line of fit should be positive, to indicate the overall upwards trend

2. Each value the model predicts should be unique
3. The model should predict results that at least partially align with pre-existing non-machine learning estimates

2.3.3 Project Risks

There should be minimal ethical issues that arise from this project due to the lack of human involvement. The only potential legal issue would be the use of data without the right approved licence in place [12]. This is mitigated by only using open data.

The first potential risk would be that the dataset being used is not sufficiently detailed in terms of number of years for training the model. In order to avoid this, the dataset should contain data for at least the last 100 years, which should bring a more reliable outcome. If this risk does arise in spite of this, then another dataset but much larger should be used. However, the downside of this approach would in turn lead to the additional risk of failure due to time-sensitivity, as switching dataset will greatly affect the development time.

The second risk therefore is the time-frame, whereby not enough time has been allowed to develop the model. Should this risk occur, then, wherever the model currently is in its development, this will be used as the final model. Plans for future improvements as well as fixes to the model will be drawn out and explained.

3 Design, Methods and Implementation

3.1 Project Design

The overall design of the project is to load the dataset, pre-process the data, train a ML model using this data, get predictions from this model, then graphing and analysing those predictions.

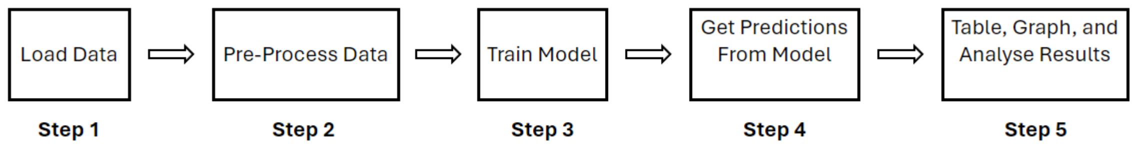

Figure 2: General Design of the Project

3.1.1 Model Design

The ML model to be used will be a multi-layered Long Short-Term Memory (LSTM) network.

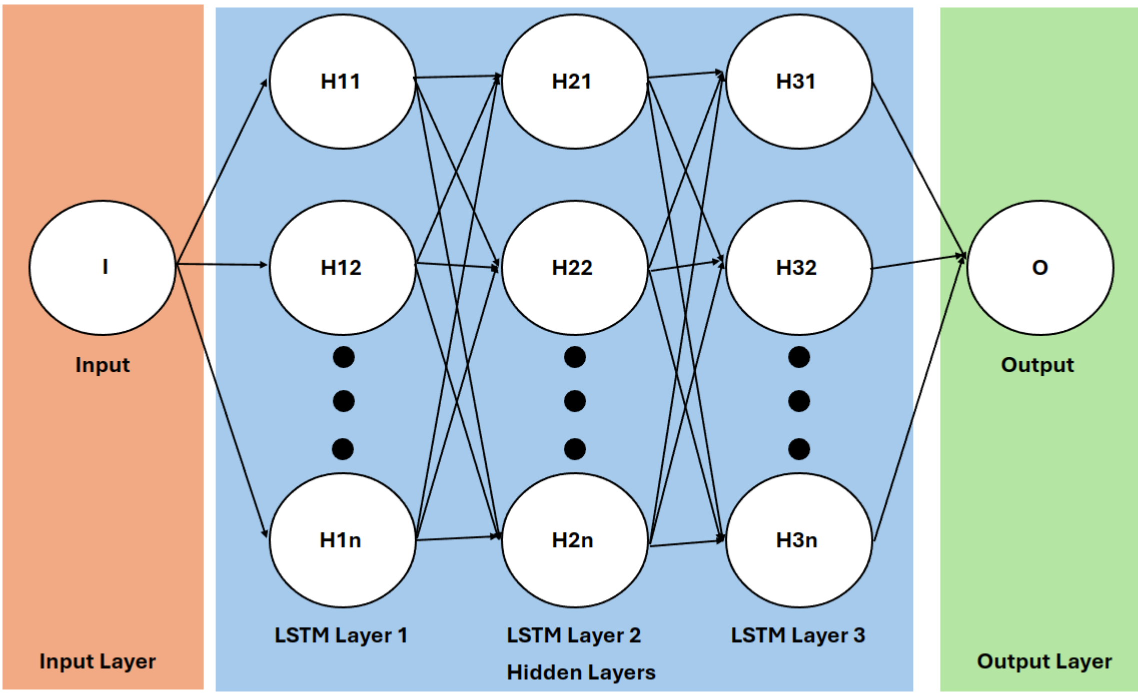

Figure 3: Model Design For This Project

Figure 3 shows the design of the model. The model has an input layer, 3 hidden layers, and an output layer. The hidden layers will be using the LSTM architecture, and the number of nodes each layer has will be descending, so that if hidden layer 1 has n nodes then hidden layer 2 will have $\frac{n}{2}$ nodes and so forth.

3.2 Project Methods

There are many possible machine learning methods that can be used to develop this project, however, some of them are more fit for the task than others. As such, it is important to identify which of them may be the most effective. The following subsection outlines the methods which will be used for this project.

3.2.1 Artificial Neural Networks

Human brains perform calculations using an interconnected network of neurons, Artificial Neural Networks (ANN) attempt to replicate this process [13]. A traditional ANN consist of three layers, an input layer, a hidden layer(s) and an output layer (see my drawings of the figures 4 and 5 below).

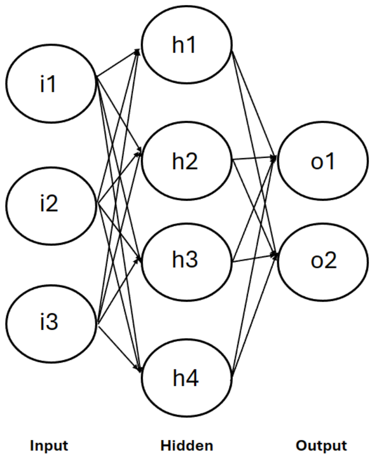

Figure 4: ANN Example Design Structure

The input and output layers do as their name suggests, whilst the hidden layer is what is the intermediary step responsible for learning and computing the data [13]. Each of these layers consist of a set of nodes which represent the neurons in the brain [13], each node is connected to other nodes, if the output of a node is above the threshold value then the node is activated [13]. The function to determine whether a node should activate is called the activation function [14]. Once a node is activated, it sends data to the next layer of the neural network, this data is then used as the input data for the next layer in the network.

If there are more than one hidden layer in Figure 5 then it would be a Deep ANN (DANN), the use of which is known as Deep Learning (DL).

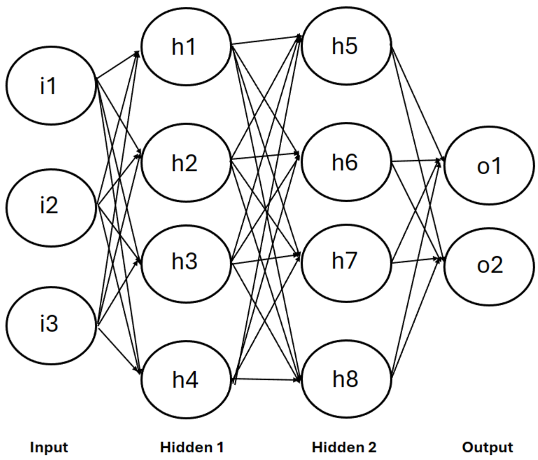

Figure 5: Example DANN Design Structure

3.2.2 Recurrent Neural Networks

Traditionally, in DL, inputs and outputs are considered independent of each other [15]. In a Recurrent Neural Network(RNN) however, takes data from previous inputs to influence the current input and output [15].

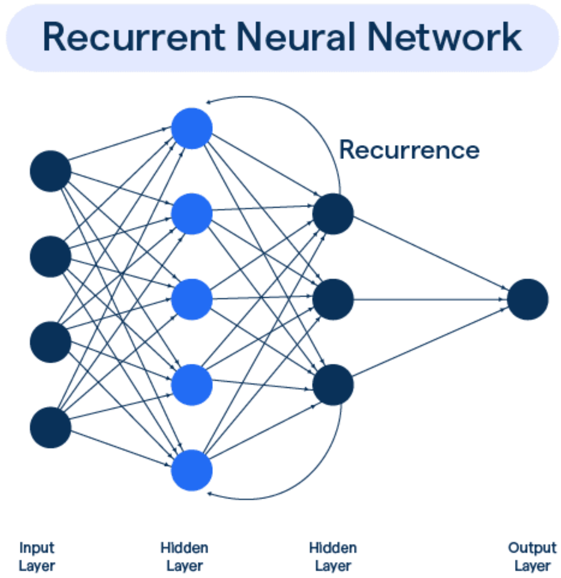

Figure 6: RNN Basic Structure [16]

The strength of RNN comes from its ability to pass information through sequential steps whilst still processing data one element at a time [17]. This means that the RNNs can more effectively model data that is not independent [17], and this also allows RNNs to be more effective when handling time series data [17]. Despite the advantages of RNNs, they do come with drawbacks. When training RNNs it is common to encounter the vanishing and exploding gradient problems [18], which are more prominent with longer sequences of data.

3.2.3 What is Long Short-Term Memory (LSTM)?

LSTM is a type of Recurrent Neural Network(RNN). Above is a diagram representing the design similar to that of the model being used in this project.

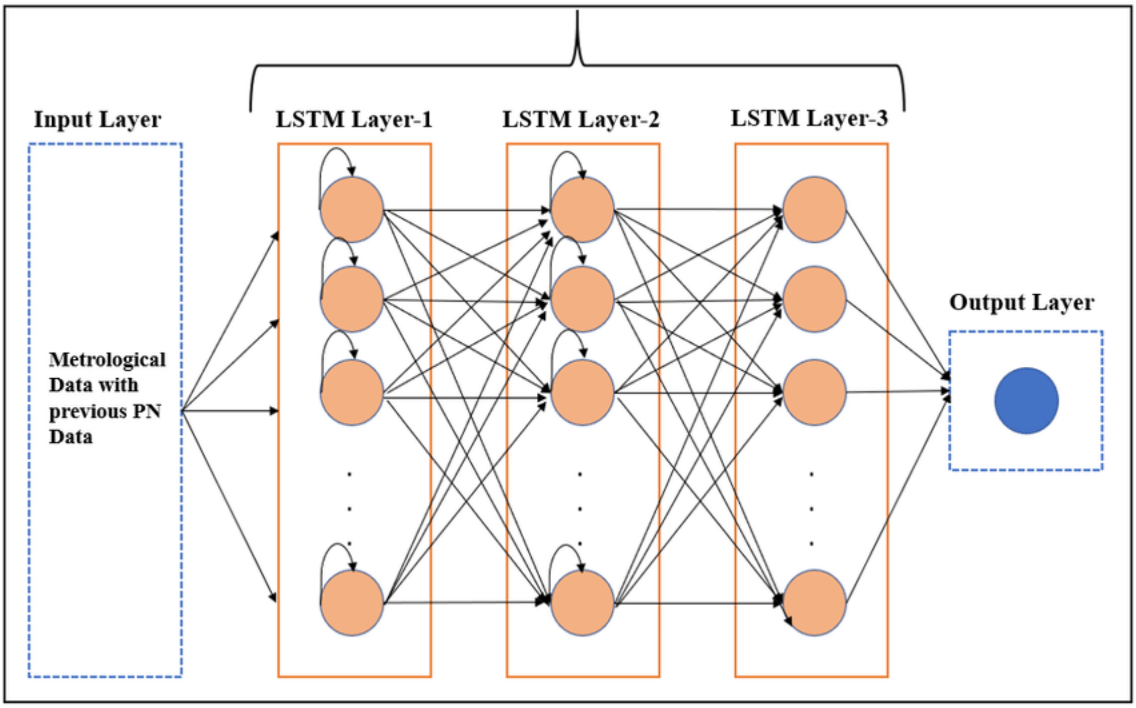

Figure 7: LTSM Architecture Example [19]

The structure of an LSTM network consists of a set of LSTM cells, each having a set of gates that control the flow information for that cell. Along with the gates, the LSTM cell also has a memory that retains information from previous time steps, this information is then used to influence the output of the cell. After which the output from one cell is passed onto the next cell, which allows the LSTM to process data over multiple time-steps [20].

3.2.4 Why Use LSTM?

LSTM is often used in time series prediction [21], which makes it appropriate as this project is working with a time series problem. The advantage LSTM has over conventional back-propagation is how it can handle the exploding and vanishing gradient problems [20], due to how it uses its gating mechanisms. The gating mechanisms are what allow gradients to back-propagate more easily through depth or time [22].

3.2.5 How Does LSTM Work?

To explain how LSTM works, let's start with explaining what the memory cells are and how they work. The memory cells are what allow for constant error flow without the disadvantages of the input weight conflict, and the output weight conflict found in the other approaches to constant error flow [20].

An LSTM consists of three types of gates, those being; the Input Gate, which is what controls how information flows in the memory cell; the Forget Gate, which is what controls how information flows out of the memory cells; and the Output Gate, which controls how the information flows out of the LSTM model [23].

LSTM gates learn to open/close using input and previous state, enabling selective retention or discarding of information, enhancing long-term dependency capture [20].

3.2.6 Activation Functions

Activation functions are what calculate the output of the node based on the individual inputs and the weights of the inputs [14]. Without an activation function, the model will just become a linear regression model.

There are many different types of activation functions, but for this project we will use the Rectified Linear Unit (ReLU) function. ReLU can be defined as

$$f(x) = max(0, x) \tag{1}$$

This means that the nodes will activate when the value of x is greater than 0.

3.3 Implementation

The project is broken down into three major stages:

- Creating & training the model
- Getting data from the model
- Analysing and graphing results from the model

3.3.1 Creating Model

The architecture of the model is as follows

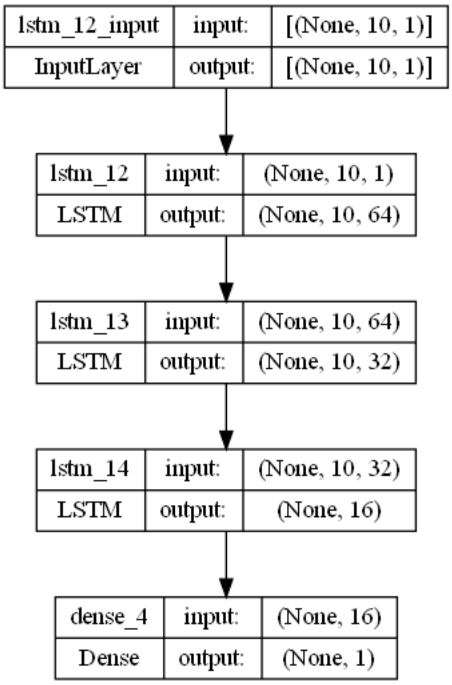

Figure 8: Visualization of the Models Architecture

As seen in the figure 8 the model consists of an input layer, 3 hidden layers and an output layer. The input and output layer both consist of one node, whilst the hidden layers contain 64, 32, and 16 nodes. The reason for the decreasing number of nodes in the hidden layers is to save on performance cost. As found in testing, having more nodes in the later layers did not result in a better result therefore, to save on performance cost, they were decreased.

3.3.2 Normalization

When developing an ML project, it is important to either standardise or normalise your data. For this project, the data is being normalised instead of standardised, as the data does not follow a Gaussian distribution. For this project, Min-Max Scaling is used.

What Min-Max Scaling does is it re-scales the data to be in the range of 0 to 1 [24]. The formula for Min-Max Scaling is

$$x_{scaled} = \frac{x - min(x)}{max(x) - min(x)} \tag{2}$$

where x is the original cell value, x_{scaled} is the scaled value of x, $min(x)$ is the minimum value of x for that column, and $max(x)$ is the maximum value of x. This is useful as it is able to maintain the shape of the original data whilst bringing the values into range, which results in better model performance [24].

3.3.3 Cross Validation

Cross Validation (CV) is what is used to gauge whether a model the output of your model is due to its design or luck [25]. Traditionally, CV works with the following process:

1. Randomly shuffle the dataset
2. Then split the data set into k amount of groups
3. For each group:
   (a) Have one group be the test data
   (b) Then have the remaining groups as the training set
   (c) Fit the model to training set
   (d) Evaluate the model to the test set
   (e) discard the model but retain the evaluation score
4. Using the sample of model evaluation scores:
   (a) Summarise the skill of the model

Unfortunately this method does not work for time series problems due to the random shuffle, as the sequence of the values is important to the problem itself. As such, instead divide the dataset into k groups, where each group consists of a continuous sequence of data, maintaining the temporal order of the data whilst still allowing the CV to take place.

3.3.4 Getting Results From Model

The model is designed to predict the next data point. To draw the next prediction, the previous prediction is added to the end of the data sequence and a new prediction is generated. As this is done for each month in the dataset, and multiple years of predictions need to be generated, it will take a significant amount of time to run. As such, when implementing this, it is important to be able to run this section of the project without having to re-run the previous training module or re-do the later graph drawing segments. As such, after the model is trained it is saved, and then the module for getting the predictions loads the model, after the predictions have been retrieved they are then saved, so that the module graphing, tabling, and analysing can load the predictions. Below is a figure of the file structure for the project.

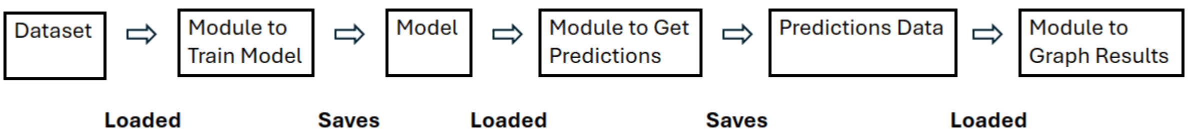

Figure 9: Project File Structure

3.3.5 Analysing and Graphing Results From Model

As there are multiple graphs and tables that need to be drawn, the module will be divided so that there is a boolean or integer value that determines whether to draw a specific graph or table. This value is changed manually in the code as any possible user for this project should have access to the code itself, as such a front end UI that handles this will not be needed.

When graphing and tabling the results themselves, it is important to understand that there are simply too many points of data for them to all be directly graphed/tabled whilst having the model still readable. Due to this, when drawing graphs it is important to graph only specific moments in time, or graph averages over longer periods of time. Though it is possible to create a large table of all the predictions, when showcasing the table it is best to just drill down and show smaller extracts of it as the entire table would be too large.

3.3.6 Dataset

The dataset used is from Kaggle [26], which draws upon research from Berkeley Earth [27]. Since 1750 CE, Berkeley Earth has been diligently measuring global temperatures, establishing itself as a reputable institution in climate research. With its extensive global coverage, the data it provides is assuredly reliable and of exceptional quality.

This dataset primarily focuses on temperature. The reason a dataset of this focus has been chosen is because having read scientific climate research, it has become apparent that increases in temperature will trigger a multitude of climate tipping points [5].

dt	LandAverageTemperature	LandAverageTemperature	LandMaxTemperat	LandMaxTemperat	LandMinTemperatu	LandMinTemperatureUı	LandAndOceanAverageTer	LandAndOceanAverageTemperatureUncertainty
1750-01-01	3.034	3.574						
1750-02-01	3.083	3.702						
1750-03-01	5.626	3.076						
1750-04-01	8.49	2.451						
1750-05-01	11.573	2.072						
1750-06-01	12.937	1.724						

Figure 10: Example of Data Set Structure [26]

The dataset is formatted with there being 9 columns. The first column in the dataset is dt (the date column), and the second column in the data set is *LandAverageTemperature*, which is what we want to predict. *LandAverageTemperature* contains the average temperature over land on the Earth's surface for that given date. There is also a column called *LandAndOceanAverageTemperatures* which contains the average temperature over the land and oceans across the earths surface. The remaining columns contain the minimum and maximum values, and uncertainties. As it is the dataset is mostly ready for processing however, one changed to be made is to convert the format of the dt column to be a numeric value, as this makes it easier for the model to process whilst training.

It is worth noting that at the earlier parts of the dataset some of the cells are missing values, this is not a major issue however, as this is only case near the beginning of the dataset where the data points shouldn't be showing any major trends, and this issue does not exist in the rest of the dataset, which is the vast majority of the data, and all of the most relevant data. Furthermore, there are very few points in the dataset where *LandAverageTemperature* is empty, as such it would be redundant to try to amend the missing data.

4 Project results, evaluation, and testing

In order to measure the effectiveness of a model, firstly, a scatter graph of the average temperatures will be drawn along with a line of best fit. If the gradient of the line of best fit is possible, we can conclude the model is to some extent functioning as predicted. I have based this result on my understanding that non-ML models currently predict some sort of upwards trend [11], and therefore, if our model does not predict an upwards trend, it will be too different to be considered a decent model.

4.1 Testing

For anything that is not the Machine Learning model itself, unit tests will be used to evaluate that they function. This includes things such as the reading the data, pre-processing, and graphing the data. Unit tests are the process of testing each component of the code individual, they are intended to run each time the code itself is run. They are useful as they effective at catching bugs early, thus ensuring that the code functions before moving on to other parts of the project.

4.2 Training LSTM

When compiling the model, the Adam optimiser is what is being used. The purpose of an optimiser is to minimise the models, and in doing so improving the models' performance. The reason for the Adam optimiser being used is as it leads to faster convergence, which is where the weights and parameters of the network have reached a stable state.

To train a model, there needs to be a way to evaluate how well the model fits the dataset. In order to do this, a loss function is used. The loss function being used is the Mean-Squared Error (MSE) function, the formula for which is

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3}$$

where y_i is the i^{th} data point value and $\hat{y}_i$ is its corresponding predicted value, and n is the number of data points. When a model as an MSE of zero there should be no error, and the closer the MSE is to zero the more the model aligns with the dataset.

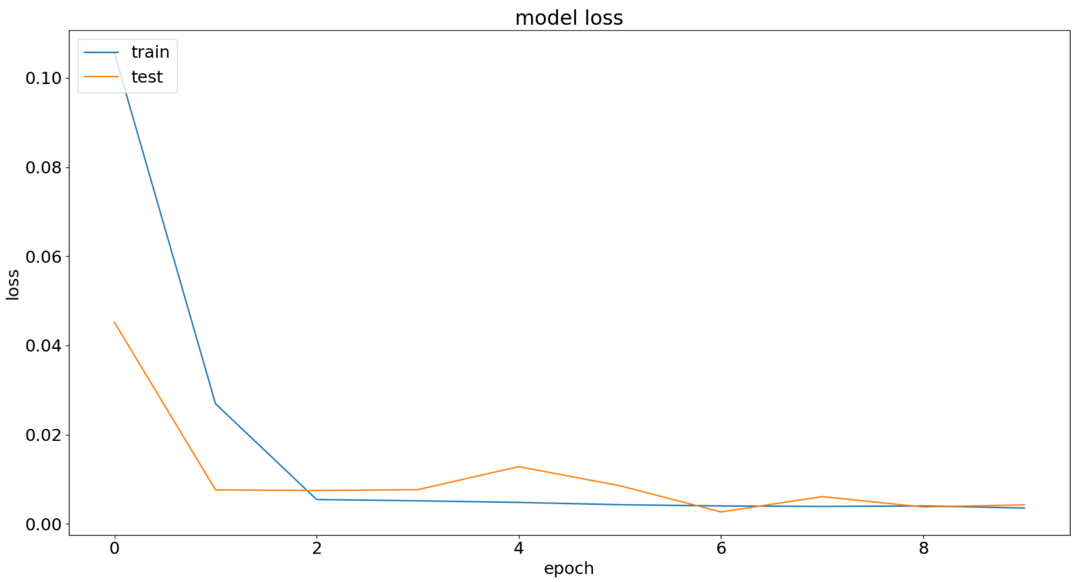

Figure 11: Loss Function Learning History

Figure 11 shows the loss function for the training and test data. From observation of Figure 11 it can be surmised that the model trained quickly, likely in part to the use of the Adam optimiser.

When fitting the model, 10 epochs are being used. An epoch is the term for one complete pass of the training data. Usually a larger number of epochs are used, but as this model fitted reasonably quickly, a smaller number was chosen. Each epoch processes a certain number of samples, and this number is called the batch size. The batch size for this network is 16 as through testing it appears to be a suitable value.

4.2.1 Evaluating LSTM Results

When evaluating the success of the model, multiple baselines should be met. The first baseline is having the data for temperature values predicted to show a positive trend. This is important as, as mentioned previously, current non-ML models predict an upwards trend [11].

Below is the LSTM Results. It is a scatter graph with each point being the average of the next 12 individual predictions the model made, this was done for 100 years worth of iterations.

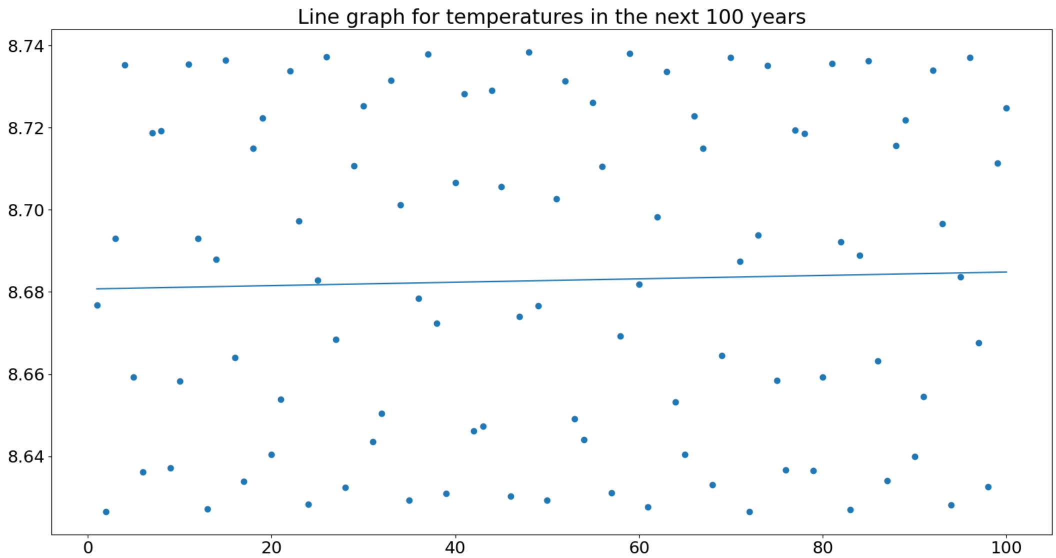

Figure 12: LSTM Results Scatter Graph

As seen in the graph above, the LSTM model was able to identify an upwards trend in the data. This can be identified by the slightly positive line of best fit. From this, we can infer that at least in some sense the model is able to understand the dataset. Another important take from Figure 12 is each different data point has a different value, which is important as it matches what we have observed in natural systems where values for climate variables fluctuate each day. Lastly, we can observe the upwards trend being only slight. The existence of a slight upwards trend aligns with non-ML models [11].

As mentioned previously, it is important the individual values from the model predicts are different, as that is what is observed in natural systems.

Date	Temperature
2016-01-01	2.10712208975851
2016-02-01	2.76488638941943
2015-03-01	4.4751373977065
2015-04-01	7.31620073646306
2015-05-01	10.2188897226452
2015-06-01	12.4602519240379
2015-07-01	13.3313622329235

Table 1: LSTM Results Individual Values Extract

Table 1 shows the first 7 values that model predicts. From Table 1 it can be observed that the model is able to identify that each value is supposed to be unique as each value in table 1 is unique. This is important as it means that the model is showing the ability to predict specific individual values.

To determine whether the results are indicative of those that will trigger a tipping point, we need to see if the average global temperature exceeds certain values. From the data set, we can work out the land average temperature in 2014 to be around 8.33125° and according to NASA, 2014 was 0.8° above the pre-industrial average [28], thus we know that if the land average temperature exceeds 9.03125° then we are in the bounds of the first major tipping point threshold, and if it exceeds 9.53125° we have exceeded the bounds of the first tipping point threshold. Note, it doesn't have to be the average of the years, just the average of a singular year for the tipping points to be met. Upon observation of Figure 12 it can be seen that the model does not predict any values that meet either tipping point threshold within the next 50 years.

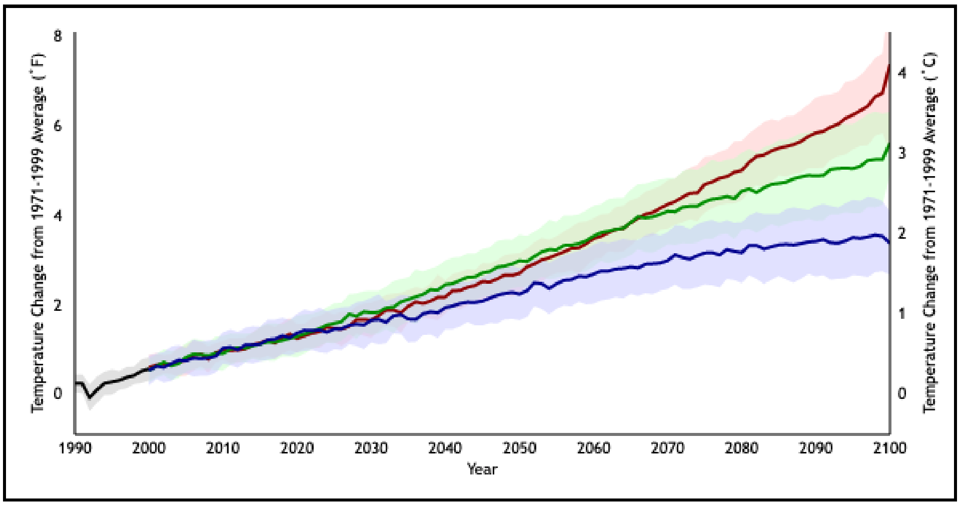

Figure 13: Pre-existing Climate Predictions [29]

Figure 13 is an example of pre-existing predictions for temperature changes in the current century, each of the lines are accounting for different scenarios. It is important to take observe none of the possible predictions shown on Figure 13 align with the prediction of the model. As such it can be reasonable to conclude that the model is not successful in predicting climate tipping points as it disagrees too heavily to pre-existing predictions, that have a scientific base to them.

4.2.2 Reasoning LSTM Results

The model's failings can be attributed to a variety of factors. Here are the factors I believe are the model's result is probably attributed to:

1. Lack of training data
2. Lack of input features
3. Design Flaw With the Project Strategy

To begin with, explaining the lack of training data. There exist two main shortcomings for the dataset being used. The first of which are a lack of data points, this is due to the dataset only having the temperature values for the first day of each month. This means that there are fewer data points than there could have been otherwise, thus resulting in a smaller training set. To offset this issue, we would have had to find more data to fill in the missing data values. However, throughout my searching I have been unable to find a dataset with those missing values. Furthermore, the dataset ends at December 2015, this is unfortunate as firstly it limits the number of data points and secondly, the data from the subsequent years could be of potentially of more importance due to the effects climate change has had over those years [11]. To alleviate this issue, one would need to find data for those years. This was not done within the time constraints, as I did not consider the potential impact that data could have had.

The lack of input features is not a necessary a fault with the dataset being incomplete, but instead a fault of only relying on the data within the dataset. As such, it is possible to find a dataset containing extra input features such as other climate variables besides temperature, and merge that dataset with the one being used, thus providing the model with more data to learn from. The reason this was not done was because I did not foresee the model struggling with predictions, as I had not considered the fact that due to the fluctuating and seasonal nature of the dataset, the model would have been unable to notice the gradual rise of the data.

This project attempted to predict the temperature for each upcoming month. However, due to the nature of how seasons work this proved difficult as it leads to the model being unable to identify the gradual trend in the data. A potential solution is to calculate the average temperature for each year from the dataset and train the model on those averages. This would result in the upwards trend in the data being more noticeable, as a consequence of such the model could have been more likely to identify the trend. The downside of this is that the model would no longer be able to predict the exact months like it can now, however, as the model is not successful in its attempts at predicting months this alternative method is preferable.

4.3 Comparing Model To Other Machine Learning Models

Despite the shortcomings of the model presented in this report, it would be unreasonable to say that the intention of using Machine Learning for the use of predicting climate tipping points is unworkable. For example, T. M. Bury, *et al* [9] were able to detect early warning signal of tipping points in their modelling. The reason for their success was based on how they handled their training in relation to the data, which came from a variety of different climate models. It is worth noting that what T. M. Bury, *et al* [9] were trying to achieve is slightly different to the model in this project. Their model finds signs of tipping points in existing data whilst this model attempts to predict future dates and uses pre-existing knowledge of what will cause tipping points, to give an overall prediction of when tipping points will occur.

Though not working with a climate system, X. Li *et al* [30] had used machine learning to detect tipping points in a complex dynamic system that is similar to that of a climate system. They had opted to use a Reservoir Computer (RC), which is a type of RNN that works differently to LSTM. This proved to be the correct decision, as their model worked very effectively.

5 Project discussion and conclusion

In this project, I describe and evaluate how, in light of the cumulative climate changes in our planet, a Machine Learning approach could help us in our understanding and potential actions for avoiding crossing this threshold. After reviewing pre-existing approaches to Machine Learning, this project was devised in an attempt to predict when tipping points would occur in climate change. Machine Learning was chosen as alternatively to the current climate models, as Machine Learning does not require the understanding and calculations of the many complex formulas and systems that define a climate. Furthermore, such an approach could possibly create a more resource-efficient alternative that could assist climate researchers in their endeavours to find a faster resolution. The model, furthermore, focused on predicting changes in temperature which scientific research inform us as will consequently lead to tipping points triggering in other variables [5]. The method decided upon was to use LSTM Recurrent Neural Network.

Through evaluation of the project's results, I determined that the model was unsuccessful, as it was unable to predict values that met the threshold upon which tipping points are activated. Due to the models' failure, it was necessary to review the possible reasons for the model's failings. The model was then compared to pre-existing Machine Learning models to draw an understanding on how they were able to overcome the pitfalls of this project.

5.1 Future Directions

The future directions of this project can be broken down into two major categories: one dealing with fixing the current project and the other one going past the scope of the project.

5.1.1 Fixing the Current Project

When fixing the current project, I will be first attempting to try to fix it whilst maintaining its ability to predict the values of each individual month. To do this, it would be advisable to increase the number of input features. This can be done by researching for another dataset containing other climate variables, such as sea ice cover or wind speed, and merging that data set with the current one. This will then create columns in the dataset that contain potentially useful data, thus increasing the number of input features. Furthermore, it will be useful to find data values for the times after 2015 as this will increase the amount of training data. Moreover, with the increase of climate change phenomena seen in those years post 2015 [11], this new data should give the model a better indication of the rising temperature levels.

If this does not prove successful, then the project shall be shifted slightly so that instead of predicting each month, it will only predict the yearly averages. This may hinder the potential utility of the model as it would be less precise. However, if it results in a functioning model, then the trade of is worth it. To implement this, an additional step to the pre-processing will be included where in the average temperature values of each year will be calculated. The model will then be trained on these average values, and will predict these average values. This will more likely result in a functioning model.

5.1.2 Going Past the Scope of the Current Project

To go past the scope of this project, the first things that will be done will be to develop the models that were originally written about in the literature review. Those being:

- Generative Adversarial Network (GAN)
- Ordinal Regression (OR) model
- Reservoir Computer (RC)

To begin, the GAN will work in a way which is similar to that of J. Sleeman *et al* [7] where a pre-existing model of tipping points will be used as training data for the discriminator. The generator will then attempt to produce tipping points in the data, and the discriminator will then discern the accuracy of these tipping points, and give feedback to the generator. This process will be repeated until the discriminator is unable to identify a difference between the generated and real tipping points.

The OR model would work in a way similar to that of P. Gutiérrez *et al*'s model [10]. OR, also called ranked learning, is a regression algorithm that is used to predict an ordinal variable, which is a variable that has an inherit order to it whilst the distance between it and other variables is unknown. OR is used to predict whether a number of independent variables can predict an ordinal dependent variable [31], which this model will take advantage of.

The RC model would work similarly to the model developed in this project but using RC instead of LSTM. Reservoir Computing is a type of Recurrent Neural Network, that maps an input signal to a Reservoir. The Reservoir is a non-linear system of higher dimension. It is a black box, meaning operations aren't visible to the user, as there is a more simple solution that is responsible for reading the state of the Reservoir. What would make RC effective for this task, is its effectiveness at handling complex systems whose state evolves with time. [30].

These models should be compared to one another, and to pre-existing models. From this, each of the models' effectiveness could be understood. Unlike the current project, these future steps would provide a variety of different Machine Learning techniques, and therefore, along with the current model, and pre-existing ML models, a rough conclusion on the effectiveness of machine learning for the process of tipping points could be drawn.

6 Bibliography

References

[1] J. C. Comiso, A. C. Bliss, R. Gersten, C. L. Parkinson, and T. Markus, "Current state of sea ice cover." https://earth.gsfc.nasa.gov/cryo/data/current-state-sea-ice-cover, 2024. Last accessed: 24-04-2024.

[2] L. Helsen, "Tackling climate change with machine learning," in *Conference on Neural Information Processing System (NeurIPS)-panelist workshop, Location: online*, 2021.

[3] Intergovernmental Panel on Climate Change (IPCC), "Intergovernmental Panel on Climate Change." https://www.ipcc.ch/. Last accessed: 26-04-2024.

[4] European Space Agency (ESA), "Understanding climate tipping points." https://www.esa.int/Applications/Observing_the_Earth/Space_for_our_climate/Understanding_climate_tipping_points. Last accessed: 26-04-2024.

[5] D. I. Armstrong McKay, A. Staal, J. F. Abrams, R. Winkelmann, B. Sakschewski, S. Loriani, I. Fetzer, S. E. Cornell, J. Rockström, and T. M. Lenton, "Exceeding 1.5 c global warming could trigger multiple climate tipping points," *Science*, vol. 377, no. 6611, p. eabn7950, 2022.

[6] A. A. Sellar, C. G. Jones, J. P. Mulcahy, Y. Tang, A. Yool, A. Wiltshire, F. M. O'Connor, M. Stringer, R. Hill, J. Palmieri, S. Woodward, L. de Mora, T. Kuhlbrodt, S. T. Rumbold, D. I. Kelley, R. Ellis, C. E. Johnson, J. Walton, N. L. Abraham, M. B. Andrews, T. Andrews, A. T. Archibald, S. Berthou, E. Burke, E. Blockley, K. Carslaw, M. Dalvi, J. Edwards, G. A. Folberth, N. Gedney, P. T. Griffiths, A. B. Harper, M. A. Hendry, A. J. Hewitt, B. Johnson, A. Jones, C. D. Jones, J. Keeble, S. Liddicoat, O. Morgenstern, R. J. Parker, V. Predoi, E. Robertson, A. Siahaan, R. S. Smith, R. Swaminathan, M. T. Woodhouse, G. Zeng, and M. Zerroukat, "Ukesm1: Description and evaluation of the u.k. earth system model," *Journal of Advances in Modeling Earth Systems*, vol. 11, no. 12, pp. 4513–4558, 2019.

[7] J. Sleeman, D. Chung, A. Gnanadesikan, J. Brett, Y. Kevrekidis, M. Hughes, T. Haine, M.-A. Pradal, R. Gelderloos, C. Ashcraft, *et al.*, "A generative adversarial network for climate tipping point discovery (tip-gan)," *arXiv preprint arXiv:2302.10274*, 2023.

[8] met office, "New flagship climate models - ukesm1 and hadgem3-gc3.1." "Accessed on 21/11/2023".

[9] T. M. Bury, R. Sujith, I. Pavithran, M. Scheffer, T. M. Lenton, M. Anand, and C. T. Bauch, "Deep learning for early warning signals of tipping points," *Proceedings of the National Academy of Sciences*, vol. 118, no. 39, p. e2106140118, 2021.

[10] P. Gutiérrez, C. Hervás-Martínez, F. Fernández-Navarro, A. Dicaire, M. Pérez-Ortiz, J. Sánchez-Monedero, P. A. Gutiérrez, and L. Summerer, "Climate tipping points: Detection and analysis of patterns using an ordinal regression approach," 2014.

[11] IPCC, "Ipcc sixth assessment report." ""https://www.ipcc.ch/assessment-report/ar6/", Accessed on 24/11/2023".

[12] U. Goverment, "The rights granted by copyright," 2024. https://www.gov.uk/guidance/the-rights-granted-by-copyright", Accessed on 29/04/2024.

[13] A. Krogh, "What are artificial neural networks?," *Nature biotechnology*, vol. 26, no. 2, pp. 195–197, 2008.

[14] S. Sharma, S. Sharma, and A. Athaiya, "Activation functions in neural networks," *Towards Data Sci*, vol. 6, no. 12, pp. 310–316, 2017.

[15] IBM, "Recurrent neural network." https://www.ibm.com/topics/recurrent-neural-networks, Accessed on 07/04/2024.

[16] B. A. C. Maker, "Recurrent neural network." https://botpenguin.com/glossary/recurrent-neural-network, Accessed on 07/04/2024.

[17] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv:1506.00019*, 2015.

[18] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.

[19] O. Surakhi, M. A. Zaidan, P. L. Fung, N. Hossein Motlagh, S. Serhan, M. .Alkhanafseh, R. Ghoniem, and T. Hussein, "Time-lag selection for time-series forecasting using neural network and heuristic algorithm," *Electronics*, vol. 10, 10 2021.

[20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[21] P. Malhotra, L. Vig, G. Shroff, P. Agarwal, *et al.*, "Long short term memory networks for anomaly detection in time series.," in *Esann*, vol. 2015, p. 89, 2015.

[22] A. Gu, C. Gulcehre, T. Paine, M. Hoffman, and R. Pascanu, "Improving the gating mechanism of recurrent neural networks," in *International Conference on Machine Learning*, pp. 3800–3809, PMLR, 2020.

[23] X. Zhu, P. Sobihani, and H. Guo, "Long short-term memory over recursive structures," in *International conference on machine learning*, pp. 1604–1612, PMLR, 2015.

[24] Google, "Normalization." "https://developers.google.com/machine-learning/data-prep/transform/normalization", Accessed on 14/04/2024.

[25] S. Bates, T. Hastie, and R. Tibshirani, "Cross-validation: what does it estimate and how well does it do it?," *Journal of the American Statistical Association*, pp. 1–12, 2023.

[26] "Climate change: Earth surface temperature data." "https://www.kaggle.com/datasets/berkeleyearth/climate-change-earth-surface-temperature-data/", Accessed on 24/11/2023.

[27] Berkley, "Data overview," 2022. "https://berkeleyearth.org/data/", Accessed on 24/11/2023.

[28] NASA, "2014 warmest year in modern record." "https://climate.nasa.gov/news/2221/2014-warmest-year-in-modern-record/", Accessed on 22/04/2024.

[29] NASA, "Climate change: Global temperature projections." "https://www.climate.gov/news-features/understanding-climate/climate-change-global-temperature-projections", Accessed on 23/04/2024.

[30] X. Li, Q. Zhu, C. Zhao, X. Qian, X. Zhang, X. Duan, and W. Lin, "Tipping point detection using reservoir computing," *Research*, vol. 6, p. 0174, 2023.

[31] V. E. Johnson and J. H. Albert, *Ordinal data modeling.* Springer Science & Business Media, 2006.