Cartography M.Sc.

Master thesis

Predicting, understanding, and visualizing fire dynamics with neural networks

Larissa Saad

2021

Predicting, understanding, and visualizing fire dynamics with neural networks

Submitted for the academic degree of Master of Science (M.Sc.)
Conducted at the Institute of Cartography and the Institute of Photogrammetry and Remote Sensing, Department of Geosciences
Technical University of Dresden

Author: Larissa Saad
Study course: Cartography M.Sc.
Supervisor: JProf. Dr. Matthias Forkel (TU Dresden)
Reviewer: Tichaona Tavare Mukunga MSc. (TU Wien)

Chair of the Thesis
Assessment Board: Prof. Dr.-Ing.habil. Dirk Burghardt (TU Dresden)

Date of submission: 03.11.2021

Statement of Authorship

Herewith I declare that I am the sole author of the submitted Master's thesis entitled:

„Predicting, understanding, and visualizing fire dynamics with neural networks“

I have fully referenced the ideas and work of others, whether published or unpublished. Literal or analogous citations are clearly marked as such.

Dresden, 03/11/2021 Larissa Saad

Acknowledgment

I would like to express my gratitude to my supervisor JProf. Dr. Matthias Forkel for his guidance throughout this research, his help, and patience with all the challenges I faced while writing this thesis. Gratitude also goes to my reviewer Tichaona Tavare Mukunga for his continuous help and valuable insights.

I would like to take this opportunity to thank the Cartography MSc consortium and Erasmus + for accepting me into this program and granting me this scholarship which allowed me to fulfill my dream of studying abroad. I heartily thank Juliane Cron for her help through the application process and during the whole program.

My deepest love and gratitude goes to my husband who always gave me strength through the hardest times. To my family in Syria and especially my mother who always encouraged me to follow my dreams. To my brother who put his faith in me and always supported my endeavors. To all my friends here and in Syria who stood by my side and turned the struggles into beautiful memories.

Abstract

As machine learning techniques are contributing to scientific research and advancement, the interpretability and visualization of these algorithms grow in importance. These techniques have introduced many improvements to advance our understanding of fire regime dynamics outperforming process-based approaches. Neural networks have achieved great accuracy with fire modeling, however, challenges arise with unbalanced time series. In this thesis, LSTM neural networks, which are designed for sequence modeling and handling unbalanced data, are investigated to explore their ability to predict fire ignition points. The research is conducted for a small area in western Africa using monthly meteorological variables and fAPAR as an indicator for vegetation for a period spanning from 2003 to 2016. The chosen methodology is based on training one LSTM for each pixel independently. Datasets are pre-processed, structured as a multivariate time series and then arranged to fit LSTM 3D data format. The network architecture was chosen by conducting multiple experiments. The pixel-based LSTM was able to capture the seasonal and spatial varieties with RMSE value computed at 3.333. However, it underestimated the high values of ignitions during the peak of fire season and was not able to record sudden events. To better understand LSTM behavior, multiple interpretation techniques were investigated to evaluate their abilities to determine the most important features and visualize their dependencies. Permutation feature importance gave an overview of overall feature importance while variance-based feature importance was able to map the spatial distribution of each feature. SHAP summary plots gave a detailed interpretation of feature importance of precedent time steps. The most important features to predict fire ignitions were found to be fAPAR, precipitation and maximum temperature. Recent conditions were found more important north of the study area, whereas, in the middle and southern regions, precedent year conditions were of higher importance. SHAP dependence plots were able to depict feature-output relationships. Using these plots, it was observed that LSTM represented the fire-predictor relationship correctly only for a few variables. For feature interactions, a 3D extension of SHAP dependence plot with added color visual variable was found to be the best visualization technique. Visualization of LSTM helped with understanding how the model is learning and which variables were modeled correctly. From here, further improvements could be applied leading to increasing trust in machine learning approaches.

Key words: Machine learning, LSTM, fire ignitions, interpretability, data visualization

Table of Contents

List of Figures

List of Tables

List of Equations

List of abbreviations

ML: Machine Learning

LSTM: Long-short term memory neural network

tmax: Maximum Temperature

tmin: Minimum Temperature

Pre: Precipitation

DTR: Diurnal Temperature Range

NI: Nesterov Index

Wind, Wspeed: Wind Speed

fAPAR: Fraction of Absorbed Photosynthetically Active Radiation

1 Introduction

The advancement of machine learning algorithms in all fields has grown a new branch of scientific research. Now machine learning techniques have become an essential element for solving problems, unraveling hidden patterns, and discovering underlying relationships. The growing volume of geospatial data collected from remote sensing satellites, location detection systems, and social media is presenting formidable challenges to Geographic Information Scientists, Cartographers, and data analysts leading to the investigation of various Artificial Intelligence (AI) and deep learning approaches in an attempt to bridge the gap between GIS, Cartography, and data science (Wilkening, 2019).

Many researchers have started using machine learning techniques to facilitate creating maps by automatically detecting geographical features such as mountains, forests, borderlines, and human settlement patterns from scanned historical maps (Chiang et al., 2020; Schnürer et al., 2021; Uhl et al., 2020), or by automatic extraction of terrain features from digital elevation models (DEMs) (Torres et al., 2020). Other important AI applications in Cartography are the classification of map types (Yang et al., 2020; Zhou et al., 2018), creating maps by automatically classifying remote sensing images (Zou et al., 2015), aligning vector data with geographical features automatically (Duan et al., 2017), and map generalization for vector and raster data (Chen et al., 2020; Yan et al., 2019).

Despite the advances in machine learning approaches, trusting these models has been a point of debate. Deep complex models are considered black boxes which means the algorithms do not provide a clear explanation of why they made a certain decision. Therefore, many researchers have preferred applying traditional simpler models at the expense of accuracy. But since this complexity is what gives the extraordinary predictive abilities for machine learning models, others have developed multiple techniques to interpret these models and visualize what is happening inside (Molnar, 2020). To visualize a machine learning model means to visualize the relationships between each factor in the model and the output prediction using multivariate or multi-dimensional data visualization techniques.

One of the ongoing research questions in the environmental remote sensing field is the suitability of machine learning models or process-oriented models to represent the different relationships between fire and the factors controlling its occurrence. Process-oriented fire models are widely used to predict different aspects of fire regimes (Hantson et al., 2016). However, recent studies have found disagreement among available models when predicting future fire trends (Andela et al., 2017; Forkel et al., 2019b). This is due to inaccurately represented some fire-predictor relationships (Forkel et al., 2019a).

Therefore, ongoing research is headed towards using complex machine learning techniques to better understand the dependencies between fire and its driving factors.

Feed-forward Artificial Neural Networks (ANNs) have displayed great accuracies with predicting burned areas (Joshi et al., 2021; Özbayoğlu et al., 2012). However, new satellite-derived datasets allow to also estimate other attributes of fire regime such as fire occurrence, i.e. the number of ignition points (Andela et al., 2019). Recently, more advanced types of ANNs such as Long Short-Term Memory neural networks (LSTM) have been used to predict and understand the controls on environmental dynamics (Besnard et al., 2019), but it has not been widely used to predict fire dynamics.

In this thesis, remote sensing datasets are used to develop a machine learning model using LSTM neural networks to predict fire occurrence. LSTM promising features in predicting time series data have encouraged this research. Furthermore, this research will focus on the available visualization techniques which can be used to interpret LSTM neural networks and their capability to visualize the relationships between the variables and the model's output.

Following this introduction, this chapter discusses the motivation and reasons behind the choice of this type of neural networks supported by a literature overview of previous research and studies. Chapter 2 describes the methodology used to choose the structure of LSTM and the available techniques to explain and visualize the model's output. Chapter 3 shows the results of predicting fire ignitions with LSTM and evaluates its performance. This chapter also compares the abilities of visualization techniques to display LSTM feature importance and dependencies. Finally, chapter 4 concludes this thesis and gives a summary of future work.

1.1 Fire in the Earth system

Fire is one of the main components of the Earth system. It has been a part of the natural cycle for a long time, dating back to the emergence of terrestrial plants around 420 million years ago (Scott et al., 2006). Wildfire has an important impact on the major global cycles that regulate climate, which includes energy fluxes, hydrologic cycles and biogeochemical cycles (Harrison et al., 2010). Fire can impact the atmospheric chemistry through trace gases and aerosols emissions. The presence of these gases is believed to influence energy fluxes through affecting radiation's scattering and absorption. This also has an influence on cloud cover and albedo, and therefore, precipitation (Lasslop et al., 2019).

As a part of the carbon cycle, natural wildfire causes a sudden release of the carbon dioxide stored in vegetation. Wildfire releases around 2 to 4 Pg (Peta Gram) Carbon per

year in addition to several greenhouse gases (Bowman et al., 2009). Combustion affects the soil properties in terms of nutrients supply, as fire alters the nitrogen and phosphorus cycle. This in turn exerts influence on fuel quantity, and by that, biomass structure and land cover distribution and composition are affected. Fire also impacts deforestation, plant mortality and reproduction. Fire affects human lives either by direct mortality due to fires expansion to urban regions or by influencing air quality. In addition to CO, biomass burning produces toxic matters and pollutants such as benzene which could lead to major health effects (Voulgarakis et al., 2015).

Fire occurrence depends on the existence of three main controls. A minimum temperature which the fuel should reach for the fire to start. Simultaneously, Fuel moisture content should be lower than a certain threshold (Albini, 1976). When the moisture content of the fuel is high, all energy will be utilized to vaporize the moisture, and therefore, ignition would fail (Viegas, 1997). This is related to the general climate temperature and the prolonged dry period without any kind of precipitation. Even when all weather conditions are suitable for combustion, a sufficient amount of fuel on site is crucial to sustaining a fire. In case of non-continuous vegetation, ignitions might happen but the fire would be extinct in a short time (Thonicke et al., 2001).

To represent each factor, numerous variables can be used. Therefore, multiple studies have highlighted the most important drivers using satellite data and machine learning techniques (Aldersley et al., 2011; Archibald et al., 2009; Bistinas et al., 2014; Forkel et al., 2019a). The climate variables which were identified as important fire controls globally are maximum temperature and diurnal temperature range (DTR). DTR is considered as a proxy for vapor pressure deficit which controls the drying rate of dead fuel (Bistinas et al., 2014). Dryness-related variables such as precipitation and the number of wet days per month were highly important in tropical forest regions (Forkel et al., 2019a). Antecedent dry-day period was also found to significantly influence fire occurrence and burned area (Aldersley et al., 2011; Kuhn-Régnier et al., 2020).

For fuel presence and productivity, the most important variables are vegetation type, fuel litter and accumulation (Forkel et al., 2019a). According to Kuhn-Régnier et al. (2020), shorter timescale conditions are more important in the tropics, and among fuel-related vegetation predictors, the Fraction of Absorbed Photosynthetically Active Radiation (fAPAR) appeared to be the most important predictor. fAPAR is considered as a measure of the solar radiation absorbed by live leaves for the photosynthesis activity and can be used as an indicator of vegetation cover. fAPAR ranges between 0 and 1, where a value of zero indicates no flammable vegetation (Knorr et al., 2014).

To better understand the dynamics between all the elements several models have been built to simulate these connections and discover the underlying relationships among them (Hantson et al., 2016). Dynamic Global Vegetation Models (DGVMs) and Earth system

models try to combine all biogeochemical cycles and disturbances, such as wildfire. The comprehension of these dynamics is essential to understand the local and global changes in natural cycles especially due to human interferences.

1.2 Predicting fire with process-oriented models

Prediction of wildfire is extremely difficult due to the complexity of factors controlling the occurrence and spread of fire. Those physical factors are now well known after years of descriptive research, however, configuring the exact physical relationships between fire and its predictors is not yet fully comprehended. As fire is a product of not only direct relationships with multiple environmental and anthropogenic variables, but also with their mutual influence at an exact point in time. Several models have been developed to study the relative importance of each factor to different aspects of wildfire and to predict its occurrence, risk and danger (Chuvieco et al., 2010).

The most ubiquitous models are fire models coupled with dynamic global vegetation models (DGVMs) or terrestrial ecosystem models (TEMs). The complexity of these models varies from simple empirical models (Reick et al., 2013; Thonicke et al., 2001) which were coupled with different DGVMs (Levis et al., 2004; Pechony et al., 2009; Sitch et al., 2003), to process-based models of medium or high complexity (Arora et al., 2005; Lehsten et al., 2010; Li et al., 2012, 2013; Melton et al., 2016; Pfeiffer et al., 2013; Thonicke et al., 2010; Venevsky et al., 2002; Yue et al., 2014). The process-oriented models try to describe the environmental processes using a set of equations derived from physical relationships. These models evolved over time and represented more complex environmental and anthropogenic variables and now they are able to predict all aspects of fire regimes such as burned areas, fire occurrence, fire size, spread and speed (Hantson et al., 2016).

Process-oriented fire models simulate the predictor-response relationships in different ways. Hence, current fire models show different results in prediction of future trends. Whereas satellite-derived datasets show a declining trend in burned areas globally (Andela et al., 2017; Forkel et al., 2019b), current fire-enabled dynamic global vegetation models (DGVMs) do not produce this apparent decline, while some models underestimate it, others show an increase in global burned areas. Using identical forcing datasets to compare these models behaviour when predicting future scenarios (Rabin et al., 2017) showed the ability of fire models to simulate burned areas spatial pattern, however, the size of the total burned areas differ significantly (Hantson et al., 2020). This means that the relationships between fire and its driving factors are not yet understood and correctly represented. Forkel et al. (2019a) found that DGVMs are able to reproduce the sensitivities between burned areas and climate variables, however, they underestimate the relationship with socio-economics drivers and do not simulate vegetation distribution and fuel correctly.

Driven from the ideas that process-based fire models are still poorly representing fire behaviour, trend and extreme events under changing conditions of the climate (Sanderson et al., 2020), ongoing research is focusing more on using data-driven empirical models especially machine learning models from simple models to a combination of deep Artificial Neural Networks (ANNs). The field of artificial intelligence has presented many solutions to complex problems as these algorithms account for non-linearities which could better describe the relationships between fire and its drivers and help to understand the complexity of the conditions of fire occurrence and spreading.

1.3 Predicting fire with machine learning models

The increase of data availability due to remote sensing techniques have allowed the development of environmental data-driven approaches which require significant amounts of data and long time series to learn and produce results. A variety of machine learning (ML) models have been used in literature. Random Forests (RFs) are the most used technique to study the fire-driver relationships and further explore which factors are more important to fire regimes on a sub-continent scale (Archibald et al., 2009; Kim et al., 2019) and global scale (Aldersley et al., 2011; Forkel et al., 2019a). RFs account for non-linearity and this gives them an advantage over traditional regression models which presume the fire-predictor relationship to be linear.

Comparative studies have been conducted to establish the most accurate technique to predict wildfire. Several researchers have compared the traditional regression models with higher order ML techniques such as neural networks and random forests (Guo et al., 2016; Jafari Goldarag et al., 2016). In these studies, regression models failed to capture the patterns of fire probabilities. Another study has compared five data mining algorithms including Multiple Regression model, Support Vector Machines (SVMs), Random Forests and a neural network to predict forest fire using only meteorological data. In this study, SVMs performed the best and were capable of predicting small fires, however, for large fires, they had lower accuracy (Cortez et al., 2007). In another research, Song et al. (2020) applied a linear model, a regression tree and a neural network to forecast monthly wildfire predictions on a global scale. The neural network outperformed the latter techniques when using the same predictor variables.

Different types of Artificial Neural Networks have been also used in several research to predict wildfires. On a regional scale, Maeda et al. (2009) used a feed forward neural network with different architectures and a backpropagation algorithm to predict forest fires in the Brazilian Amazons. In this study, a simple network with one hidden layer and four neurons achieved satisfactory results with fire risk spatial distribution areas consistent with fire season observations. Multiple studies have also obtained high accuracy mapping forest fire probability and burned forest areas using feed forward multilayer perceptron (MLP) networks (Özbayoğlu et al., 2012; Satir et al., 2016). On a global scale, Joshi et al. (2021) presented a global model to predict burned areas using a multilayer feed forward

neural network. In this study, the dataset was split into geographical regions that share common features (e.g., same vegetation structure, level of human influence). Then, the model was trained to detect fire drivers in each region. The model achieved high levels of accuracy with global spatial correlation of 0.92. On the other hand, the model failed to distinguish fire extremes at an annual regional scale and was able to capture only 23% of the observed global decline in burned areas.

Neural networks have achieved great accuracy with modelling and predicting spatial patterns of different aspects of fire regimes. However, challenges arise when modelling a time series with extreme or sudden events. For this purpose, more advanced types of NNs could perform better and might be able to model more complex relationships.

1.4 Long-short term memory (LSTM) neural networks

Long-short term memory neural networks (LSTM) are a special type of Recurrent Neural Networks (RNNs). The basic difference between RNNs and feed-forward neural networks is the presence of feedback loops (Figure 1). This means, in RNNs, each hidden neuron takes current input from the previous layer and also what it has learned from the prior inputs. This allows RNNs to use previous knowledge when making future events predictions. The concept of the loop can be understood by unrolling the RNN. The loop represents a chain or sequence of copies of the same network, and this makes RNNs more appropriate for sequence modelling such as speech recognition, handwriting detection, sentiment analysis and time series forecasting. The main pitfall of RNNs is the vanishing gradient problem when dealing with long sequences (Goyal et al., 2015, pp. 129–134). To solve this problem, Hochreiter et al. (1997) proposed to use a new memory cell structure with input, output and forget gates that allow better control over which information to preserve and which to forget. The structure of the LSTM unit can be seen in Figure 2. LSTM allows RNNs to remember their input over a long period of time.

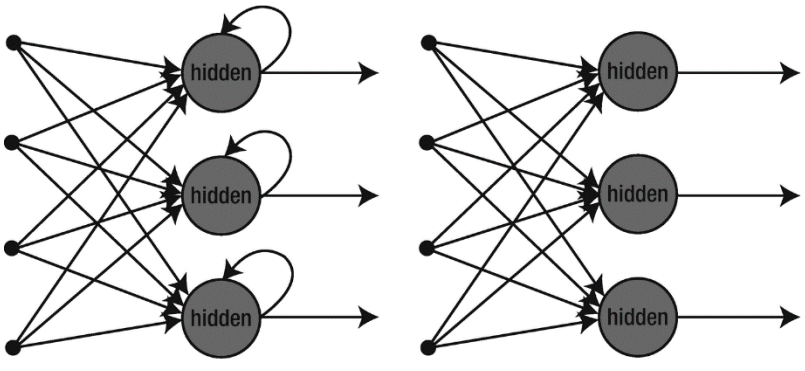

Figure 1: Architectural difference between feed-forward neural networks (b) and recurrent neural networks (a) (Goyal et al., 2015, pp. 122–123)

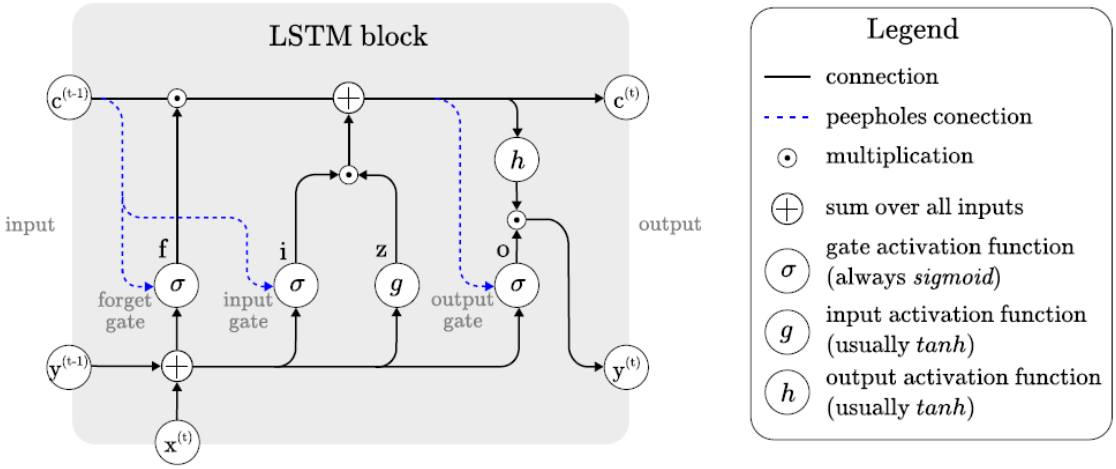

Figure 2: LSTM cell structure (Van Houdt et al., 2020)

Multi-layer Perceptrons (MLPs) can be applied to sequence prediction problems, however, they have certain limitations (Brownlee, 2020, pp. 7–9) that can be compensated by using RNNs. LSTMs have proven to solve many sequence prediction problems that cannot be solved by MLPs (Gers et al., 2001). Time series forecasting is a challenging problem for most algorithms given the fact that here the temporal structure and the order of steps should be preserved. Multiple techniques have been used to study time series processes. The Autoregressive Integrated Moving Average (ARIMA) model is often used, however, this technique tends to center past events around the mean which prevents the prediction of extreme values (Hua et al., 2019). Furthermore, Support Vector Machines (SVMs) have been extensively studied for time series analysis but it is presented with multiple technical challenges (Sapankevych et al., 2009).

LSTM has been used in different fields to handle imbalanced or noisy time series issues (Giles et al., 2001; Han et al., 2004) and has given promising results. It has also been employed for anomaly detection (Taylor et al., 2016) giving its ability to preserve long term dependencies and remembering past events.

In the field of fire prediction, LSTM has not been widely used. Liang et al., (2019) employed three types of neural networks including LSTMs to estimate wildfire scale using only meteorological predictors. LSTM gave the highest accuracy with 90.9%. Perumal et al. (2020) compared two types of RNNs and their ability to model the duration and direction of wildfire. LSTM and Gated Recurrent Unit (GRU) which is a variant of LSTM that is able to capture dependencies of different time scales were tested. In this study, GRU performed better than LSTM for longer time series. Kong et al. (2018) used MODIS data to calculate the global environmental monitoring index (GEMI) for each pixel. Each pixel then had one time series and one LSTM was trained for each one independently. Afterwards, the prediction was depicted in one burned area map. The proposed approach gave effective and stable results for online disturbance detection.

LSTMs have shown impressive results in predicting time series data and extreme events. Therefore, in this thesis, LSTMs are used to predict fire occurrence with different frequencies over a period of time to explore their predictive ability and study their efficiency to correctly capture the relationships and sensitivities between fire and driving factors.

1.5 Visualization of machine learning and neural network models

Trusting machine learning models has always been a point of debate in scientific fields. Given the fact that most of these algorithms do not explain how they made their decisions or explicitly demonstrate the functions they concluded to represent relationships and since these functions are completely driven from the datasets provided, arguments have been made that using different data might lead to different results because they are not built on clear physical relations. However, the achievements and accuracy of these models have proven that they are learning in a correct manner, and this encouraged researchers to use them in multiple fields including Cartography especially with the widespread use of geo-spatial data.

To help with understanding what is happening inside these black boxes, recent developments have succeeded in making these models interpretable by using data visualization techniques to facilitate human interpretation and give confidence when adopting a data-driven approach. To visualize a machine learning model means to visualize the relationships between each factor in the model and the output prediction in an n-dimensional space. Visualization of these dependencies helps with understanding how the model is working, why the model is making this decision, and if the model is working correctly or failing. The complexity of decomposing these relationships depends on the model type and its ability for interpretation.

Miller (2019) gave a non-mathematical definition of interpretability 'Interpretability is the degree to which a human can understand the cause of a decision'. Therefore, if a model is of high interpretability, this means that predictions or decisions made by the model and the reasons behind them are easy for humans to understand (Molnar, 2020). In this context, we can distinguish basically between two types of models, Interpretable Models, and black-box models. Interpretable Models, such as linear regression, logistic regression, Naive Bayes and decision trees, provide an understandable way of how the algorithm created the model. With these models, we can understand how the trained model makes predictions, how the model coefficients or weights affect the predictions and why the model predicted a certain value or class for one instance and for a group of instances (Molnar, 2020).

Blackbox models, such as neural networks, are more difficult to explain. One prediction in a neural network might go through millions of mathematical operations and this makes it impossible for humans to simply interpret the behavior. Therefore, multiple Model-Agnostic Methods were developed to separate the explanation from the machine learning model. These techniques can be applied to any model in theory but some technical limitations might prevent this. The nature of LSTM data format limits the application of many interpretation techniques. Therefore, in this thesis, available approaches are studied and compared to investigate which of these methods can be applied to interpret and visualize LSTM neural networks in an n-dimensional space.

1.6 Research objectives and questions

The research will be divided into two main objectives and derived sub-objectives:

a- The first objective of this research is to predict wildfire occurrence using remote sensing data by applying Long Short-Term Memory (LSTM) neural networks and investigate its potential to understand the controls on fire dynamics.

Recently, deep learning methods such as long short-term memory networks (LSTM) have been used to predict and understand the controls on environmental dynamics. However, this kind of neural networks have not been used to predict fire dynamics. Therefore, we aim at:

1. Explore LSTM predictive ability of fire ignition points
2. Detect LSTM ability to correctly capture the relationships between fire and driving factors

b- The second objective is to compare available methods' abilities to visualize and interpret LSTM neural networks.

Visualizing the machine learning model's dependencies in a correct way is essential for better interpretation and understanding of results. Therefore, this research will focus on:

1. Investigate current interpretation techniques and their ability to characterize global and local feature importance, the spatial distribution of feature importance and predictor-response relationships and interactions.
2. Analyzing the most effective visualization techniques for this type of models

Therefore, this research will try to answer the following questions:

Q1: What are the opportunities and limitations of using LSTM neural networks to predict fire occurrence?

Q2: What is the ability of LSTM to record the relationships of fire drivers?

Q3: What is the best available method to interpret and visualize LSTM neural networks in an efficient and understandable way?

This research is intended to contribute to the Remote Sensing field, especially, environmental research which uses satellite data to observe, analyze, model, and predict changes in ecosystems. The results of this thesis will provide useful insights for wildfire modelling about the potential and predictive power of LSTM neural networks. Furthermore, this study will be beneficial to the area of data visualization, specifically cartographers, statisticians, and data scientists working with multivariate visualization techniques and machine learning.

2 Methodology

This chapter explains the steps taken to design the neural network structure starting from available datasets, required data pre-processing techniques, and choosing the network's hyper-parameters based on experimentation. To investigate if the network is representing the correct dependencies, multiple visualization techniques are studied to assess which available approach works best with this type of neural networks.

2.1 Fire predictors and Datasets

The main variables of fire occurrence can be generally summarized as the presence of an ignition source, availability of fuel and weather conditions. In this research, the meteorological variables considered are 2-metre maximum and minimum temperature, total precipitation, and wind speed. All meteorological variables were obtained from the CRU JRA V2.0 dataset (Harris, 2019). The units of measurements are for temperature variables in Kelvins, for precipitation in kg/m^2, and for wind speed in m/s. All variables are provided on a 0.5 deg latitude x 0.5 deg longitude grid. DTR was calculated based on daily maximum and minimum temperature obtained from the same dataset.

To account for the antecedent dry-day period, many fire danger indices were used in literature. In this thesis, the Nesterov Index (Nesterov, 1949) is used. It is a simple daily fire danger rating index that requires daily air temperature, dew point temperature and precipitation as input data. This index accumulates weather-related conditions to measure the period of consecutive days without precipitation. When the daily precipitation exceeds 3mm, it is then set to zero. The Nesterov index is calculated as follows:

$$NI = \sum_{i=1}^{w} T_i(T_i - D_i)$$

Equation 1: The Nesterov Index

Where: W = number of days since last rainfall > 3 mm, T = midday temperature (°C), D = dew point temperature (°C)

The Nesterov Index was also calculated based on meteorological data obtained from CRU JRA V2.0. For fuel presence and accumulation, fAPAR variable was obtained from the MOD15A2H dataset (Myneni et al., 2015), with spatial resolution of 0.25 deg latitude x 0.25 deg longitude grid.

To train and validate the neural network, the Global Fire Atlas dataset (Andela et al., 2019) is used. The dataset contains monthly data of ignition count estimated by Number/ month. The algorithm for fire ignition detection is based on MODIS datasets of moderate resolution, this means that the smallest fire detected is one MODIS pixel (approximately 21 ha) (Andela et al., 2019). The monthly fire atlas data has a spatial resolution of 0.25 deg latitude x 0.25 deg longitude grid and spans over the years 2003-2016.

A list of all the variables used can be found in (Table 1). All variables are given in temporal resolution of one month and covers the period from January 2003 till December 2016. For other important variables such as plant functional type, biomass and fuel litter, only static maps were available. The use of such types of variables could cause bias in the LSTM neural network, therefore, these factors were excluded. An ignition source is considered to be always available, whereas the concentration will be on studying the surrounding conditions.

Table 1: Overview of the variables and available datasets

Variable	Description	Data source	Spatial Resolution	Temporal Resolution
Predictors				
Tmax	Mean of monthly maximum temperature	CRU JRA v2.0 (Harris, 2019)	0.5x0.5	Monthly
Tmin	Mean of monthly minimum temperature	CRU JRA v2.0 (Harris, 2019)	0.5x0.5	Monthly
Pre	Total precipitation	CRU JRA v2.0 (Harris, 2019)	0.5x0.5	Monthly
NI	Nesterov Index	CRU JRA v2.0 (Harris, 2019)	0.5x0.5	Monthly
Wind	Wind Speed	CRU JRA v2.0 (Harris, 2019)	0.5x0.5	Monthly
fAPAR	Fraction of Absorbed Photosynthetically Active Radiation	MODIS: MOD15A2H (Myneni et al., 2015)	0.25x0.25	Monthly
Target variables				
Fire ignitions	Count of ignition points per cell	Fire Atlas (Andela et al., 2019)	0.25x0.25	Monthly

2.2 Study area

For this thesis, a region in the African continent spanning from 4° to 16° North of the Equator and 18° West to 4° East was chosen (Figure 3). This area includes two main climate zones, a Tropical savanna climate zone that corresponds to the Köppen climate classification categories 'Aw' (for a dry winter), and a Semi-Arid climate zone in the north where it is closer to the Sahara desert (Transition between humid climate and desert climate) (W. Köppen, 1936). In the southern parts of the region humid forests are the dominant vegetation. In tropical savanna climate, the vegetation is generally characterized by tree-studded grasslands and the tall, coarse grass called savanna. In semi-arid regions, short vegetation like grass or shrubs are usually found (Figure 4). Fire occurrence is frequent in this region. Fire season usually extends from September till June every year as it corresponds with the dry season.

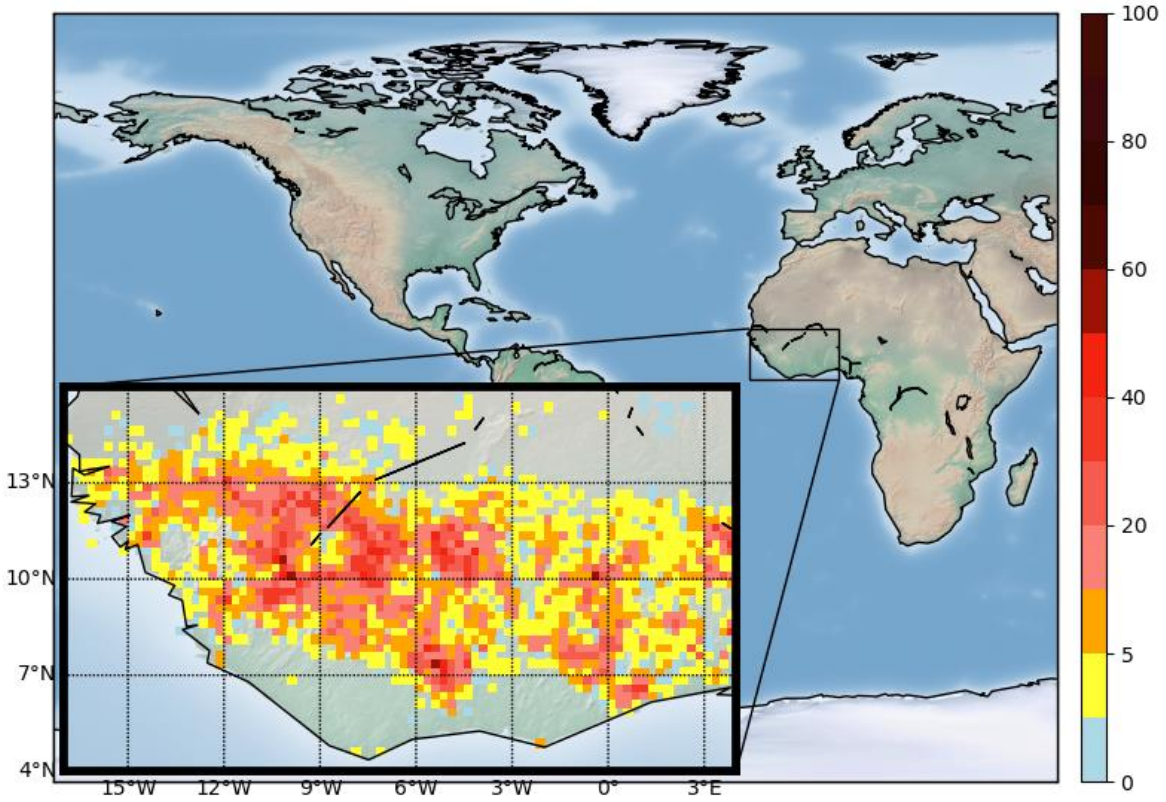

Figure 3: Study Area with a sample of ignition count for one month

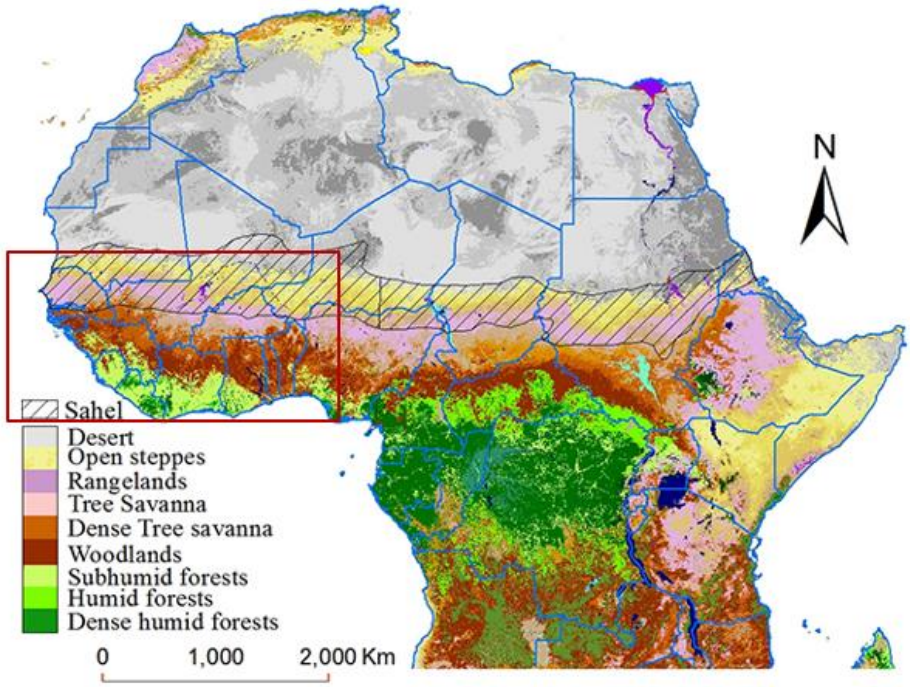

Figure 4: Land cover map in the Sahel zone. The red rectangle was added to indicate the location of the study area in this thesis. Source: From GLC 2000, EU-JRC data (Mbow, 2017)

2.3 Data pre-processing

When building a machine learning model, data preprocessing is the first step to clean, format, and organize raw data to make it suitable for the neural network. The steps here involve homogenizing the datasets to similar spatial and temporal resolution, data integration, data cleaning to handle missing values, data reduction to eliminate correlated variables and data transformation to transform the data to an appropriate form. These steps are highly important to avoid misleading results.

2.3.1 Data aggregation and resampling

The original datasets of Tmax, Tmin, Wind and NI cover the period from 1901 to 2018. As a first step, the datasets were sliced to match the study period. All datasets are available in NetCDF file format. As seen in Table 1, the target variable dataset is on a different grid size from the predictor variables except for fAPAR. This makes it difficult to work out the differences or combine the datasets together. Therefore, it is necessary to remap the coarser grid into a finer one to match the target dataset. Since the resolution of the target grid is exactly half the resolution of the variable grid, this means that each cell value will

be interpolated to the neighboring four cells. For this, the bilinear interpolation remapping method using Climate Data Operators (CDO) (Kaspar et al., 2010) is used. All the variables were remapped to 0.25 deg latitude x 0.25 deg longitude grid.

The diurnal temperature range (DTR) is the difference between the daily maximum and minimum temperature. DTR was calculated for the daily data and then aggregated to monthly time steps to match the target dataset. Afterward, DTR was also remapped to the new grid using bilinear interpolation.

After matching all the variables to the same spatial and temporal resolution, all files were merged into one NetCDF file which contained all 7 predictor variables and the target one. Furthermore, a mask was applied to exclude all ocean pixels to reduce the size of the dataset.

2.3.2 Missing values

The presence of missing values in a dataset could cause serious problems for data analysis and the neural network. Inappropriate handling of missing values could cause bias and misleading results (García et al., 2015). Therefore, it is important to understand the dataset in order to handle the missing values in the most efficient way. First, the missing values were calculated in all variables over the whole study area (Table 2) after excluding the ocean region. We can observe that the missing values are only present in the ignitions and fAPAR features.

One of the most common methods in dealing with missing values is simply dropping them. However, in this case this is not possible. Firstly, each pixel represents a time series and dropping these values would ruin the continuity of the data, and secondly, the limited size of the dataset. Nonetheless, after further studying the data, it is safe to fill the missing values with zeros. The missing data represent no fire occurrence in the ignitions variable, and since a part of the study region is a desert, this implies no plants exist in these areas, therefore, no fAPAR.

Table 2: Exploring the dataset to allocate the missing values

Variable	Count of Missing Values
DTR	0
fAPAR	524669
Ignitions	2025479
NI	0
Pre	0
Tmax	0
Tmin	0
Wind	0

2.3.3 Multicollinearity test

Multicollinearity is the occurrence of high intercorrelations among two or more independent variables in a multiple regression model. Multicollinearity is different than correlation. Whereas correlation is a linear relationship between two variables, multicollinearity can happen between two variables or between one variable and a linear combination of others (Alin, 2010). Multicollinearity makes the regression estimate highly unstable. This instability will increase the variance of estimates and make them unreliable (Donald et al., 1967). If the estimates are not reliable, the prediction accuracy for the model will not be trustworthy and this can lead to skewed or misleading results when we try to understand the importance of each predictor variable to predict a target variable (Shrestha, 2020).

In literature, there are different techniques to detect multicollinearity (Daoud, 2018; Shrestha, 2020). The Variance Inflation Factor (VIF) assesses how much the variance of the estimated regression coefficient increases if the predictors are correlated. The idea of variance inflation is that first, we run an auxiliary linear regression of one of the independent variables on all the other independent variables to get a value of R-squared. Here, R^2 essentially tells us how well the regressed variable describes the movements in the other variables. High values of R^2 mean that the variable is multicollinear with linear combinations of the other variables. VIF is calculated using Equation 2:

$$VIF = \frac{1}{1 - R_i^2} \quad for\ i = 1,2, \dots, k$$

Equation 2: Variance Inflation Factor (VIF)

Where: R_i^2 is the coefficient of multiple determination of x_i on the remaining variables (Shrestha, 2020). The higher values of R^2 give smaller values of the denominator, therefore, higher value of VIF.

Table 3 shows the results of calculating VIF values for the entire dataset. We can see that the variables with the highest values of VIF are tmax, tmin, and DTR. The lowest value of VIF is 1 which indicates no correlation, on the contrary, VIF values have no upper limit. If VIF < 10, there's a moderate multicollinearity among the variables, however, it's preferable for the VIF to be lower than 5. To treat multicollinearity, the correlated variables are removed one at a time and VIFs are then recalculated. The variables are dropped according to their importance. Therefore, tmin was excluded first and VIF values were recalculated (Table 3). It is observable now that all VIF values are low and it is safe to say that there is no multicollinearity among the predictors.

Table 3: The variance inflation factors (VIFs) of the variables before and after dropping one of the correlated predictors (tmin)

Variable	VIF	VIF after dropping tmin
Wind	1.96	1.96
DTR	538577.07	1.66
fPAR	2.91	2.91
NI	1.31	1.31
Pre	2.09	2.09
tmax	2257348.76	1.49
tmin	2229555.77	

2.3.4 Data transformation

In general, linear regression models like simple linear regression or logistic regression expect the outcome variable to be normally distributed but they do not make assumptions about the distribution of the predictor variables. Non-linear regression models do not have this assumption, yet some studies have taken interest in the prediction accuracy of artificial neural networks (ANN) when the outcome variable is highly skewed. Larasati et

al., (2019) have found no significant decrease in the ANN accuracy when dealing with skewed data. On the other hand, studies have found that transforming severely skewed variables to a roughly normal distribution often results in a better performance (Kubben et al., 2020, pp. 79–81; U. A. Kumar, 2005).

Therefore, a sample of the data was taken i.e., a random pixel, and the distribution of data was visualized (Figure 5). It is visually observed that the target variable (ignitions) is extremely positively skewed. This also applies for NI and pre. This is detected in the whole dataset after running many samples. To handle the problem of skewed data, two main approaches can be followed. The first one is looking for appropriate data processing techniques and the other technique is finding an appropriate model approach. In this section, several data transformation techniques will be applied to mitigate the data imbalance as possible. The focus will be on the target variable i.e., ignitions.

Several of the most ubiquitous data transformation techniques were applied to all variables, for example, log transformation, square root transformation, box-cox transformation (Box et al., 1964), and yeo-Johnson transformation (YEO et al., 2000) (Figure 6). Prior to applying the transformations, a small value of one was added to the variable to avoid the logarithm of zero and dividing by zero in the cox-box technique as it applies the reciprocal transformation in some cases. It is clearly observed that the transformation techniques did not convert the data distribution to Gaussian, nonetheless, the box-cox and yeo-johnson transformations slightly mitigated the skewness of data. The reasons could be that the data size is small, in addition to being severely skewed with inflated zeros. Furthermore, multiple combinations of techniques were applied to test their effects, however, the final product did not improve significantly and in some cases performed worse.

The effects of these transformations on the neural network results are yet to be tested in the following sections. Therefore, no preferable transformation technique is selected at this point.

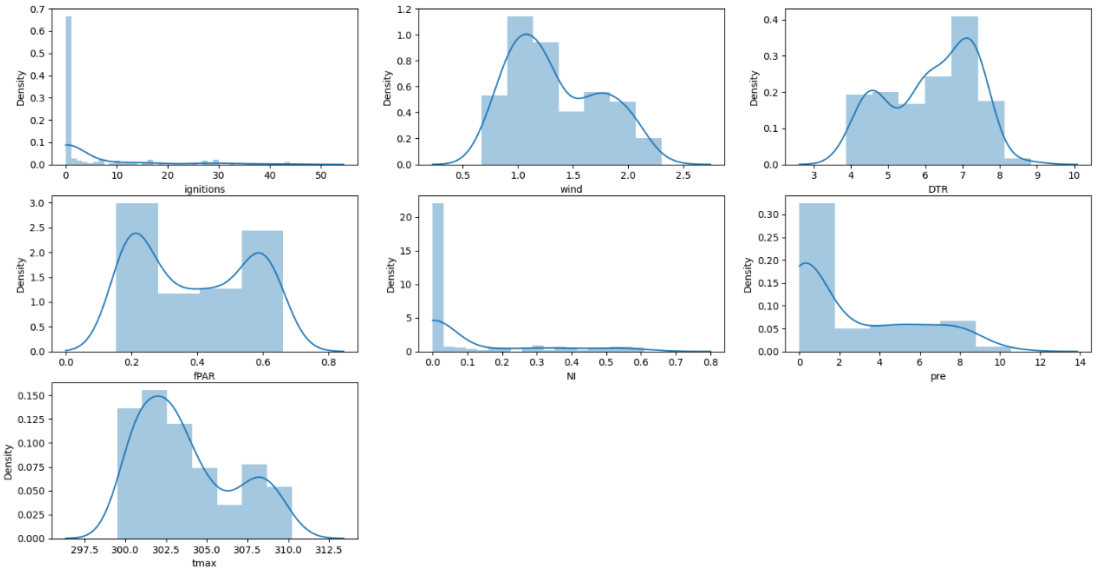

Figure 5: Histogram plots with kernel density for all variables of a random pixel in the dataset

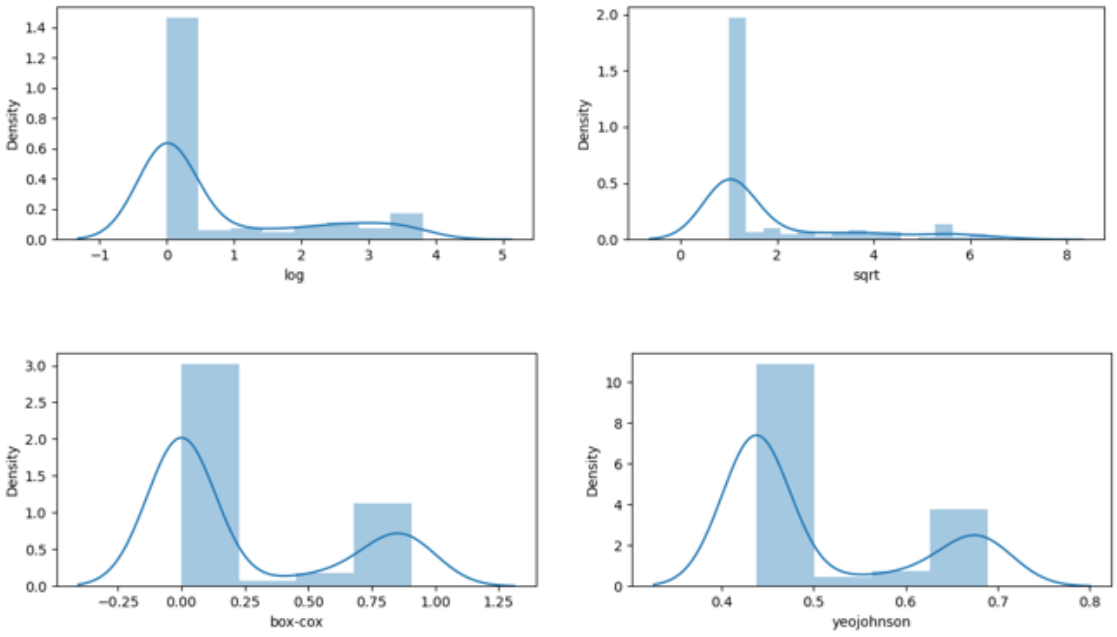

Figure 6: The effect of applying multiple transformation technique on the highly skewed output variable (ignitions count)

2.3.5 Feature scaling

When the dataset has multiple variables with different units of measurement, feature scaling facilitates direct comparison between variables. Machine learning techniques basically see only numbers. When we have large numbers in one feature and really small numbers in another, the algorithm makes an assumption that higher numbers get higher priority and this increases the difficulty of the problem being modeled and causes bias in the output.

The choice of scaling technique depends on the data distribution. Data normalization using the Min-Max scaler which scales the data within the chosen range [0,1] was applied considering that all the variables in our dataset (inputs and output) are not normally distributed. The Min-Max scaler is calculated as in Equation 3:

$$x_{new} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Equation 3: Min-Max Scaler

2.4 LSTM architectures and experiments

As explained in 1.4, LSTM is designed for sequence modeling and prediction. This means that the plain model of LSTM does not consider 2D data and therefore any spatial information will be lost (Van Houdt et al., 2020). However, due to the powerful abilities of LSTM to predict and capture long-term dependencies, several researchers have used either pure LSTM models (Arslan et al., 2019; Kong et al., 2018) or developed new models to extend the application of LSTM to handle spatio-temporal data.

The inclusion of spatial information has been done in various techniques. One of the methods is using a graph network to capture the spatial connectedness before passing the results to another neural network like LSTM (Khodayar et al., 2019; Perumal et al., 2020). Other researchers have worked on the improvement of the LSTM cell design by adding an additional cell that memorizes the spatial information (Wang et al., 2017, 2019). Moreover, the most ubiquitous method used among researchers is embedding a convolutional neural network that reads the spatial information and then passes them to an LSTM network as a sequence (A. Kumar et al., 2020; Moskolaï et al., 2020). However, these approaches were developed to work with a univariate sequence, where a time series of the same variable is fed to the network to learn from previous observations and then predict the next value. In this thesis, the aim is to predict one independent variable based on other inputs

i.e., multivariate time series. In literature, there are few available models to predict multivariate time series that consider the spatio-temporal dependencies, and this is done also by embedding different types of neural networks as a front-end of the model to read the spatial information (Gou et al., 2020; Xiao et al., 2021). Nonetheless, in this thesis, we would like to evaluate the capacity of pure LSTM to predict a multivariate time series when dealing with highly irregular data such as fire ignitions.

One approach to achieve this is by using pixel-based LSTM, where each pixel is treated as an independent multivariate time series. Using this method, we would like to better understand how LSTM will handle different frequencies of fire occurrence and the sudden changes in ignitions count along subsequent months.

2.4.1 Preparing data for LSTM

To understand how to structure the data in the correct form for LSTM, the model type needs to be determined first (Figure 7). This is basically set based on the number of input sequences and the number of steps we would like the network to predict. As mentioned before, the dataset is one NetCDF file with one outcome variable and six predictors. The format of the data for one example pixel is shown in Table 4. Each pixel is a multivariate time series with 168 time steps where the six predictors are used to forecast the next month's ignition count. This problem can be structured as a Many-to-one sequence model in which sequences of n vectors of input features are processed and then the output is produced only after the whole sequence of feature vectors has passed through. This can be formatted as follows:

$$Y(t) = f(X1(t_{1,2,..,t-1}), X2(t_{1,2,..,t-1}), \dots, Xn(t_{1,2,..,t-1}))$$

Equation 4: Many-to-one sequence representation

Where Y(t) is the outcome variable, and (X_1, X_2, ...) are the predictors for previous n time steps ($t_{1,2,...}$)

Table 4: Format of the data

longitude	Latitude	index	time	Ignition count	DTR	fPAR	pre	tmax	wind	NI
-12.597	8.858	0	2003-01-01	3.5000	5.43753	0.53677	0.05428	304.15045	1.17915	0.0
		1	2003-02-01	1.3333	5.50904	0.51708	0.70357	304.52508	2.20828	0.0
		2	2003-03-01	8.6666	4.82853	0.52482	0.47201	304.93811	2.12751	0.0
.....										
		166	2016-11-01	0.0000	3.70297	0.69636	2.75937	302.67605	0.73359	0.0
		167	2016-12-01	2.0000	4.07177	0.63130	0.53639	302.95983	0.82405	0.0

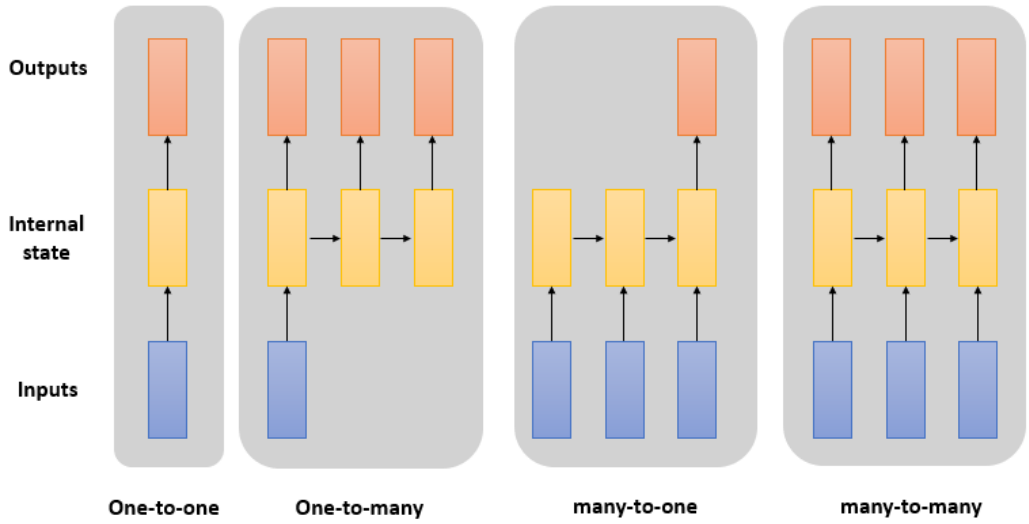

Figure 7: Different model types of LSTM. Each rectangle represents a vector and the arrows represent functions.

Regarding data transformation (2.3.4), several experiments have been conducted to determine the most suitable technique. Both the Box-cox and Yeo-Johnson transformation did not show any improvement in performance, furthermore, when the neural network overestimates some predictions and these values are out of the transformation range, the results are being returned as not a number (NAN) and causing problems with the network. On the other hand, using the log transformation, even though it did not have a major effect on the variable distribution, did improve the output results.

Following this, the data should be framed as a supervised learning problem. After data transformation and feature scaling (2.3.5), the time series is then split into training and

testing sets. For the training data, the whole time series is taken until September 2015. The testing data is considered from October 2015 until September 2016 to cover the fire season which annually starts in September. To train the model, a sliding window approach is applied to generate samples. The sliding window is defined by the window length i.e., the sequence length considered by LSTM to make a prediction, and the window horizon or the number of predicted time steps. The input data of LSTM must be three-dimensional [samples, timesteps, features]. In this case, each step of the sliding window is considered as one sample, the window length is equivalent to the time steps and the number of features is the number of predictors. Afterward, the data is split into input i.e., the predictors, and output i.e., ignitions.

2.4.2 Metrics of evaluation

To assess the network performance, the Root Mean Square Error (RMSE), Root Mean Absolute Error (MAE) and Coefficient of determination (R^2) metrics are used.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{N}}$$

Equation 5: Root Mean Square Error (RMSE)

$$MAE = \frac{\sum_{i=1}^{n}|y_i - \hat{y}_i|}{N}$$

Equation 6: Mean Absolute Error (MAE)

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y}_i)^2}$$

Equation 7: Coefficient of determination (R2)

Where y_i is the actual value and $\hat{y}_i$ is the predicted value of sample i, $\bar{y}_i$ is the mean value of all samples

2.4.3 Experiments

The selection of the best LSTM structure usually involves performing multiple experiments with different hyperparameters using a trial and error approach, where the error is measured after each attempt to quantify the implications of using one specific parameter. Those parameters include the number of hidden layers, number of LSTM units in each

layer, time lags, different activation functions, different loss functions, and different optimizers.

Activation functions in neural networks determine how the weights are summed up in the node to produce an output. When the activation function is nonlinear, it helps the model to account for non-linear relationships between the input variables and the output. The choice of the activation function has a large impact on the neural network performance. Therefore, three functions were tested, the logistic function (Sigmoid), the Hyperbolic Tangent function (Tanh), and the rectified linear activation function (ReLU). In this research, the ReLU (Figure 8) outperformed the other functions and improved the neural network by speeding up the training.

Neural networks learn using gradient descent algorithms. Gradient descent is an optimization algorithm used to minimize the values of the loss function by updating the network weights iteratively until it finds the minima of the function. The algorithm calculates the gradient in each iteration towards the direction of the steepest ascent. The size of the step that the algorithm takes in each iteration to reach the local minima is called the learning rate (Goodfellow et al., 2017, pp. 294–310). There are multiple optimization algorithms with adaptive learning rates, however, the Adaptive Moment Estimation (Adam) optimizer has been adapted in many studies as it outperformed the other algorithms (Ruder, 2016). In this thesis, the Adam optimizer was used with a learning rate of 0.001.

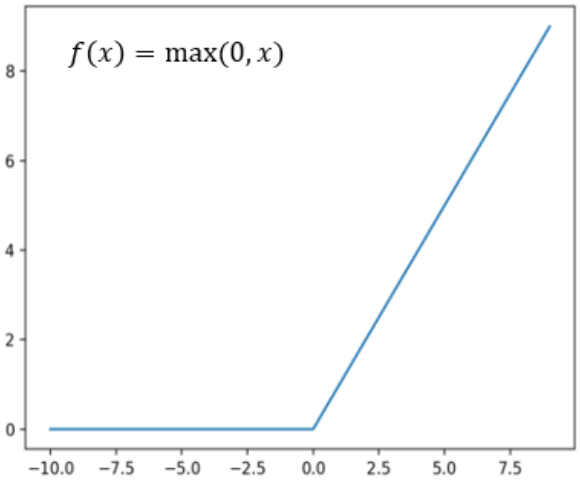

Figure 8: ReLU activation function. The function is half rectified, this means it outputs zero across half its domain, therefore, any negative input given to the ReLU activation function turns the value into zero immediately (Goodfellow et al., 2017, pp. 193–195). It is also less susceptible to vanishing gradients that prevent deep models from being trained.

To avoid overfitting, a dropout layer was used which is a simple but remarkably effective regularization method (Srivastava et al., 2014). This technique temporarily skips or drops out randomly a number of neurons to prevent them from learning an interdependent set of features weights which could lead to overfitting.

To systematically choose the other parameters, we start with a simple structure with one LSTM layer of 64 units and the Relu activation function. The loss function is the Root Mean Squared Error and the optimizer is Adam optimizer with a learning rate of 0.001. Since the model is training each pixel independently, running the network takes a significant amount of time. Therefore, batches of the data were selected from different locations of the study area which represent about 800 pixels. Table 5 shows the RMSE and MAE values with different experiments. For the number of hidden layers, It is observed that the error increased with the increased depth of the network. In the author's opinion, this indicates either that the model is overfitting or the relationship does not require a very complex model to be explained. The table also demonstrates the results for the experiments with different time step lengths and different activation functions. The different time step length affects the performance of LSTM as it accounts for the number of previous time steps used to make a prediction. Taking into account the previous year when predicting the next month has yielded the least error. Furthermore, the mean absolute error (MAE) loss function has decreased the error. MAE loss function is usually useful when the data has outliers as it is more robust to higher values. This makes sense given the nature of the ignition data which suddenly displays really high values in particular months.

Different types of LSTM could have considerable effect on the output. So far, the structure used is a vanilla LSTM which is defined with one input layer, one fully connected LSTM hidden layer and a fully connected output layer. This type of LSTM ,also called unidirectional, preserves information of the past because the only inputs it has seen are from the past. Bidirectional LSTMs are an extension of traditional LSTMs which learn the entire sequence in both forward and backward direction before making a prediction. Table 5 shows that using Bidirectional LSTM did not improve the results in this case. The final chosen hyperparameters of LSTM can be found in Table 6.

Table 5: Setup tests results for different parameters to determine LSTM structure

Experiment	RMSE	MAE
Different Number of hidden layers		
One LSTM layer	**5.915882**	2.801811936
Two hidden layers	6.239518	2.867196477
Three hidden layers	6.499306	3.01
Different time lags		
One layer 6 months	5.915882	2.801811936
One layer 12 months	**5.760793**	2.71326
One layer 18 months	5.815041	2.759578
One layer 24 months	5.766851	2.734103
Different Activation Functions		
RMSE	5.760793	2.71326
MAE	**5.715536**	2.6674773
Different types of LSTM		
Vanilla LSTM	**5.715536**	2.6674773
Bidirectional LSTM	5.807794393	2.707526329

Table 6: LSTM neural network hyperparameters used in this thesis

hyperparameter	Value
Learning rate	0.001
Batch size	12
Window size	12
Loss function	Mean Absolute Error (MAE)
Activation function	Relu
Optimizer	Adam
Hidden layers	1
Input data size	(12,12,7)
Drop out	True (0.2)
Feature scaling	True [0,1]

2.5 Evaluation

The evaluation of LSTM neural network performance involves comparing it to other baseline models. In this thesis, the performance of LSTM is compared to the performance of linear regression and ridge regression. The linear regression algorithm tries to find the linear relationship between the variables and the output. Ridge regression is an extension of linear regression which belongs to a class of regression tools that use L2 regularization.

This regularization technique adds a penalty to the linear regression to avoid overfitting. The comparison between the three models is based on the value of RMSE and MAE for the entire study area, and scatter plots to display the behaviour of the models for each month. The three models are applied using the same variables and conditions.

2.6 Visualization techniques

In machine learning, it is equally important to have a model with good results and interpretable predictions. This can be achieved by applying specific techniques to find out which patterns the algorithms are learning and which features are affecting its decisions. Feature importance values indicate which variable has the highest impact on the output of the model. The purpose is to get a better understanding of the model's logic to determine whether the predictions are sensible. It can be also used to select the most significant variables which helps in reducing the complexity of the model while keeping the same prediction accuracy.

There are multiple techniques to explain machine learning models. Some of these techniques are model specific and can be applied only on interpretable models as explained in 1.5. For complex models, global and local model-agonistic methods are applied. Most of these techniques apply independent explanation approaches. Global methods interpret the average behaviour of the model whereas local methods explain individual predictions (Molnar, 2020). When the goal is to measure the effect magnitude of each feature for the entire model, permutation feature importance is applied. Whereas, to describe the general relationship between one feature and the predicted values, Partial Dependence Plots (PDPs), Individual Conditional Expectation (ICE) or Accumulated Local Effects (ALE) plots can be employed. Feature interaction techniques are also used to describe how two features affect each other.

LSTM neural networks are complex models. Therefore, only model-agonistic methods can be applied. However, given that LSTM takes data in 3D, this limits the implementation of some approaches from a technical point of view. In this section, multiple methods that can be applied to LSTM neural networks are explained. The implementation and limits of these techniques are further discussed to select the most appropriate one for LSTMs.

2.6.1 Permutation feature importance

The concept of measuring feature importance with permutation was first introduced for Random Forests (Breiman, 2001). Afterwards, Fisher et al., (2019) introduced a model-

agnostic approach based on the same idea. The theoretical principle is rather simple. To measure the importance of one feature to the model output, we shuffle the values of this feature randomly then calculate the error increase. If the variable is of high importance, the error would increase significantly because the model relies on this feature to make a prediction (Molnar, 2020).

The advantages of this approach are that it works with any model type, is easy to understand and implement, gives global insight for the whole output, and takes into account all interactions among all variables. On the other hand, the disadvantages can be summarized as follows (Molnar, 2020):

1. It is not clear if the error should be measured on training or test set
2. This approach does not give in depth insight on features' interactions or the accurate relationship between the feature and model output
3. This method describes only the error and does not give a clear explanation how the model's output variance differs by permuting one feature
4. Shuffling the features randomly will give different results if the calculations were performed more than once. The optimal solution here is to repeat the calculations several times then compute the mean but this requires higher computation time and effort.
5. If there is correlation between variables, the results might be bias. When permuting one feature that is correlated with another, in this case the importance will be split between the two features.

In this thesis, permutation feature importance was applied for the whole area. For each predictor in each pixel, the values were shuffled randomly then the error was measured. This procedure was repeated six times. The importance of each feature was determined based on the magnitude of the error increase related to the original error of the model.

2.6.2 Variance-based Feature Importance

This approach was proposed by (de Sá, 2019) to determine the relative importance of features in neural networks generally. The concept of this method depends on capturing the weights of the neurons connected to each feature. When making a decision or a prediction with the neural network, the variables with the highest importance will get the highest change in neurons' weights while training the model and this will determine the contribution of each feature to the final output. This technique measures the variance of each neuron' weight changes regardless of their values, and by measuring the total variance of the weights for each node connected to one feature, the relative importance can be calculated.

The application of this technique is rather simple. It works by only adding an extension to the neural network. This extension is readily available by the author using Python programming language. After modifying the network, the training was run again for the entire study area. This method works only with neural networks and is not general for all machine learning models.

2.6.3 Explaining model predictions through explanation method

Model-agnostic methods separate the explanations from the machine learning model where they use simpler algorithms independent of the model to explain the feature importance and the relationships between them. The biggest advantage of these methods is that they are flexible and can work with any model type (Molnar, 2020).

Current methods, such as DeepLIFT (Shrikumar et al., 2017), LIME (Ribeiro et al., 2016), Layer-Wise Relevance Propagation (Bach et al., 2015) and Classic Shapley Value Estimation (Lipovetsky et al., 2001; Štrumbelj et al., 2014) use the same explanation method called Additive Feature Attribution Method. This method explains the model output by assigning an effect for each feature then summing the effects of all values attributed to all features (Lundberg et al., 2017). In their paper, Lundberg and Lee (2017) introduced a new approach that unifies the attributes of all methods mentioned before into one game theoretic approach to explain the output of any machine learning model.

SHAP (SHapley Additive exPlanation) explains each prediction individually by calculating a value for each feature. These values are called Shapley values and were originally created by Shapely (1953) using a cooperative game theory approach that assigns a payout to each player based on their contribution to the entire output (Molnar, 2020). This approach was adapted for machine learning models to calculate the features' contribution to the model's output by considering them as players in the game which is making predictions. These values could be positive or negative indicating in which direction they are pushing the output.

SHAP introduces multiple explanation techniques for different models. For example, LinearExplainer is used for linear models, TreeExplainer is designed for models that are based on a tree-like decision tree, random forest, gradient boosting and many more explainers. In this thesis, the focus will be on explainers which work with neural networks and specifically LSTM. SHAP Explainers that are compatible with deep machine learning models are Kernel SHAP and Deep SHAP. Kernel SHAP is a model-agonistic approximation method which works with all models, however, kernel SHAP requires data of 2D shape [samples, features]. This technique can be applied for LSTM in case it considers one time step only to predict the next one. Since our LSTM model takes into account the previous

twelve months to make a prediction, the kernel SHAP cannot be applied here. For 3D data, DeepSHAP explainer can be used.

SHAP explains each prediction locally but these values can be aggregated to represent global views of feature importance. Therefore, DeepSHAP explainer is applied first to one prediction, then explanations for multiple pixels are combined to examine which type of SHAP plots are appropriate for pixel-based LSTM. Furthermore, other visualization techniques are investigated to depict feature importance, feature dependence, and feature interactions.

3 Results and discussion

3.1 Model evaluation

The structure of LSTM was selected based on multiple experiments (2.4.3). The selected LSTM hyperparameters (Table 6) are used then to predict one year in advance for the whole study area. The prediction covers the fire season starting from October 2015 until September 2016. The neural network was trained pixel-wise which means each multivariate time series with 168 monthly data points represents one pixel of the study area. A visual comparison between the predicted data and the Fire Atlas data can be seen in Figure 9.

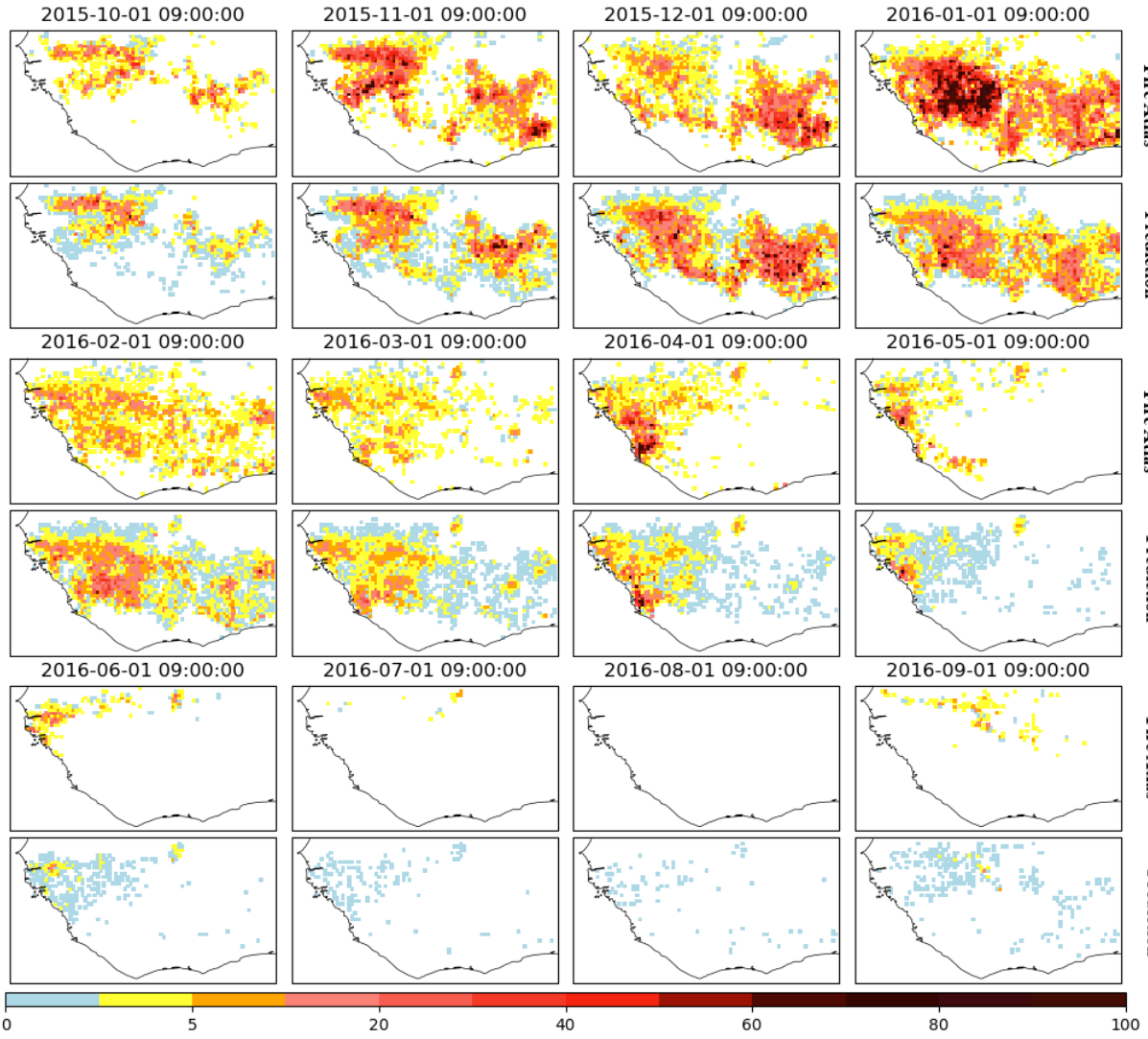

Figure 9: Comparison between the results of LSTM neural network and Fire Atlas data for one year prediction

In general, the model was able to depict the spatial pattern of fire ignitions even though no spatial information was passed to the neural network. Figure 9 shows that the network has predicted the fire season pattern in all months successfully. However, LSTM was not able to forecast the extremely high values during the peak of the season. In most of the cases, LSTM underestimated fire ignitions and predicted values around the mean. This can be clearly observed in November, December and January (Figure 10). During the months when there is a small number of fires or no fire at all, in June, July and August, LSTM predicts small numbers of fire occurrences in scattered locations. Those random fire ignitions, which are shown in blue, range only between 0 and 1 ignition points (Figure 9).

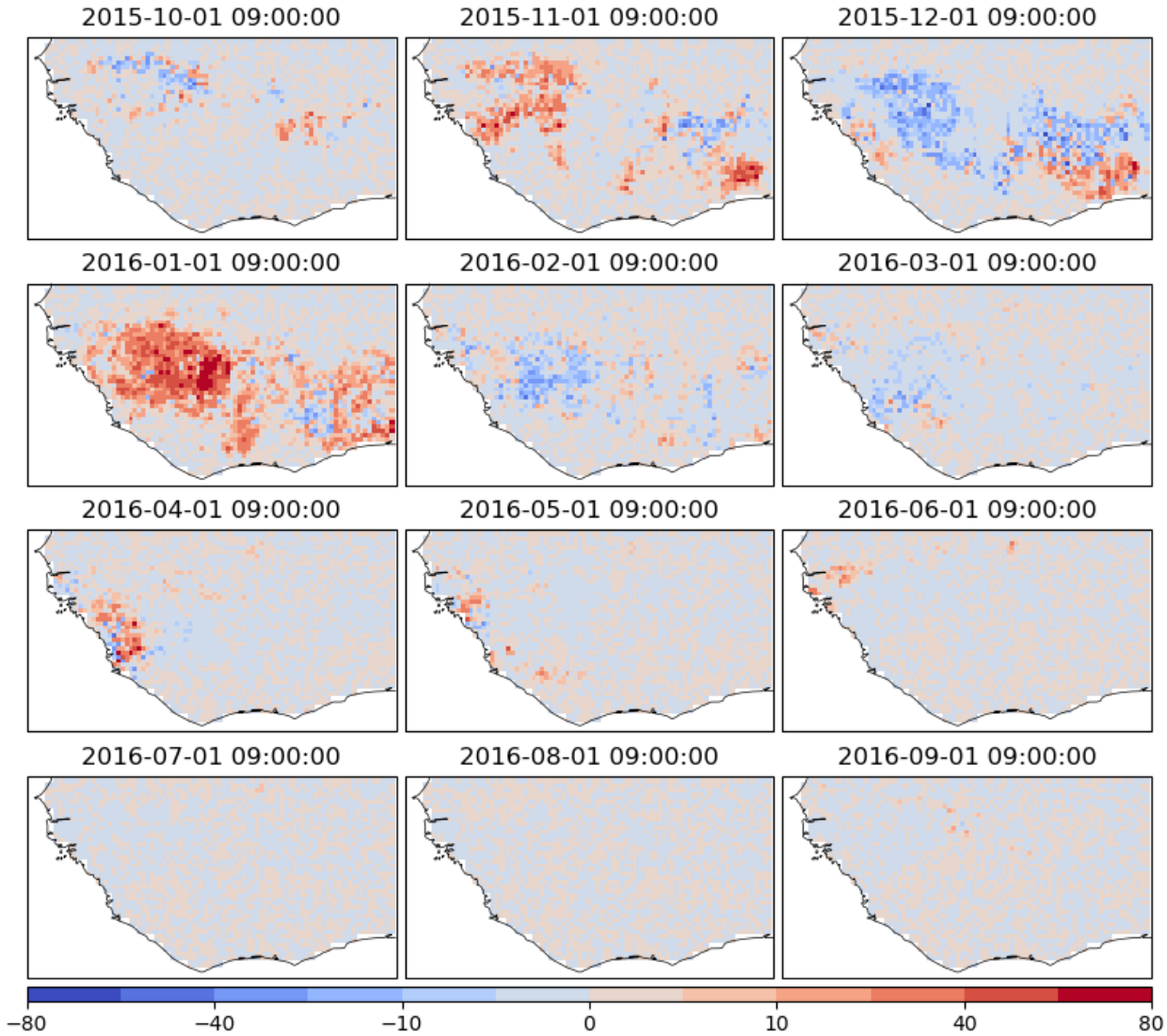

Figure 10: The difference between original and predicted fire ignitions for one year prediction

To evaluate the model, fire ignitions are predicted using two baseline models then compared to LSTM results.

Table 7 shows the mean error for the whole study area obtained by those three models. Furthermore, comparative scatter plots of the models' outputs with the original values for each month is shown in Figure 11. Each point represents one prediction for one pixel. The x-axis represents the real values whereas the y-axis depicts the predictions by different models. The dashed black line represents the ideal case where the predictions match the real values perfectly and the error is zero. The distance between each dot and the dashed line shows the magnitude of the error.

In general, the linear regression performed the worst with really larger errors, this can be seen especially in February (Figure 11). We can also observe that the Ridge Regression behaviour is similar to LSTM in some months. However, it tends to overestimate the zero values when there is no fire, for example, this is noticed from June till September. Even though LSTM is underestimating high fire values, the model is showing more robust behaviour towards small to medium numbers of fires than the linear models. Overall, LSTM performed better and this indicates that the relationships between fire ignitions and the predictors are non-linear and cannot be modeled with simple linear models.

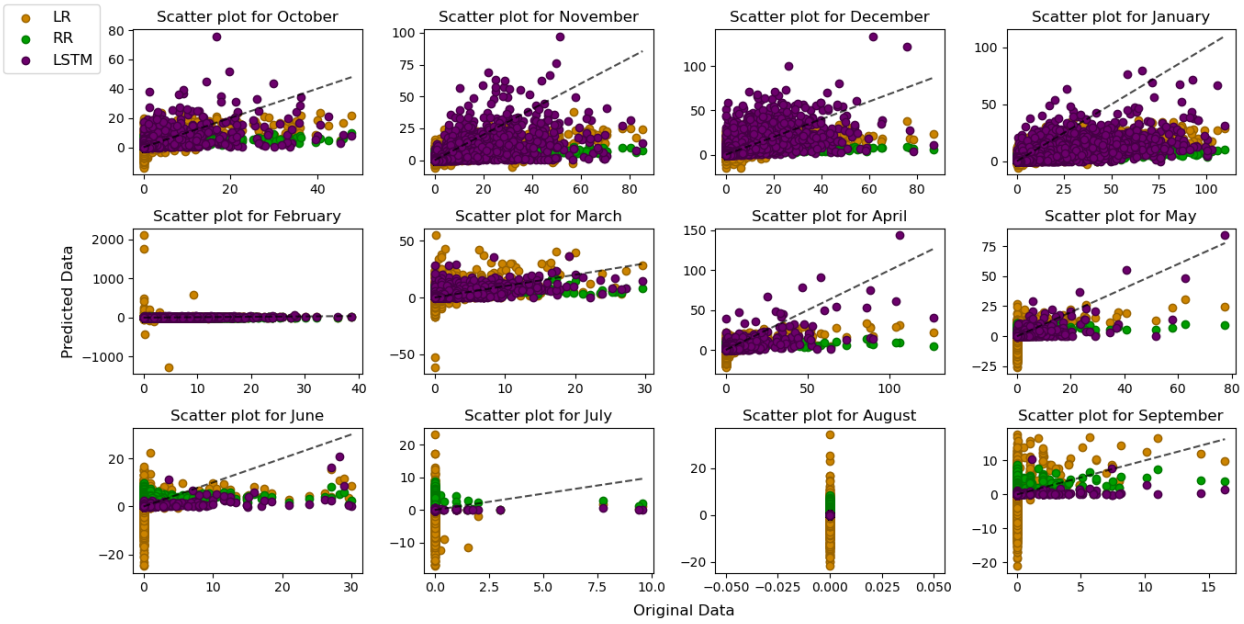

Figure 11: Comparative scatter plots of LSTM predictions (purple) with two baseline models, Linear Regression (LR) (orange) and Ridge Regression (RR) (green)

Table 7: Comparison of RMSE and MAE for one year prediction for the entire study area

Prediction for one year	RMSE	MAE
LSTM	3.333	1.509
Linear Regression	4.48353	2.8453
Ridge Regression	4.0209	2.5585

To better understand LSTM behaviour, a pixel-based map of the coefficient of determination (R^2) values for 12 months predictions was created (Figure 12). For each pixel one value of R^2 is given and this value shows how well the 12 predictions match the original values. The coefficient of determination is a measurement used to explain how much variability of one factor can be caused by its relationship to another related factor. The goodness of fit is represented as a numerical value. A value of 1.0 indicates a perfect fit, and the model is reliable for future forecasts. While a value of 0 would indicate that the calculation fails to accurately model the data at all. Figure 12 shows that only 13% of the pixels had high values of R^2, meanwhile, most of the data had medium or small fit values. It is also noticeable that there are pixels with negative values of R^2. The negative values indicate poor performance and the regression model is predicting a trend that is completely different from the trend of the data.

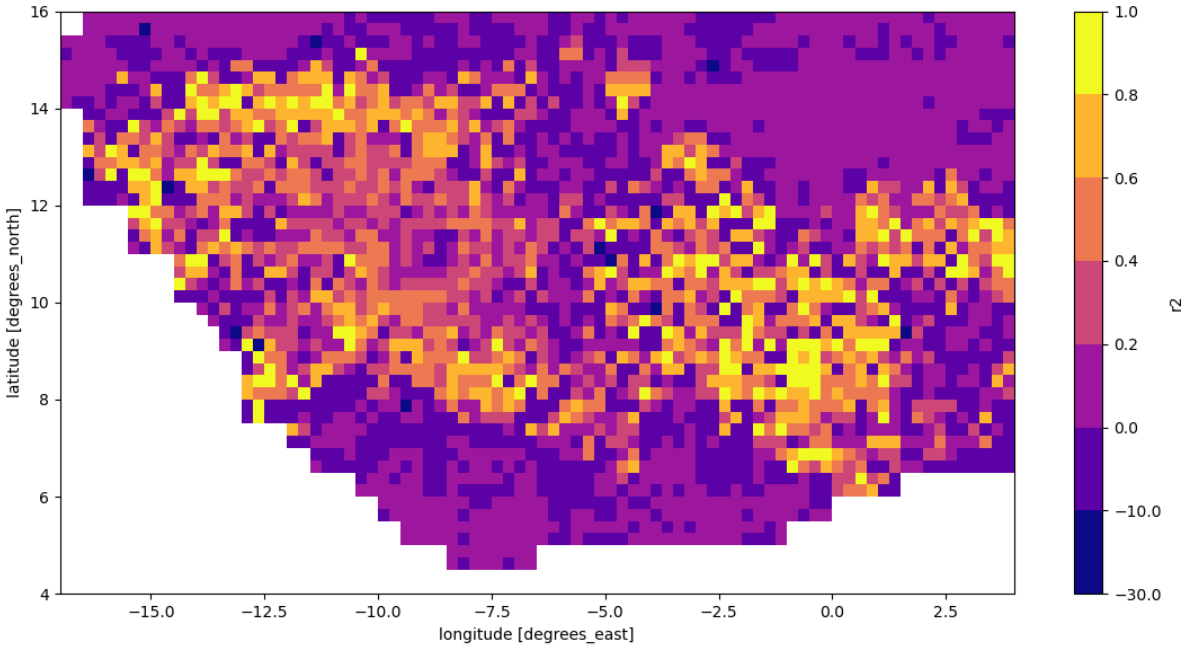

Figure 12: Pixel-based map of coefficient of determination (R^2) values for one year prediction

The previous map shows that LSTM performed really well in some areas and the regression model is representing the relationships between the predictors and the output correctly. In others the predictions did not fit the regression model and this means that future predictions cannot be reliable here. For more details, different samples of the data were selected and further studied to comprehend the effect of fire frequencies on LSTM performance. The first samples were selected where R^2 values are high. Those pixels are chosen mainly from the middle area. Figure 13 shows an apparent frequency in fire ignitions which ranges between 20 to 35 ignition points every year. In these samples, the values of R^2 are close to one and this means that LSTM successfully predicted ignitions count for the whole year including the peak of fire season. By observing this pattern in many other pixels, it can be concluded that when there is a repeated manner in the time series, LSTM is learning the conditions which created this pattern, and any changes in these terms are reflected immediately in the predictions.

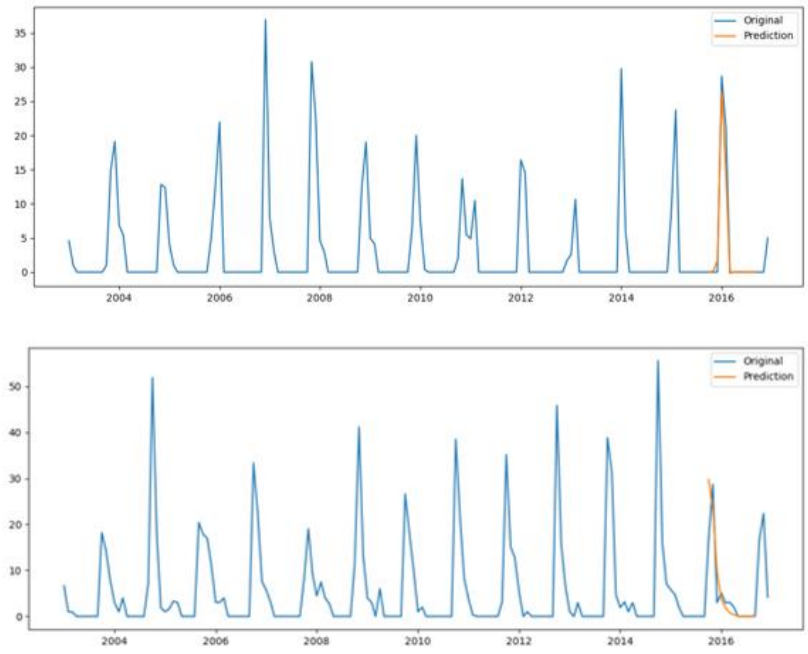

Figure 13: Two samples where R^2 values are high and fire ignitions are more frequent and annual

The other interesting case is when R^2 values are high but the fire occurrence is rare (Figure 14). LSTM is successfully predicting no fires even though, in the lower sample, a high ignition count was recorded in the previous two seasons. When fire frequency is low, LSTM was also capable of predicting the correct values.

The last case represents two samples with different fire occurrence frequencies but with a severe case in the last fire season. Figure 15 shows that LSTM was not able to predict this sudden uprise in the time series. In the upper sample, there is a pattern almost repeated every year where the ignition count ranges between 20 and 40. In the last fire season, ignition points reached 100 all of a sudden. LSTM in this case predicted a value around the mean. This can be understood since LSTM has never seen this situation in this time series so the network has not learned these conditions. In the lower sample, LSTM predicted no fire occurrence. In these cases, the values of R^2 were recorded as negative because the prediction is completely different than the original data trend.

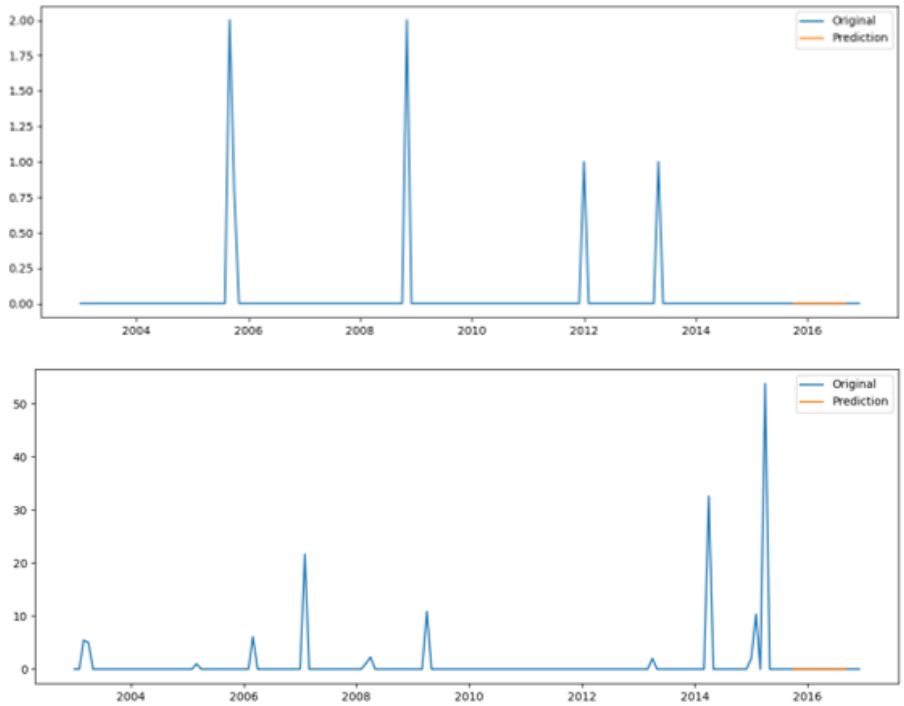

Figure 14: Two samples where R^2 values are high and fire ignitions are rare

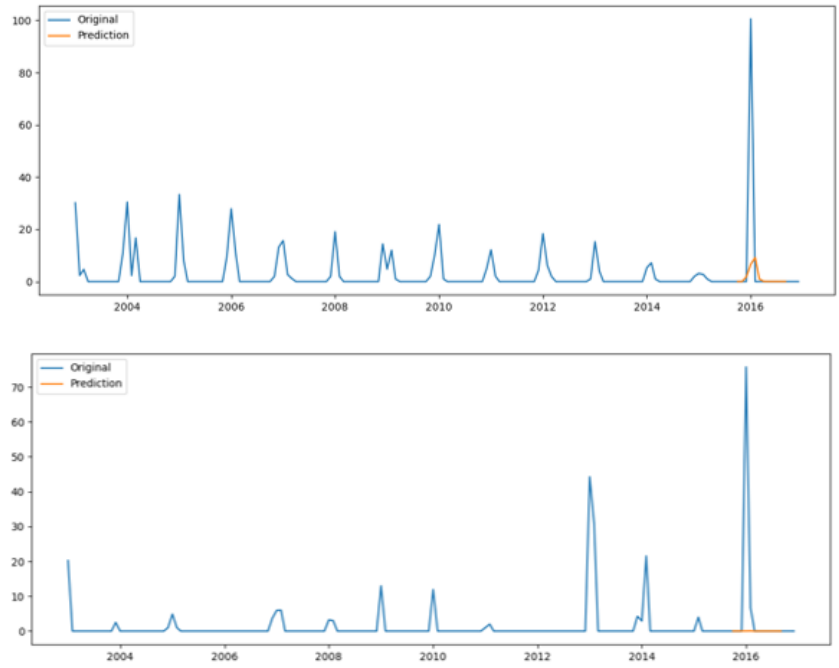

Figure 15: Two samples where R^2 values are low. Fire occurrence has different frequencies but with extreme values in the last fire season.

In summary, when fire occurrence was more frequent, LSTM produced accurate results (Figure 13). However, when a sudden uprise in the ignition count happened, the neural network was not able to forecast this event (Figure 15). Even though LSTM is known for its powerful prediction ability for extreme events, in this case, it was not able to project them correctly. The reason behind this could be the limited length of the time series in which the network did not see this pattern of events that led to this extreme event, and therefore could not learn from it.

3.2 Comparison of visualization techniques for explainable LSTM-based fire modelling

LSTM data format which should be in 3D imposes technical limitations for applying some model-agonistic explanation methods. In this thesis, three visualization approaches with different techniques and algorithms that can be applied to interpret LSTM neural networks are studied. The goal is to investigate each method's ability to depict overall feature importance, the spatial distribution of feature importance and predictor-response relationships.

Permutation feature importance is the simplest method that can be applied to any machine learning model. However, this technique only gives the overall importance of each feature without any further information about the relationships between the predictors and the output or the spatial distribution of feature importance. The results can be visualized as a simple bar chart as seen in Figure 16.

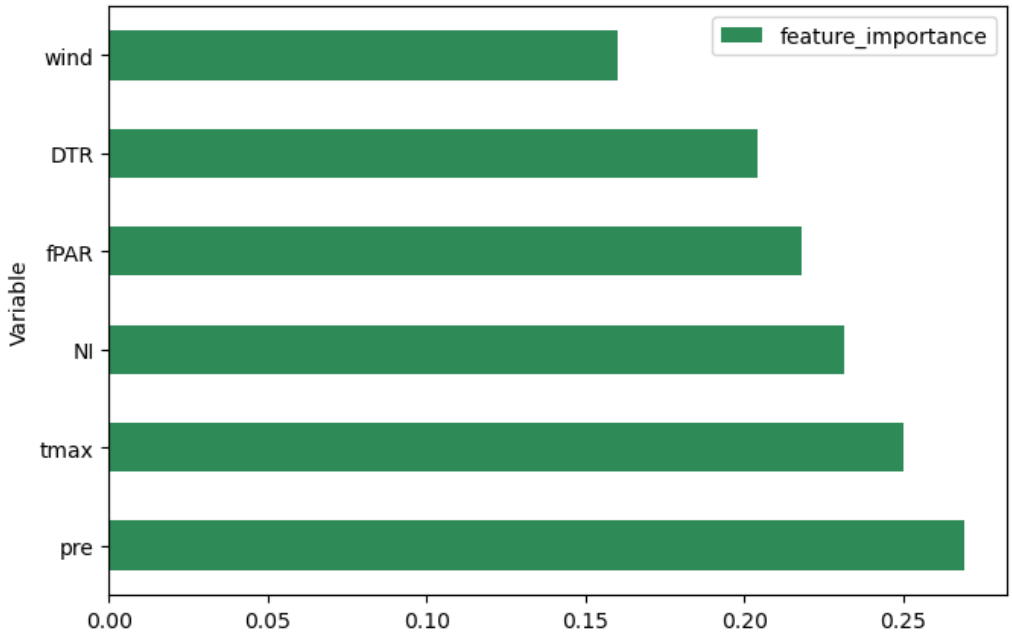

Figure 16: Permutation feature importance for the entire study area. The error increase is represented as percentage of the original RMSE of the model

Variance-based feature importance gives a rank for each variable in the ML model. Since the approach in this thesis uses an independent LSTM for each pixel, feature importance for the whole study area cannot be concluded. Therefore, the relative importance of each feature in each pixel is determined, then a map for each feature was created to show the spatial distribution of feature importance (Figure 17). The relative importance of each feature is calculated as a percentage, where 1 is given to the feature with the highest importance.

The advantages of this approach is that it is easy to implement and can give a general view for the model output. On the other hand, it does not give any visualizations about the features' relationships or interactions. Furthermore, given the stochastic nature of neural networks, running the model multiple times might lead to different weights and in this case different results, therefore, it is advisable here to repeat the training multiple times then taking the average result.

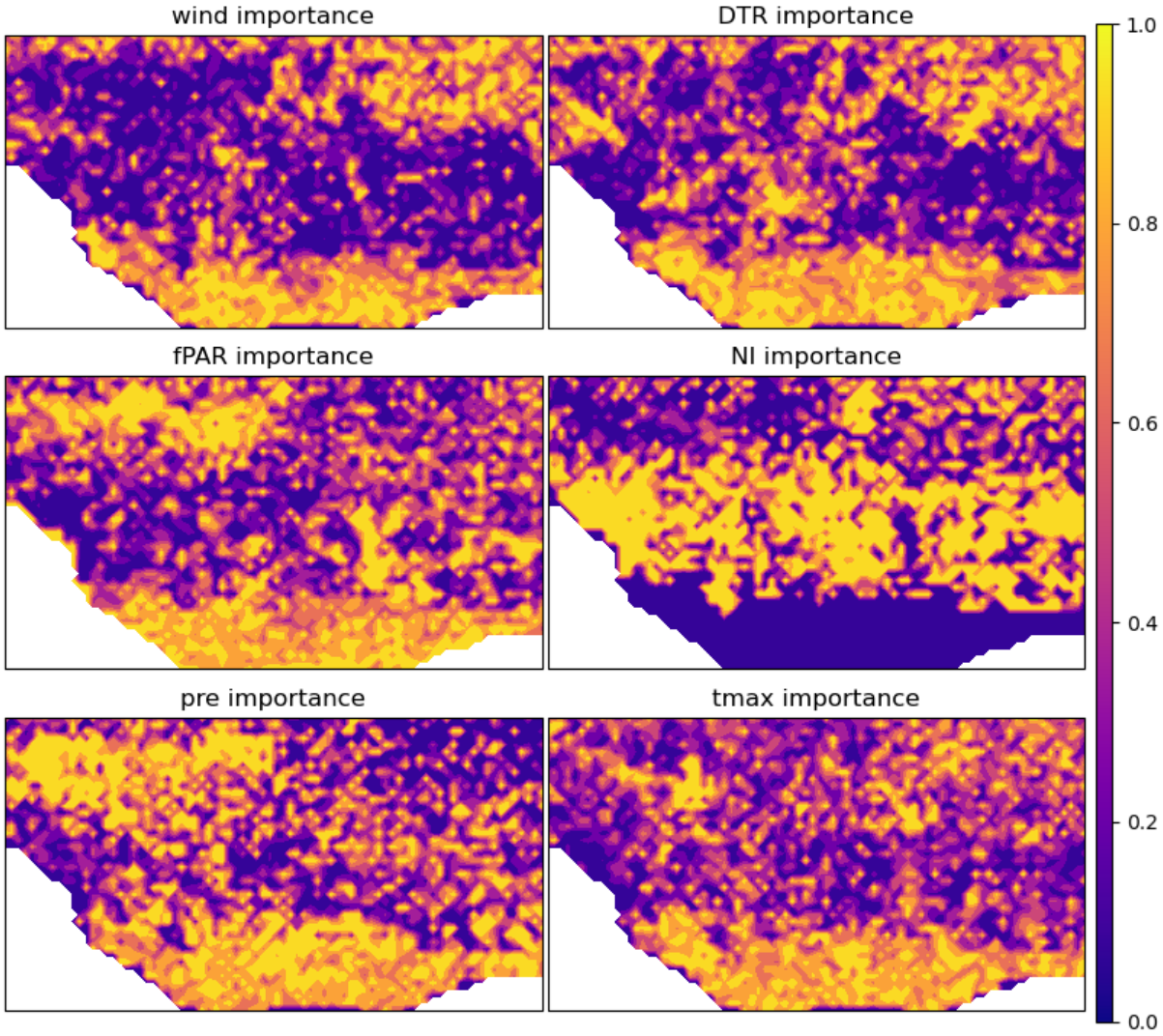

Figure 17: Variance-based feature importance for the entire study area. The relative importance of each feature in each pixel is represented as a percentage.

The variety of SHAP plots helps in interpreting all model decisions, starting from one prediction to the entire study area. To understand how SHAP works with LSTM in this thesis, firstly an individual prediction is explained. SHAP computes a base value which is the average model output over the training dataset. Then SHAP values explain how each feature pushes the model output from the base value towards the output prediction. Figure 18 shows a SHAP force plot of how the first time step affects the prediction of the first month (October) in one pixel. Features pushing the prediction higher than the base value are colored in red, whereas features pushing the prediction lower are in blue. The magnitude of shap value of each feature explains the feature importance or its responsibility for a change in the model output. SHAP also takes into account the order of feature introduction as well as the interactions between features, helping us better

understand the model performance. The sum of shap values equals the difference between the expected model output and the current model output.

It can be seen in Figure 18 that the base value or average ignition points in this case is 0.1325 and the output of the model is 0.18. On the one hand, most of the features are pushing the prediction higher than the base values and this means that these features are contributing in predicting higher ignition points at this pixel. fAPAR here has the largest effect based on the SHAP value magnitude. On the other hand, precipitation can be seen in blue color which indicates pushing the model towards lesser ignition points. This is reasonable considering that the presence of rainfall increases vegetation wetness and therefore the likelihood for fire occurrence is lower.

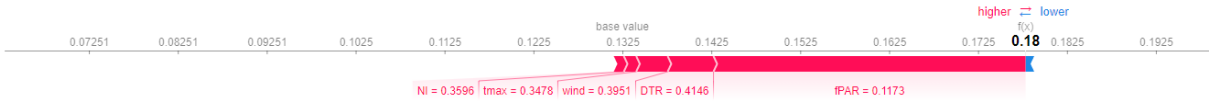

Figure 18: SHAP values for the first time steps for predicting one month (October) for one pixel. This prediction was explained by Deep SHAP. Red feature attributions push the score higher, while blue feature attributions push the score lower.

Force plots can also explain how the previous time steps affect the prediction. For the same month of October, the twelve force plot explanations such as the one shown in Figure 18 are taken, rotated 90 degrees, and then stacked horizontally (Figure 19). This type of plots is interactive where it is possible to see the SHAP value for each feature in each time step. When plotted in a static manner, they do not give much information. However, using the same plot we can see how each feature is affecting the output individually over the whole twelve time steps (Figure 20). Figure 20 shows an example of fAPAR impact on the model output. In this pixel, fAPAR values for the previous two months push the prediction lower, whilst fAPAR values for the previous year are more important and are pushing the output towards higher ignition values. Furthermore, using this type of plots, the effect of one feature on any other one can be visualized.

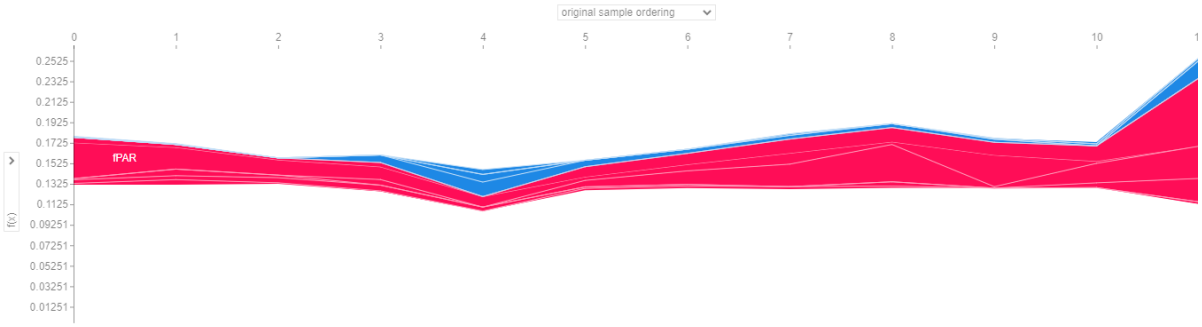

Figure 19: SHAP values for the previous twelve time steps for predicting one month (October) for one pixel. This prediction was explained by Deep SHAP.

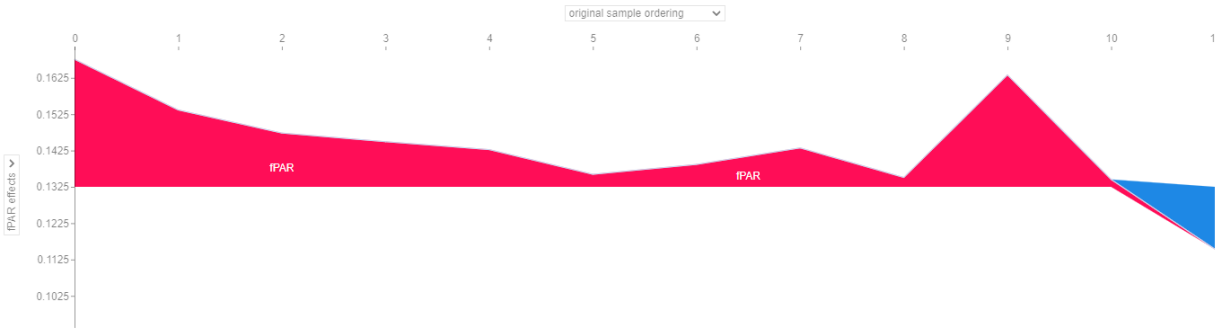

Figure 20: SHAP values to explain fAPAR feature effect for the previous twelve time steps for predicting one month (October) for one pixel.

The previous examples were demonstrated for one pixel. However, the goal here is to visualize feature importance for the entire study area. SHAP can be used for global explanations by running SHAP for every pixel. The result is a 3D matrix of SHAP values where each row in this matrix represents one row in the data and each column represents one feature. In Figure 21, a force plot for a small subset of the study area is displayed. The plot is condensed and does not clearly show which feature is more important than the other. Even though it is interactive, the plot does not give clear information. Therefore, for displaying feature importance for a large area, force plots are not the correct type of data visualization.

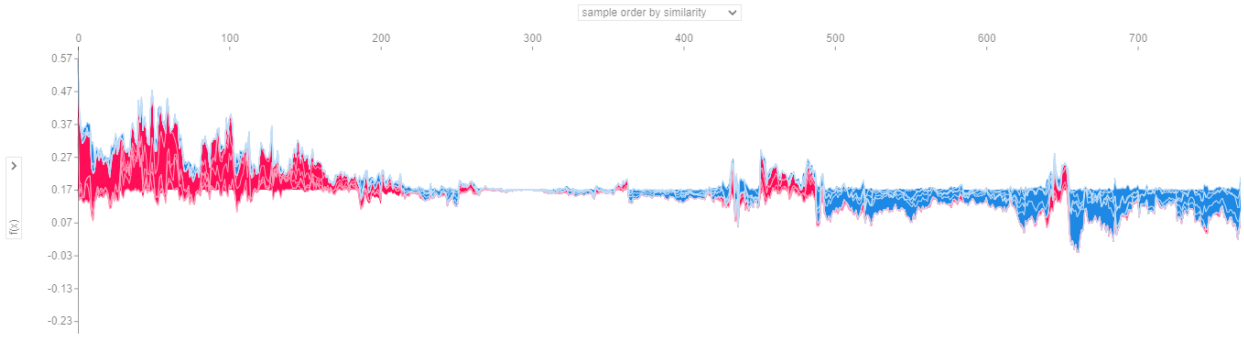

Figure 21: Force plot for a small subset of the study area showing feature importance

SHAP provides another method to classify the features by the sum of the magnitudes of SHAP values. One type of plots is called SHAP feature importance which is an alternative to permutation feature importance where it sums the absolute SHAP values per feature. The features with higher magnitude are more important. The other type is called SHAP summary plots. The summary plot shows the feature importance and SHAP values distribution at the same time. Each dot represents one row in the data and has three characteristics. The position on the y-axis represents which feature it is depicting. The position on the x-axis shows whether the effect of this point caused a higher or lower model prediction. The color of the point indicates whether the feature value was high or low for that instance of the dataset. Furthermore, the features are ordered according to their importance.

SHAP summary plots are able to visualize only 2D data, therefore, overall feature importance considering all twelve time steps was not possible, taking into account that for LSTM, SHAP gives a 3D explanation matrix. Therefore, SHAP values were extracted for each precedent month in each pixel. Figure 22 shows SHAP summary plots for the precedent one month. SHAP feature importance gives an overview of features' ranking but does not contain more information than the importance. On the other hand, the summary plot shows more information. For example, it can be seen that the high values of precipitation (red color) are pushing the model prediction towards lower values, meanwhile the lower values (blue color) lead the output to higher ignition points. Whereas DTR values are behaving in the exact opposite way. Using this type of plot gives an indication of the type of relationship between the feature and the model's output. SHAP summary plots are useful to understand feature importance for each precedent time step for the model output.

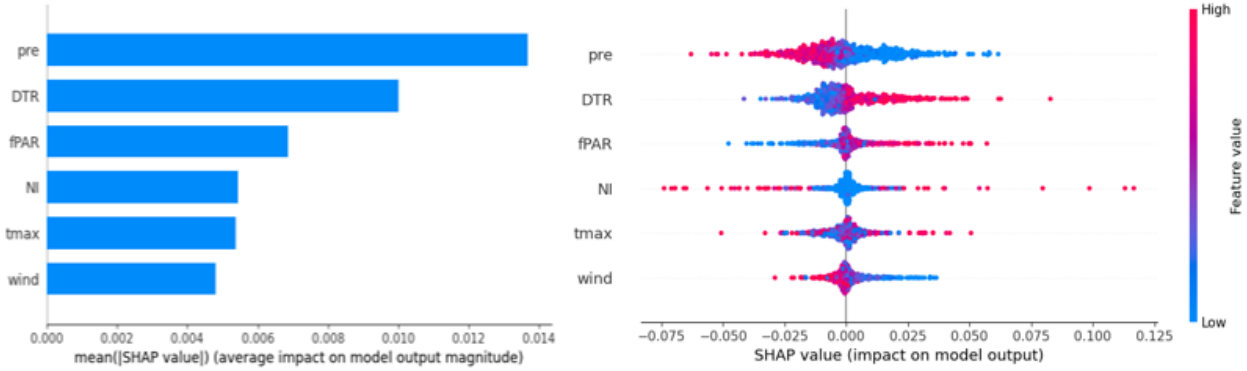

Figure 22: SHAP feature importance plot (left) and SHAP summary plot (right) for the precedent month

SHAP introduces many visualization techniques which helps interpreting LSTM model. The application of SHAP is not as simple as the other approaches. It requires computational time and a good understanding of which SHAP kernel is the most appropriate for the model type. The variety of SHAP plots helps in interpreting all model decisions, starting from one prediction to the entire study area. Furthermore, SHAP is built on a strong theoretical base which makes this approach more reliable than others (Molnar, 2020).

In summary, for pixel-based LSTM model, the opportunities and limitations of visualization techniques depend on the purpose of this visualization. For general understanding of the most important features, permutation feature importance is the easiest and simplest method. SHAP summary plots can also be used but just in case LSTM does not take into account multiple time steps to predict the next one. To map the spatial distribution for feature importance, using variance-based feature importance method, it was possible to depict the relative importance in each pixel. However, using SHAP values, it was not possible to create a map to show the spatial patterns. SHAP can depict the spatial distribution only if a convolutional neural network is used as a front-end to the LSTM model.

Force plots are suitable for explaining one decision at a time by using multiple visual variables simultaneously such as color, size and orientation in one simple figure to represent the magnitude of the feature effect and its trend (Figure 18). Those types of plots are suitable for LSTM because they accept 3D data and can also explain the effect of multiple previous time steps on one prediction (Figure 19). However, for many samples, those plots become inapprehensible as they are stacked and clustered by explanation similarity to find groups of similar instances (Figure 21). Even though interactivity is added to facilitate information extraction, this type of plots is not suitable for a large number of

instances. Using SHAP summary plots, it is possible to extract feature importance for each precedent time step.

3.3 Importance of predictor variables

Using permutation feature importance for the entire study area, precipitation was found to be the highest important variable for predicting ignitions followed by maximum temperature. Whereas, Nesterov index, fPAR and diurnal temperature range appear to have smaller effect on the model output respectively. Finally, the wind predictor generated the smallest error difference and this indicates that wind plays the smallest role in fire ignition.

When comparing the patterns of feature importance generated using variance-based feature importance method (Figure 17) to the land cover map (Figure 4), we can visually distinguish three regions. Those regions represent three different land covers following a gradual change in climate zones from south to north (2.2). In the southern areas where humid forests are, fPAR, precipitation, maximum temperature and DTR have the highest importance, whereas the nesterov index has no role at all in this region. This can be explained by the continuous rainfall in these areas which will prevent the Nesterov index from accumulating for long periods of time. In the middle region, the Nesterov index appears to have the highest importance among other variables which do not seem to exhibit any clear spatial pattern. In the northern regions, precipitation comes as the highest important variable in the western part in addition to fPAR in limited areas. In the north eastern parts, DTR and wind appear to have higher importance.

For further understanding of feature importance in those regions, SHAP was implemented to extract feature importance for the twelve precedent time steps required to make a prediction (Figure 23).

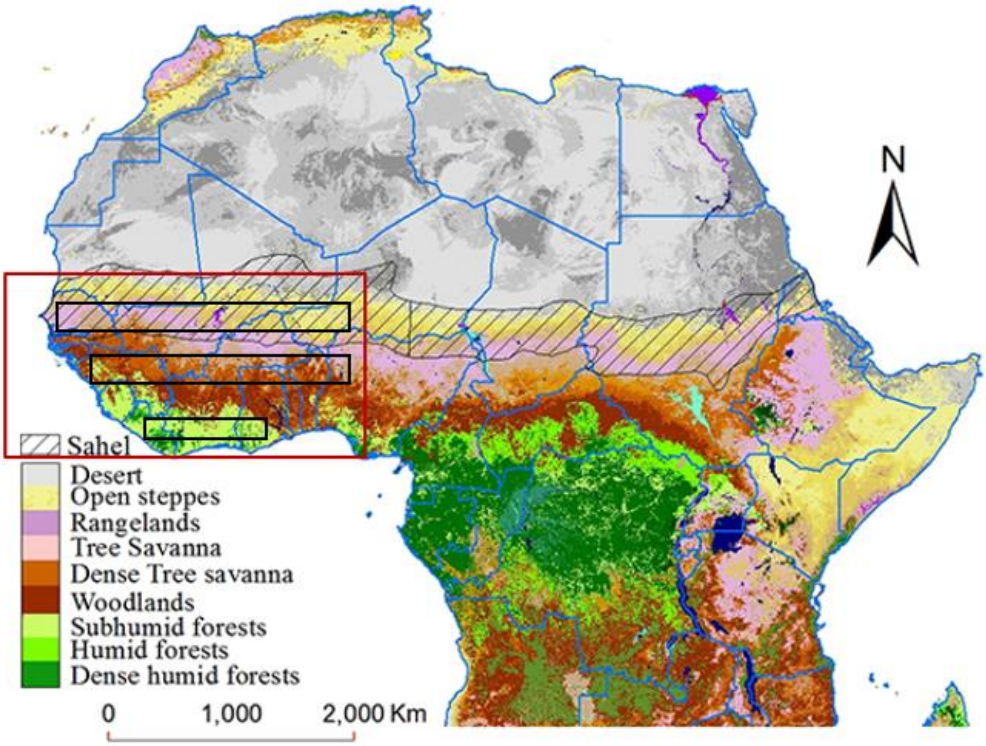

Figure 23: Sub-regions for SHAP feature importance analysis (black rectangles), taken from land cover map in the Sahel zone (Mbow, 2017). The red rectangle represents the entire study area.

In the southern subregion where humid and subhumid forests are prevalent, Figure 24 and Figure 25 show that the antecedent conditions > 10 months appear to be of higher importance for the model output where pre and fAPAR are equally important followed by DTR. Overall feature importance is considered by taking the absolute SHAP values (Figure 24) whereas Figure 25 shows the detailed positive and negative impact for each feature. For recent conditions, pre, wind and fAPAR ranked the highest respectively for the preceding one month, whereas this order is reversed in the following month. Wind has a high influence in recent conditions but this impact decreases gradually, on the other hand, DTR shows the opposite behaviour. In this sub-region, the Nesterov index has no influence.

The negative and positive effect of the features in each month is almost similar except for some differences. For example, in the preceding 4^{th} and 5^{th} months, the maximum temperature had a low positive effect but ranked the higher in the negative impact. This means that the temperature during these months contributed to a smaller ignition count because the temperature in these areas are at the lowest during these months.

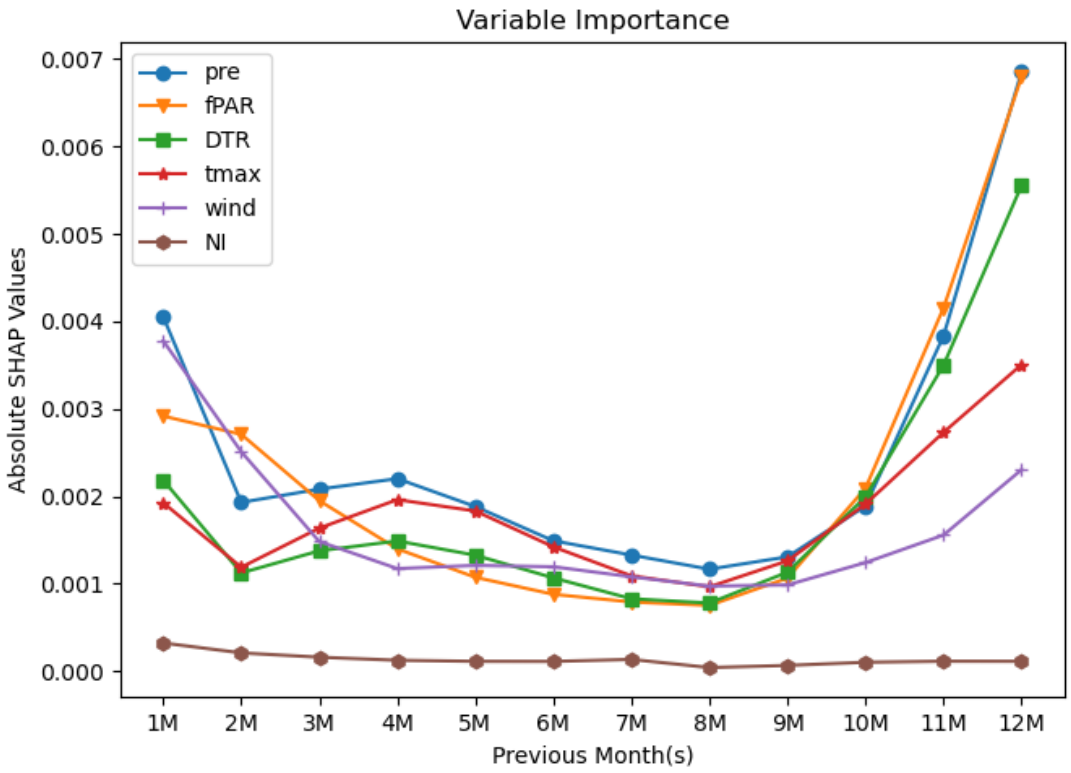

Figure 24: Feature importance for the previous 12 months in the southern sub-region

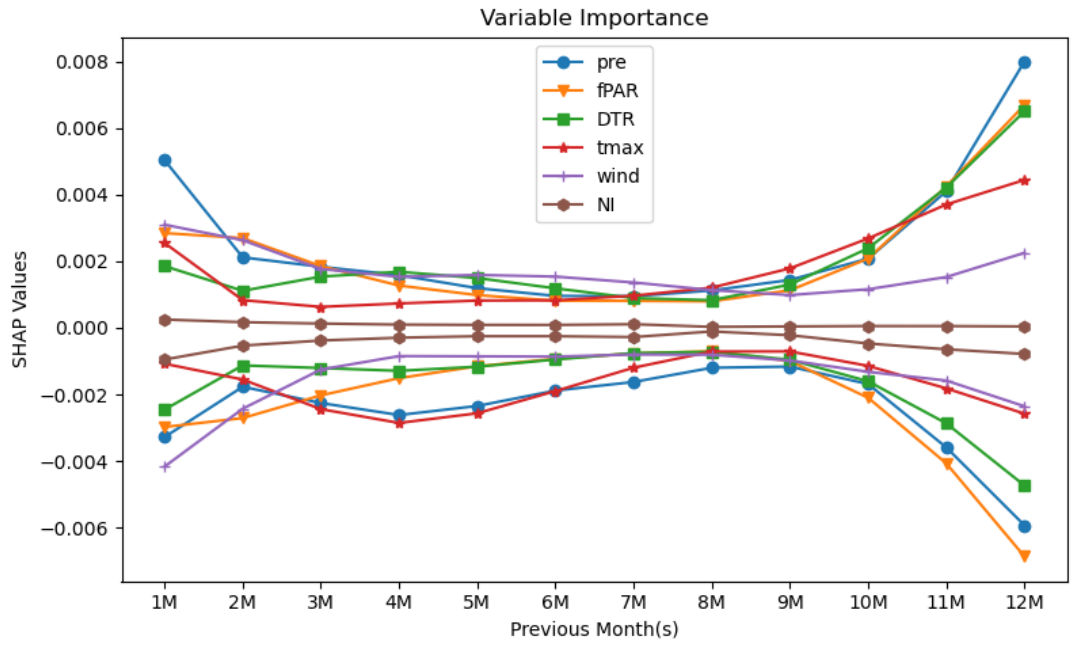

Figure 25: The negative and positive effect of each feature for the previous 12 months in the southern sub-region

In the middle region where the vegetation is generally characterized by woodlands and dense tree savanna, fAPAR is the dominant feature for precedents < 4 months and > 8 months (Figure 26 and Figure 27). For the other months, pre ranked the most important variable. It can also be observed here that the antecedent year conditions play an important role in predicting fire occurrence, especially fAPAR. Furthermore, the influence of tmax appears significantly during the preceding 8th to 10th months. The Nesterov index has slightly higher influence here, especially for antecedent conditions > 10 months. All features have similar positive and negative influence (Figure 27) but the positive impact is generally higher.

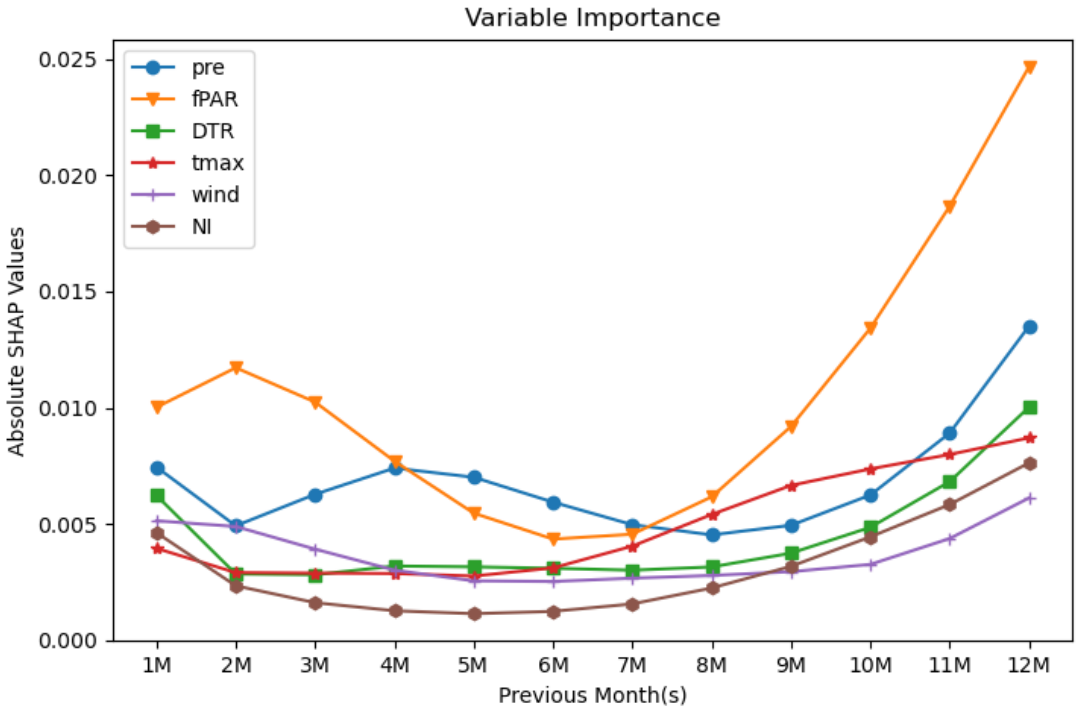

Figure 26: Feature importance for the previous 12 months in the middle sub-region

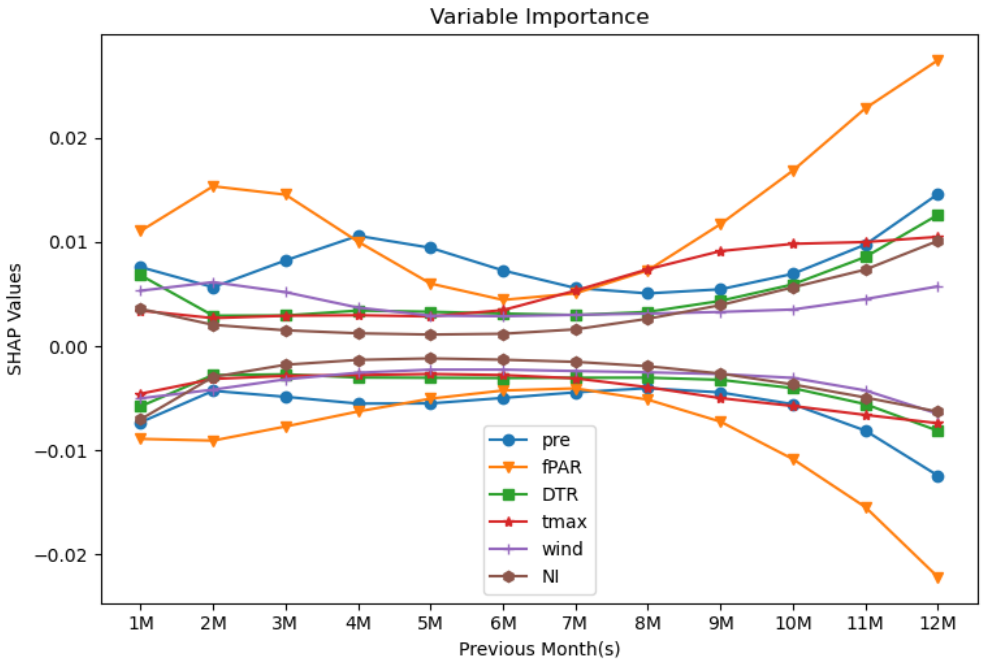

Figure 27: The negative and positive effect of each feature for the previous 12 months in the middle sub-region

In the northern sub-region where the transition between humid climate and desert climate begins, the dominant vegetation types are rangelands and open steppes. In this part, fAPAR and precipitation are also the dominant variables (Figure 28). The precedent conditions < 4 months are more important whereby vegetation presence has a much higher positive impact on fire ignitions (Figure 29). On the other hand, the Nesterov index affects more negatively towards less ignition points in the precedent one month. Generally, the importance of the Nesterov index and tmax increase in the preceding conditions > 6 months. This can be explained by the proximity to the desert, therefore, higher temperature and more dry days starts to have bigger contributions to fire occurrence. DTR and wind have the lowest effect on fire ignitions in this region.

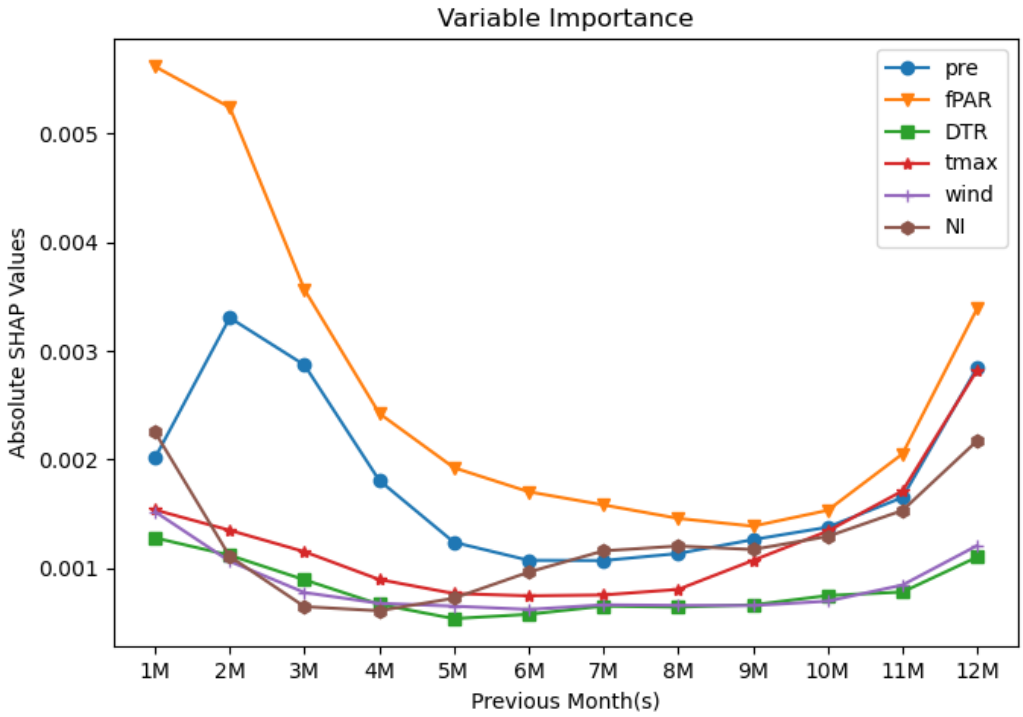

Figure 28: Feature importance for the previous 12 months in the northern sub-region

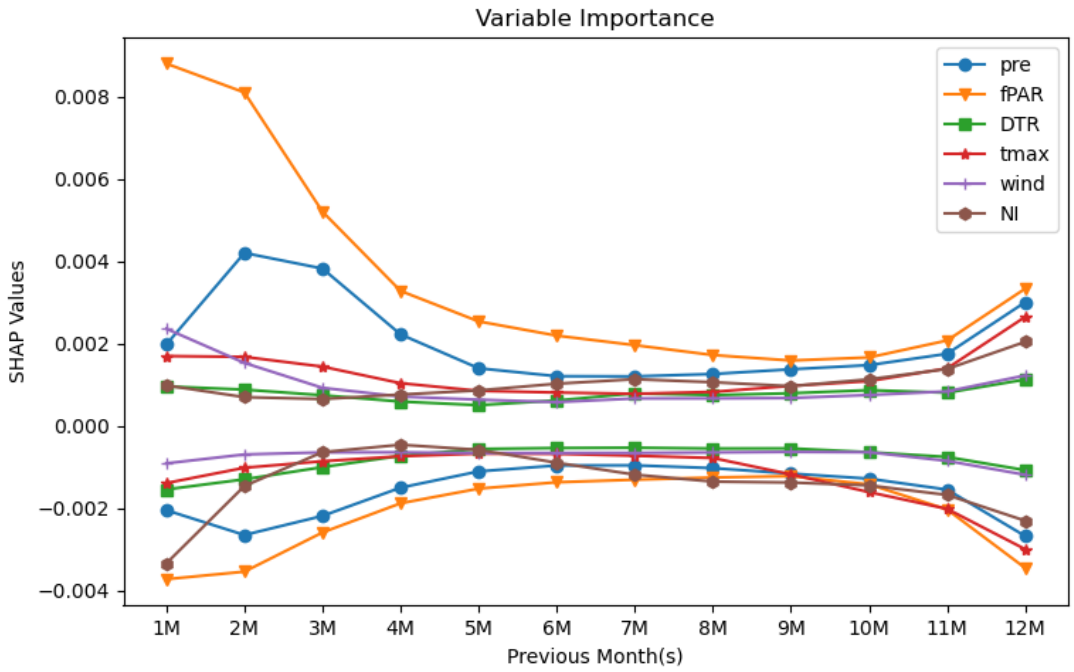

Figure 29: The negative and positive effect of each feature for the previous 12 months in the northern sub-region

The spatial distribution of feature importance is almost compatible between the variance-based approach and SHAP. Pre and fPAR were mapped as important in all regions using both methods. In the south, both methods showed that NI has no influence, furthermore, the maps showed that all other variables exhibited high importance, this is further explained by SHAP where each feature ranked as the highest in different months. In the middle region, NI appears to be the most important variable whereas using SHAP its importance increases slightly only for precedent conditions > 10 months. For wind and DTR, SHAP is depicting these features as the least important in the north whereas the maps show higher importance in the northeastern part.

3.4 Predictor-response relationships

SHAP was the only method used in this thesis able to depict the predictor-response relationships and interactions. SHAP dependence plots are an alternative to Partial Dependence plots (PDPs) and Accumulated Local Effects (ALE). These plots are the simplest way to explain the relationship between one feature and the output and they are simple to explain to a non-technical audience. The x-axis represents the real values of the feature, whereas the y-axis values show the corresponding SHAP values. Figure 30 shows how the SHAP values change with the values of each feature representing how the model depends on that feature. It can be seen that when fPAR and DTR values increase, SHAP values increase, or in other words, the number of ignition points. fAPAR represents the existence of vegetation and this agrees with the fact that when the area is covered with a continuous layer of vegetation, the probability of fire occurrence increases. Meanwhile, the rise in precipitation quantities leads to a decrease in ignition points. On the other hand, the maximum temperature does not show any clear relationship in this model. For the wind variable, there is a slight decline with higher values but it is not significant. The Nesterov index does not form any type of relationship and the values appear to be in disarray.

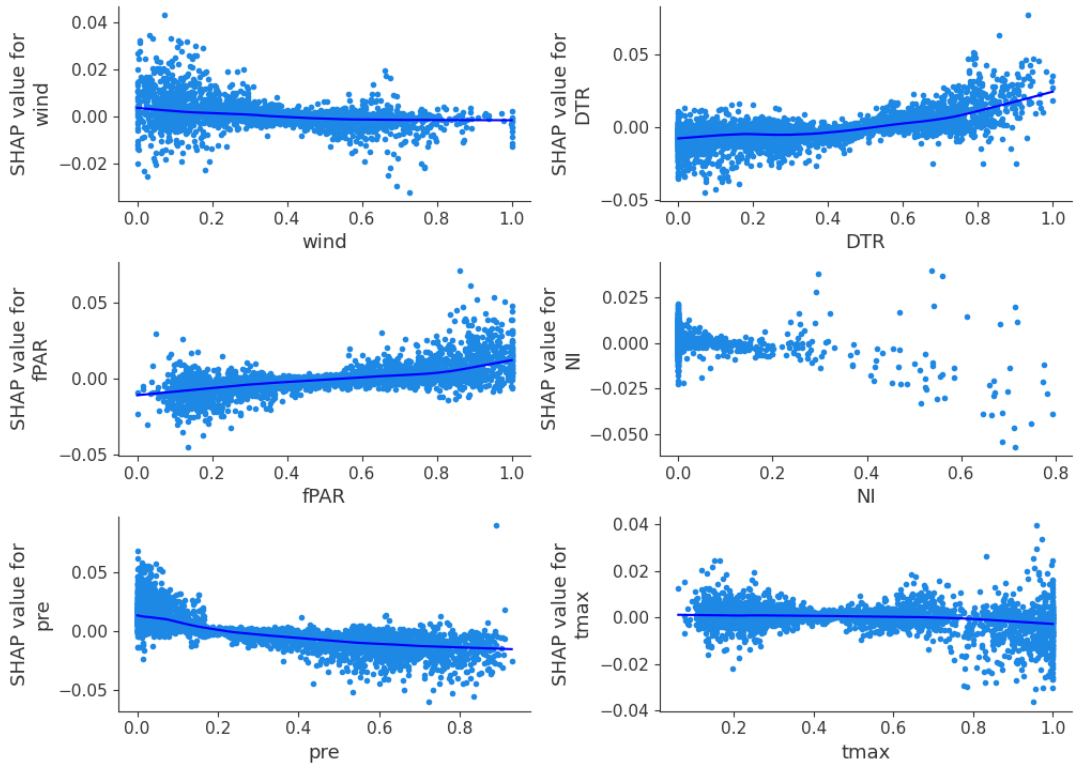

Figure 30: SHAP dependence plots

The dependence plots also have the advantage of adding interaction attributes to the features. By adding a color which indicates how each feature is affected by the values of another feature (Figure 31). For example, the relationship between precipitation and DTR inside the machine learning model can be seen clearly. The higher values of DTR (red) affect the lower values of precipitation. Whilst the small values of DTR have more interactions with the higher values of precipitation. However, this relationship is not always clear. Adding the additional color feature to this plot helped with compiling information without over-complicating the figure or affecting its simplicity in conveying the correct idea. This technique is functional when the relationship between the two features is very clear, however, in most cases the colors get clustered and it becomes very hard to explain the interaction effect.

Therefore, spreading the points in a 3D space contributes to the visual interpretation of the relationship (Figure 32). Some of the interactions shown in Figure 31 are not completely clear. For example, the interaction between precipitation and fPAR is clustered in some areas and the colors are mixed together. Therefore, to better understand the relationship and interactions, the 3D plot better visualizes the distribution of feature interaction. Even the understandable relationships like precipitation with DTR can be better comprehended when scattered in 3D space. SHAP interaction plots can help with understanding how features interact with each other inside the neural network but these relationships cannot be directly depicted like dependence plots.

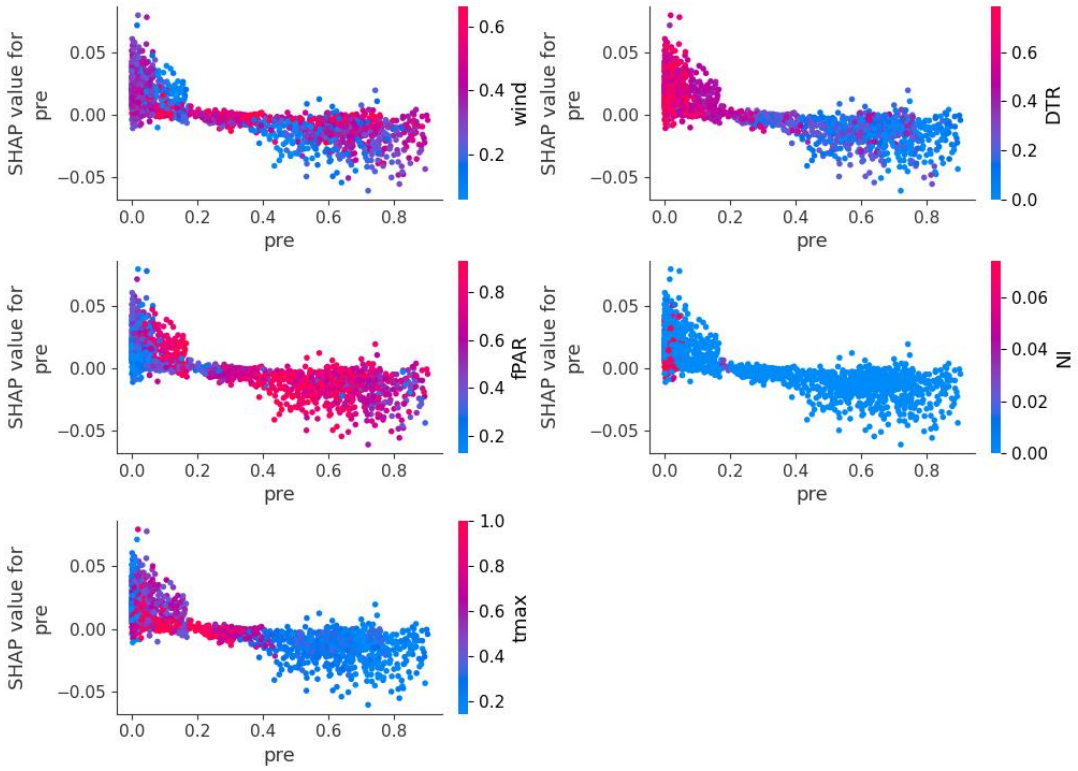

Figure 31: SHAP interaction plots for precipitation with the other variables

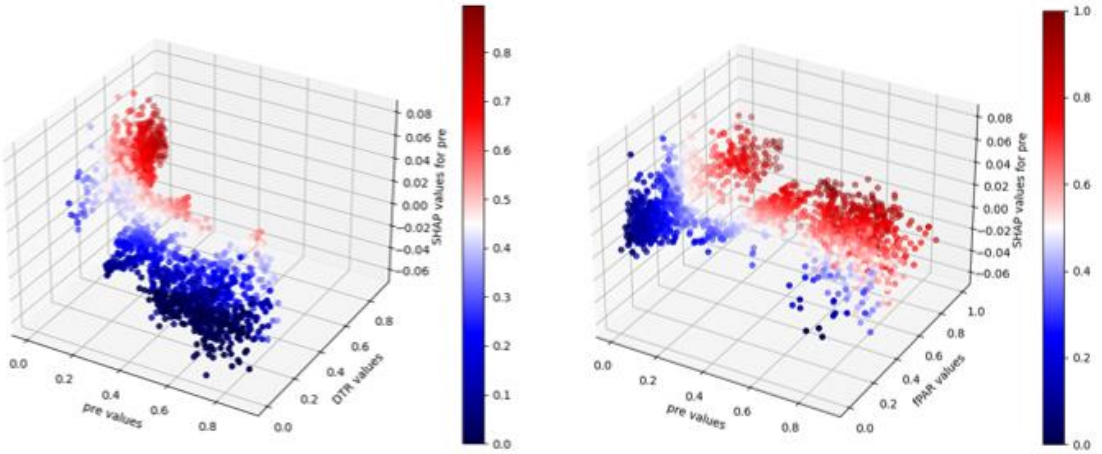

Figure 32: 3D interaction plots. DTR-pre interaction plot (left) and fPAR-pre interaction plot (right). The x-axis and y-axis represent the features' values. The z-axis represents SHAP values for precipitation. Each point is colored based on the SHAP value attributed to a feature.

4 Conclusions

Prediction of different aspects of fire regime and understanding the relationships between fire and its predictors have been improved using different types of machine learning models. Advanced types of neural networks such as LSTM have not been widely studied for predicting fire dynamics. The first objective of this thesis is to explore the opportunities and limitations of using LSTM neural networks to predict fire occurrence and capture the fire-predictor relationships. The selected methodology is based on using a pixel-based LSTM with five meteorological predictors and fAPAR as an indicator for vegetation. The availability of data and the type of the neural network have imposed many limitations. The inclusion of all important fire drivers as recommended in the literature was not possible. LSTM requires all the variables to be as a time series covering the same study period to avoid bias results. Based on the available data, the study covered 14 years of monthly data or 168 time steps. All datasets were aggregated and resampled to 0.25 deg latitude x 0.25 deg longitude grid then merged into one NetCDF file. Data pre-processing involved filling the missing values, removing correlated variables and data enhancement by applying the log transformation to mitigate data skewness. To prepare the data for LSTM, each pixel was structured as a multivariate time series then split into train and test sets. LSTM architecture was selected based on multiple experiments performed by testing different hyperparameters. The chosen architecture is a vanilla LSTM with one hidden layer which takes the previous twelve time steps to make a prediction.

The pixel-based LSTM captured the seasonal and spatial varieties with RMSE value computed at 3.333 for the entire study area. The error is calculated as the mean RMSE for all pixels. This has shown that the LSTM-based machine learning model has produced acceptable results with few input predictors. LSTM has a big potential for modeling fire dynamics with its ability to remember past events for a long time. On the other hand, LSTM underestimated the high values of ignitions during the peak of fire season. In many cases, LSTM predicted no fire when a sudden uprise in the last fire season occurs leading to major differences and producing low R^2 values. In this case, LSTM was not able to capture the extreme values and performed better during the months of lower fire occurrence. One reason behind this could be the limited length of the time series. Since each pixel is trained independently, the length of the time series is considered rather short. Machine learning techniques require large amounts of data for training and testing in order to learn efficiently.

For the second objective, permutation feature importance, variance-based feature importance and SHAP were applied to explore their abilities to determine the importance of each predictor for fire ignitions and visualize the fire-predictor relationships. LSTM 3D data format which considers previous time steps to make a prediction have imposed many

technical restrictions. No general approach was able to visualize local and general feature importance. For general interpretation, permutation feature importance gave an overview of the most important variables for the entire study area. Using variance-based feature importance, the spatial distribution of each feature was mapped. SHAP summary plots are suitable to give more details and depict feature importance for each precedent time step.

The most important features to predict fire ignitions were mainly fAPAR, precipitation and maximum temperature. The order of importance for other variables differs based on location and precedent month. In the south and middle regions, precedent year conditions (preceding 11^{th} and 12^{th} months) had a higher effect than recent conditions for all variables. Whereas in the north, recent conditions < 4 months were found more important especially for fAPAR, pre and NI. The Nesterov Index had an impact on fire ignitions only in the north and middle regions for the preceding six months and further, meanwhile, it exhibited no influence in the south. The impact of DTR and wind decreased gradually from south to north.

To visualize LSTM inner relationships and interactions, SHAP dependence plots are advisable for feature-output relationships. Interpreting LSTM neural network showed that LSTM was able to model the fire-predictor relationship correctly only for precipitation, DTR and fAPAR (Figure 30). Whereas, for maximum temperature and wind the relationship was vague. The higher and lower values both affected the model in a negative and positive way depicting a straight line, therefore, it is unclear how any changes in these features would affect fire ignitions. The nesterov index did not play a major role for LSTM and no clear relationship was concluded from the model. For feature interactions, a 3D extension of SHAP dependence plot with added color visual variable was found to be the best visualization technique. SHAP helps with understanding how one feature is affecting the other inside the model but this relationship cannot be depicted directly.

For further developments, these results can be improved in many ways. For pixel-based LSTM, a longer time series is preferrable, as machine learning models require so much data to learn efficiently. For example, daily data could be used instead of monthly but this is limited by datasets availability and by which fire attribute is studied. For fire ignition count, to the author's best knowledge, the Fire Atlas (Andela et al., 2019) is the only available dataset that provides 14 years of continuous monthly fire ignition data, and this confined data usage to monthly. More advanced types of LSTM could be used for further studies such as Attention-LSTM neural networks. This type of neural networks mechanisms allow the model to assign larger weights to specific time steps and important features while training. After specifying which variable dominates at which time step (3.3), attention can be focused on these features to develop the network and investigate any improvement in its performance. The main limitation of using LSTM in fire modeling is that spatial information is fundamental for many fire attributes. Prediction of fire ignition points was possible in this thesis but for predicting fire spread or burned areas, the spatial

connectedness is essential. Reading the spatial information with convolutional neural networks and then passing the information to LSTM is one of the most common methods used in research and, in the author's opinion, it could improve modeling fire dynamics substantially.

For more understandable LSTM visualization, using one comprehensive model facilitates implementing different visualization techniques. The pixel-based model has imposed many technical and computational limitations.

Visualization techniques have contributed to better understanding of the machine learning model and presented useful insights for further developments. This study contributes to the environmental remote sensing field that focuses on modeling wildfire as a part of the Earth system. It also addresses the importance of appropriate data visualization techniques to increase the trust in machine learning models and encourage their applications in Cartography.

References

Albini, F. A. (1976). Estimating wildfire behaviour and effects. *Ogden, Uttah: USDA Forest Service, Intermountain Forest and Range Experiment Station.*, *Report No. INT-30.*

Aldersley, A., Murray, S. J., & Cornell, S. E. (2011). Global and regional analysis of climate and human drivers of wildfire. *Science of the Total Environment*, *409*(18), 3472–3481. https://doi.org/10.1016/j.scitotenv.2011.05.032

Alin, A. (2010). Multicollinearity. *Wiley Interdisciplinary Reviews: Computational Statistics*, *2*(3), 370–374. https://doi.org/10.1002/wics.84

Andela, N., Morton, D. C., Giglio, L., Chen, Y., Van Der Werf, G. R., Kasibhatla, P. S., DeFries, R. S., Collatz, G. J., Hantson, S., Kloster, S., Bachelet, D., Forrest, M., Lasslop, G., Li, F., Mangeon, S., Melton, J. R., Yue, C., & Randerson, J. T. (2017). A human-driven decline in global burned area. *Science*, *356*(6345), 1356–1362. https://doi.org/10.1126/science.aal4108

Andela, N., Morton, D. C., Giglio, L., Paugam, R., Chen, Y., Hantson, S., Van Der Werf, G. R., & Anderson, J. T. (2019). The Global Fire Atlas of individual fire size, duration, speed and direction. *Earth System Science Data*, *11*(2), 529–552. https://doi.org/10.5194/essd-11-529-2019

Archibald, S., Roy, D. P., van Wilgen, B. W., & Scholes, R. J. (2009). What limits fire? An examination of drivers of burnt area in Southern Africa. *Global Change Biology*, *15*(3), 613–630. https://doi.org/10.1111/j.1365-2486.2008.01754.x

Arora, V. K., & Boer, G. J. (2005). Fire as an interactive component of dynamic vegetation models. *Journal of Geophysical Research: Biogeosciences*, *110*(G2), n/a-n/a. https://doi.org/10.1029/2005jg000042

Arslan, N., & Sekertekin, A. (2019). Application of Long Short-Term Memory neural network model for the reconstruction of MODIS Land Surface Temperature images. *Journal of Atmospheric and Solar-Terrestrial Physics*, *194*(August), 105100. https://doi.org/10.1016/j.jastp.2019.105100

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K. R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, *10*(7), 1–46. https://doi.org/10.1371/journal.pone.0130140

Besnard, S., Carvalhais, N., Altaf Arain, M., Black, A., Brede, B., Buchmann, N., Chen, J., Clevers, J. G. P. W., Dutrieux, L. P., Gans, F., Herold, M., Jung, M., Kosugi, Y., Knohl, A., Law, B. E., Paul-Limoges, E., Lohila, A., Merbold, L., Roupsard, O., … Reichstein, M. (2019). Memory effects of climate and vegetation affecting net ecosystem CO2 fluxes in global forests. *PLoS ONE*, *14*(2), 1–22. https://doi.org/10.1371/journal.pone.0211510

Bistinas, I., Harrison, S. P., Prentice, I. C., & Pereira, J. M. C. (2014). Causal relationships vs. emergent patterns in the global controls of fire frequency. *Biogeosciences Discussions*, *11*(3), 3865–3892. https://doi.org/10.5194/bgd-11-3865-2014

Bowman, D. M. J. S., Balch, J. K., Artaxo, P., Bond, W. J., Carlson, J. M., Cochrane, M. A., D'Antonio, C. M., DeFries, R. S., Doyle, J. C., Harrison, S. P., Johnston, F. H., Keeley, J. E., Krawchuk, M. A., Kull, C. A., Marston, J. B., Moritz, M. A., Prentice, I. C., Roos, C. I., Scott, A. C., … Pyne, S. J. (2009). Fire in the earth system. *Science*, *324*(5926), 481–484. https://doi.org/10.1126/science.1163886

Box, G. E., & Cox, D. R. (1964). An analysis of transformations. *Ournal of the Royal Statistical Society*, *26.2*(Series B (Methodological)), 211–243. https://doi.org/https://doi.org/10.1111/j.2517-6161.1964.tb00553.x

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*, 5–32. https://doi.org/https://doi.org/10.1023/A:1010933404324

Brownlee, J. (2020). *Long Short-Term Memory Networks With Python Develop Sequence Prediction Models With Deep Learning*.

Chen, Z., Ma, X., Yu, W., & Wu, L. (2020). An integrated graph Laplacian downsample (IGLD)-based method for DEM generalization. *Earth Science Informatics*, *13*(4), 973–987. https://doi.org/10.1007/s12145-020-00482-5

Chiang, Y. Y., Duan, W., Leyk, S., Uhl, J. H., & Knoblock, C. A. (2020). Historical map applications and processing technologies. In *Using historical maps in scientific studies* (pp. 9–36). Springer, Cham.

Chuvieco, E., Aguado, I., Yebra, M., Nieto, H., Salas, J., Martín, M. P., Vilar, L., Martínez, J., Martín, S., Ibarra, P., de la Riva, J., Baeza, J., Rodríguez, F., Molina, J. R., Herrera, M. A., & Zamora, R. (2010). Development of a framework for fire risk assessment using remote sensing and geographic information system technologies. *Ecological Modelling*, *221*(1), 46–58. https://doi.org/10.1016/j.ecolmodel.2008.11.017

Cortez, P., & Morais, A. (2007). A Data Mining Approach to Predict Forest Fires using Meteorological Data. *Proceedings of 13th Portugese Conference on Artificial Intelligence*, 512–523. http://www.dsi.uminho.pt/~pcortez/fires.pdf

Daoud, J. I. (2018). Multicollinearity and Regression Analysis. *Journal of Physics: Conference Series*, *949*(1). https://doi.org/10.1088/1742-6596/949/1/012009

de Sá, C. R. (2019). Variance-Based Feature Importance in Neural Networks. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11828 LNAI*(October), 306–315. https://doi.org/10.1007/978-3-030-33778-0_24

Donald, F., & Glauber, R. (1967). Multicollinearity in Regression Analysis : The Problem Revisited. *The Review of Economic and Statistics*, *49*(1), 92–107.

https://www.jstor.org/stable/1937887

Duan, W., Knoblock, C. A., Feldman, D., Uhl, J. H., & Leyk, S. (2017). Automatic Alignment of Geographic Features in Contemporary Vector Data and Historical Maps. *GeoAI '17 Proceedings of the 1st Workshop on Artificial Intelligence and Deep Learning for Geographic Knowledge Discovery*, 45–54.

Fisher, A., Rudin, C., & Dominici, F. (2019). All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *Journal of Machine Learning Research*, *20*, 1–81. http://jmlr.org/papers/v20/18-760.html.

Forkel, M., Andela, N., P Harrison, S., Lasslop, G., Van Marle, M., Chuvieco, E., Dorigo, W., Forrest, M., Hantson, S., Heil, A., Li, F., Melton, J., Sitch, S., Yue, C., & Arneth, A. (2019a). Emergent relationships with respect to burned area in global satellite observations and fire-enabled vegetation models. *Biogeosciences*, *16*(1), 57–76. https://doi.org/10.5194/bg-16-57-2019

Forkel, M., Dorigo, W., Lasslop, G., Chuvieco, E., Hantson, S., Heil, A., Teubner, I., Thonicke, K., & Harrison, S. P. (2019b). Recent global and regional trends in burned area and their compensating environmental controls. *Environmental Research Communications*, *1*(5), 051005. https://doi.org/10.1088/2515-7620/ab25d2

García, S., Luengo, J., & Herrera, F. (2015). Data Preprocessing in Data Mining. In *Intelligent Systems Reference Library* (Vol. 72).

Gers, F., Eck, D., & Jürgenr, S. (2001). Applying LSTM to Time Series Predictable through Time-Window Approaches. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *2130*(August 2001), v. https://doi.org/10.1007/3-540-44668-0

Giles, C. L., & Lawrence, S. (2001). *Noisy Time Series Prediction using Recurrent Neural Networks and Grammatical Inference* (Vol. 44). http://www.stern.nyu.edu/

Goodfellow, I., Bengio, Y., & Courville, A. (2017). *Deep Learning*. 800.

Gou, Y., Zhang, T., Liu, J., Wei, L., & Cui, J. H. (2020). DeepOcean: A general deep learning framework for spatio-temporal ocean sensing data prediction. *IEEE Access*, *8*, 79192–79202. https://doi.org/10.1109/ACCESS.2020.2990939

Goyal, P., Pandey, S., & Jain, K. (2015). *Deep Learning for Natural Language Processing*. https://doi.org/10.1007/978-1-4842-3685-7

Guo, F., Zhang, L., Jin, S., Tigabu, M., Su, Z., & Wang, W. (2016). Modeling anthropogenic fire occurrence in the boreal forest of China using logistic regression and random forests. *Forests*, *7*(11), 1–14. https://doi.org/10.3390/f7110250

Han, M., Xi, J., Xu, S., & Yin, F. (2004). *Prediction of Chaotic Time Series Based on the*

Recurrent Predictor Neural Network. *52*(12), 3409–3416.

Hantson, S., Arneth, A., Harrison, S. P., Kelley, D. I., Prentice, I. C., Rabin, S. S., Archibald, S., Mouillot, F., Arnold, S. R., Artaxo, P., Bachelet, D., Ciais, P., Forrest, M., Friedlingstein, P., Hickler, T., Kaplan, J. O., Kloster, S., Knorr, W., Lasslop, G., … Yue, C. (2016). The status and challenge of global fire modelling. *Biogeosciences Discussions*, *2016*(January), 1–30. https://doi.org/10.5194/bg-2016-17

Hantson, S., Kelley, D. I., Arneth, A., Harrison, S. P., Archibald, S., Bachelet, D., Forrest, M., Hickler, T., Lasslop, G., Li, F., Mangeon, S., Melton, J. R., Nieradzik, L., Rabin, S. S., Colin Prentice, I., Sheehan, T., Sitch, S., Teckentrup, L., Voulgarakis, A., & Yue, C. (2020). Quantitative assessment of fire and vegetation properties in simulations with fire-enabled vegetation models from the Fire Model Intercomparison Project. *Geoscientific Model Development*, *13*(7), 3299–3318. https://doi.org/10.5194/gmd-13-3299-2020

Harris, I. C. (2019). CRU JRA v2.0: A forcings dataset of gridded land surface blend of Climatic Research Unit (CRU) and Japanese reanalysis (JRA) data. *Centre for Environmental Data Analysis*. https://catalogue.ceda.ac.uk/uuid/7f785c0e80aa4df2b39d068ce7351bbb

Harrison, S. P., Marlon, J. R., & Bartlein, P. J. (2010). Fire in the Earth System. In J. Dodson (Ed.), *Changing Climates, Earth Systems and Society* (pp. 21–48). Springer Netherlands. https://doi.org/10.1007/978-90-481-8716-4_3

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hua, Y., Zhao, Z., Li, R., Chen, X., Liu, Z., & Zhang, H. (2019). Deep Learning with Long Short-Term Memory for Time Series Prediction. *IEEE Communications Magazine*, *57*(6), 114–119. https://doi.org/10.1109/MCOM.2019.1800155

Jafari Goldarag, Y., Mohammadzadeh, A., & Ardakani, A. S. (2016). Fire Risk Assessment Using Neural Network and Logistic Regression. *Journal of the Indian Society of Remote Sensing*, *44*(6), 885–894. https://doi.org/10.1007/s12524-016-0557-6

Joshi, J., & Sukumar, R. (2021). Improving prediction and assessment of global fires using multilayer neural networks. *Scientific Reports*, *11*(1), 1–14. https://doi.org/10.1038/s41598-021-81233-4

Kaspar, F., Schulzweida, U., & Müller, R. (2010). *" Climate data operators " as a user-friendly processing tool for CM SAF ' s satellite-derived climate monitoring products.* https://doi.org/10.13140/RG.2.2.20422.68165

Khodayar, M., Wang, J., & Manthouri, M. (2019). Interval Deep Generative Neural Network for Wind Speed Forecasting. *IEEE Transactions on Smart Grid*, *10*(4), 3974–3989. https://doi.org/10.1109/TSG.2018.2847223

Kim, S. J., Lim, C. H., Kim, G. S., Lee, J., Geiger, T., Rahmati, O., Son, Y., & Lee, W. K. (2019). Multi-temporal analysis of forest fire probability using socio-economic and environmental variables. *Remote Sensing*, *11*(1). https://doi.org/10.3390/rs11010086

Knorr, W., Kaminski, T., Arneth, A., & Weber, U. (2014). Impact of human population density on fire frequency at the global scale. *Biogeosciences*, *11*(4), 1085–1102. https://doi.org/10.5194/bg-11-1085-2014

Kong, Y. L., Huang, Q., Wang, C., Chen, J., Chen, J., & He, D. (2018). Long short-term memory neural networks for online disturbance detection in satellite image time series. *Remote Sensing*, *10*(3). https://doi.org/10.3390/rs10030452

Kubben, P., Dumontier, M., & Dekker, A. (2020). Fundamentals of Clinical Data Science. In *The Routledge Companion to Media and Fairy-Tale Cultures*. https://doi.org/10.4324/9781315670997-60

Kuhn-Régnier, A., Voulgarakis, A., Nowack, P., Forkel, M., Prentice, I. C., & Harrison, S. P. (2020). Quantifying the Importance of Antecedent Fuel-Related Vegetation Properties for Burnt Area using Random Forests. *Biogeosciences Discussions*, *November*, 1–24.

Kumar, A., Islam, T., Sekimoto, Y., Mattmann, C., & Wilson, B. (2020). ConvCast: An embedded convolutional LSTM based architecture for precipitation nowcasting using satellite data. *PLoS ONE*, *15*(3). https://doi.org/10.1371/journal.pone.0230114

Kumar, U. A. (2005). Comparison of neural networks and regression analysis: A new insight. *Expert Systems with Applications*, *29*(2), 424–430. https://doi.org/10.1016/j.eswa.2005.04.034

Larasati, A., Hajji, A. M., & Dwiastuti, A. (2019). The relationship between data skewness and accuracy of Artificial Neural Network predictive model. *IOP Conference Series: Materials Science and Engineering*, *523*(1). https://doi.org/10.1088/1757-899X/523/1/012070

Lasslop, G., Coppola, A. I., Voulgarakis, A., Yue, C., & Veraverbeke, S. (2019). Influence of Fire on the Carbon Cycle and Climate. *Current Climate Change Reports*, *5*(2), 112–123. https://doi.org/10.1007/s40641-019-00128-9

Lehsten, V., Harmand, P., Palumbo, I., & Arneth, A. (2010). Modelling burned area in Africa. *Biogeosciences*, *7*(10), 3199–3214. https://doi.org/10.5194/bg-7-3199-2010

Levis, S., Bonan, G. B., Vertenstein, M., & Oleson, K. W. (2004). The Community Land Model's dynamic global vegetation model (CLM-DGVM): Technical description and user's guide. *NCAR Tech. Note TN-459+ IA 50*.

Li, F., Levis, S., & Ward, D. S. (2013). Quantifying the role of fire in the Earth system - Part 1: Improved global fire modeling in the Community Earth System Model (CESM1). *Biogeosciences*, *10*(4), 2293–2314. https://doi.org/10.5194/bg-10-2293-2013

Li, F., Zeng, X. D., & Levis, S. (2012). A process-based fire parameterization of intermediate complexity in a dynamic global vegetation model. *Biogeosciences*, *9*(7), 2761–2780. https://doi.org/10.5194/bg-9-2761-2012

Liang, H., Zhang, M., & Wang, H. (2019). A Neural Network Model for Wildfire Scale Prediction Using Meteorological Factors. *IEEE Access*, *7*, 176746–176755. https://doi.org/10.1109/ACCESS.2019.2957837

Lipovetsky, S., & Conklin, M. (2001). Analysis of regression in game theory approach. *Applied Stochastic Models in Business and Industry*, *17*(4), 319–330. https://doi.org/10.1002/asmb.446

Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. *31st International Conference on Neural Information Processing Systems*, 4768–4777.

Maeda, E. E., Formaggio, A. R., Shimabukuro, Y. E., Arcoverde, G. F. B., & Hansen, M. C. (2009). Predicting forest fire in the Brazilian Amazon using MODIS imagery and artificial neural networks. *International Journal of Applied Earth Observation and Geoinformation*, *11*(4), 265–272. https://doi.org/10.1016/j.jag.2009.03.003

Mbow, C. (2017). The Great Green Wall in the Sahel. *Oxford Research Encyclopedia of Climate Science*, *August 2017*, 1–24. https://doi.org/10.1093/acrefore/9780190228620.013.559

Melton, J. R., & Arora, V. K. (2016). Competition between plant functional types in the Canadian Terrestrial Ecosystem Model (CTEM) v. 2.0. *Geoscientific Model Development*, *9*(1), 323–361. https://doi.org/10.5194/gmd-9-323-2016

Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, *267*, 1–38. https://doi.org/10.1016/j.artint.2018.07.007

Molnar, C. (2020). *Interpretable Machine Learning*. Lulu.com.

Moskolaï, W. R., Abdou, W., Dipanda, A., & Kolyang, D. T. (2020). Application of LSTM architectures for next frame forecasting in Sentinel-1 images time series. *ArXiv*, 13.

Myneni, R., Knyazikhin, Y., & Park, T. (2015). MOD15A2H MODIS/Terra Leaf Area Index/FPAR 8-Day L4 Global 500m SIN Grid V006 [Data set]. *NASA EOSDIS Land Processes DAAC*. https://doi.org/https://doi.org/10.5067/MODIS/MOD15A2H.006

Nesterov, V. G. (1949). Flammability of the Forest and Methods for Its Determination (Gorimost lesa i metodi eio opredelenia) (in Russian). *USSR State Ind. Press, Moscow*.

Özbayoğlu, A. M., & Bozer, R. (2012). Estimation of the burned area in forest fires using computational intelligence techniques. *Procedia Computer Science*, *12*, 282–287. https://doi.org/10.1016/j.procs.2012.09.070

Pechony, O., & Shindell, D. T. (2009). Fire parameterization on a global scale. *Journal of Geophysical Research Atmospheres*, *114*(16), 1–10.

https://doi.org/10.1029/2009JD011927

Perumal, R., & van Zyl, T. L. (2020). Comparison of Recurrent Neural Network Architectures for Wildfire Spread Modelling. *ArXiv*. https://doi.org/10.1109/SAUPEC/RobMech/PRASA48453.2020.9078028

Pfeiffer, M., Spessa, A., & Kaplan, J. O. (2013). A model for global biomass burning in preindustrial time: LPJ-LMfire (v1.0). *Geoscientific Model Development*, *6*(3), 643–685. https://doi.org/10.5194/gmd-6-643-2013

Rabin, S. S., Melton, J. R., Lasslop, G., Bachelet, D., Forrest, M., Hantson, S., Kaplan, J. O., Li, F., Mangeon, S., Ward, D. S., Yue, C., Arora, V. K., Hickler, T., Kloster, S., Knorr, W., Nieradzik, L., Spessa, A., Folberth, G. A., Sheehan, T., … Arneth, A. (2017). The Fire Modeling Intercomparison Project (FireMIP), phase 1: Experimental and analytical protocols with detailed model descriptions. *Geoscientific Model Development*, *10*(3), 1175–1197. https://doi.org/10.5194/gmd-10-1175-2017

Reick, C. H., Raddatz, T., Brovkin, V., & Gayler, V. (2013). Representation of natural and anthropogenic land cover change in MPI-ESM. *Journal of Advances in Modeling Earth Systems*, *5*(3), 459–482. https://doi.org/10.1002/jame.20022

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should i trust you?" Explaining the predictions of any classifier. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *13-17-Augu*, 1135–1144. https://doi.org/10.1145/2939672.2939778

Ruder, S. (2016). An overview of gradient descent optimization algorithms. *Insight Centre for Data Analytics, NUI Galway*. http://caffe.berkeleyvision.org/tutorial/solver.html

Sanderson, B., & Fisher, R. (2020). A fiery wake-up call for climate science. *Nature Climate Change*, *10*(3), 173–174. https://doi.org/10.1038/s41558-020-0712-5

Sapankevych, N., & Sankar, R. (2009). Time series prediction using support vector machines: A survey. *IEEE Computational Intelligence Magazine*, *4*(2), 24–38. https://doi.org/10.1109/MCI.2009.932254

Satir, O., Berberoglu, S., & Donmez, C. (2016). Mapping regional forest fire probability using artificial neural network model in a Mediterranean forest ecosystem. *Geomatics, Natural Hazards and Risk*, *7*(5), 1645–1658. https://doi.org/10.1080/19475705.2015.1084541

Schnürer, R., Sieber, R., Schmid-Lanter, J., Öztireli, A. C., & Hurni, L. (2021). Detection of Pictorial Map Objects with Convolutional Neural Networks. *Cartographic Journal*, *58*(1), 50–68. https://doi.org/10.1080/00087041.2020.1738112

Scott, A. C., & Glasspool, I. J. (2006). The diversification of Paleozoic fire systems and fluctuations in atmospheric oxygen concentration. In *PNAS* (Vol. 103). www.pnas.orgcgidoi10.1073pnas.0604090103

Shapley, L. S. (1953). A Value for n-Person Games. In *Contributions to the Theory of Games 2.28* (pp. 307–318). Princeton University Press. https://doi.org/https://doi.org/10.1515/9781400881970-018

Shrestha, N. (2020). Detecting Multicollinearity in Regression Analysis. *American Journal of Applied Mathematics and Statistics*, *8*(2), 39–42. https://doi.org/10.12691/ajams-8-2-1

Shrikumar, A., Greenside, P., & Kundaje, A. (2017). Learning important features through propagating activation differences. *34th International Conference on Machine Learning, ICML 2017*, *7*, 4844–4866.

Sitch, S., Smith, B., Prentice, I. C., Arneth, A., Bondeau, A., Cramer, W., Kaplan, J. O., Levis, S., Lucht, W., Sykes, M. T., Thonicke, K., & Venevsky, S. (2003). Evaluation of ecosystem dynamics, plant geography and terrestrial carbon cycling in the LPJ dynamic global vegetation model. *Global Change Biology*, 161–185.

Song, Y., & Wang, Y. (2020). Global wildfire outlook forecast with neural networks. *Remote Sensing*, *12*(14). https://doi.org/10.3390/rs12142246

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, *15*(56), 1929–1958. https://doi.org/10.1016/0370-2693(93)90272-J

Štrumbelj, E., & Kononenko, I. (2014). Explaining prediction models and individual predictions with feature contributions. *Knowledge and Information Systems*, *41*(3), 647–665. https://doi.org/10.1007/s10115-013-0679-x

Taylor, A., Leblanc, S., & Japkowicz, N. (2016). Anomaly detection in automobile control network data with long short-term memory networks. *Proceedings - 3rd IEEE International Conference on Data Science and Advanced Analytics, DSAA 2016*, 130–139. https://doi.org/10.1109/DSAA.2016.20

Thonicke, K., Spessa, A., Prentice, I. C., Harrison, S. P., Dong, L., & Carmona-Moreno, C. (2010). The influence of vegetation, fire spread and fire behaviour on biomass burning and trace gas emissions: Results from a process-based model. *Biogeosciences*, *7*(6), 1991–2011. https://doi.org/10.5194/bg-7-1991-2010

Thonicke, K., Venevsky, S., Sitch, S., & Cramer, W. (2001). The role of fire disturbance for global vegetation dynamics: Coupling fire into a dynamic global vegetation model. *Global Ecology and Biogeography*, *10*(6), 661–677. https://doi.org/10.1046/j.1466-822X.2001.00175.x

Torres, R. N., Fraternali, P., Milani, F., & Frajberg, D. (2020). Mountain summit detection with Deep Learning: evaluation and comparison with heuristic methods. *Applied Geomatics*, *12*(2), 225–246. https://doi.org/10.1007/s12518-019-00295-2

Uhl, J. H., Leyk, S., Chiang, Y. Y., Duan, W., & Knoblock, C. A. (2020). Automated extraction of human settlement patterns from historical topographic map series using weakly

supervised convolutional neural networks. *IEEE Access*, *8*, 6978–6996. https://doi.org/10.1109/ACCESS.2019.2963213

Van Houdt, G., Mosquera, C., & Nápoles, G. (2020). A review on the long short-term memory model. *Artificial Intelligence Review*, *53*(8), 5929–5955. https://doi.org/10.1007/s10462-020-09838-1

Venevsky, S., Thonicke, K., Sitch, S., & Cramer, W. (2002). Simulating fire regimes in human-dominated ecosystems: Iberian Peninsula case study. *Global Change Biology*, *8*(10), 984–998. https://doi.org/10.1046/j.1365-2486.2002.00528.x

Viegas, D. X. (1997). Modelling surface-fire behaviour. *Forest Fire Risk and Management*, 65–80.

Voulgarakis, A., & Field, R. D. (2015). Fire Influences on Atmospheric Composition, Air Quality and Climate. *Current Pollution Reports*, *1*(2), 70–81. https://doi.org/10.1007/s40726-015-0007-z

W. Köppen. (1936). Handbuch der Klimatologie. *Das Geographische System Der Klimate*, *43*(12), 935. https://doi.org/10.2307/200498

Wang, Y., Long, M., Wang, J., Gao, Z., & Yu, P. S. (2017). PredRNN: Recurrent neural networks for predictive learning using spatiotemporal LSTMs. *Advances in Neural Information Processing Systems*, *2017-Decem*(Nips), 880–889.

Wang, Y., Zhang, J., Zhu, H., Long, M., Wang, J., & Yu, P. S. (2019). Memory in memory: A predictive neural network for learning higher-order non-stationarity from spatiotemporal dynamics. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2019-June*, 9146–9154. https://doi.org/10.1109/CVPR.2019.00937

Wilkening, J. (2019). Towards Spatial Data Science: Bridging the Gap between GIS, Cartography and Data Science. *Abstracts of the ICA*, *1*(403), 1–2. https://doi.org/10.5194/ica-abs-1-403-2019

Xiao, Y., Yin, H., Zhang, Y., Qi, H., Zhang, Y., & Liu, Z. (2021). A dual-stage attention-based Conv-LSTM network for spatio-temporal correlation and multivariate time series prediction. *International Journal of Intelligent Systems*, *36*(5), 2036–2057. https://doi.org/10.1002/int.22370

Yan, X., Ai, T., Yang, M., & Yin, H. (2019). A graph convolutional neural network for classification of building patterns using spatial vector data. *ISPRS Journal of Photogrammetry and Remote Sensing*, *150*(March), 259–273. https://doi.org/10.1016/j.isprsjprs.2019.02.010

Yang, Z., Gui, Z., Wu, H., & Li, W. (2020). A Latent Feature-Based Multimodality Fusion Method for Theme Classification on Web Map Service. *IEEE Access*, *8*, 25299–25309. https://doi.org/10.1109/ACCESS.2019.2954851

YEO, I.-K., & JOHNSON, R. A. (2000). *A new family of power transformations to improve normality or symmetry*. *2*, 991–994. https://doi.org/https://doi.org/10.1093/biomet/87.4.954

Yue, C., Ciais, P., Cadule, P., Thonicke, K., Archibald, S., Poulter, B., Hao, W. M., Hantson, S., Mouillot, F., Friedlingstein, P., Maignan, F., & Viovy, N. (2014). Modelling the role of fires in the terrestrial carbon balance by incorporating SPITFIRE into the global vegetation model ORCHIDEE - Part 1: Simulating historical global burned area and fire regimes. *Geoscientific Model Development*, *7*(6), 2747–2767. https://doi.org/10.5194/gmd-7-2747-2014

Zhou, X., Li, W., Arundel, S. T., & Liu, J. (2018). *Deep Convolutional Neural Networks for Map-Type Classification*. http://arxiv.org/abs/1805.10402

Zou, Q., Ni, L., Zhang, T., & Wang, Q. (2015). Remote Sensing Scene Classification. *IEEE Transactions on Geoscience and Remote Sensing Letters*, *12*(11), 2321–2325.