Karlsruhe Institute of Technology

AI Climate

Designing and Explaining Temporal Deep Learning Models for Wildfire Danger Prediction

—

Bachelor's Thesis

Pauline Becker

2476877

At the Institute of Theoretical Informatics ITI

Chair for AI in Climate and Environmental Sciences

Reviewer: Prof. Dr. Peer Nowack, Prof. Dr. Pascal Friederich

Supervisor: Prof. Dr. Peer Nowack, Dr. Carolina Natel

07.08.2025

Abstract

Wildfires in the Mediterranean region have become more frequent and intense in recent years, closely linked to the effects of climate change. Accurately predicting areas at risk is essential for supporting land and forest management, improving preparedness, and mitigating wildfire impacts. In this work, we address this challenge by developing seven temporal Deep Learning (DL) architectures and a Random Forest baseline to forecast next-day wildfire danger in this highly fire-prone region.

We used the Mesogeos Datacube (Kondylatos, Prapas, Camps-Valls, & Papoutsis, 2023), a comprehensive wildfire dataset integrating meteorological, vegetation, topographical, and human-related variables at a 1 km $\times$ 1 km daily resolution from 2006 to 2022. As a key contribution, we apply multiple Explainable Artificial Intelligence (XAI) techniques to reveal the models internal decision-making.

Our results demonstrate that all DL models achieved strong and comparable predictive performance (F_1-Score > 0.8), significantly outperforming the Random Forest baseline. SHapley Additive exPlanations (SHAP) and Integrated Gradients (IG) consistently highlighted temperature, humidity, precipitation, and vegetation indices as key contributors to model predictions, largely consistent across all models. We assessed each models physical interpretability using a consistency score, quantifying how well SHAP-derived attributions align with established firedriver relationships and conducted a detailed timelag analysis to investigate the impact of extended temporal contexts on predictive performance. While attention-based models achieved higher consistency scores and effectively leveraged extended temporal contexts, recurrent models plateaued or degraded with longer input sequences, and Random Forest exhibited the highest consistency despite its lower predictive performance, underscoring the trade-off between accuracy and physical alignment.

Exemplary for the case studies, two major Spanish fire events occurring at nearly the same time and location revealed contrasting patterns: one linked to an extreme heatwave and drought that the model explained well, and a false negative associated with lightning-driven ignition, which proved more challenging to predict. Finally, we observed a pronounced geographical imbalance in the dataset, with wildfire events concentrated in coastal regions and negative samples dominating inland areas, which challenges model generalization in underrepresented inland fire scenarios.

These findings underscore the potential of temporal DL models combined with XAI to support accurate and reliable wildfire danger forecasting in fire-prone regions.

Acknowledgments

This work was performed on the HoreKa supercomputer funded by the Ministry of Science, Research and Arts Baden-Württemberg and by the Federal Ministry of Education and Research.

I gratefully acknowledge the support of OpenAI, Perplexity, and SciSpace which were helpful in refining my English writing and providing useful code examples contributing significantly to the clarity and precision of this thesis.

In addition, I would like to express my sincere gratitude to Peer Nowack and Carolina Natel for their invaluable guidance and continued support throughout the course of this thesis. Their expertise, encouragement, and insightful feedback have been instrumental in shaping the direction and quality of this work.

I am also sincerely thankful to Lina Rennstein for her patience and support in answering my questions about HoreKa, reviewing my thesis, and providing feedback on the UI design my plots.

Finally, I would like to thank my family, friends, and my partner Julian for their unwavering support, understanding, and encouragement. Their presence throughout this journey has meant the world to me, and I am truly grateful for their constant belief in me.

Contents

List of Figures

List of Tables

Acronyms

AI Artificial Intelligence.

ALE Accumulated Local Effects.

ANOVA Analysis of Variance.

AUPRC Area Under the Precision-Recall Curve.

CNN Convolutional Neural Network.

COP-DEM Copernicus Global Digital Elevation Model.

CSV Comma-separated values - Data format.

CV Cross-Validation.

d2m Dewpoint temperature at 2m.

dem Elevation.

DL Deep Learning.

EDO European Drought Observatory.

EFFIS European Forest Fire Information System.

FFN Feedforward Neural Network.

FN False Negative.

FP False Positive.

FWT Fire-Weather Type.

GRN Gated Residual Network.

GRU Gated Recurrent Unit.

GTN Gated Transformer Network.

IG Integrated Gradients.

ITI Institute of Theoretical Informatics.

lai Leaf Area Index.

lc_agriculture Land cover class: agriculture.

lc_forest Land cover class: forest.

lc_grassland Land cover class: grassland.

lc_settlement Land cover class: settlement.

lc_shrubland Land cover class: shrubland.

lc_water_bodies Land cover class: water bodies.

lc_wetland Land cover class: wetland.

lst_day Day land surface temperature.

lst_night Night land surface temperature.

LSTM Long Short-Term Memory.

MDI Mean Decrease in Impurity.

ML Machine Learning.

MLP Multilayer Perceptron.

MODIS Moderate Resolution Imaging Spectroradiometer.

MSE Mean Squared Error.

ndvi Normalized Difference Vegetation Index.

OOB Out-of-Bag.

PDP Partial Dependence Plot.

PR Precision.

PR-RE Precision-Recall.

REC Recall.

RF Random Forest.

rh Relative Humidity.

RNN Recurrent Neuronal Network.

roads_distance Distance from Roads.

SHAP SHapley Additive exPlanations.

slope Slope of the Area.

smi Soil Moisture Index.

sp Surface Pressure.

ssrd Surface Solar Radiation Downward.

t2m Temperature at 2m above the Earths surface.

TFT Temporal Fusion Transformer.

TN True Negative.

TP True Positive.

tp Total Precipitation.

VPD Vapor-Pressure Deficit.

wind_speed Wind Speed.

XAI Explainable Artificial Intelligence.

1. Introduction

Historically, wildfires have long been considered a carbon neutral process on extended time scales, as vegetation regrowth can eventually offset carbon emissions from combustion (M. W. Jones et al., 2022; Prapas, Ahuja, et al., 2023). However, anthropogenic climate change is increasingly disrupting this balance, driving more frequent, intense, and widespread fire events worldwide. With each incremental rise in global temperature, conditions favorable to fire ignition and spread are projected to intensify (M. W. Jones et al., 2022). For instance, in Mediterranean Europe, mean seasonal fire danger is expected to increase by 2–4% per decade under high-emission scenarios, expanding fire risk to regions such as western and central France where fuel availability will likely not be a limiting factor (Dupuy et al., 2020).

Fires also interact with the climate system through several feedback mechanisms. They directly reduce terrestrial carbon storage, emit greenhouse gases, and influence the atmospheric concentration of aerosols, which affect radiative forcing. In addition, post-fire changes in vegetation cover alter surface albedo, further modulating regional and global climate (Lasslop et al., 2019; Potter et al., 2020). Climatecarbon cycle feedbacks involving fire are estimated to contribute approximately 6 ppm of additional atmospheric CO_2 per 1°C of global mean temperature increase (Harrison et al., 2018; M. W. Jones et al., 2022).

Beyond their effects on ecosystems and climate, wildfires also pose significant environmental and public health risks. High levels of fine particulate matter ($PM_{2.5}$) generated during fire events are associated with severe health outcomes, including increased rates of infant mortality (Cascio, 2018; Pullabhotla et al., 2023). In addition, wildfires strain healthcare systems by causing direct fire-related injuries and fatalities as well as smoke-related illnesses (Cascio, 2018). Wildfire smoke also negatively impacts mental health and individual well-being (B. A. Jones, 2017). Severe fires can degrade water quality by introducing sediment, nutrients, heavy metals, and other contaminants into aquatic systems (Bladon et al., 2014) and they can damage buildings and critical infrastructure.

Given the anticipated and substantial changes in climate over the coming century, there is an urgent need to reassess existing wildfire prediction, adaptation, and mitigation strategies (Alonso et al., 2024). In recent years, ML has emerged as a powerful tool for wildfire modeling (Jain et al., 2020), often outperforming traditional process-based models in predictive accuracy (Di Giuseppe et al., 2025). While conventional models are effective at capturing regional and interannual fire dynamics, they tend to struggle with finer spatial and temporal resolution, limiting their ability to represent complex interactions between fire drivers such as climate conditions, fuel availability, topography, and human activity (Prapas, Ahuja, et al., 2023). ML models, by contrast, excel at leveraging large, heterogeneous datasets to uncover these complex relationships and enhance predictive accuracy.

Moreover, the integration of XAI techniques with ML enables researchers not only to predict wildfire danger but also to interpret the underlying drivers and mechanisms. Recent studies have successfully combined XAI with ML algorithms to analyze fire dynamics and improve our understanding of model behavior, as discussed in Section 1.1. Two factors play a central role in this context: the predictive accuracy of wildfire forecasting models and their physical interpretability, which allows linking model outputs to underlying environmental processes (F. Li et al., 2024). Based on these recent developments, our goal is to develop highly accurate DL models for wildfire danger prediction while employing XAI approaches to explain and validate their outputs.

In this study, we focus on predicting next-day fire danger within a representative fire-prone region

of the Mediterranean. We leverage the Mesogeos dataset (Kondylatos, Prapas, Camps-Valls, & Papoutsis, 2023), which spans 2006–2022 and provides daily data at a 1 km × 1 km grid resolution. This dataset integrates heterogeneous variables encompassing fuel characteristics, meteorology, topography, and human factors. More details on the dataset and the extraction workflow are provided in Section 3. To capture the temporal dimension of fire danger, we incorporate a 30-day time lag of predictors into our models, thereby embedding the spatio-temporal context essential for wildfire modeling.

We evaluate seven DL architectures designed to capture temporal and spatio-temporal dependencies, including four Transformer-based variants (a standard Transformer, Gated Transformer Network (GTN), and Temporal Fusion Transformer (TFT)), as well as Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Multilayer Perceptron (MLP), and benchmark their performance against a less complex baseline Random Forest (RF) model. Three of these models, namely LSTM, Transformer, and GTN, are adapted from Track A: Wildfire Danger Forecasting of Kondylatos, Prapas, Camps-Valls, and Papoutsis (2023), while the remaining architectures represent extensions developed as part of this thesis. Detailed information about the different model architectures can be found in Section 4.2. To ensure generalizability, we employ cross-validation across multiple temporal folds.

As a key part of this thesis, we employ XAI techniques to identify the most influential drivers of fire danger and quantify their temporal contributions. Within case studies of specific fire events we assess model behavior in challenging prediction scenarios. An overview of the XAI methods used in this study and their underlying principles is provided in Section 4.3.

By combining state-of-the-art DL models with XAI, this work aims to improve our understanding of wildfire dynamics and contribute to the development of interpretable, data-driven tools for fire danger prediction.

The subsequent sections of this thesis are structured as follows: Chapter 2 provides the climatic and environmental context for wildfire dynamics in the Mediterranean, while Chapter 3 offers a detailed description of the Mesogeos dataset, including the data extraction workflow and a comparison with existing datasets, and further outlines the ML pipeline used in this study, encompassing the cross-validation setup, and the used evaluation metrics. Additionally, Section 4.2 outlines the implemented ML architectures. An overview of the XAI methods employed and their underlying principles is presented in Section 4.3. Chapter 5 details the implementation of the ML pipeline. The results of our experiments, including model performance, interpretability analyses, and case studies, are presented in Section 6, and key findings, limitations, and future research directions are summarized in Chapter 7.

1.1. Related Work

Wildfire forecasting and wildfire susceptibility mapping based on remote sensing and Earth observation data have become essential tools for proactive wildfire management (Jain et al., 2020).

The potential of DL in modeling such complex spatiotemporal dynamics is highlighted by Reichstein et al. (2019), who advocate for data-driven learning to better capture nonlinear interactions in Earth System Science, including wildfires. Several region-specific studies have further explored these dynamics. For example, Iban and Sekertekin (2022) investigated wildfire prediction in Turkey, Cilli et al. (2022) in Italy, Abdollahi and Pradhan (2023) in Australia, and G. Zhang et al. (2019) in China. While these works provide valuable insights, they often limit their scope to smaller geographical areas or lower spatial resolution.

Efforts to scale up to global datasets have also emerged. Notably, the SeasFireDataset (Alonso et al., 2024) has been used to compare wildfire activity across different regions worldwide. Ji et al. (2024) introduced the Seas Fire Datacube for such comparative studies, while Michail et al. (2024) proposed a Spatio-Temporal Deep Neural Network architecture. Prapas, Ahuja, et al. (2023) focused on a U-Net-based model, and Prapas, Bountos, et al. (2023) advanced wildfire forecasting by incorporating teleconnection signals into a Transformer-based architecture for improved subseasonal-to-seasonal prediction.

Some recent studies, such as those by F. Li et al. (2024) and Di Giuseppe et al. (2025), directly compare DL models with traditional baselines, including Decision Trees (DT), Random Forests (RF), XGBoost, and Artificial Neural Networks (ANN) , as well as conventional Fire Weather Indices, in the context of major fire events, including those in Los Angeles at the start of 2025.

In the Mediterranean context, Kondylatos, Prapas, Ronco, et al. (2022) and Shams Eddin et al. (2023) used datasets similar to those in this work, but focused on much coarser spatial resolutions (25×25 km instead of 1×1 km) and were limited to smaller areas such as Greece or the eastern Mediterranean.

Moreover, existing Explainable Artificial Intelligence (XAI) applications in wildfire prediction remain limited. Most rely on basic architectures such as Feedforward Neural Network (FFN) (Abdollahi & Pradhan, 2023) or Random Forests (Cilli et al., 2022), and typically restrict their analyses to specific subregions (Kondylatos, Prapas, Ronco, et al., 2022).

The objective of this thesis is to improve upon these limitations by training and interpreting multiple DL architectures across the entire Mediterranean region, while still enabling detailed regional XAI analyses. Compared to studies focused on small subregions, this broader geographical scope helps mitigate overfitting and regional bias, increasing the robustness and generalizability of the results. At the same time, we avoid the high heterogeneity and domain shift challenges often encountered in global-scale datasets, which can obscure regional patterns and reduce predictive reliability. By focusing on the Mediterranean as a whole, we strike a balance between model generality and regional relevance, thereby enhancing the practical value of the models for operational wildfire management. In contrast to prior studies, we perform a systematic XAI analysis across all models using multiple explanation techniques, enabling a direct comparison of interpretability between architectures. Our overarching goal is to advance the development of robust and accurate wildfire prediction models grounded in the principles of XAI.

2. Fire Science

This chapter provides an overview of the climatic and environmental context relevant to wildfire dynamics, with a particular focus on the Mediterranean region. It first introduces the primary drivers of wildfire occurrence, including fuel availability, meteorological conditions, human influences, and topographical factors. Building on this foundation, the chapter examines why the Mediterranean is especially prone to wildfires, considering both its characteristic climate patterns and socio-economic pressures. Finally, it explores projected climate change impacts on the region, highlighting how intensifying heatwaves, prolonged droughts, and phenomena such as hydroclimate whiplash are expected to exacerbate fire danger and reshape fire regimes across the Mediterranean Basin.

2.1. Wildfire Contributing Factors

Wildfire occurrence results from a complex interplay of diverse factors, ultimately requiring an ignition source to trigger fire emergence. The most important drivers of fire activity can be broadly categorized into four groups: fuel availability, meteorological conditions, human influences, and topographical characteristics (Abdollahi & Pradhan, 2023). The variables used in this study were also grouped accordingly and can be found in Table A.1 in the appendix. Furthermore, the general direction of influence (i.e., whether higher values increase or decrease the likelihood of fire occurrence) for each variable is summarized in Table A.2 in the appendix.

Fuel

Fuel availability refers to the type, quantity, and condition of combustible vegetation present in a landscape. This includes live and dead plant material such as grasses, shrubs, and trees, whose moisture content, density, and spatial continuity determine their flammability and the potential for fire spread (Sun et al., 2021). For ignition to occur, fuels must be sufficiently dry to support combustion. Fuel moisture is primarily governed by environmental variables such as temperature, solar radiation, humidity, and precipitation (Prestemon et al., 2013). In particular, the duration and intensity of preceding precipitation events strongly influence the moisture content of surface fuels (Bradshaw et al., 1984). Fuel characteristics are dynamic and can change seasonally or in response to climatic conditions, with low fuel moisture and high biomass loads generally increasing fire risk (Wang & Wang, 2020).

Meteorology

Meteorological conditions encompass the weather and climate variables that influence fire ignition and behavior. Key factors include temperature, relative humidity, wind speed and direction, and precipitation. High temperatures and low humidity dry out fuels, making them more susceptible to ignition, while wind can dramatically accelerate fire spread and alter fire direction (Abdollahi & Pradhan, 2023; Bowman et al., 2009; Dorph et al., 2022). Antecedent weather, such as wet periods that promote vegetation growth followed by drought, can further intensify fire risk by increasing available fuel loads (Bowman et al., 2009). Extended droughts or heatwaves, often linked to climate change, have been shown to increase both the frequency and severity of wildfires globally (Abatzoglou et al., 2019; Brown et al., 2004; M. W. Jones et al., 2022).

Human Factors

Human activity is a dominant ignition source in many fire-prone regions, especially in the Mediterranean, where arson and negligence (e.g., stubble burning) are common (Palaiologou et al., 2021). Data from the European Fire Database indicate that more than 70% of wildfires in Southern Europe between 2006 and 2010 were human-induced (Ganteaume et al., 2013). Key socio-economic and infrastructural factors such as population density, proximity to roads, and land use patterns are related to the spatial distribution and probability of ignitions (Dorph et al., 2022; Ganteaume et al., 2013). Moreover, the expansion of settlements into wildland areas, known as the wildland-urban interface, has significantly increased the risk of human-caused fires (Abdollahi & Pradhan, 2023).

Topographical Factors

Topographical features such as slope, aspect, and elevation play a critical role in shaping wildfire behavior. Fires tend to move more rapidly uphill due to preheating of fuels, while aspect affects solar radiation and thus vegetation dryness (Tavakkoli Piralilou et al., 2022). South- and west-facing slopes in the Northern Hemisphere typically experience higher temperatures and lower moisture, making them more fire-prone (Rothermel, 1983). Elevation can affect both fuel types and local weather patterns, further modulating fire risk (Sun et al., 2021). The complexity of terrain can also influence wind patterns, creating conditions that either promote or hinder fire spread (Moritz et al., 2010).

2.2. Wildfires in the Mediterranean Region

The Mediterranean region is particularly vulnerable to wildfire due to a combination of climatic and socio-economic factors. One contributing factor is the frequent fulfillment of the so-called 30-30-30 rule during summer months, referring to days when surface temperatures exceed $30°C$, relative humidity drops below 30%, and wind speeds surpass 30km/h (Farid et al., 2024; Francisco Seijo, 2017). These conditions form what is often called the "megafire triangle" (Farid et al., 2024) and are strongly associated with explosive fire behavior and extreme wildfire events. At the same time, the region hosts a dense seasonal population due to tourism, accounting for 32% of global international travel activity (World Tourism Organisation, 2018). This adds pressure to already fire-prone landscapes through increased human presence and infrastructure exposure.

Especially southern Europe has seen a sharp rise in burned area, and wildfires are increasingly affecting the Mediterranean basin (San-Miguel-Ayanz et al., 2022).This trend is strongly linked to anthropogenic climate change, which has made fire weather more severe since the 1980s and is expected to intensify even under ambitious mitigation scenarios (Abatzoglou et al., 2019; Jolly et al., 2015; M. W. Jones et al., 2022). Moreover, 2017 has been one of the most devastating wildfire seasons on record in some southern Mediterranean countries, with a remarkable increase in burnt area compared to the previous decade: 535% for Portugal, 160% for France, 105% for Italy, and 95% for Spain (Dupuy et al., 2020).

Fire occurrence in the Mediterranean shows distinct temporal patterns. Human-caused fires tend to peak in spring (March–April) and summer (August), often in the afternoon, reflecting links to agricultural or recreational activities (Vazquez & Moreno, 1998).The influence of human activity on fire occurrence is closely linked to regional socio-economic conditions. Factors such as high unemployment rates or agricultural practices can significantly increase the likelihood of both intentional and accidental ignitions (Ganteaume et al., 2013). In contrast, lightning-induced fires,

which are largely climate-driven, are more concentrated in remote or mountainous areas, with peaks in July–August and clustered late-afternoon ignition times, especially in alpine regions like Switzerland (Conedera et al., 2006). As climate change continues to affect atmospheric instability and fuel aridity, lightning-ignited fires are expected to become more prominent and potentially expand beyond their historical hotspots (Ganteaume et al., 2013).

2.3. Projected Climate Change Impacts on Mediterranean Wildfires

Climate change is projected to substantially increase wildfire regimes in the Mediterranean Basin by amplifying both the climatic and ecological conditions conducive to fire (Cramer et al., 2018; Cunningham et al., 2024; Di Virgilio et al., 2019; Dupuy et al., 2020; Ruffault, Curt, Martin-StPaul, et al., 2018; Ruffault, Curt, Moron, et al., 2020). One phenomenon that is believed to be amplified by climate change is *hydroclimate whiplash* (Swain et al., 2025), which occurs when periods of unusually wet conditions are followed by very dry conditions. The initial wet phase supports vegetation growth and leads to a rapid accumulation of biomass, increasing the amount of available fuel. During the subsequent dry period, this biomass loses moisture quickly, resulting in highly flammable landscapes and an elevated risk of severe wildfire activity (Di Giuseppe et al., 2025; Swain et al., 2025).

Another critical mechanism involves the role of heatwaves, during which high temperatures combined with elevated Vapor-Pressure Deficit (VPD) accelerate vegetation drying and plant mortality over short time frames (Adams et al., 2017; Park Williams et al., 2013). These conditions are expected to occur more frequently and persist for longer periods in future climate scenarios (Cochard, 2021). The results of Ruffault, Curt, Moron, et al. (2020) indicate that, since large wildfires generally develop under heat-induced Fire-Weather Types (FWTs), future increases in the frequency and intensity of these FWTs are likely to substantially raise the potential for extreme wildfire events. In addition, heatwaves contribute to the depletion of fuel induced by climate change, which has been linked to exponential increases in burned area. As drought intensifies fire spread and the potential for rapid expansion grows with increasing fire size, the combination of persistent heat and fuel scarcity may further amplify fire danger in the region (Cunningham et al., 2024; Ruffault, Curt, Martin-StPaul, et al., 2018).

Beyond the biophysical drivers, the socio-economic and ecological consequences of increased wildfire activity in the region are anticipated to be profound. The Mediterranean Basin is already recognized as a climate change hotspot, and future risks are expected to affect multiple interconnected systems (Cramer et al., 2018; Ruffault, Curt, Moron, et al., 2020). Among the most vulnerable sectors are food production and security, public health, and water resources.

Regional projections consistently point to southern Europe as a critical area of concern. Under high-emission scenarios, projections indicate an average increase in seasonal fire danger of 24% per decade in Mediterranean regions, reaching up to 7% per decade in parts of France (Dupuy et al., 2020). The spatial distribution of fire-prone areas is also expected to shift to areas where fuel loads are not likely to be limiting, with expansion into regions that are currently less affected such as western and central France, and central-eastern Europe (Dupuy et al., 2020).

Collectively, these findings underscore the urgency of advancing fire modeling approaches, particularly those based on ML, that combine high predictive accuracy with strong physical interpretability (F. Li et al., 2024), in order to reliable assess future wildfire risk across the Mediterranean landscape.

3. Data & Methods

The Mesogeos dataset (Kondylatos, Prapas, Camps-Valls, & Papoutsis, 2023) was developed to facilitate the advancement of data-driven wildfire modeling specifically for the Mediterranean region. While traditional process-based models have been widely applied to various fire-related tasks, they often tend to struggle with finer spatial and temporal resolution, limiting their ability to represent complex interactions between fire drivers such as climate conditions, fuel availability, topography, and human activity (Prapas, Ahuja, et al., 2023). Data-driven approaches, by contrast, are better suited to model these intricate relationships. This thesis build up on the Mesogeos datacube, which is publicly available at Kondylatos, Prapas, Camps-Valls, and Papoutsis (2023).

3.1. Data Sources

The classification of the key drivers of fire occurrence into the three main categories, human activity, weather conditions, and vegetation was developed based on the concepts outlined in Hantson et al. (2016). Weather and vegetation are particularly important because they directly impact how quickly fuels dry out, which in turn affects both the likelihood and spread of fires. Beyond these natural factors, human activities, such as deliberate or accidental ignition, land use changes, and population density also strongly shape fire behavior and fire regimes. In the context of this study, no raw data were collected or compiled independently. Instead, the Mesogeos dataset (Kondylatos, Prapas, Camps-Valls, & Papoutsis, 2023) was utilized, which integrates and harmonizes a wide range of data sources relevant to wildfire modeling. Meteorological variables, including temperature, wind speed, wind direction, dewpoint temperature, surface pressure, relative humidity, precipitation, and surface solar radiation were sourced from the ERA5-Land dataset (Muñoz-Sabater et al., 2021), which provides hourly historical land-based weather data from 1950 to the present. Information related to vegetation status and drought conditions was obtained using daytime and nighttime land surface temperature (Wan et al., 2015), the Normalized Difference Vegetation Index (ndvi) (Didan, 2015), and the Leaf Area Index (lai) (Myneni et al., 2002) from Moderate Resolution Imaging Spectroradiometer (MODIS), along with soil moisture estimates from the European Drought Observatory (EDO) (Cammalleri et al., 2017). Indicators of human presence and activity, such as population density and proximity to roads, were drawn from Worldpop (Tatem, 2017). Terrain characteristics, including elevation, slope, aspect, and curvature, were incorporated using data from the Copernicus Global Digital Elevation Model (COP-DEM) (Franks & Rengarajan, 2023). Additionally, land cover classifications were taken from the Copernicus Climate Change Service (Potapov et al., 2022). Fire-specific data, such as burned areas, were retrieved from European Forest Fire Information System (EFFIS), while ignition points and ignition dates were estimated using the MODIS Active Fire (AF) product (Giglio et al., 2016).

3.2. Datacube

The Mesogeos dataset (Kondylatos, Prapas, Camps-Valls, & Papoutsis, 2023) is organized as a spatio-temporal datacube, with three dimensions representing longitude, latitude, and time. It provides 27 variables capturing key wildfire drivers, including meteorological conditions, vegetation dynamics, land cover types, and human activity patterns. However, following the recommendations of Kondylatos, Prapas, Camps-Valls, and Papoutsis (2023), only 24 variables were used for wildfire danger forecasting, excluding burned areas, aspect, and curvature. In addition to these well-known fire predictors, Mesogeos includes separate layers for historical burned areas, ignition events, and

the associated burned area sizes. The dataset offers a fine spatial resolution of 1 km×1 km and a daily temporal resolution, covering a continuous period from 2006 to 2022. Geographically, it spans the broader Mediterranean region, encompassing approximately 4714 km in longitude and 1753 km in latitude, comprising a total of 6026 days of observations. Given this spatial and temporal extent, each dynamic variable in the dataset comprises a total of 47,796,706,692 individual data points. The variables used in this study, along with their abbreviations, are summarized in Figure 3.1. A comprehensive overview of all variables, including their data sources, original temporal and spatial resolutions, and measurement units, is provided in Table A.3 in the appendix, as presented by Kondylatos, Prapas, Camps-Valls, and Papoutsis (2023).

Abbreviation	**Description**
	Dynamic Variables
d2m	Dewpoint temperature at 2m (K)
lai	Leaf Area Index
lst_day	Day land surface temperature (K)
lst_night	Night land surface temperature (K)
ndvi	Normalized Difference Vegetation Index
rh	Relative humidity (%)
smi	Soil Moisture Index
sp	Surface pressure (Pa)
ssrd	Surface solar radiation downwards (J/m^2)
t2m	Temperature at 2m above the Earth's surface (K)
tp	Total precipitation (m)
wind_speed	Wind speed (m/s)
	Static Variables
dem	Elevation (m)
lc_agriculture	Land cover class: agriculture (%)
lc_forest	Land cover class: forest (%)
lc_grassland	Land cover class: grassland (%))
lc_settlement	Land cover class: settlement (%)
lc_shrubland	Land cover class: shrubland (%)
lc_sparse_vegetation	Land cover class: sparse vegetation (%)
lc_water_bodies	Land cover class: water bodies (%)
lc_wetland	Land cover class: wetland (%)
population	Population ($people/km^2$)
roads_distance	Distance from roads (km)
slope	Slope (rad)

Table 3.1.: Abbreviations and Descriptions of the Variables used in the ML Models.

3.3. Comparing with Existing Data Sets

Mesogeos is a comprehensive and flexible dataset specifically designed to support a wide range of ML tasks in the field of wildfire modeling. Unlike many existing datasets, which are often tailored to address only a single application (Kondylatos, Prapas, Ronco, et al., 2022; F. Li et al., 2024; Singla et al., 2021), Mesogeos adopts a more general purpose approach. It is provided as a cloud-optimized spatio-temporal datacube, where each entry is directly linked to a specific date, longitude, and latitude. This structure not only enables easy selection of data across time and space but also makes it straightforward to compute new variables or integrate additional information. The adaptability of Mesogeos facilitates the creation of custom ML datasets and opens possibilities for extending its use to different research objectives.

Importantly, it represents the first dataset of this spatial and temporal resolution specifically built for wildfire studies in the Mediterranean region.

A related initiative is the SeasFire Data Cube (Alonso et al., 2024), which similarly adopts a spatio-temporal structure to support global analyses of fire dynamics. However, in contrast to Mesogeos, SeasFire aggregates data at a coarser spatial resolution of 0.25° and a temporal resolution of eight days, limiting its applicability for fine-scale regional studies, particularly in heterogeneous

landscapes like the Mediterranean. While SeasFire provides valuable insights into broader global fire patterns, Mesogeos offers substantially finer detail and temporal continuity, making it more suitable for localized wildfire danger forecasting tasks.
A potential extension of this thesis could involve adapting and scaling the Mesogeos datacube structure to a global context, as in Alonso et al. (2024). This would facilitate the development of high-resolution wildfire modeling capabilities beyond the Mediterranean, supporting broader applications in other fire-prone regions around the world.

3.4. Dataset Extraction

The methods used to extract the dataset and the underlying assumptions were adopted from (Kondylatos, Prapas, Camps-Valls, & Papoutsis, 2023). Nevertheless, we include this methodology in the thesis to clarify the origin of key assumptions, such as the use of ignition point proxies and the chosen ratio of positive to negative samples, which directly influence model training and interpretation. To train a binary classifier for predicting high wildfire danger, a dataset was constructed to distinguish between positive and negative samples based on wildfire occurrences. A fire event was considered indicative of high fire danger if it ultimately exceeded a burned area of 30 hectares (Kondylatos, Prapas, Camps-Valls, & Papoutsis, 2023). Since the exact ignition locations are not available, the centroid of each burned area polygon, obtained from the *burned_areas_shapefiles*, was used as a proxy for the ignition point.
To integrate these ignition points into the datasets gridded structure at a 1×1 km resolution, the nearest grid cell to each centroid was identified and treated as the ignition source. This spatial anchoring ensured consistent data extraction for model training. To avoid border effects and ensure spatial consistency, ignition points located too close to the edges of the data cube were excluded, as the required input patch (125×125 pixels, equivalent to a 62 km buffer in each direction) would not fit entirely.
For each valid ignition source, a time series window of 30 days preceding the ignition day was extracted, spanning from $t-30$ to $t-1$, where t denotes the day of ignition. The ignition day itself t was excluded from the predictor variables to ensure a strict separation between input features and the target event. Consequently, the final input sequence corresponds to $t-30$ (30 days before ignition) up to $t-1$ (the day immediately before ignition). The input comprised all dynamic variables and static features from the Mesogeos dataset (Kondylatos, Prapas, Camps-Valls, & Papoutsis, 2023), with static variables repeated across time steps, while burned areas and ignition points were excluded to prevent data leakage. The selected positive samples were compiled into a unified DataFrame and exported as a CSV file for downstream use. Importantly, ignition coordinates were also excluded from the predictive feature set to further avoid information leakage.
Selecting appropriate negative samples posed a greater challenge due to the inherently stochastic nature of wildfire occurrences (Kondylatos, Prapas, Ronco, et al., 2022). The absence of a fire event on a given day does not necessarily imply low fire danger, making it unreliable to treat no-fire pixels as negative examples. To mitigate this risk, negative samples were only drawn from locations and days where no fires occurred within a 62 km radius, thereby avoiding areas with potentially dangerous but unrecorded conditions. Given the relatively small number of fire events, we adopted a commonly used sampling strategy (Huot et al., 2021; Kondylatos, Prapas, Ronco, et al., 2022; G. Zhang et al., 2019), selecting twice as many negative samples as positive ones to enhance the training set while maintaining a balanced class distribution. The input data were normalized prior to being fed into the models.
The classifier outputs a softmax probability, which is interpreted as the predicted level of fire danger for each location and time step.

Missing values in the time series primarily result from limitations in the satellite-based input data, such as cloud cover or low spatial resolution, which hinder the reliable observation of environmental variables on certain days. To handle these NaN values before the samples were used to train the models, missing entries were first filled using the temporal mean across the entire time series for each feature. If missing values still remained after this step, they were replaced with zero, corresponding to the normalized mean of each variable.

3.5. Machine Learning Pipeline for Wildfire Danger Prediction

To forecast wildfire danger based on the Mesogeos dataset (Kondylatos, Prapas, Camps-Valls, & Papoutsis, 2023), a comprehensive ML pipeline was developed as illustrated in Figure 3.1. Samples were extracted from the Mesogeos datacube as described in Section 3.4. The extracted samples include variables grouped into four categories: Fuel, Meteorology, Human Factors, and Topographical Factors. These features were combined into a unified sample representation for each instance.

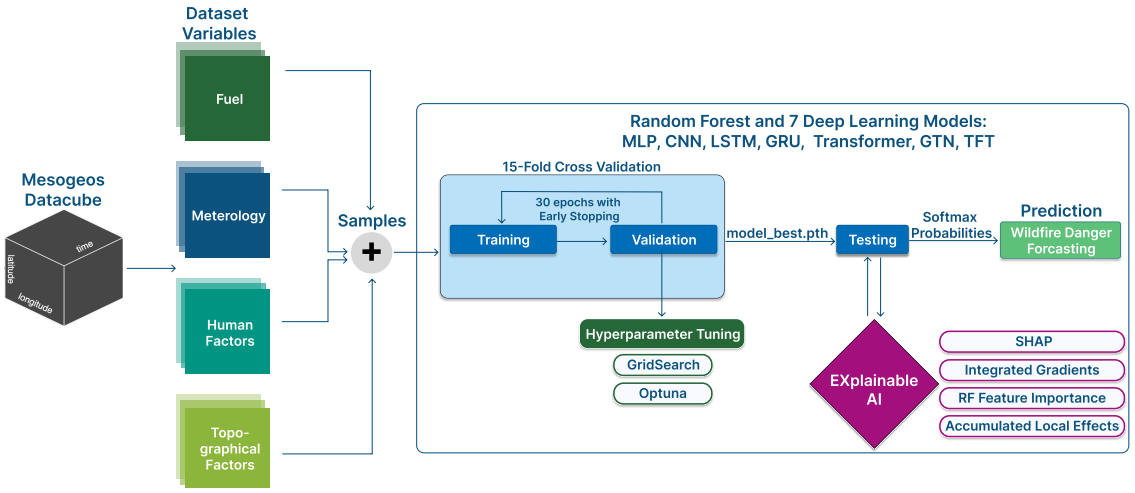

Figure 3.1.: **Overview of the ML Pipeline.** Samples extracted from the Mesogeos datacube are used in a cross-validation setup with Hyperparameter Optimization. Final models are evaluated on the Test Set and interpreted using XAI techniques.

Subsequently, a 15−Fold Cross-Validation strategy was employed to split the dataset into training, validation, and testing sets, as described in Section 3.6. In each fold, models were trained for a maximum of 30 epochs using early stopping based on the validation loss.

To identify optimal model configurations, hyperparameter tuning was applied using GridSearch and Optuna (Akiba et al., 2019). The best model checkpoint, denoted as model_best.pth, was stored based on validation performance.

These best-performing models were then evaluated on the held-out test data to assess their ability to generalize across time and space and to produce reliable wildfire danger forecasts. To enable an interpretable output, each model produces a probability distribution over two classes via a Softmax layer, where the resulting probabilities reflect the predicted level of fire danger, with higher values indicating a greater likelihood of significant fire expansion.

To further analyze and interpret model predictions and to uncover which input features most strongly influenced the predicted wildfire danger, a variety of Explainable Artificial Intelligence

(XAI) methods were applied. These include SHapley Additive exPlanations (SHAP)(Lundberg & Lee, 2017), Integrated Gradients (IG) (Sundararajan et al., 2017), Random Forest Feature Importance, and Accumulated Local Effects (ALE) (Apley & Zhu, 2020; Molnar, 2025).

3.6. Cross Validation for Robust Model Evaluation

Cross-Validation (CV) is a crucial step in assessing the generalization capability of ML models and avoiding misleading performance estimates due to overfitting (Jain et al., 2020; Tonini et al., 2020). In wildfire prediction research, CV is increasingly used to strengthen the reliability of ML-based susceptibility assessments (Abdollahi & Pradhan, 2023; Tonini et al., 2020; G. Zhang et al., 2019). We implemented a 15−fold CV strategy on the Mesogeos dataset (Kondylatos, Prapas, Camps-Valls, & Papoutsis, 2023), splitting the available data into disjoint training, validation, and testing subsets, rotating the years 2006–2022 so that each fold uses 14 years for training, one year for validation, and two consecutive years for testing as described in Figure 3.2. Validation was always performed in the year immediately preceding the two test years, ensuring a temporally plausible setup and reducing information leakage, as recommended by Roberts et al. (2017).

Figure 3.2.: **Overview of the 15-fold Temporal CV Setup.** For each fold, one year is used for Validation (orange), two consecutive years for Testing (red), and the remaining years for Training (blue).

The goal of this CV was to assess whether model performance remains consistent across different temporal splits. By confirming stable and robust performance across folds, it provides a solid foundation for the final evaluation and the subsequent XAI analysis, which was conducted on the fixed chronological split used in the Mesogeos study (2006–2019 training, 2020 validation, 2021 − 2022 testing).

3.7. Model Evaluation Metrics

To evaluate the performance of ML models in predicting large fire events, we rely on two key metrics: the F_1-Score (Sasaki, 2007) and the Area Under the Precision-Recall Curve (AUPRC). Both are well-suited for imbalanced classification tasks, where positive samples (e.g., wild fires) are relatively rare compared to negative samples (Raschka, 2014).

F_1-Score

The F_1-Score is a harmonic mean of two fundamental metrics: *Precision (PR)* and *Recall (REC)*. It provides a balanced measure that accounts for both false positives and false negatives.

- **Precision (PR)** measures the proportion of correctly predicted positive instances among all instances that were predicted as positive:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.1}$$

- **Recall (REC)** quantifies the proportion of correctly predicted positive instances among all actual positive instances:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.2}$$

- F_1-**Score** combines both Precision and Recall into a single metric:

$$F_1-\text{Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.3}$$

Here, *True Positive (TP)* refers to the number of correctly predicted wildfire occurrences, whereas *False Positive (FP)* denotes the number of instances where a wildfire was predicted but did not actually occur, and *False Negative (FN)* represents the number of actual wildfires that were not detected by the model.

A high Precision indicates that most predicted wildfire events indeed correspond to real occurrences, while a high Recall implies that the model successfully identifies the majority of actual wildfires. A higher F_1-Score indicates a more balanced model performance in terms of Precision and Recall.

Area Under the Precision-Recall Curve (AUPRC)

The AUPRC is calculated as the area under the Precision-Recall (PR-RE) curve. The PR-RE curve illustrates the trade-off between Precision and Recall across different classification thresholds. It represents the Precision vs. Recall values corresponding to all possible classification thresholds applied to the model predictions. A higher AUPRC indicates that the model maintains both high Precision and Recall across a range of thresholds, reflecting its robustness in distinguishing relevant fire events from background noise. Compared to overall accuracy, the AUPRC is generally regarded a superior, as more informative and discriminative, single number evaluation metric, particularly in settings with class imbalance, where accuracy may be misleading.

4. Machine Learning (ML) Methods

ML is a subfield of Artificial Intelligence (AI) that enables systems to learn from data and make predictions or decisions without being explicitly programmed for each individual task. By analyzing patterns in historical data, ML models aim to generalize from known examples to unseen situations. ML has become a powerful tool across a wide range of disciplines, including natural language processing, computer vision, finance, and healthcare. In the context of this thesis, it is particularly relevant for applications in climate and environmental sciences.

ML techniques are commonly divided into supervised and unsupervised learning. Supervised learning involves training models on labeled data, where both the input features and the correct output values are known, with the goal of learning a mapping that can be used to predict the outcomes on new data. This approach is widely used for tasks such as classification (e.g., predicting whether a wildfire will ignite) and regression (e.g., forecasting temperature or humidity levels).
Unsupervised learning deals with unlabeled data and aims to uncover hidden patterns or structures within it. This is particularly useful for clustering similar observations or for dimensionality reduction, which simplifies high-dimensional datasets for analysis.

4.1. ML Models vs. Physical Models for Fire Modeling

Physical or process-based, wildfire models simulate fire behavior by explicitly representing the physics of combustion, heat transfer, and atmospheric interactions. Prominent examples include FARSITE (Finney, 1998) and the Rothermel model (Rothermel, 1983). Those require detailed input on fuel types, topography, and meteorological conditions. Although such models provide mechanistic insight and are widely used in operational fire management, they are computationally expensive and highly dependent on the availability and precision of the input data, which limits their utility for large-scale or real-time forecasting (Lever et al., 2023; F. Li et al., 2024).

In contrast, ML models learn patterns and dependencies directly from historical data, allowing them to capture complex and non-linear relationships between fire occurrence and its driving factors. ML-based approaches, such as Random Forests and deep neural networks, are typically more computationally efficient and flexible, integrating heterogeneous data sources and producing near real-time predictions. Recent work demonstrates that ML models can outperform traditional fire danger indices and physical models in terms of predictive accuracy (Kong, 2024; F. Li et al., 2024).

For the above reasons, we use ML models for wildfire danger forecasting. However, XAI remains essential to assess their physical interpretability (F. Li et al., 2024), as demonstrated in our XAI analysis (see Section 6.4).

4.2. ML Models

This thesis focuses on supervised learning, specifically a binary classification task in which one class indicates increased wildfire danger and the other represents low-risk conditions. The models are trained on labeled input data to learn patterns that distinguish between these two classes. A RF as a classical ensemble method is examined, alongside DL models such as MLP, LSTM, GRU, CNN, Transformer, TFT, and GTN.

4.2.1. MLP

MLP is a fundamental class of feedforward neural networks that is widely used for classification and regression tasks. First introduced by Rosenblatt (1958), it consists of an input layer, one or more hidden layers, and an output layer. Each layer is fully connected to the next, and information flows in one direction, from input to output, as illustrated in Figure 4.1.

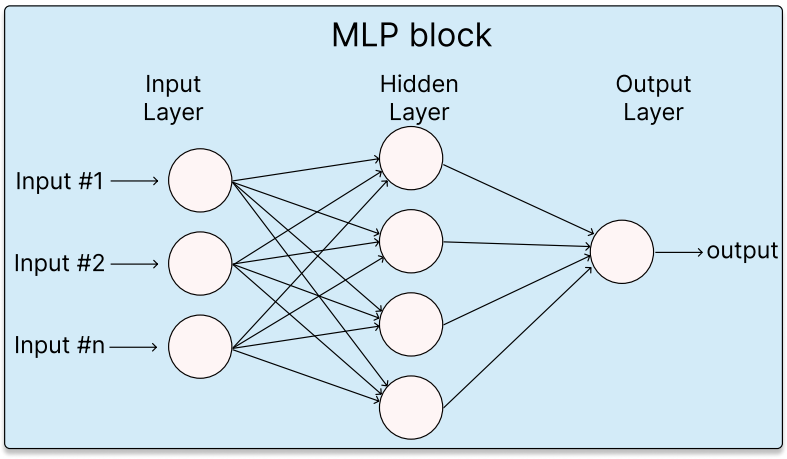

Figure 4.1.: **Simplified Architecture of the Implemented MLP** The network consists of an input layer, one or more hidden layers, and an output layer. Each neuron in a layer is fully connected to all neurons in the next layer.

A MLP processes structured data and learns non-linear relationships by passing weighted inputs through non-linear activation functions. The MLP training process can be divided into three main steps:

Step 1: Forward Propagation

During forward propagation, input data is passed sequentially through the network layers. In each hidden layer, a neuron computes a weighted sum of the inputs, adds a bias, and applies a non-linear activation function (Taud & Mas, 2018):

$$z = \sum_i w_i x_i + b$$

$$a = \phi(z)$$

This process is repeated through all layers until the output layer produces the final prediction (Bishop, 1995).

Step 2: Loss Function

The loss function quantifies the discrepancy between the network's output and the true labels. For binary classification tasks, the binary cross-entropy loss is typically used:

$$L = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

Step 3: Backpropagation

To minimize the loss, the model applies backpropagation to compute the loss gradients with respect to all weights and biases (Rumelhart et al., 1986). Using the chain rule, the error is propagated backward from the output layer to the input. The weights are updated using gradient descent:

$$w_i = w_i - \eta \cdot \frac{\partial L}{\partial w_i}$$

This ensures that the weights are adjusted in the direction that reduces the loss.

Training performance depends on hyperparameters such as learning rate, number of layers, and number of neurons per layer. Deeper MLP can model complex relationships but are more susceptible to overfitting (Bishop, 1995).

Variable Definitions

- x_i: Input feature (e.g., metrology variable) where the index i runs over all input features or neurons of the previous layer
- w_i: Trainable weight for input x_i
- b: Bias term added to the weighted sum
- ϕ: Activation function (e.g., ReLU, sigmoid, tanh)
- z: Linear combination before activation, $z = \sum_i w_i x_i + b$
- a: Activation output, $a = \phi(z)$
- $\hat{y}_i$: Predicted output for the i-th sample
- y_i: Ground-truth label for the i-th sample
- L: Loss function (e.g., Cross Entropy Loss or Mean Squared Error (MSE))
- η: Learning rate
- N: Number of Samples

4.2.2. Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) networks (Hochreiter & Schmidhuber, 1997) are a variant of Recurrent Neuronal Network (RNN) which are specifically designed to overcome limitations of traditional RNNs in learning long-term dependencies in sequential data. While standard RNNs can process time-series data by maintaining a hidden state across time steps, they typically suffer from the vanishing gradient problem (Hochreiter, 1991; Kolen & Kremer, 2001), which hampers their ability to learn from events that occurred far in the past. This phenomenon arises during backpropagation when gradients become increasingly small through successive layers, ultimately preventing the network from updating earlier weights effectively. This makes it difficult for them to capture dependencies across extended sequences, which is an essential requirement for applications such as wildfire danger forecasting using time-series data.

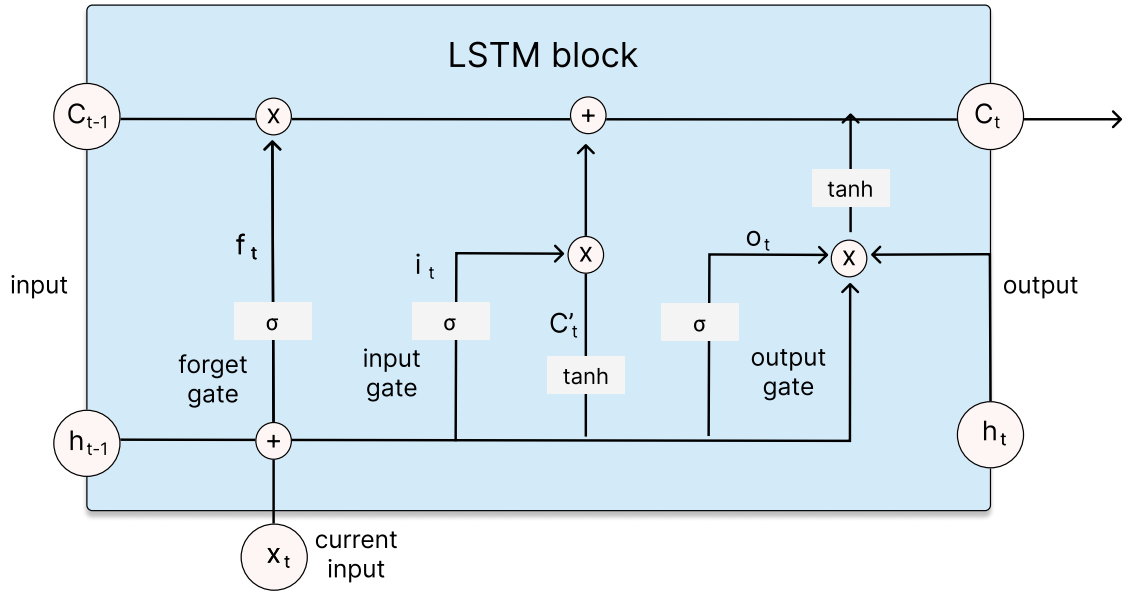

Figure 4.2.: **Simplified Architecture of the Implemented LSTM**. Each LSTM cell maintains a cell state and uses forget, input, and output gates to regulate information flow, enabling the modeling of long-term dependencies in sequential data.

LSTMs mitigate this issue by introducing a more elaborate internal structure that is shown in each LSTM cell contains a cell state that acts as a long-term memory and a set of gates, an input gate, an output gate and a forget gate, that regulate information flow (Van Houdt et al., 2020). The cell receives two main inputs at each time step: the current input vector x_t and the hidden state h_{t-1} , which is also known as the output from the previous time step. Based on these, the LSTM performs the following operations:

Forget gate: f_t decides which information from the previous cell state C_{t-1} should be retained or discarded. It is computed as

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f),$$

where σ denotes the sigmoid activation function. Values close to 0 "forget" the corresponding cell components, while values close to 1 retain them.

Input gate: i_t regulates which new information should be written to the cell state. It consists of two parts: the gate itself

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i),$$

and a candidate vector $\tilde{C}_t$ containing potential updates to the cell state, typically computed using a tanh activation function:

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c).$$

The cell state is then updated as

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t.$$

Output gate: o_t determines how much of the updated cell state should influence the output hidden state:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o),$$

$$h_t = o_t \cdot \tanh(C_t).$$

Through this architecture, LSTMs can maintain relevant information across longer sequences and learn complex temporal dependencies (Hochreiter & Schmidhuber, 1996). This makes them particularly well-suited for fire danger forecasting, where the model must integrate and remember evolving signals from meteorological, ecological, and anthropogenic variables over time.

Variable Definitions

- x_t: Input vector at time step t (e.g., today's features)
- h_t: Hidden state (output) at time step t
- c_t: Cell state (long-term memory) at time step t
- f_t, i_t, o_t: Activation vectors for the forget, input, output gates
- $\tilde{c}_t$: Candidate cell state (new memory content)
- W_f, W_i, W_c, W_o: Weight matrices for the forget, input, candidate, and output gates
- b_f, b_i, b_c, b_o: Bias vectors for the respective gates
- σ: Sigmoid activation function
- tanh: Hyperbolic tangent activation function

LSTM networks have become widely adopted in real-world applications. For example, they were used to significantly improve machine translations in Google Translate (Metz, 2016). Beyond time-series forecasting, LSTMs are also used in a variety of other domains such as Computer Vision (Van Houdt et al., 2020), and Natural Language Processing (Gers & Schmidhuber, 2001).

4.2.3. Gated Recurrent Unit (GRU)

GRU, introduced by Cho et al., 2014, are a type of RNN designed to capture temporal dependencies without relying on a separate memory cell as in LSTM units. As showed in Figure 4.3, GRUs use two gates, the *reset gate* and the *update gate*, to regulate the flow of information and determine which components of the past hidden state to retain or overwrite based on the current input.

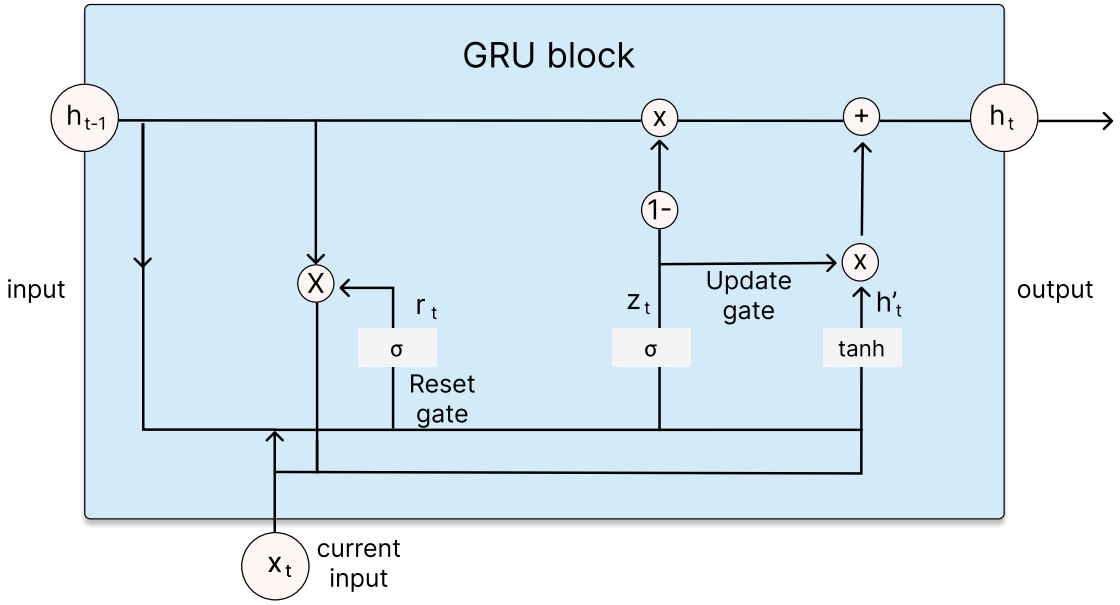

Figure 4.3.: **Simplified Architecture of the Implemented GRU**. GRU cells use reset and update gates to control the contribution of past hidden states and current inputs, providing a computationally efficient alternative to LSTMs for sequence modeling.

At each time step t, the reset $\mathbf{r}_t$ and update $\mathbf{z}_t$ gates are calculated based on the current input and the previous hidden state:

$$\mathbf{r}_t = \sigma(W_r[h_{t-1}, x_t]), \quad \mathbf{z}_t = \sigma(W_z[h_{t-1}, x_t]), \tag{4.1}$$

The reset gate determines to what extent the previous hidden state is ignored when computing the candidate activation. The update gate controls how much of the new candidate activation should be incorporated into the current hidden state.

The candidate activation $\tilde{h}_t$ is computed by applying the reset gate to the previous hidden state before combining it with the current input:

$$\tilde{h}_t = \tanh(W_h[r_t \odot h_{t-1}, x_t]), \tag{4.2}$$

where $\odot$ denotes element-wise multiplication and W_h is another weight matrix. This vector represents the potential new content to be added to the hidden state.

Finally, the new hidden state h_t is a linear interpolation between the previous hidden state and the candidate activation, weighted by the update gate:

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t. \tag{4.3}$$

This gating mechanism enables the GRU to adaptively retain long-term dependencies or focus on recent inputs, depending on the context. The architecture thus provides a computationally efficient alternative to LSTM networks while maintaining comparable performance on many sequence modeling tasks.

Variable Definitions

- x_t: Input vector at time step t

- h_{t-1}: Hidden state from the previous time step
- h_t: Hidden state at the current time step
- r_t, z_t: Reset and Update gate
- $\tilde{h}_t$: Candidate activation
- W_r, W_z, W_h: Trainable weight matrices for reset, update, and candidate computation
- σ: Sigmoid activation function
- tanh: Hyperbolic tangent activation function
- $\odot$: Element-wise multiplication

4.2.4. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) are a class of DL models that are particularly effective for processing grid-like data such as images. Although CNNs were originally developed for image classification (Krizhevsky et al., 2017), their architectural structure also makes them suitable for time-series forecasting, including wildfire danger prediction. In this context, the temporal axis can be represented as the height of an image and the feature dimension as the width. The overall architecture of the implemented CNN is illustrated in Figure 4.4.

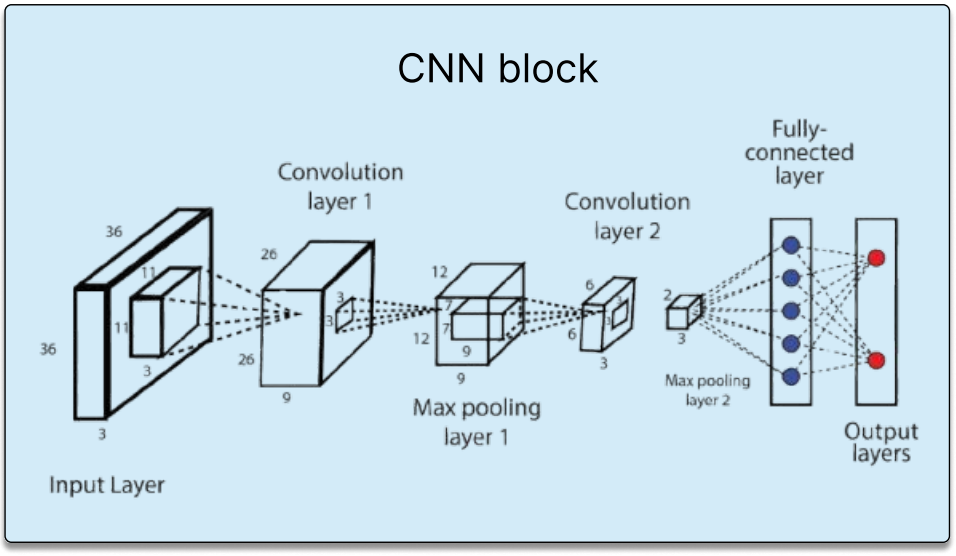

Figure 4.4.: **Architecture of the Implemented CNN**. The model consists of stacked convolution and pooling layers for feature extraction, followed by fully connected layers for final classification.

Convolutional Layers

The core operation in a CNN is the convolution, which involves applying learnable filters (kernels) across the input matrix R to extract spatial or temporal patterns (O'Shea & Nash, 2015). Each kernel contains a set of weights that are learned during training.

Mathematically, for a kernel $K \in \mathbb{R}^{k_h \times k_w}$ and an input matrix $X \in \mathbb{R}^{H \times W}$, the convolution at a specific location (i, j) involves computing the weighted sum between the kernel weights and the

corresponding input patch:

$$Y(i,j) = \sum_{m=0}^{k_h-1} \sum_{n=0}^{k_w-1} K(m,n) \cdot X(i+m, j+n)$$

This operation generates a *feature map* that highlights certain temporal or spatial structures, depending on what the kernel has learned.
To introduce non-linearity into the model and enable the learning of complex patterns, convolutional and fully connected layers are typically followed by a non-linear activation function.

Pooling Layers

Pooling layers follow convolutional layers to progressively reduce the spatial dimensions and computational complexity of the model. The most common types are:

- **Max Pooling**: Given a pooling window $W \times W$, the output at each location is computed as:

  $$Y(i,j) = \max_{(m,n) \in W \times W} X(i+m, j+n)$$

  This selects the maximum value in the region and is particularly effective at preserving dominant features.

- **Average Pooling**: In contrast, average pooling computes the mean value in each window:

  $$Y(i,j) = \frac{1}{|W \times W|} \sum_{(m,n) \in W \times W} X(i+m, j+n)$$

  This operation provides a smoother summary of the region and is less sensitive to outliers.

Transition to Fully Connected Layers

After several convolution and pooling operations, the resulting tensor is flattened into a one-dimensional vector and passed to fully connected layers for the final classification or regression. The network is trained using forward and backward propagation, adjusting kernel and layer weights to minimize a loss function.

Variable Definitions

- X: Input matrix (e.g., time-lagged feature matrix)
- K: Convolutional kernel (filter) with trainable weights
- $Y(i,j)$: Output value at location (i,j) in the feature map
- k_h, k_w: Kernel height and width
- $W \times W$: Pooling window size
- $\max(\cdot)$: Max operation over a region
- $\text{ReLU}(x)$: Rectified Linear Unit activation, $\max(0,x)$
- $|W \times W|$: Number of elements in the pooling window

4.2.5. Transformer and Gated Transformer Network (GTN)

Unlike RNNs, Transformers introduced by Vaswani et al. (2017) process entire input sequences in parallel. This is achieved by encoding positional information explicitly and using self-attention mechanisms to model dependencies in sequential data, capturing relationships between all positions in a sequence regardless of their distance.

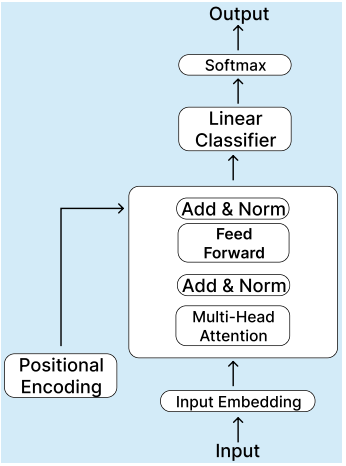

Figure 4.5.: **Simplified Architecture of the Implemented Transformer**. The model leverages multi-head self-attention and feedforward layers within stacked encoder and decoder blocks to capture dependencies across all time steps in parallel (Vaswani et al., 2017).

The Transformer model, as shown in Figure 4.5, is composed of two main parts:

- **Encoder:** A stack of N identical layers, each containing a multi-head self-attention mechanism followed by a position-wise feedforward neural network.
- **Decoder:** A stack of N layers with masked multi-head attention to prevent attending to future positions, followed by multi-head attention over the encoder outputs and a feedforward network.

Positional Encoding

Since the model does not include any recurrence or convolution, positional encodings first introduced by Gehring et al. (2017) are added to the input embeddings to provide information about the relative (Shaw et al., 2018) or absolute (Vaswani et al., 2017) position of tokens in the sequence. These encodings are combined with the input embeddings and are learned or defined using bounded, no-linear functions.

Attention Mechanism

At the core of the Transformer lies the attention mechanism, specifically Scaled Dot-Product Attention. It enables the model to compute contextual relationships between different positions in the sequence. For a given input vector E, the model computes:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

The dot product QK^T produces a score matrix that reflects the relevance between all pairs of positions. After scaling by the square root of the dimension d_k and applying softmax normalization, the resulting attention weights are used to compute a weighted sum over the value vectors V.

Multi-Head Attention

To allow the model to jointly attend to information from different representation subspaces, the Transformer uses multi-head attention. This involves running multiple self-attention operations in parallel, each with its own set of learnable projections for Q, K, and V. The outputs of all heads are concatenated and linearly transformed:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O$$

Each attention head focuses on different aspects of the sequence, such as syntactic or semantic relationships, and contributes a proposed update to the original embedding. These updates are then combined and added back to the input embedding via residual connections and layer normalization.

Feedforward Network

Following the attention mechanism, each position in the sequence is passed through a fully connected feedforward neural network, also referred to as a MLP as introduced in Chapter 4.2.1. This network is applied independently and identically to every vector in the sequence.

The underlying intuition is that the MLP learns to store specific facts or patterns that are useful for the prediction task. Each input vector is transformed in a way that emphasizes particular features or interactions relevant to its position. This component introduces additional modeling capacity and enables the Transformer to learn complex, non-linear functions independently at each position in the sequence.

Variable Definitions

- E: Embedding vector representing input features at a given time step
- $Q = EW^Q$: query matrix
- $K = EW^K$: key matrix
- $V = EW^V$: value matrix
- W^Q, W^K, W^V, W^O: Learnable projection matrices
- d_k: Dimensionality of the key vectors

Gated Transformer Network (GTN)

While the standard Transformer model captures temporal dependencies through self-attention across input sequences, the GTN extends this architecture by introducing an additional attention mechanism across feature dimensions. In this setup, the model processes both the temporal sequence and the feature channels independently via two separate Transformer encoders. The outputs from both streams are combined using a soft attention gating mechanism, which learns to weigh their relative importance for the final prediction. This dual-path design allows the GTN to selectively emphasize informative features while preserving temporal patterns, offering increased flexibility in handling heterogeneous meteorology and fuel variables. The general architecture of GTN is illustrated in Figure 4.6.

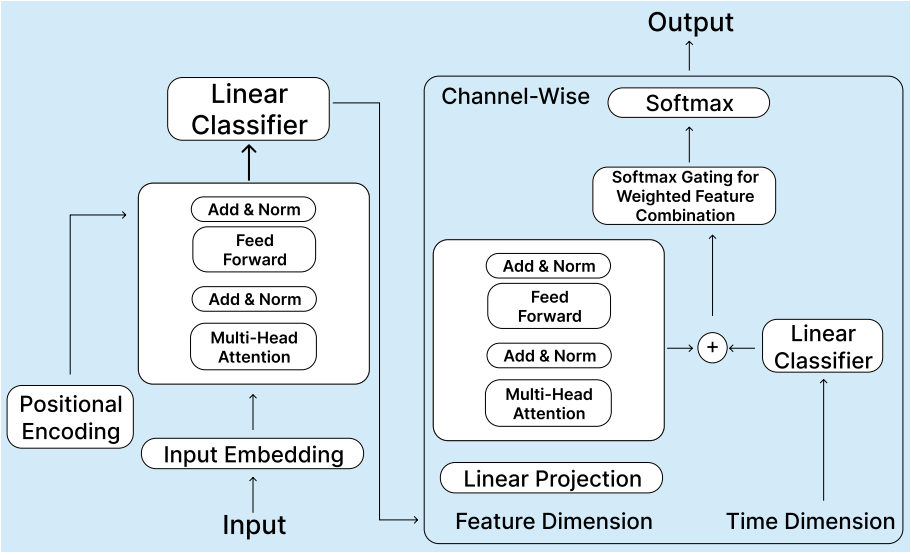

Figure 4.6.: **Simplified Architecture of the Implemented GTN**. The model introduces parallel attention paths across time and feature dimensions, which are combined through a softmax gating mechanism for weighted feature integration.

4.2.6. Temporal Fusion Transformer (TFT)

The Temporal Fusion Transformer (TFT), introduced by Lim et al. (2021), is a DL architecture specifically designed for interpretable multi-horizon time series forecasting. In contrast to classical Transformer models and the GTN used in this work (see Section 4.2.5), the TFT combines separate processing paths for static and temporal covariates with a series of gating and attention mechanisms.

The implementation presented in this thesis follows the original TFT architecture closely but is simplified for the specific requirements of wildfire danger classification. In particular, certain advanced components such as the full Variable Selection Networks, temporal fusion decoder, and context gating mechanisms are not included. Instead, the focus lies on a robust encoding of dynamic and static inputs and a lightweight attention mechanism over time.

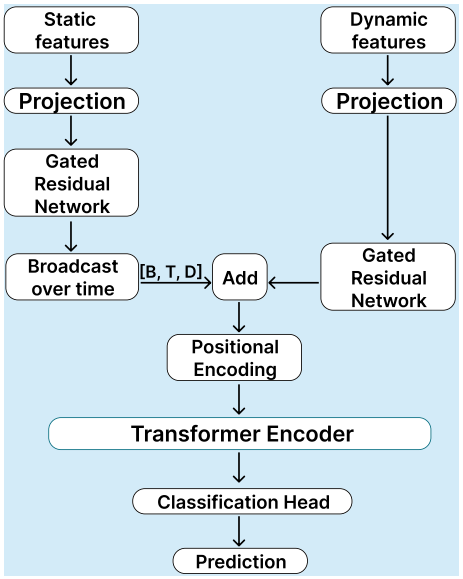

Figure 4.7.: **Simplified Architecture of the Implemented TFT**. Static and dynamic inputs are processed separately through linear projections and GRNs. Static inputs are broadcast over time and added to the dynamic representation before entering the Transformer Encoder.

As illustrated in Figure 4.7 the implemented *TFTNet* model processes static and dynamic inputs separately using linear projections followed by GRN. Static features are first projected and passed through a GRN. The resulting static context is then broadcast across all time steps and added to the dynamic representation. Dynamic features are similarly processed through a projection and GRN pipeline. The combined representation is enriched with positional encodings and passed through a Transformer Encoder to capture temporal dependencies. The final classification is derived from the last time step of the encoded sequence.

In contrast to the GTN, which uses parallel encoders and softmax-based fusion across time and feature dimensions, TFT focuses solely on temporal attention and uses GRNs to modulate feature relevance. Static features are processed separatly and broadcast across time. Unlike the original Transformer architecture (Vaswani et al., 2017), which includes both an encoder and decoder for sequence-to-sequence tasks, this model omits the decoder entirely, as it is not required for single-step binary classification.

4.2.7. Random Forest (RF)

Random Forest (RF) represent an ensemble-based extension of traditional decision trees, designed to overcome their tendency to overfit and improve generalization performance on unseen data. While decision trees offer high interpretability and fast inference, their predictive accuracy is often limited (Ho, 1995). RFs address these limitations by combining the predictions of multiple decision trees that are trained on randomly sampled subsets of the data and feature space (Breiman, 2001).

The method builds on two core principles: bagging (Breiman, 1996) and random feature selection. In the bagging process, multiple training subsets are generated by sampling from the original training data with replacement. Each tree in the forest is trained on one such bootstrap sample, which typically includes about two-thirds of the original data points; the remaining third, known as the Out-of-Bag (OOB) samples, are later used for internal validation (Tibshirani et al., 2009). Notably, sampling with replacement allows individual data points to appear multiple times within the same bootstrap dataset.

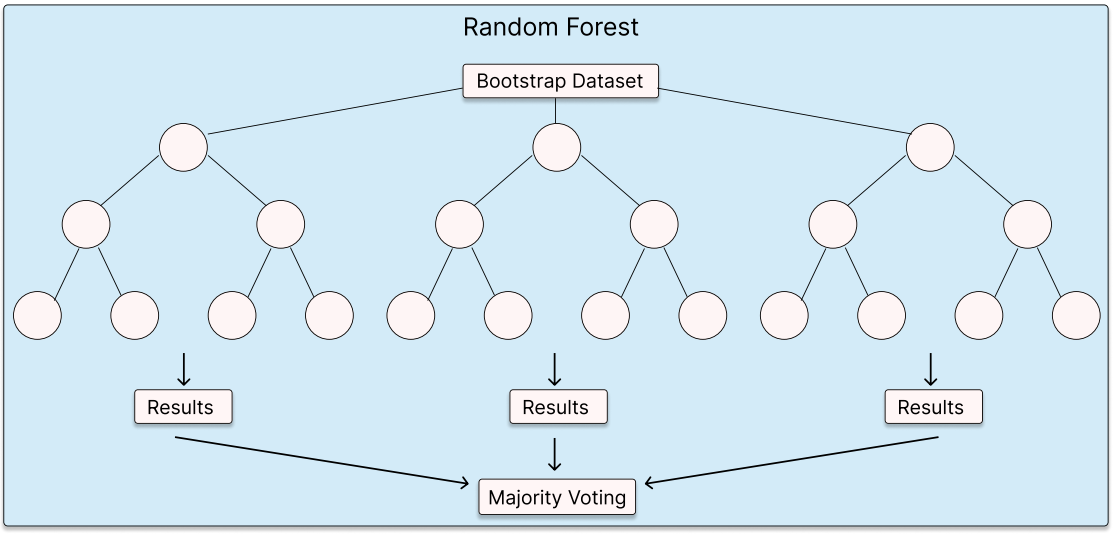

Figure 4.8.: **Simplified Architecture of the Implemented RF**. Multiple decision trees are trained on different bootstrap samples with random feature selection to reduce correlation between trees. Predictions from all trees are combined via majority voting to improve generalization performance.

As outlined in Figure 4.8 each decision tree is then independently trained by recursively splitting the training data to maximize class separation, which is typically measured by criteria such as Gini impurity, information gain, or entropy. Random feature selection introduces additional randomness by limiting the number of features that each split in a tree can consider. At each node of a tree, a random subset of available input features is selected, and the best split is determined only among this subset. A common heuristic is to choose the $\sqrt{p}$ features from the total of the p input features (Liaw & Wiener, 2002). This random feature selection reduces the correlation between trees and increases the diversity of the ensemble, which is crucial for improving predictive performance.

During inference, a majority voting scheme is used: for a given input sample, each tree provides a prediction, and the final output corresponds to the class that receives the highest number of votes among all trees.

Model performance of a Random Forest is commonly estimated using the OOB error, which is calculated by testing each data point on the subset of trees that did not see it during training and measures the misclassification rate of the OOB samples throughout the forest. This built-in validation mechanism provides an unbiased estimate of the model's generalization accuracy without requiring a separate validation set.

4.3. Explainable Artificial Intelligence (XAI)

Despite their impressive predictive power, many ML models, especially complex ones such as deep neural networks, are often criticized for acting as "black boxes", as they yield accurate predictions but offer little transparency into their decision-making process.

To address this limitation, this thesis employs XAI techniques to uncover how input features influence predictions. Understanding the model's reasoning is essential for trust, transparency, and scientific insight, particularly in critical applications like wildfire danger forecasting.

Three state-of-the-art methods are used: SHAP, a game-theoretic model-agnostic approach, Integrated Gradients (IG), a gradient-based technique tailored for deep neural networks and ALE

which quantifies how features influence the model prediction on average, even in the presence of correlated inputs.

All three methods are passive XAI approaches, meaning they do not require any modification to the underlying model architecture or training process. Additionally, they belong to the XAI class of explanation by attribution, which involves assigning importance scores to input features based on how much each feature contributes to the models output (Y. Zhang et al., 2021).

Despite their shared foundation, the methods differ in explanatory scope: SHAP provides local explanations for individual predictions, IG offers semi-local insights by averaging attributions along paths from a baseline, and ALE yields global explanations by estimating the average marginal effect of each feature across the entire input space (Kim et al., 2025; Y. Zhang et al., 2021).

4.3.1. SHapley Additive exPlanations (SHAP) Values

SHapley Additive exPlanations (SHAP) values provide a unified and theoretically grounded approach to interpreting the predictions of the model based on concepts from cooperative game theory (Lipovetsky & Conklin, 2001). Each input feature is viewed as a player in a game, and the model output corresponds to the payout generated by their cooperation. The goal is to fairly distribute this payout among all features, reflecting their individual contribution.

Mathematically, SHAP values correspond to Shapley values (Shapley & Lloyd, 1953), which provide a unique solution for allocating rewards in cooperative games. The contribution of each feature is defined as its average marginal effect on the model output across all possible feature orderings. In the context of ML, this results in attributing the prediction to the input features by measuring how the expected output changes when conditioning on each feature (Lundberg & Lee, 2017). This enables SHAP to explain how a specific model prediction deviates from the baseline output that would be made in the absence of any feature information.

However, as highlighted by Molnar (2025), Shapley values are often misinterpreted. The Shapley value of a feature does not represent the difference in prediction obtained by removing that feature during model training. Instead, it should be understood as the estimated contribution of a given feature value to the difference between the model's prediction for the current instance and the mean model prediction, conditional on the current set of feature values.

At its core, SHAP represents the prediction of a model as a sum of individual feature contributions. The model output is approximated by an interpretable surrogate model g defined in a simplified binary input space $z' \in \{0,1\}^M = \mathcal{Z}$, where each binary value represents the presence or absence of a feature. The surrogate model takes the additive form:

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i, \tag{4.4}$$

where ϕ_i represents the attribution of the i-th feature and ϕ_0 is the model output when all features are absent.

Given the exponential computational complexity of calculating exact Shapley values, SHAP introduces several approximation techniques. Among these, Kernel SHAP has emerged as one of the most widely adopted methods, as it frames the estimation process as a locally weighted linear regression problem. In this study, Kernel SHAP was chosen because it offers more accurate attributions with fewer model evaluations compared to other sampling-based approaches (Abdollahi &

Pradhan, 2021). Moreover, it has been successfully applied in prior research, including in Abdollahi and Pradhan (2023).

Kernel SHAP is a model-agnostic approximation technique for computing SHAP values when the underlying model is treated as a black box. It leverages the general SHAP formulation above and estimates the attributions ϕ_i by fitting a surrogate model g to locally approximate the output of the true model f. Specifically, Kernel SHAP formulates this as a weighted linear regression problem in the binary input space:

To recover the true Shapley values, the SHAP framework, as introduced by (Lundberg & Lee, 2017), identifies a unique weighting kernel $\pi_{x'}$ and a loss function L such that the following weighted least-squares objective is minimized:

$$L(f, g, \pi_{x'}) = \sum_{z' \in \mathcal{Z}} \left[f(h_x(z')) - g(z') \right]^2 \pi_{x'}(z'), \tag{4.5}$$

In this formulation, $h_x(z')$ maps the simplified binary input z' back to a full input sample for the original model f, $\pi_{x'}(z')$ is a specially designed kernel that assigns higher weights to simplified inputs z' that are more similar to the original instance x', and $g(z')$ denotes the interpretable linear surrogate model in SHAP form.

By minimizing this weighted loss, Kernel SHAP estimates the SHAP values ϕ_i such that they satisfy the three key properties described by Lundberg and Lee (2017):

- **Local accuracy:** The sum of the SHAP values exactly matches the model output for the instance being explained.
- **Missingness:** Features that do not influence the prediction receive an attribution of zero.
- **Consistency:** If a model changes such that the marginal contribution of a feature increases (or stays the same) across all inputs, its SHAP value does not decrease.

These properties ensure that SHAP values derived via Kernel SHAP are the unique solution within the class of additive feature attribution methods that is both theoretically sound and practically interpretable (Lundberg & Lee, 2017).

Variable Definitions

- f: Original prediction model treated as a black box.
- g: Linear surrogate model approximating f locally.
- $g(z')$: Prediction of the surrogate model for a simplified binary input z'.
- z': Simplified binary input vector ($z'_i = 1$ if feature i is included, else 0).
- $\mathcal{Z}$: Set of all possible binary feature inclusion vectors.
- ϕ_i: SHAP value (attribution) assigned to feature i.
- ϕ_0: Baseline model output when all features are absent.
- $h_x(z')$: Mapping function from simplified input z' back to the full feature space instance for evaluation by f.
- x: Instance being explained in the original input space.
- x': Baseline input (e.g., zero vector) representing absence of information.
- $\pi_{x'}(z')$: Weighting kernel that emphasizes samples z' closer to the explained instance x'.
- $L(f, g, \pi_{x'})$: Weighted least-squares loss function used in Kernel SHAP to fit g to f locally.

4.3.2. Integrated Gradients (IG)

The Integrated Gradients (IG) method was introduced by Sundararajan et al. (2017) to provide feature attributions for DL models. The goal is to quantitatively determine the contribution of each input feature to the models prediction, based on a set of two fundamental axioms:

- **Sensitivity:** An attribution method satisfies sensitivity if, for any input and baseline that differ in one feature and result in different predictions, the differing feature is assigned a non-zero attribution.
- **Implementation Invariance:** Two models are considered functionally equivalent if they produce identical outputs for all possible inputs. If two models are functionally equivalent, then the attribution method should assign the same attributions to both models, regardless of implementation differences (e.g., different architectures).

Unlike classical gradient-based methods, IG combine the sensitivity of techniques such as DeepLIFT (Shrikumar et al., 2019) with the implementation invariance of standard gradients. Integrated Gradients offer a clear advantage over naive gradient methods, as they avoid issues related to non-linear activation saturation (e.g., ReLU) and provide robust, interpretable explanations for deep and complex neural networks (Sundararajan et al., 2017).

The core idea is to compute gradients along a straight-line path from a baseline input x' to the actual input x, and to accumulate these gradients. This results in stable and interpretable attribution scores for each input feature.

Mathematically, the attribution for the i-th input feature is defined as:

$$\text{IntegratedGrads}_i(x) := (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \cdot (x - x'))}{\partial x_i} \, d\alpha \tag{4.6}$$

In this work, the Integrated Gradients method is implemented using the *Captum* library, a PyTorch-based framework (Paszke et al., 2017) for model interpretability. The library provides a high-level interface for computing attributions across various neural network architectures without requiring changes to the underlying model.

Variable Definitions

- F: Prediction function computed by the model.
- x: Actual input vector.
- x': Baseline input vector with no relevant information (e.g., zero vector).
- x_i: The i-th feature value of the actual input.
- x_i': The i-th feature value of the baseline input.
- α: Interpolation parameter scaling between baseline x' and actual input x.
- $\frac{\partial F}{\partial x_i}$: Gradient of the model output with respect to the i-th feature.

4.3.3. Accumulated Local Effects (ALE)

Accumulated Local Effects (ALE) plots are a model-agnostic method to estimate the influence of input features on model predictions by computing local changes in the prediction function and accumulating them across intervals. Introduced by Apley and Zhu (2020), ALE plots are

designed to overcome the limitations of Partial Dependence Plots (PDPs), especially in the presence of correlated features, where PDPs can produce misleading interpretations (Apley & Zhu, 2020; Molnar, 2025; Salih, 2024).

For a numerical feature x_j, the ALE method, as introduced by Apley and Zhu, 2020, partitions the feature range into intervals based on quantiles. Within each interval, the model output is evaluated at the upper and lower bounds of the interval for all data points in that bin, and the average prediction difference is computed. These local differences approximate the feature's marginal effect within that interval. The accumulated sum of these average differences across intervals gives the uncentered ALE estimate:

$$\hat{f}_{j,\mathrm{ALE}}(\mathbf{x}) = \sum_{k=1}^{k_j(x)} \frac{1}{n_j(k)} \sum_{i:x_j^{(i)} \in N_j(k)} \left[\hat{f}(z_{k,j}, \mathbf{x}_{-j}^{(i)}) - \hat{f}(z_{k-1,j}, \mathbf{x}_{-j}^{(i)}) \right], \tag{4.7}$$

To ensure interpretability, the ALE function is centered so that its average over the data is zero:

$$f_{j,\mathrm{ALE}}(\mathbf{x}) = \hat{f}_{j,\mathrm{ALE}}(\mathbf{x}) - \frac{1}{n} \sum_{i=1}^{n} \hat{f}_{j,\mathrm{ALE}}(x_j^{(i)}). \tag{4.8}$$

The name Accumulated Local Effects derives from its two core components. "Local Effects" refer to the average change in prediction within each interval, and "Accumulated" refers to the summation of these effects across intervals to reflect the total feature effect at a given value.

Variable Definitions

- x_j: The j-th numerical feature under analysis.
- $\mathbf{x}$: Feature vector of a single instance.
- $\hat{f}(\cdot)$: Prediction function of the trained model.
- $k_j(x)$: Index of the interval (bin) to which x_j belongs.
- $N_j(k)$: Set of instances whose x_j values fall into the k-th interval.
- $n_j(k) = |N_j(k)|$: Number of instances in the k-th interval.
- $z_{k,j}$: Upper boundary of the k-th interval for feature x_j.
- $\mathbf{x}_{-j}$: Feature vector excluding x_j, used to isolate its marginal effect.
- n: Total number of instances in the dataset used for centering the ALE estimate.

5. Implementation

This chapter provides an overview of the implementation of the ML models and XAI techniques described in Chapter 4. It covers the general setup and training logic shared across models, the modular project structure, and the integration of hyperparameter tuning using Optuna. Furthermore, it outlines how model performance monitoring and evaluation is facilitated through tools such as TensorBoard and systematic logging.

All code related to this work is publicly available on GitHub and can be accessed at https://github.com/paulinebecker2002/mesogeos.

To facilitate maintainability and allow for future extensions, the implementation adopts a modular project architecture in line with best-practices for ML codebases, the structure of which is illustrated and explained below. The core functionality is distributed across different Python modules, such as *train.py*, *trainer.py*, and *model.py*, each with a clearly defined responsibility. This modularity allows flexible experimentation, making it straightforward to integrate additional features such as hyperparameter optimization via Optuna, learning rate scheduling, and comprehensive training monitoring. Tools such as TensorBoard and structured logging are used to track training progress and model performance consistently. The modular design also ensures that the pipeline remains adaptable between different model architectures with minimal adjustments.

The project is structured as illustrated in Listing 5.1. The main components include:

```
mesogeos/
|    ml_tracks/
|   └── a_fire_danger/
|       ├── a_danger_forecasting/     # Contains classification data files
|       ├── configs/                  # Model-specific configs (MLP, GRU, etc.)
|       ├── dataloaders/              # Data loading utilities
|       ├── datasets/                 # Dataset preprocessing and splitting
|       ├── integrated_gradients/     # IG computation & plotting
|       ├── logger/                   # Manages logging and TensorBoard
|       ├── models/                   # Model architectures & metrics
|       ├── saved/                    # Stores all model outputs and artifacts
|       │   ├── ale/                  # Stores ALE plots
|       │   ├── ig/                   # Stores IG values and plots
|       │   ├── model/                # Stores trained model checkpoints
|       │   ├── log/                  # Log files and used config files
|       │   ├── shap_plot/            # SHAP values (CSV/NPZ) and plots
|       ├── shap_local/               # SHAP computation & plotting
|       ├── trainer/                  # Handle the execution of training loops
|       ├── tester/                   # Evaluation and testing scripts
|       ├── utils/                    # Helper utilities
|       ├── train.py                  # Entry point for training
|       └── test.py                   # Entry point for testing
|-- notebooks/                        # Notebooks for exploring the datacube
|-- outputs/                          # Analysis outputs and notebooks
|-- requirements.txt                  # Python dependencies
|-- README.md                         # Project documentation
```

Listing 5.1: Model Directory Structure

5.1. Model Architecture

To demonstrate the typical model structure used in this work, we present three representative architectures: a LSTM network, a CNN and a Transformer. All models are implemented using the PyTorch framework (Paszke et al., 2017) and follow the same modular design to ensure compatibility with the overall training and evaluation pipeline.

LSTM

The LSTM model captures temporal dependencies in sequential data using a single LSTM layer followed by a feedforward classifier. The model expects input sequences of shape *[batch_size, sequence_length, input_dim]* and uses only the final hidden state of the LSTM for classification.

```
class SimpleLSTM(nn.Module):
    def __init__(self, input_dim=24, output_lstm=128, dropout=0.5):
        super().__init__()
        self.lstm = nn.LSTM(input_dim, output_lstm, num_layers=1, batch_first=True)
        self.ln1 = torch.nn.LayerNorm(input_dim)
        ...
```

Listing 5.2: SimpleLSTM: Model Initialization.

As described in Listing 5.2, the *___init___* method, initializes the LSTM with the given input dimension and output size (number of hidden units). A layer normalization is applied before passing the data into the recurrent layer to stabilize training. The LSTM output is processed by a three-layer fully connected network with dropout and ReLU activations to compute the final class probabilities.

The forward method defines how input data flows through the LSTM model during training and inference and is defined as follows:

```
def forward(self, x):
    x = self.ln1(x)
    lstm_out, _ = self.lstm(x)
    x = self.fc_nn(lstm_out[:, -1, :])
    return x
```

Listing 5.3: SimpleLSTM: Forward Pass.

First, the input sequence x is normalized using a *LayerNorm* operation to stabilize training and improve convergence. This normalized input is then passed to a single-layer LSTM, which captures temporal dependencies across the sequence. The LSTM returns the full sequence of hidden states, but only the hidden state corresponding to the last time step (*lstm_out[:, -1, :]*) is selected. This vector serves as a fixed-size representation of the entire input sequence and is passed through a feed-forward neural network *fc_nn*, which consists of fully connected layers with dropout and ReLU activations. Finally, the output is a two-dimensional vector corresponding to the prediction classes. This architecture is particularly effective for sequence-level classification tasks, where only the final output of the sequence is relevant for the prediction.

CNN

The CNN model processes time-series data by applying 2D convolutions over the input matrix, treating time and feature dimensions as spatial dimensions. The model uses two convolutional layers followed by max-pooling and a dense layer.

```
class SimpleCNN(nn.Module):
    def __init__(self, input_channels=1, seq_len=30, num_features=24,
        dim=128, dropout=0.5):
        super(SimpleCNN, self).__init__()
        self.conv1 = nn.Conv2d(input_channels, 16, kernel_size=(3, 3),
        padding=1)
        self.conv2 = nn.Conv2d(16, 32, kernel_size=(3, 3), padding=1)
        ...
```

Listing 5.4: SimpleCNN: Model Initialization.

The convolutional layers apply local feature extractors on the reshaped input data. Max-pooling reduces the spatial dimensions and acts as a downsampling mechanism. After flattening the feature maps, the result is passed through a fully connected layer for classification.

The forward method is shown in Listing 5.5.

```
def forward(self, x):
    x = x.unsqueeze(1)
    x = self.pool(torch.relu(self.conv1(x)))
    x = self.pool(torch.relu(self.conv2(x)))
    x = self.flatten(x)
    x = self.dropout(x)
    return self.fc(x)
```

Listing 5.5: SimpleCNN: Forward Pass.

The input tensor is first reshaped to *[batch_size, 1, seq_len, num_features]*. The convolutions and max-pooling layers transform the data into a compact representation suitable for classification.

Transformer and Gated Transformer Network (GTN)

The Transformer model leverages self-attention mechanisms to capture long-range dependencies in both the temporal and feature dimensions of the input data. We consider two variants: a standard Transformer model without channel attention, which applies self-attention solely over the temporal dimension, and a GTN with *channel_attention=True*, which adds an additional feature-wise attention branch combined through a learned gating mechanism.

```
class TransformerNet(nn.Module):
    def __init__(self, seq_len=30, input_dim=24, d_model=256,
                 nhead=8, dim_feedforward=512, num_layers=4,
                 dropout=0.1, channel_attention=False):
        super().__init__()
        self.lin_time = nn.Linear(input_dim, d_model)
        self.pos_encoder = PositionalEncoding(d_model=d_model, dropout=dropout)
        self.transformer_encoder_time = nn.TransformerEncoder(
            nn.TransformerEncoderLayer(d_model, nhead, dim_feedforward, dropout),
            num_layers=num_layers
        )
        ...
```

Listing 5.6: TransformerNet: Model Initialization.

As shown in Listing 5.6, the model first applies a linear projection *lin_time* to map input features into a $|d_model|-$dimensional embedding space, followed by sinusoidal positional encoding

(Vaswani et al., 2017) to preserve temporal order information. The embedded sequence is processed by a stack of *TransformerEncoder* layers, each consisting of multi-head self-attention and feed-forward sublayers.

In the GTN, a channel-attention branch transposes the input and applies a second transformer encoder along the feature dimension, allowing the model to learn inter-feature dependencies. The outputs of both branches are projected via linear layers (*out_time* and *out_channel*) and combined through a gating mechanism to adaptively weight temporal and feature contributions adaptively.

The forward method is defined in Listing 5.7.

```
def forward(self, x_):
    x = torch.tanh(self.lin_time(x_))
    x = self.pos_encoder(x)
    x = self.transformer_encoder_time(x)
    x = x[0, :, :]

    if self.channel_attention:
        y = torch.transpose(x_, 0, 2)
        y = torch.tanh(self.lin_channel(y))
        y = self.transformer_encoder_channel(y)
        ...
    return self.classifier(x)
```

Listing 5.7: TransformerNet: Forward Pass.

The temporal encoder outputs a sequence representation from which the first token is selected as the global summary vector. When channel attention is enabled, this is combined with the feature-wise encoder output using learned gating weights. Finally, the concatenated or temporal-only representation is passed through a linear classifier to predict the target class probabilities.

By combining temporal attention, optional feature-wise attention, and positional encoding, this architecture effectively models both sequential dependencies and feature interactions.

5.2. Hyperparameter Optimization

To achieve optimal model performance, hyperparameter optimization was conducted using two distinct strategies: Grid Search and Optuna (Akiba et al., 2019).

Optuna

For a more efficient and flexible hyperparameter tuning approach, Optuna was integrated into the training pipeline. Optuna is a software framework for automatic hyperparameter optimization, featuring a *define-by-run* interface that allows dynamic construction of the search space (Akiba et al., 2019).

The Optuna search space included:

- `hidden_dims`: [128, 64], [256, 128], [512, 256]
- `batch_size`: 128, 256, 512, and 1024
- `learning rate`: log-uniform distribution between 1e−5 and 1e−2
- `dropout`: uniform distribution between 0.0 and 0.7
- `weight_decay`: log-uniform distribution between 1e−6 and 1e−2

- `gamma` (learning rate scheduler): uniform distribution between 0.1 and 0.9

Each trial dynamically updates the training configuration, initializes the model, and trains it using the same data pipeline and trainer class used in regular training. The objective function returns the validation AUPRC score, which is used by Optuna to guide the optimization.

To efficiently explore the search space, Optuna uses the *Tree-structured Parzen Estimator* sampler by default. Instead of testing all combinations exhaustively, it builds a probabilistic model of the objective function and prioritizes regions in the search space that have previously led to good results. This enables Optuna to focus the search on promising hyperparameter configurations and avoid less effective ones. Poor-performing trials can also be pruned early, saving computation time.

Grid Search

Hyperparameters are parameters that govern the training process and model structure but are not learned directly from the data. For the Random Forest model, hyperparameter tuning was done using *GridSearchCV* from the *scikit-learn* library. For DL models such as CNNs or LSTMs, a custom Slurm-based script was used to run exhaustive grid search combinations on the HoreKa cluster.

Listing 5.8 shows a relevant excerpt of the HoreKa job script used to tune the DL models:

```
for lr in 0.0005 0.001 0.002
do
  for dr in 0.0 0.025 0.05 0.075
  do
    for bs in 128, 256
    do
      python train.py \
        --config $CONFIG_TRAIN_PATH \
        --lr $lr \
        --dr $dr \
        --bs $bs

      MODEL_PATH="$SAVE_DIR/$MODEL_NAME/$RUN_ID/model_best.pth"

      python test.py \
        --config $CONFIG_TEST_PATH \
        --mp $MODEL_PATH
    done
  done
done
```

Listing 5.8: GridSearch script used on HoreKa for CNN model.

This approach allowed running multiple training jobs with different hyperparameter combinations in parallel on the cluster.

5.3. Implementation of XAI Methods

To enable interpretation and comparison of feature attributions across model architectures, two XAI methods were implemented: SHAP and IG. Both approaches follow a modular and extensible design, with parallel code structures and dedicated subdirectories (*shap_local* and *integrated_gradients*) that include scripts for computing and visualizing attributions. This setup ensures consistent usage across all DL models and the RF baseline.

SHAP

The SHAP implementation is model-type aware and automatically selects the appropriate explainer. For instance, the Random Forest model uses *TreeExplainer*, whereas DL models use *KernelExplainer* with custom model wrappers. SHAP values are computed on the test set and saved along with feature inputs, labels, and coordinates. Visualization utilities include among others grouped feature importance, waterfall plots, and beeswarm plots. Listing 5.9 shows exemplarly the integration of the SHAP plots from the SHAP library.

```
expl = shap.Explanation(values=shap_values, data=input_tensor.numpy(), feature_names=
    feature_names)
shap.plots.beeswarm(expl, max_display=25, show=False)
plt.savefig("shap_beeswarm_plot.png", dpi=300)
```

Listing 5.9: SHAP Beeswarm Plot for class 1 values.

Integrated Gradients (IG)

IG were implemented using the *captum.attr.IntegratedGradients* module from the open-source Captum library. IG values were computed for all DL models in which the input tensor is passed directly to the IG explainer along with a baseline input that represents the absence of features.

Since all input variables were standardized prior to model training (zero mean and unit variance), a tensor filled with zeros was used as the baseline input. This corresponds to the expected value of the features and is a common choice when data normalization has been applied. Listing 5.10 shows the typical IG computation for the DL models.

```
def get_baseline(input_tensor):
    return torch.zeros_like(input_tensor)

input_tensor.requires_grad_()
baseline = get_baseline(input_tensor)
ig = IntegratedGradients(model)
attributions = ig.attribute(input_tensor, baseline, target=target_class)
```

Listing 5.10: Integrated Gradients computation with a zero baseline.

The resulting attributions are saved per test sample and used to generate visualizations similar to SHAP, such as temporal heatmaps, bar plots, and beeswarm plots. All visualizations support filtering by predicted class and enable direct comparisons across different model architectures.

6. Results & Discussion

This chapter presents the evaluation of all ML models on wildfire danger prediction, combining both quantitative performance metrics and XAI analyses. We first compare predictive performance across models and assess their statistical significance. Subsequently, we investigate spatial patterns of predicted fire probabilities, analyze the impact of temporal context length, and conduct a detailed feature attribution study using SHAP and IG. Finally, we examine case studies, assess model alignment with physical domain knowledge, and discuss limitations to provide a comprehensive understanding of model behavior and reliability.

6.1. Performance on Wildfire Danger Prediction

All models demonstrate strong classification performance, with F_1-Scores exceeding 0.75 on the test set. Aligned with findings from prior wildfire prediction studies, such as Kondylatos, Prapas, Ronco, et al. (2022) and F. Li et al. (2024), DL models, achieve consistently higher results than baseline models like a Random Forest. Additionally, Transformer-based architectures achieve consistently high results across all phases, while more complex attention-based models such as the TFT or GTN do not exhibit statistically significant improvements over the baseline Transformer.

Compared to the results reported in Track A: Wildfire Danger Forecasting from Kondylatos, Prapas, Camps-Valls, and Papoutsis (2023), our models surpass the performance of the LSTM, Transformer, and GTN baselines (F_1-Scores of approximately 0.78) through the application of cross-validation and extensive hyperparameter tuning. Furthermore, by incorporating additional architectures (CNN, GRU, MLP, TFT, RF), we achieve higher F_1-Scores exceeding 0.80 for nearly all DL models, with all architectures except for the RF and TFT outperforming the baselines provided in Kondylatos, Prapas, Camps-Valls, and Papoutsis (2023).

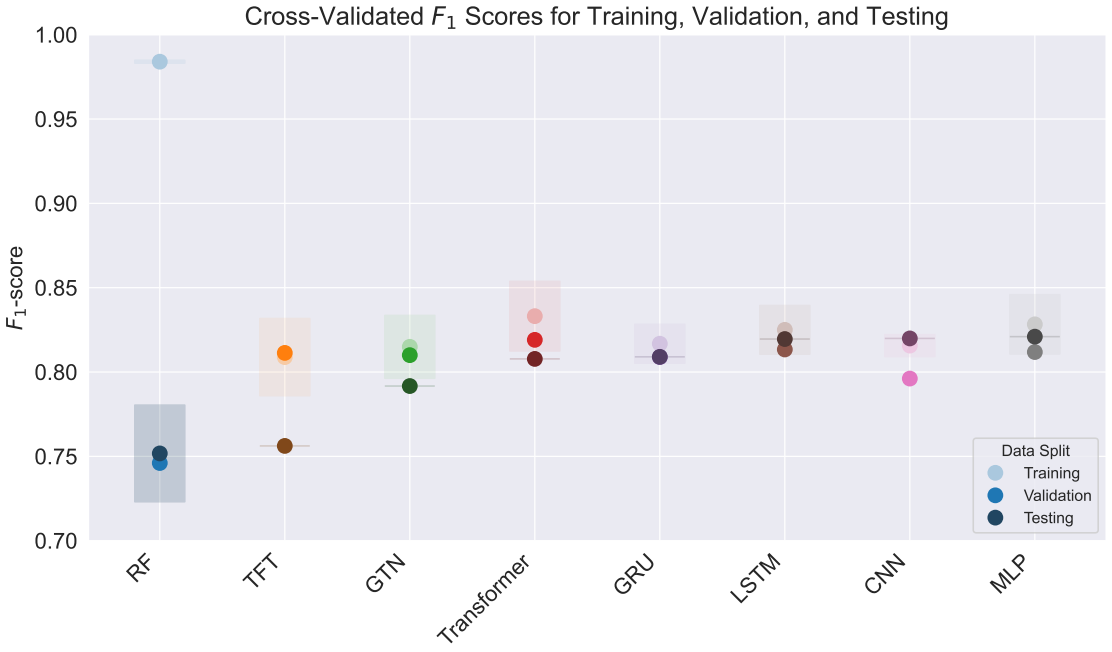

Figure 6.1.: **F_1-Scores across Training, Validation, and Test Sets.** The figure shows three points per model with shaded boxes indicating the standard deviation across cross-validation splits for training and testing. The Random Forest model exhibits strong overfitting compared to the neural network architectures.

RF shows relatively high F_1-Scores on the training set (Figure 6.1) but lower performance on validation and test data, indicating moderate overfitting. In contrast, all DL models display narrow error bars, denoting standard deviation across cross-validation splits, suggesting that they generalize well and exhibit robust behavior across data partitions.

Beyond overall performance, we also examine recall (fire detection) and precision (minimizing false alarms). High-recall models such as the TFT (REC: 0.831) and GTN (REC: 0.813) detect most fires but trade this for lower precision, resulting in more false alarms. Prioritizing recall is crucial, as missing an actual fire (FN) poses a greater risk than issuing additional warnings.

In contrast, the Transformer (PR: 0.805) combines strong precision with solid recall, offering well-balanced performance. Similarly, the MLP (PR: 0.772) achieves high precision but at the cost of missing more fire events.

Additionally, AUPRC scores are largely consistent with the F_1-Score results and are shown in Figure B.1 in the appendix.

Following the approach of Nowack et al. (2020), we assess the significance of performance differences across models using a Analysis of Variance (ANOVA) *F-Test* and a KruskalWallis test (Kruskal & Wallis, 1952). As both tests reject the null hypothesis that all models perform equally well, differences in the Testing Set are highly statistically significant, with p-value of 7.135×10^{-61} for a standard one-way ANOVA F-test and a p-value of 3.599×10^{-9} for a non-parametric KruskalWallis-test, confirming that at least one model differs strongly from the others.

Since the KruskalWallis test indicates significant differences, we conduct a post-hoc Dunn test (Dunn, 1964) to determine which specific pairs of models differ significantly. The results, visualized as a heatmap in Figure B.2 in the in appendix reveal that the Random Forest model significantly underperforms compared to all neural network models, while the performance differences among the DL models are not statistically significant.

Nonetheless, the Transformer model achieves slightly better results in terms of $F1-$Score on the original fixed chronological split used in the Mesogeos study (2006–2019 training, 2020 validation, 2021–2022 testing) compared to all other architectures, and is therefore selected as the primary model for further XAI-based interpretability analyses.

6.2. Distribution of Fire Danger Probabilities

In addition to the quantitative performance metrics, we visualize the spatial softmax probabilities predicted by the Transformer model on the test dataset, as illustrated in Figure 6.2.

This visualization provides an intuitive understanding of model behavior. As expected, the majority of positive fire events are associated with high softmax probabilities (i.e., red points), while most negative samples are associated with low probabilities (i.e., blue points). This suggests that the model successfully distinguishes between high and low fire danger instances.

However, spatial discrepancies emerge: the model tends to be more accurate in coastal regions, with particularly strong predictive accuracy observed in regions such as Croatia, Albania, and Greece, as well as on the island of Sicily and the northern coast of Algeria. This spatial heterogeneity is further reflected by the presence of blue points among predicted positives in the interior, and red points appearing among negatives near the coast.

One possible explanation is that wildfires occur more frequently near the coast in the Mediterranean from $2006-2022$, resulting in a higher concentration of training samples in these regions. We tested

this hypothesis by calculating the Euclidean distance between each positive sample and the nearest point along the coast. Fires occurring within a 100 km radius of the nearest coast were classified as coastal. The results confirm this assumption: 72.5% of positive fire events are located near the coast, while only 27.5% occur inland. In contrast, inland areas contain significantly fewer positive samples and are dominated by negative examples. While the overall class distribution in the dataset is already imbalanced with approximately two thirds of samples labeled as negative and one third as positive, this imbalance is further exacerbated geographically. If most of the positive samples are concentrated in coastal areas and the majority of negative samples are located inland, the model may struggle to generalize well to underrepresented inland fire scenarios.

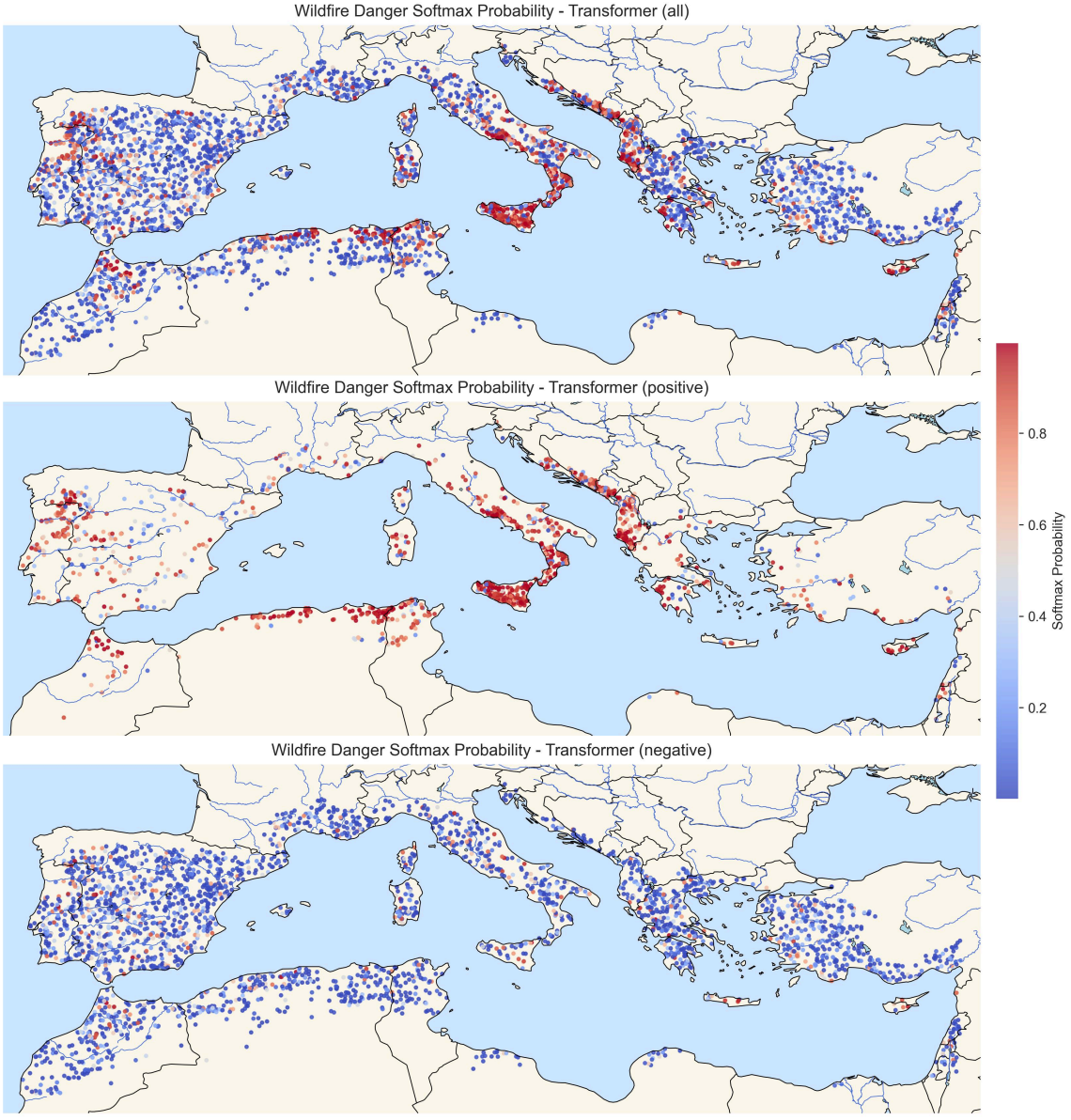

Figure 6.2.: **Transformer Model Predictions for Softmax Fire Probabilities on Test Data.** Top: all test samples. Middle: only negative samples. Bottom: only positive samples. Color indicates predicted fire probability, with red indicating high danger and blue indicating low danger.

6.3. Impact of Temporal Context Length on Model Performance

To evaluate how the length of historical input (time lag) influences model performance, we compare the F_1-Scores of Transformer and LSTM models across varying input durations from 5 to 30 days. As shown in Figure 6.3, both models perform well across all time lags, but the Transformer consistently outperforms the LSTM, regardless of the input length or data split (training, validation, testing).

The Transformer benefits from longer temporal contexts, showing a near-linear performance increase up to 30 days. This suggests that attention-based models are better equipped to leverage long-range dependencies. In contrast, the LSTM's performance saturates around 15 days and declines thereafter, likely due to the models limited ability to retain long-term dependencies, constrained by the forget gate mechanism inherent in its architecture.

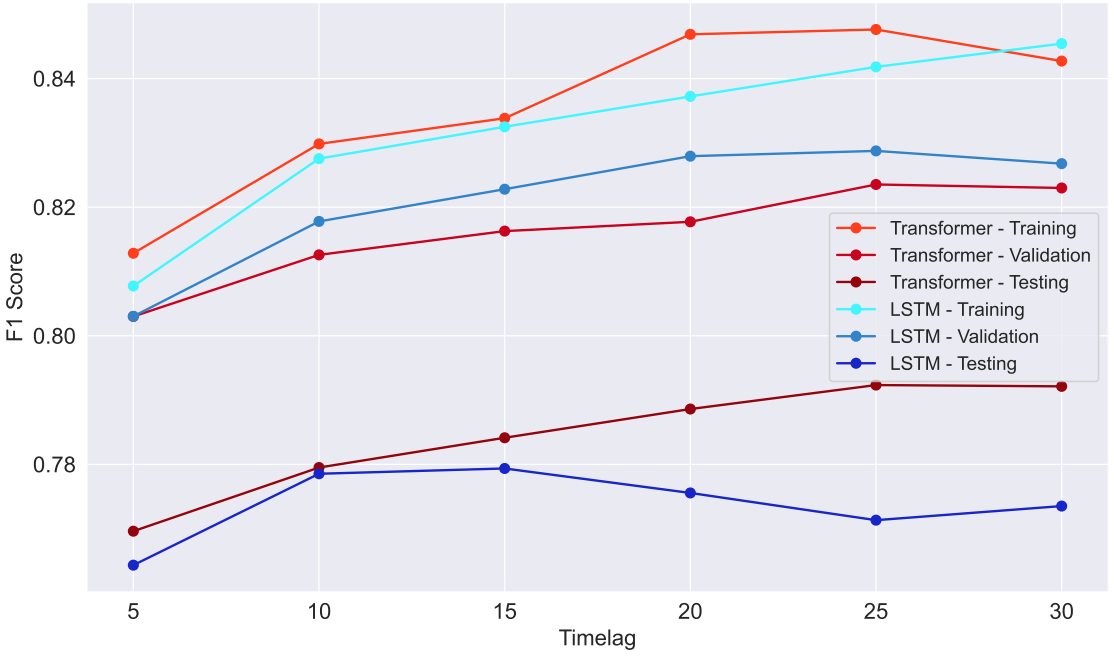

Figure 6.3.: F_1-Scores across different time lags (530 days) for LSTM and Transformer. Each model is evaluated on three splits (training, validation, testing), color-coded by tone: red shades for Transformer, blue shades for LSTM. Performance generally increases with time lag for Transformers, while LSTM plateaus after 15 days.

These observations support the hypothesis that Transformer architectures are inherently more effective at extracting useful temporal patterns from longer daily sequences. For applications where extended historical context is available, Transformer-based models are preferable. However, shorter contexts still yield competitive results for both architectures.

These findings are in line with prior work, such as Prapas, Bountos, et al. (2023), who conducted a similar time lag sensitivity analysis using the SeasFire cube (Alonso et al., 2024), evaluating forecasting windows up to 128 days with 8-day temporal resolution at global scale. Although not directly comparable due to differences in temporal granularity and prediction targets, their findings similarly indicate that Transformer-based models experience a slower decline in performance and eventually reach a plateau. Similarly, Michail et al. (2024) showed that models trained with longer

time series achieve better and more stable performance but eventually saturate when attempting very long-range forecasting, particularly for recurrent architectures such as GRUs or LSTMs.

6.4. SHAP Analysis

In this section, we used SHAP introduced by Lundberg and Lee (2017) to interpret model predictions and identify the most influential features driving fire probability estimates. Global and local analyzes were performed to reveal general patterns of feature importance across models, while case-specific samples were examined in greater detail to uncover recurring patterns in well-predicted fires and to identify systematic challenges in cases where the models failed.

Importantly, we analyzed both individual lagged variables (e.g., *t2m_t-1*) to capture short-term dynamics before fire ignition, as well as temporally aggregated features over the 30-day time period prior to fire to assess longer-term environmental signals. Additionally, we benchmark model explanations against established fire science to assess physical consistency and domain alignment.

6.4.1. SHAP Feature Importance

To gain a deeper understanding of the model decision processes, SHAP Beeswarm plots were generated by aggregating SHAP values across all 30 time steps for each of the 24 features. Figure 6.4 compares the SHAP summary plots for both the Transformer and Random Forest models. Each point represents a single sample, colored by feature value (red = high, blue = low), with features ordered by average importance.

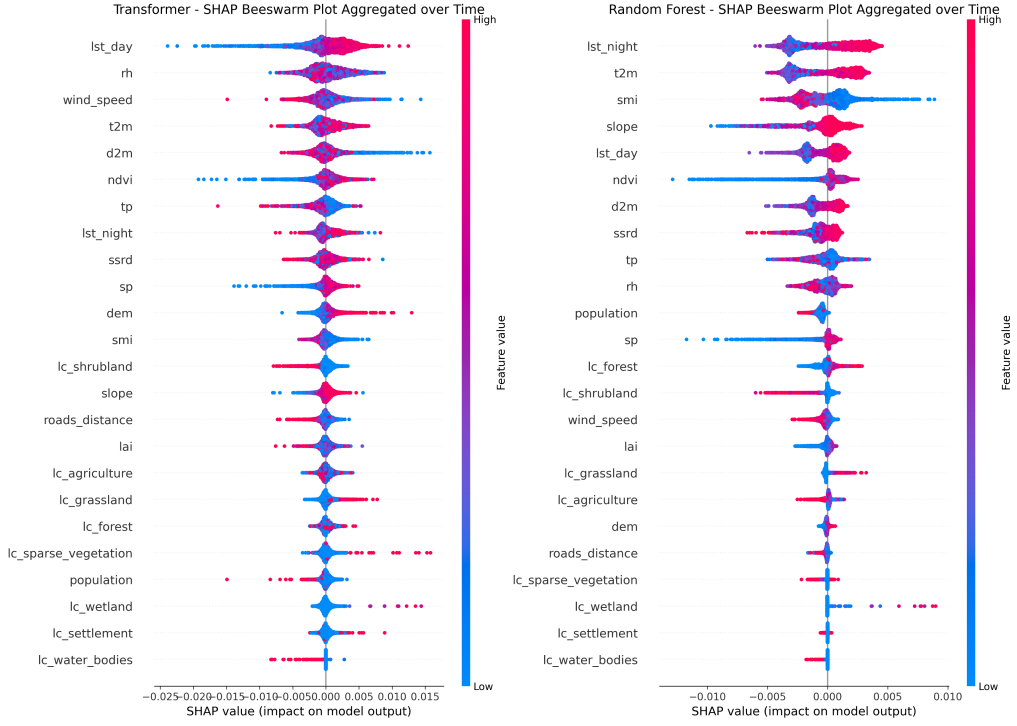

Figure 6.4.: **Comparison of SHAP Beeswarm Plots for the Transformer (left) and Random Forest (right).** SHAP values are aggregated over time steps, and colored by feature value (red = high, blue = low). Features are sorted by mean importance.

Both models exhibit broadly consistent patterns: dynamic weather and vegetation features dominate model predictions, whereas static variables such as land cover categories and population

density play a minor role. In particular, features such as *Surface Pressure (sp)*, *Wind Speed (wind_speed)*, (*Normalized Difference Vegetation Index (ndvi)*), *Total Precipitation (tp)*, and *Distance from Roads (roads_distance)* show similar directional effects and importance across both models.

However, key differences emerge. *Temperature at 2m above the Earths surface (t2m)* and *Night land surface temperature (lst_night)* are highly influential for the Random Forest, showing wide SHAP value distributions and clear feature effects, but are less impactful for the Transformer. Conversely, *Dewpoint temperature at 2m (d2m)* ranks highly for the Transformer and aligns with physical expectations (Chuvieco et al., 2023), but is less relevant for the Random Forest. Notably, *Soil Moisture Index (smi)* is one of the top predictors in the Random Forest, where higher *smi* reduces fire risk.

To further assess the robustness of the explainability results, we computed feature attributions for the Transformer model using IG as show in Figure B.3 in the appendix. Although SHAP and IG are not directly comparable in scale, the direction of the feature effects on the prediction remains consistent across both methods. IG were not applied to the Random Forest model, as it does not support gradient-based interpretation. The results confirm the SHAP findings, with similar dynamic features, such as *lst_day*, *Relative Humidity (rh)*, and *wind_speed*, as most relevant. For the majority of variables (e.g., *Elevation (dem)*, *ndvi*, *sp*), IG and SHAP reveal a consistent direction of effect, where higher feature values correspond to higher attribution values or vice versa.

6.4.2. Comparison of SHAP with Random Forest Feature Importance

To complement the SHAP analysis, we also consider the aggregated Feature Importance from the Random Forest model as shown in 6.5 similar to the approach of Cilli et al. (2022). Interestingly, almost all features overlap with the SHAP results. Particularly temperature variables such as the day and night land surface temperature (*lst_night*, *lst_day*), and the temperature at 2m above the Earth's surface (*t2m*) are the top-ranked features in both XAI methods. In addition, fuel-related indicators such as the (*Normalized Difference Vegetation Index (ndvi)*) and (*Leaf Area Index (lai)*) appear relatively important, although these variables also exhibit the largest discrepancies between the two XAI methods. By contrast, land cover classes (e.g., *Land cover class: forest (lc_forest)*, *Land cover class: shrubland (lc_shrubland)*, *Land cover class: settlement (lc_settlement)*) and human-related variables (e.g., *population*) contribute very little to the models decisions, possibly due to coarser information in these variables, as these values only change annually.

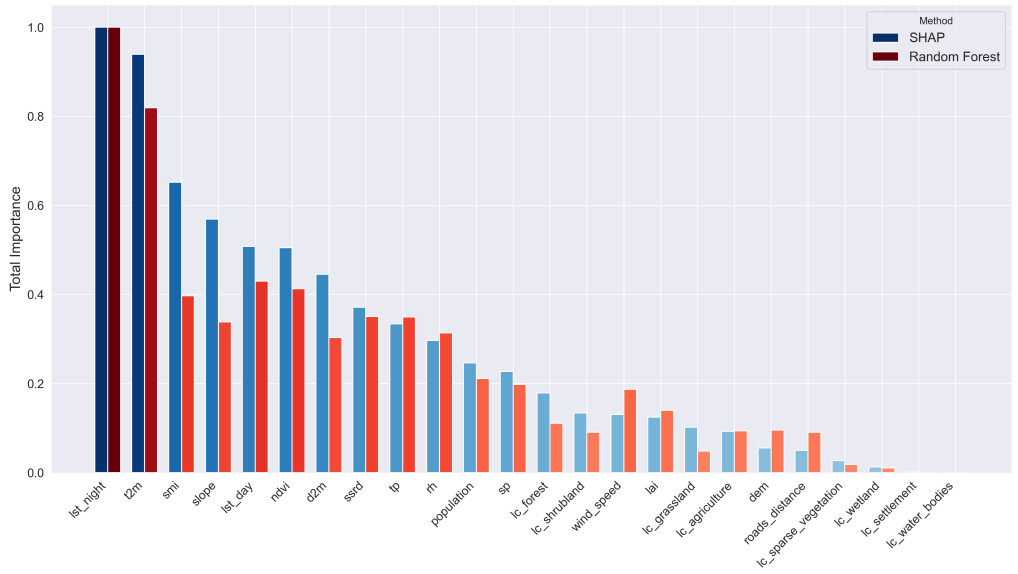

Figure 6.5.: **Comparison of Aggregated Feature Importance from SHAP and Random Forest Feature Importance.** SHAP values are aggregated as the sum of absolute attributions over all samples and time steps, while Random Forest Feature Importance are obtained from the *feature_importances_* attribute of *scikit-learn*'s *RandomForestClassifier*, based on the MDI criterion. Since SHAP and RF Feature Importance lie on different scales, a min-max normalization was applied.

6.4.3. SHAP-based Feature Analysis across Models

To assess the general relevance of input features for fire classification, we computed the mean absolute SHAP values for each feature, averaged across all models. As displayed in descending order in Figure 6.6, higher values indicate that a feature has, on average, a stronger influence on the model's predictions.

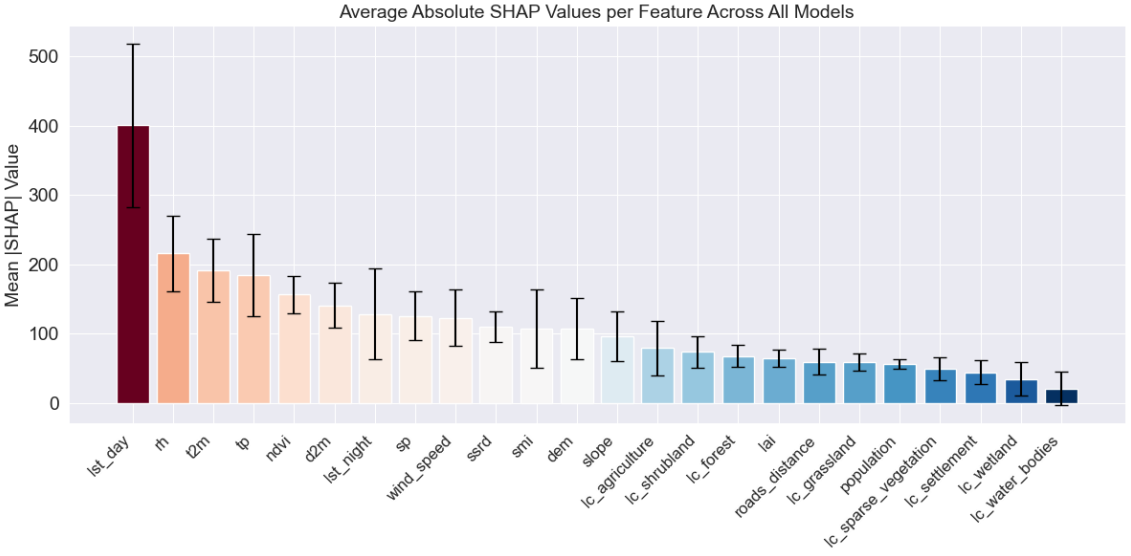

Figure 6.6.: **Average Absolute SHAP Values per Feature across all Models.** Bars show the mean absolute SHAP value, and error bars denote the standard deviation across models.

The most influential feature in all models is clearly the day land surface temperature (*lst_day*),

followed by several key meteorological drivers such as (*rh*), temperature at 2 meters (*t2m*),(*tp*), and (*ndvi*).

In contrast, static variables, especially land cover categories such as *lc_shrubland*, *lc_forest*, or *Land cover class: wetland (lc_wetland)*, tend to show much lower SHAP values, along with consistently small standard deviations across models, indicating consistently low influence across models. This can be partly explained by the phenomenon of explainable variance: dynamic meteorological variables (e.g., temperature, *rh*, *wind_speed*) exhibit higher variability and thus explain more of the short-term variation in fire occurrence, as they directly drive ignition and spread conditions. Only when these factors align, does fire become likely.

However, such favorable ignition conditions must coincide with flammable fuels. If the land cover corresponds to non-burnable areas, such as water bodies, no fire will occur, regardless of meteorological extremes. In this sense, land cover acts more as a precondition or binary constraint, while meteorological drivers explain the variance in observed fire patterns. Their low SHAP values do not imply irrelevance, but rather that their filtering effect is more binary: if the wrong fuel type is present, the output is always "no fire", making them less influential in explaining variation within actual fire zones. Moreover, there is likely considerable uncertainty in land cover datasets, which could further contribute to this behavior by reducing the reliability of land cover as a predictor.

To complement this analysis, Figure 6.7 shows a heatmap of relative feature ranks in the models. Here, each cell represents the rank of a feature for a given model. Unlike the barplot in Figure 6.6, this representation is ordinal and does not consider the magnitude of SHAP values.

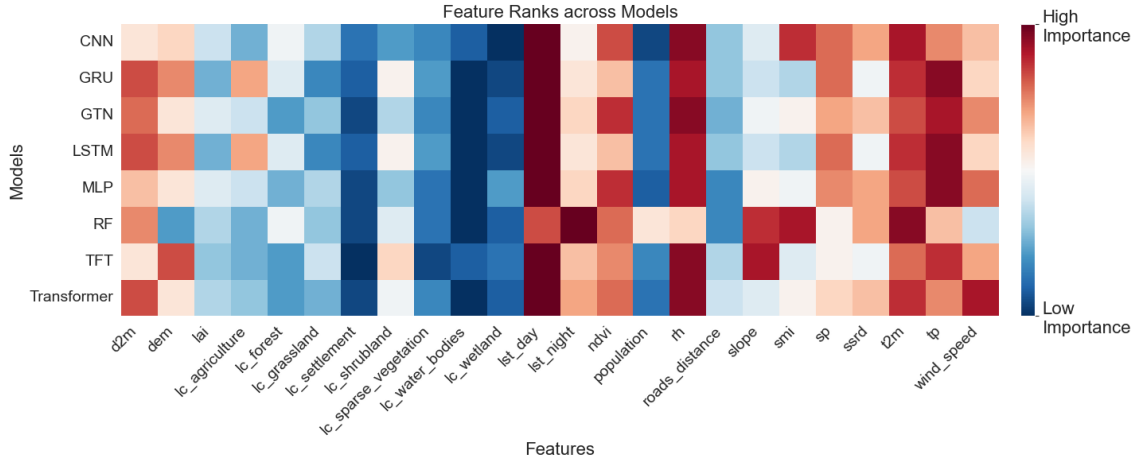

Figure 6.7.: **Feature Ranks across Models.** Lower ranks (dark red) indicate higher importance, while higher ranks (blue) reflect lower relevance. Ranks are based on mean SHAP values per feature and model.

Variables such as *lst_day*, *rh*, *t2m* and *tp* are among the most relevant across nearly all models, confirming their importance from the barplot view in Figure 6.6.

LSTM and GRU show nearly identical feature importance patterns, reflecting their similar recurrent architectures (see Chapter 4.2). Both emphasize meteorological inputs while down-weighting static variables.

Interestingly, surface pressure (*sp*) is ranked as important by most neural networks, but not by TFT or RF. Similarly, solar surface radiation (*Surface Solar Radiation Downward (ssrd)*) is assigned high importance in CNN, GTN,MLP RF, and Transformer models, but is largely disregarded by

GRU, LSTM, and TFT. The feature *wind_speed* stands out in the Transformer, where it receives significantly higher importance than in any other architecture. In contrast, *Slope of the Area (slope)* ranks highly only in TFT and RF, while *smi* is particularly influential in RF and CNN but less so in neural models.

In general, the RF model exhibits a distinct feature weighting pattern compared to the DL models. It assigns disproportionately high importance to variables such as *smi*, *slope*, *t2m*, and *lst_night*, while down-weighting others like *lst_day*, *rh*, and *wind_speed* which are considered highly influential by most neural architectures. This discrepancy may be explained by underlying differences in model architectures. Tree-based models such as Random Forest conduct implicit feature selection via hierarchical splitting, often prioritizing variables with strong individual predictive power. In contrast, DL models form layered representations that capture interactions and nonlinear combinations across features. As a result, Random Forest tends to concentrate attribution on a few dominant variables, whereas neural models distribute importance more evenly across multiple interacting inputs (Harris & Grzes, 2019; Kong, 2024).

6.4.4. Temporal Dynamics of SHAP Feature Importance Across Models

To analyze how individual features contribute to predictions of fire probability between models, SHAP comparison plots were generated for selected input variables and visualized in different temporal contexts in Figure 6.8. Each row corresponds to a feature and the SHAP distributions are shown for all models under three temporal framings: t-1, t-30, and aggregated over 30 days. Positive SHAP values indicate an increased of fire, and negative values suggest a reduced risk. It is important to note that the x-axis scales differ between subplots due to variations in SHAP value magnitude, which are typically more pronounced in t-1 plots compared to the aggregated ones.

Several consistent patterns emerge:

First, models tend to exhibit greater consistency and directional consistent for t-1 features than at earlier or aggregated time points. For example, day land surface temperature (*lst_day*) shows a strong and consistent positive relationship with fire probability across all models at t-1, whereas this effect is diminished at t-30. Similar temporal trends are observed for (*rh*), and *wind_speed*, where models exhibit clearer directional relationships immediately prior to ignition. These findings are further supported by an IG analysis as show in Figure B.4 in the appendix, which reveals comparable temporal patterns for key predictive features. The IG method was applied to all models except for the RF model, which is not compatible with gradient-based attribution technique. It is important to note that the absolute magnitudes of SHAP and IG values are not directly comparable, as they are derived from fundamentally different interpretability frameworks.

Second, time aggregation can soften sharp feature effects. This is especially evident in features like *lst_day*, or for (*rh*) which show a strong effect at t-1 become more heterogeneous when averaged across all 30 days. This suggests that relevant fire predictors may only become impactful shortly before ignition.

Conversely, a few features, such as (*smi*) and the land cover category *Land cover class: grassland (lc_grassland)* only reveal a clear predictive relationship when aggregated over time. These features may not exhibit strong temporal variability, but their cumulative effect contributes meaningfully to the model output.

Third, certain static vegetation-related feature, such as the forest cover class *lc_forest*, exhibit inconsistent SHAP effects across both models and time scales. These inconsistencies may indicate

model uncertainty in capturing reliable relationships between fuel types and fire occurrence, likely due to the lack of temporal dynamics in these static input layers or because there is likely considerable uncertainty in land cover datasets as explained in Section 6.4.3. IG analysis further confirms this instability in directional attribution for *lc_forest* across architectures as show in Figure B.4 in the appendix.

Interestingly, *wind_speed* exhibits one of the few notable disagreements between SHAP and IG: while SHAP values at t-1 suggest a strong inverse relationship, IG reveals a less pronounced or opposite effect. However, when temporally aggregated, the two explanation methods converge toward more aligned attributions, indicating that some of these inconsistencies may be driven by local model sensitivity near ignition time.

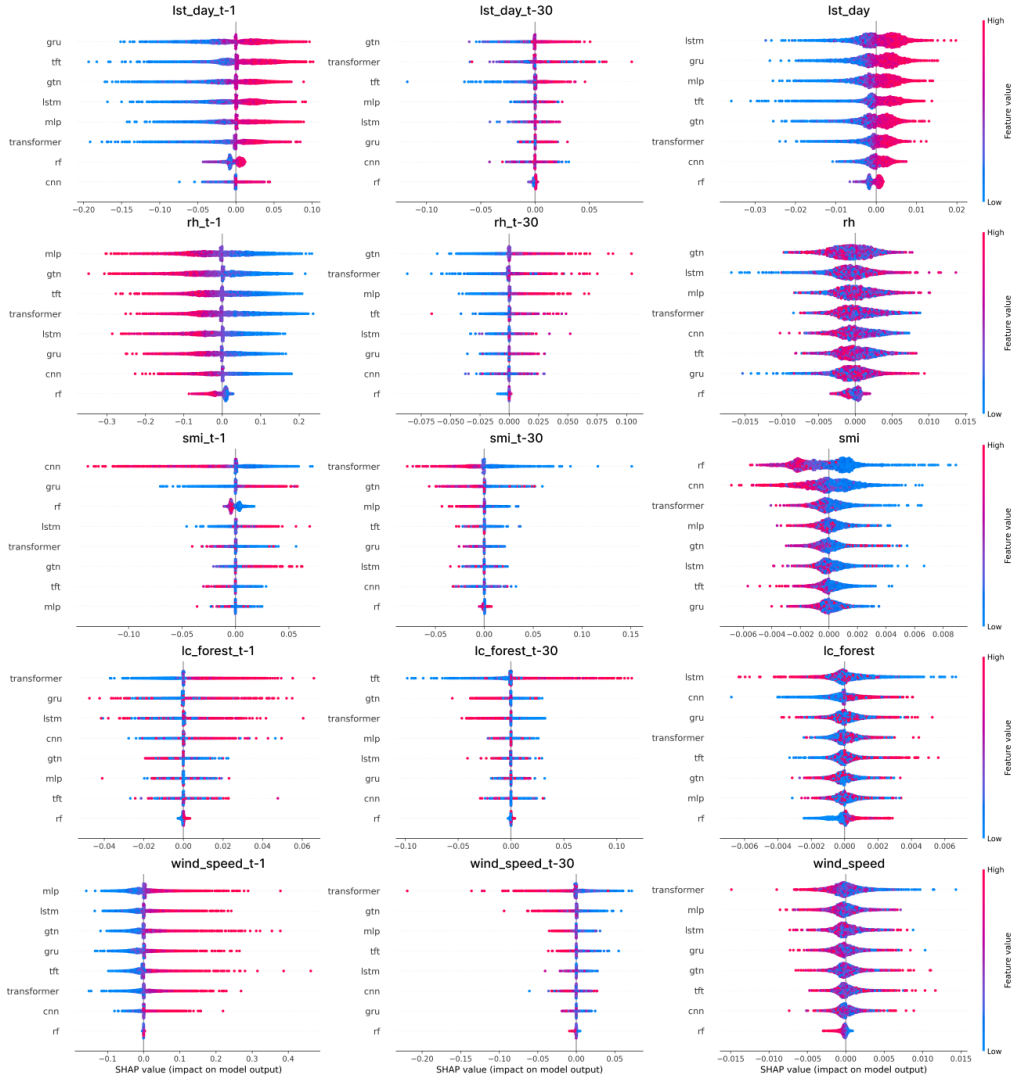

Figure 6.8.: **SHAP Comparison Plots across all Models for selected Features.** Left: SHAP values for input features at t-1 (one day prior to ignition). Middle: SHAP values for the same features at t-30. Right: SHAP values aggregated across all 30 time steps. Colors indicate normalized feature values (red = high, blue = low).

6.4.5. SHAP-Based Analysis of Model Alignment with Physical Domain Knowledge

Building on the work of F. Li et al. (2024), who emphasized the importance of evaluating models beyond predictive accuracy, we assess the physical interpretability of our ML models. Therefore, we computed a physical consistency score. For each feature and sample, we compared the direction of the SHAP value (positive or negative) with the corresponding normalized input value, classified as high (> 0) or low (< 0) against the expected physical relationship (positive or negative). The expected sign of each relationship was derived from established fire behavior literature and is documented in Table A.2 in the appendix.

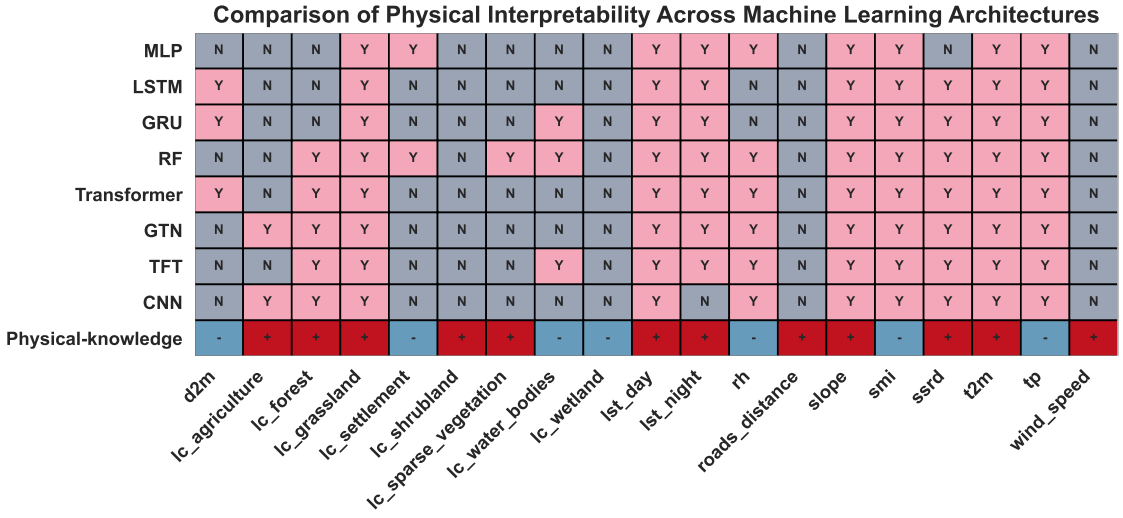

Figure 6.9.: **SHAP-based Physical Consistency Matrix.** Each cell indicates whether the model's SHAP-based explanation for a given feature was physically consistent ("Y") or not ("N"). The bottom row shows the expected physical relationship between each variable and fire occurrence, with "+" indicating a positive relationship and "−" a negative one.

A SHAP value greater than zero indicates that the feature contributes to the prediction of fire occurrence (class 1), while a SHAP value below zero suggests a contribution to the absence of fire (class 0). A sample was considered physically consistent if, for positively related features, high input values were associated with positive SHAP values or low inputs with negative SHAP values. For negatively related features, physical consistency was assigned when high inputs led to negative SHAP values or low inputs to positive SHAP values. We then aggregated the number and proportion of physically consistent samples for each feature. Figure 6.9 summarizes this benchmarking across models by comparing the model explanations to established physical relationships.

For example, there are clear positive relationships for temperature variables like *t2m*, *d2m*, *lst_day*, and *lst_night* that promote fuel drying and thus fire ignition, aligning with findings by Chuvieco et al. (2023) and Di Giuseppe et al. (2025). *wind_speed* is another strong positive driver, as it accelerates the spread of fire and oxygen supply, increasing the probability of fire (Pimont et al., 2012; Weise & Biging, 1997). Land cover types that offer continuous or dry fuel loads, such as *Land cover class: agriculture (lc_agriculture)*, *lc_forest*, and *lc_grassland* show as well positive associations with fire risk (Mukunga et al., 2023; Radeloff et al., 2023). In contrast to the aforementioned positive drivers, we also observe clear negative relationships with variables associated with climate wetness and non-flammable land cover types. Following the approach of F. Li et al. (2024), we

incorporate predictor values from one month prior to fire events, thus capturing the antecedent fuel conditions and moisture dynamics. Since increased fuel moisture reduces flammability and inhibits ignition, we assume negative associations between large probability of fire and variables related to wetness such as the *smi*, *rh*, and *tp* (Andela & van der Werf, 2014; Chuvieco et al., 2023; Holsten et al., 2013; Zhao et al., 2025). Similarly, *Land cover class: water bodies (lc_water_bodies)* and *lc_wetland* show negative effects, as moist or aquatic environments are less flammable and may act as natural firebreaks (Chuvieco et al., 2023; Zhao et al., 2025).

However, the relationship between fire occurrence and certain climate or environmental variables is complex, making it difficult to assign a consistent directional effect to some variables. Therefore, several features, most notably *ndvi*, population density, *Elevation (dem)*, *Surface Pressure (sp)*, and *Leaf Area Index (lai)* were not included in Table A.2 due to ambiguous or highly context dependent relationships. For instance, the absence of a clear relationship for population density is explained by its highly context-dependent and non-monotonic effects on burned area, which vary across regions and land-use types (Bistinas et al., 2013). Similarly, L.-M. Li et al. (2009) identified non-linear relationships between population density and forest fire probability, consistent with findings that fire risk tends to increase with population density, particularly in rangelands, but declines beyond region-specific population thresholds or in cropland areas (Bistinas et al., 2013).

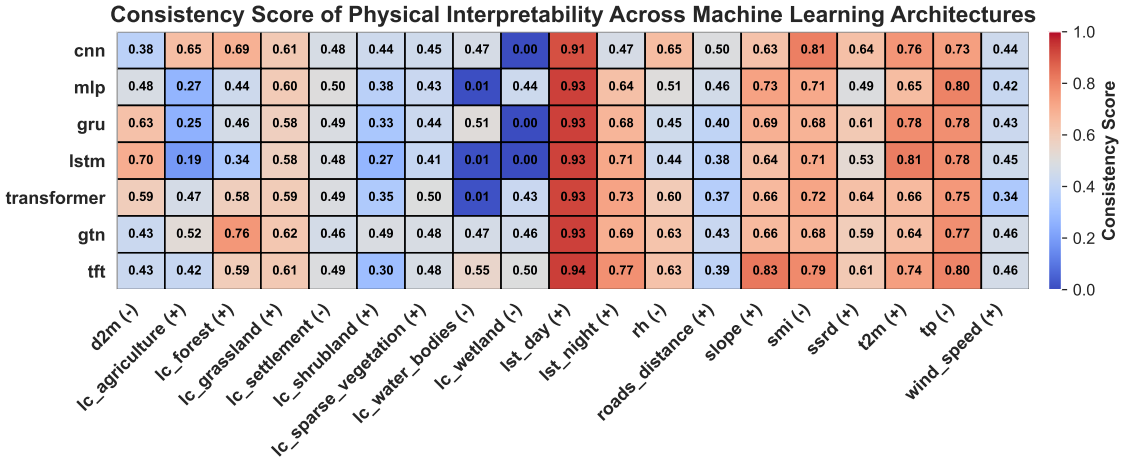

Figure 6.10.: **Matrix of Continuous Physical Consistency Scores.** Each cell represents the proportion of samples for which the SHAP-based explanation for a given feature was physically consistent with domain knowledge. Darker shades indicate higher consistency.

Compared to other ML architectures, more complex DL models such as Transformer and GTN demonstrated a relatively high degree of physical interpretability, correctly capturing 11 out of 19 expected relationships. In contrast, simpler models like MLP or LSTM identified fewer physically consistent patterns. Interestingly, the Random Forest benchmark model outperformed all others in terms of physical interpretability, capturing 13 out of 19 relationships, including several land cover classes representing different fuel types (e.g., *lc_settlement*, *lc_shrubland*, *lc_water_bodies*) that no other model was able to correctly represent. A subset of variables, mainly static features (e.g., *lc_wetland*, *lc_shrubland*, *wind_speed*) remained constant throughout the year, were not correctly captured by any model, suggesting their limited relevance for the final prediction. These results emphasize the need to assess the reliability of the model beyond the performance metrics alone (Eyring et al., 2019; F. Li et al., 2024).

To further distinguish how consistently these relationships were reflected across all samples, a

continuous physical consistency score was calculated. This score, as shown in Figure 6.10 reflects the proportion of samples for which the SHAP value of each feature is aligned with its expected physical effect.

6.5. Feature Sensitivity via Accumulated Local Effects (ALE)

To investigate non-linear effects of individual input features on the model output, we generated ALE plots for selected variables, including day land surface temperature (*lst_day*), *rh*, *ssrd*), and *tp* that exhibit particularly nonlinear relationships. ALE plots quantify the average change in the prediction as the feature value varies locally, providing a model-agnostic view of nonlinear feature effects (Apley & Zhu, 2020).

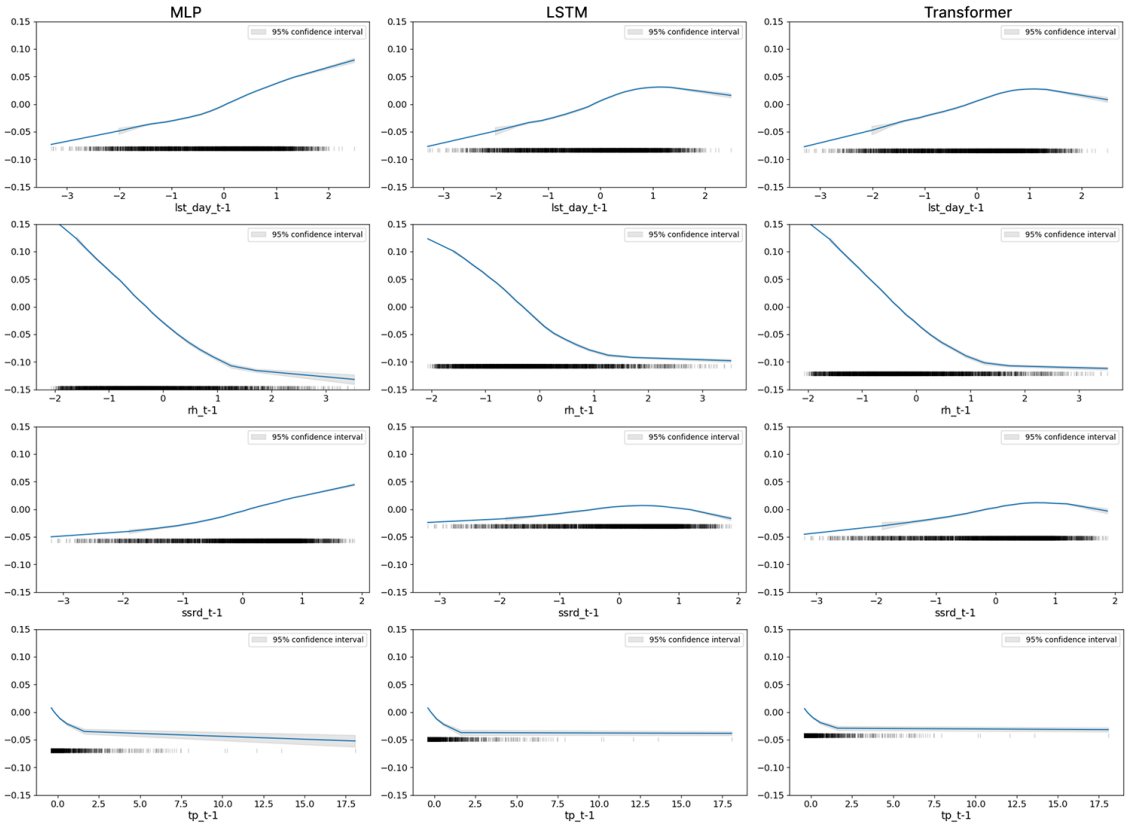

Figure 6.11.: **ALE Plots for Four Key Environmental Variables across three DL Architectures.** Each subplot shows the marginal effect of a single feature on the predicted fire probability. All plots are centered and y-axis standardized to the range [0.15, 0.15] to enable comparability across models.

As shown in Figure 6.11, *rh* and *lst_day* emerge as the most influential variables, consistently producing large ALE amplitudes. *rh* shows a strong negative relationship with fire probability, with a clear saturation effect, that beyond a certain level, further increases have no additional impact. *tp*) exhibits a similar pattern, sharp initial decrease in predicted risk, followed by a plateau, particularly visible in LSTM and Transformer models.

In contrast, *ssrd* has a weaker influence overall. While complex models like LSTM and Transformer show slight non-linear trends, the effect remains small, and MLP responses are nearly flat. Daytime land surface temperature (*lst_day*), however, demonstrates strong and varied effects: LSTM, GRU, Transformer, and GTN exhibit steep non-linear increases, while MLP and TFT show simpler, near-linear responses.

6.6. Case Studies of Model Predictions

To further interpret the model decisions at the sample level, we use SHAP *waterfall plots*, which visualize how individual features contribute to a specific prediction. These plots provide a transparent view into why the model predicted a given instance as fire or no fire. Inspired by previous work on wildfire prediction in southern Europe (Cilli et al., 2022), we illustrate the behavior of this visualization method across all four classification outcomes: True Positive, False Positive, False Negative, and True Negative.

In these plots, features are ordered by a descending absolute SHAP value, and each bar indicates the contribution of that feature to the final output of the model. The length of the bar shows the magnitude of the influence, while the sign and color (positive, red, or negative, blue) indicates the direction of the contribution toward or against the predicted fire probability. Waterfall plots should be read from top to bottom: the model starts from a base value, and each subsequent feature adjusts the classification score $f(x)$. Here, the baseline value $E(f(x))$ represents the mean model output across all samples, while the instance-specific value $f(x)$ captures the cumulative effect of the feature contributions relative to this global mean, as detailed in Section 6.4. The small gray tick marks on the left axis indicate the normalized feature values observed in the specific sample. If the final model output is greater than zero, the sample is classified as fire (class = 1); otherwise, it is labeled no fire (class = 0). These plots reflect feature importance aggregated over all 30 input days.

True Positive: Greece, 02 August 2021

This sample corresponds to the most severe fire in the test dataset, located on Elimnii Island. On August 3rd, a fire ignited near Varympompi, just north of Athens, under extreme weather: the air temperature exceeded $42°C$, relative humidity dropped to 10%, and wind speeds surpassed 15 km/h (Xanthopoulos, 2021). The top left waterfall plot in Figure 6.12 clearly captures this meteorological extremity. Key features such as *rh*, *slope*, and *ndvi* contribute strongly in the positive direction, with all top-ranked features aligned with physical expectations. The average softmax probability across all models for this instance is exceptionally high (0.986), demonstrating strong consensus in predicting a fire. It is important to note, however, that this softmax probability is computed independently of the SHAP estimation and should not be confused with the instance-specific value $f(x) = 0.688$. This example illustrates a clear alignment between real-world conditions and the models explanation.

False Positive: Greece, 30 July 2021

This false positive (top right) sample, also from Greece, shows a similar feature signature: high importance for variables like land surface temperature *lst_day*, *rh*), and land cover types (e.g., *lc_wetland* and *lc_grassland*. Despite the absence of a fire, for example the Transformer predicts a high softmax probability of fire (0.994). In particular, although meteorological drivers are influential, additional fuel-related factors such as *ndvi* and land cover also contribute to the prediction. This highlights the potential for false positives when vegetation and climatic signals resemble fire-prone conditions but do not lead to ignition.

False Negative: Spain, 15 June 2022

This sample (bottom left) corresponds to one of the largest fires in the test set. On June 15th, dry thunderstorms and resulting lightning ignited multiple wildfires in Castile and León under severe conditions (60 km/h wind) (Copernicus Emergency Management Service data, 2022; Wikipedia,

2022b). Yet, the model predicts a negative class with a low probability (0.2701). Although the prediction is incorrect, the feature contributions suggest a more nuanced picture. Some conditions are indeed conducive to fire: air temperature (*t2m*) is quite high, vegetation indicators such as *ndvi* and *smi* point to available biomass, and *rh* is low, indicating dry conditions. However, key drivers such as (*lst_day*), and (*Dewpoint temperature at 2m (d2m)*) are not particularly extreme and contribute negatively to prediction. Thus, while there is potential for fire based on some environmental variables, the overall signal is not strong enough to trigger a positive prediction. Importantly, the reported ignition event cause itself, lightning, is not explicitly represented in the dataset. Based on the available environmental and meteorological variables, the model estimates a low probability of fire. However, the dry thunderstorm provided a critical ignition source, leading to the development of large-scale fire despite otherwise unsuspicious fire-prone conditions.

True Negative: Spain, 06 July 2022

This sample (bottom right) represents a correctly classified no-fire instance in the Province of Seville, Andalusia, Spain. Despite being located in a fire-prone region, the model assigns a negative prediction with a softmax probability of 0.271. The SHAP contributions show that most features push the prediction toward the negative class, with meteorological and vegetation-related variables dominating. For example, low values of *slope*, *ndvi*, and *wind_speed* reduce fire likelihood, while some temperature related variables like *lst_night* and *ssrd* as well as moisture-related indicators such as *rh* contribute slightly positively but remain insufficient to outweigh the negative contributions. Human activity and topographic variables exert only marginal influence. This instance illustrates a typical example of a confidently predicted negative outcome, even within a typically fire-prone region during midsummer.

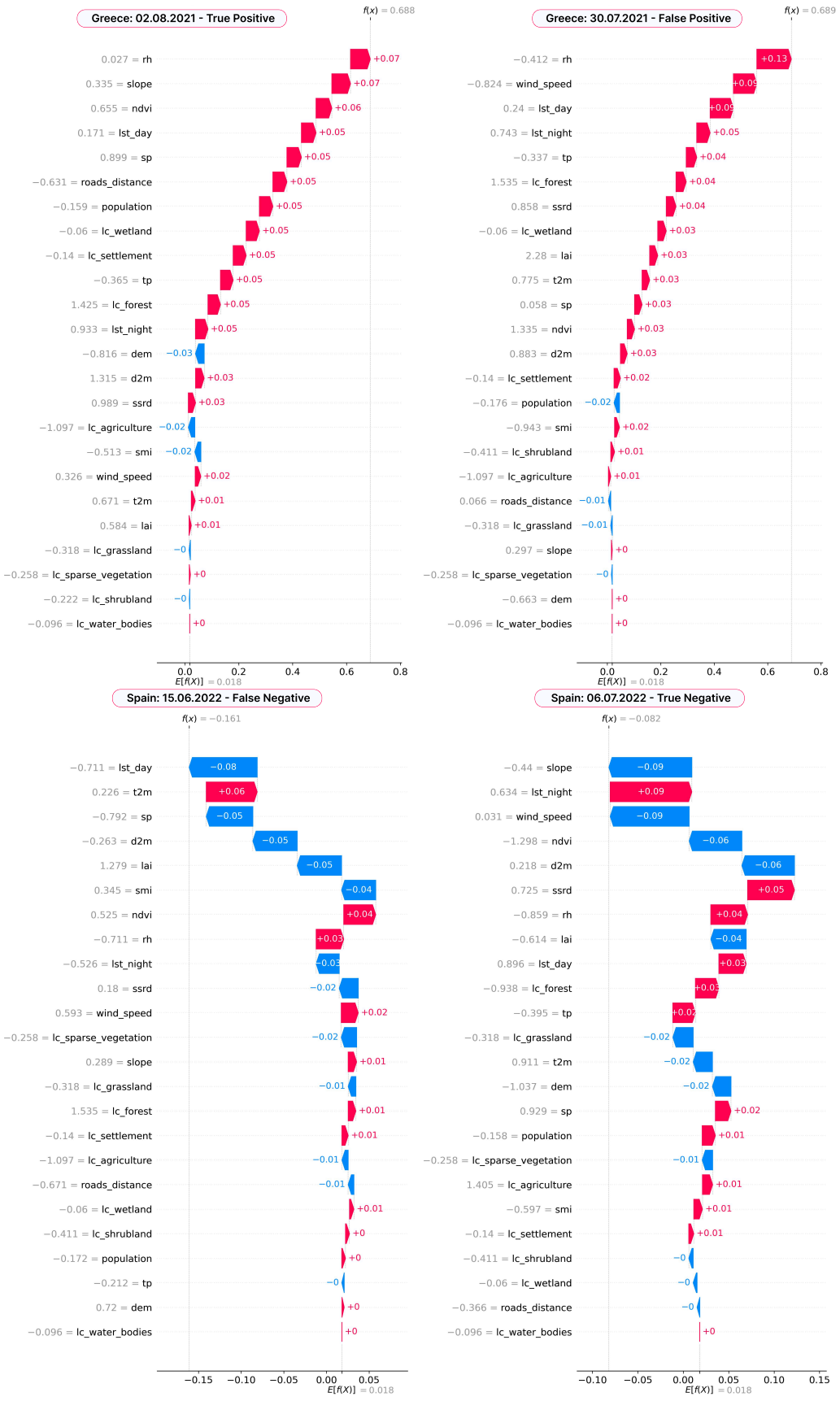

Figure 6.12.: **SHAP Waterfall Plots of Four Samples from the Test Dataset.** Top left: tp sample from Greece (02 August 2021). Top right: FP sample from Greece (30 July 2021). Bottom left: FN from Spain (15 June 2022). Bottom right: TN sample from Spain (06 July 2022). Feature contributions are aggregated over all 30 input days.

6.6.1. Comparison of Two Fire Events in Spain

To further contextualize model explanations, we contrast a false negative prediction from 15 June 2022 as described in detail in Section 6.6 with a true positive prediction from 17 July 2022.

A Copernicus Sentinel image provides a side-by-side visualization of both fire events, as shown in Figure B.5 in the appendix. Although predicted differently by the model, both fires reached similar magnitudes and occurred in the same region of Spain, with only slightly overlapping affected areas.

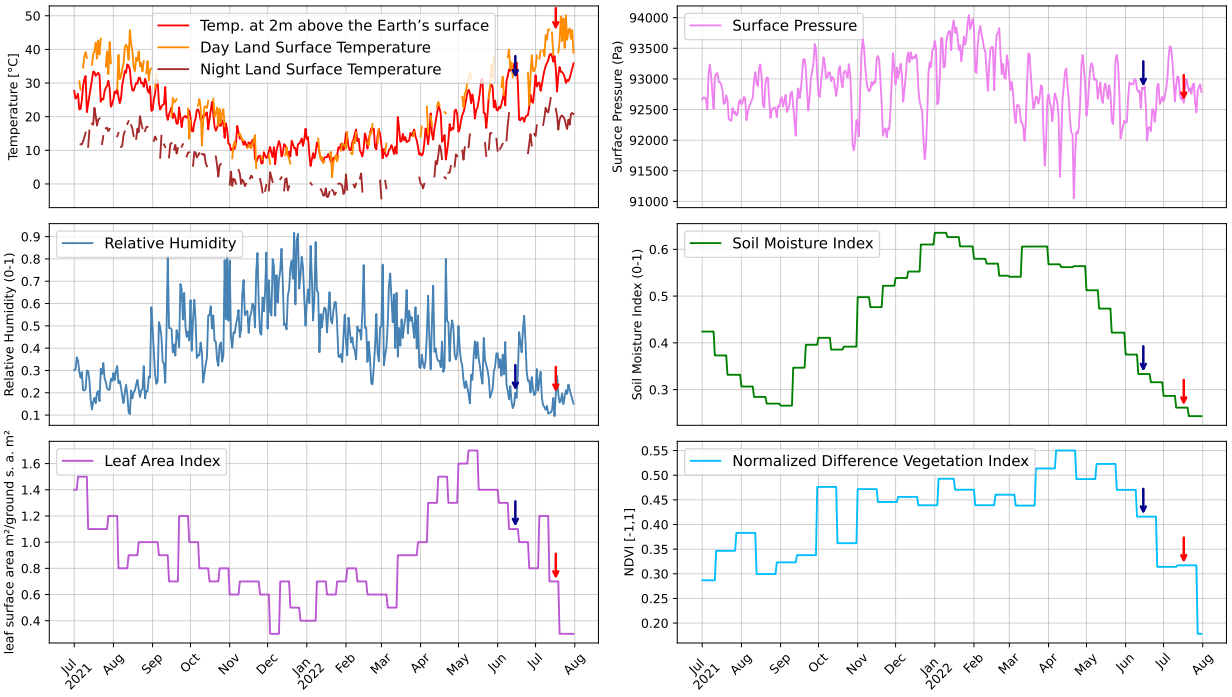

Figure 6.13.: **Temporal Evolution of Environmental Variables before the two Spain Fires 2022**. The two arrows indicate the ignition dates: blue for the June fire and red for the July fire. The data is shown at the June fire coordinates, which lie very close to the July ignition point. The plots show a sharp rise in air and surface temperatures *t2m*, *lst_day*,*lst_night* accompanied by a decrease in *smi* and *rh*.

Figure 6.13 illustrates the temporal build-up of environmental conditions prior to the June and July fires. The coordinates used for this plot correspond to the June fire location. However, as shown in Figure B.5 in the appendix, the ignition points of the two fires exhibit only marginal geographical differences. A distinct intensification of fire-conducive conditions is observed: *t2m*, *lst_day*, *lst_night* steadily increase, while *smi*, *rh*, *lai*, and *ndvi* decrease in the days preceding ignition.

While the June fire was reportedly ignited by lightning and is discussed in more detail in Section 6.6, the July fire broke out near Losacio on 17 July 2022 and, according to Copernicus emergency reports and media sources, was triggered by a heatwave and prolonged drought conditions (AEMET, 2022). It rapidly spread under extreme circumstances, killing two people and burning over 4,000 hectares (Copernicus Sentinel-2 imagery, 2022; Wikipedia, 2022a).

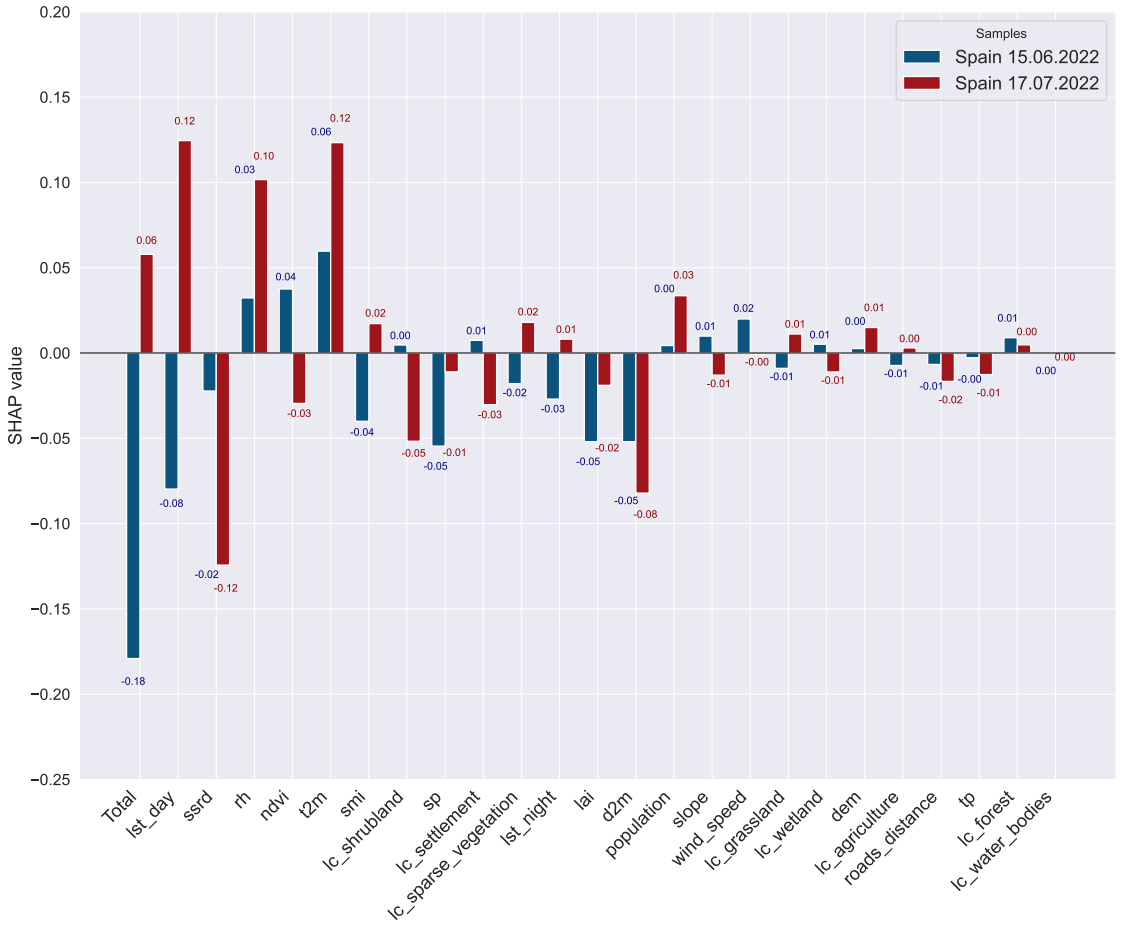

Figure 6.14.: **SHAP Comparison of the two Spain Wildfire Events (June vs. July 2022).** Feature contributions to the final prediction are shown for both events predicted by the Transformer model. Colors indicate individual SHAP values, with red for the July event (True Positive) and blue for the June event (False Negative). Features are ordered by descending absolute SHAP difference.

Notably, the total SHAP sum for the June event is strongly negative, suppressing fire probability, whereas the July fire shows a strong positive SHAP contribution. Among the top differentiating features, *lst_day* and *t2m* stand out, with much higher values in the July event, reflecting the intense heatwave during that period. *rh* plays a particularly important role: lower values in the July fire lead to a high positive SHAP contribution, while higher humidity in June is associated with a small negative SHAP value.

Interestingly, the surface pressure (*sp*) appears to contribute negatively to the estimated probability of fire in June. The absolute value was slightly lower (91884.910 Pa) than in July (92697.520 Pa), suggesting a stronger low-pressure system, typically linked to atmospheric instability and cooler conditions. However, the dataset lacks explicit ignition-related variables, such as the occurrence of lightning strikes or the weather forecast, which may help explain why lightning-induced fires are difficult for the model to detect. Although this case provides a plausible explanation for the failure of the model, it remains a single instance and would require further empirical investigation to confirm a broader pattern. Moreover, it is worth noting that the real causes of extreme fires are not known for all events, so this physical interpretation of ML model outputs is inherently limited by the availability openly accessible post-fire reports.

To ensure the robustness of these findings, we additionally computed the IG explanations for the

same two fire events, as shown in Figure B.7 in the appendix. While the absolute values differ due to conceptual differences between SHAP and IG, the direction of feature influence (positive or negative contribution) is largely consistent. For instance, key variables such as *lst_day*, *ndvi*, *smi*, and *rh* exhibiting contrasting attributions, often in opposite directions across the two events, highlight higher surface and air temperature, along with lower vegetation indices and solar radiation, as important discriminators for the July fire compared to the June fire. Moreover, *lai*, capturing vegetation density, shows predominantly negative contributions in Figure 6.14 and in Figure B.7, with a stronger negative effect for the June fire compared to the July fire. Notably, under IG, the July fire even shows a slight positive contribution. For both fires, the normalized values are high (e.g., 1.279 in the June fire (see Figure 6.12)), and even in the July fire the value is only slightly below the normalized mean of 0. This suggests that excessive vegetation density may be associated with higher wetness or humidity, thus reducing fire risk rather than promoting it. It indicates that fire probability might only increase within a certain intermediate range of vegetation density, beyond which the link to fire-prone conditions weakens or reverses. However, IG assigns small positive contributions of *lai* towards fire, suggesting subtle differences in how vegetation structure is interpreted across the two methods.

6.6.2. Effect of excluding *lst_day* from the model input

To assess the models reliance on land surface temperature, we conducted an ablation experiment in which the model was re-trained without the *lst_day* feature. Figure 6.15 illustrates the resulting SHAP values for the previously misclassified June fire (False Negative), comparing the original model (with *lst_day*) and the retrained version (without *lst_day*).

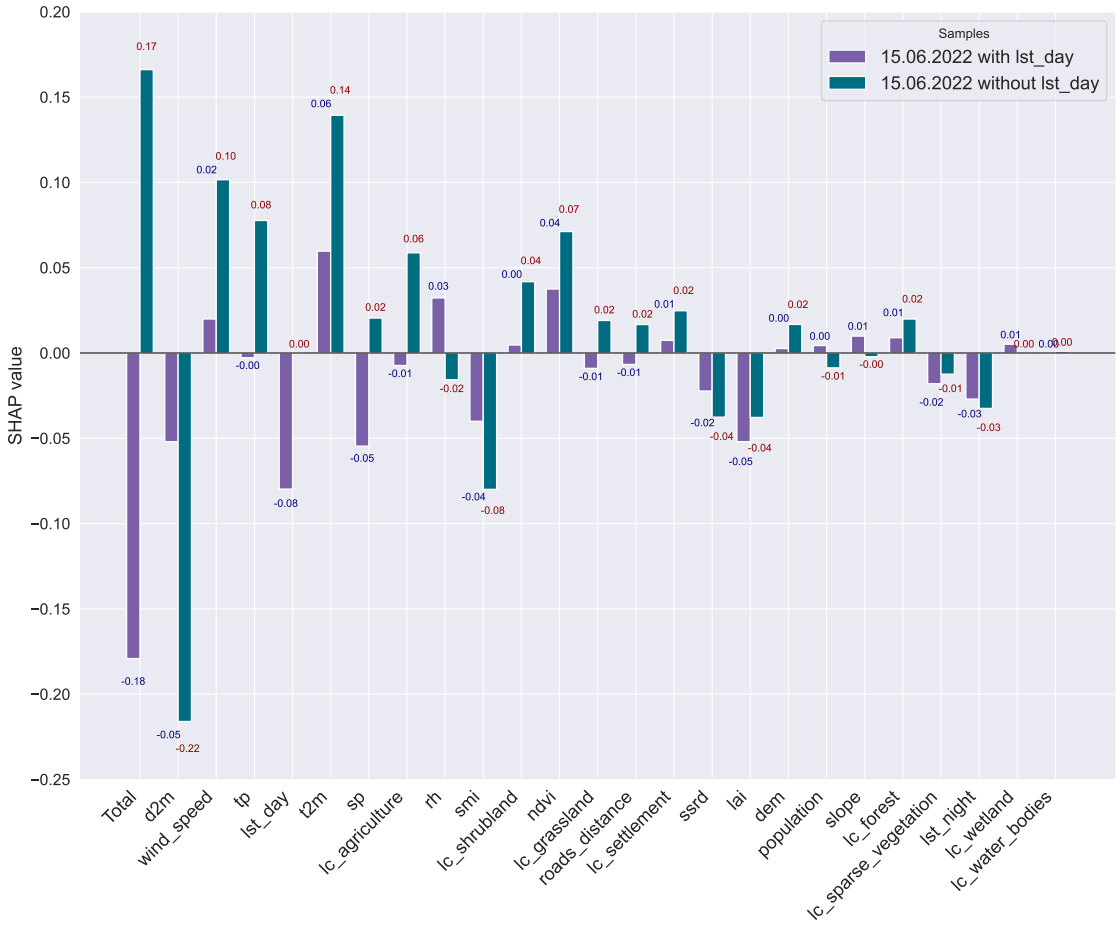

Figure 6.15.: SHAP Comparison for the False Negative June 2022 Fire Sample. Bars represent SHAP values for the prediction with (purple) and without (teal) the *lst_day* feature. The exclusion of *lst_day* leads to stronger contributions from other temperature-related variables such as *t2m* and *d2m*, resulting in a higher predicted fire probability.

Interestingly, the softmax fire probability for the June fire increases substantially, from 0.19 to 0.52, when *lst_day* is excluded. Since the predicted probability crosses the 0.5 threshold, this instance would now be classified as a fire event, although with moderate confidence. In contrast, for the correctly classified July fire, the prediction remains stable, with a slightly increased probability.

This suggests that for the June sample, the presence of *lst_day* suppressed the models fire prediction, possibly due to interactions with other correlated temperature features such as *t2m* (see Section 6.7). After the exclusion of *lst_day*, features like *t2m*, *tp*, *ndvi*, and *wind_speed* receive stronger positive attribution towards fire prediction, while variables such as *lai* and *lc_agriculture* shift from strong negative to weaker or even slightly positive attribution. The difference in the influence of surface pressure (*sp*) between the two fire events becomes negligible without *lst_day*.

However, a more general explanation of why the softmax probability changes so substantially still requires further investigation, and this observation should be considered an illustrative example rather than a conclusion that can be generalized across other cases.

A visual comparison of both fire events without *lst_day* is provided in the appendix in Figure B.6, complementing the side-by-side view shown in Figure 6.14.

6.7. Correlation Analysis of Mesogeos Variables

To better understand redundancies among meteorological inputs, we computed pairwise Pearson correlations between all temperature-related variables as shown in Figure 6.16. The analysis reveals consistently high correlations (e.g., *t2m–lst_night*: $r = 0.67$), indicating strong multicollinearity.

Additionally, surface solar radiation downward (*ssrd*) is also clearly correlated with temperature variables, which is expected due to the physical relationship between solar energy input and surface heating. Nevertheless, excluding variables such as *t2m* is not advisable, as it remains one of the more robust temperature indicators, whereas land surface temperature products (e.g., *lst_day*, *lst_night*) frequently contain missing values.

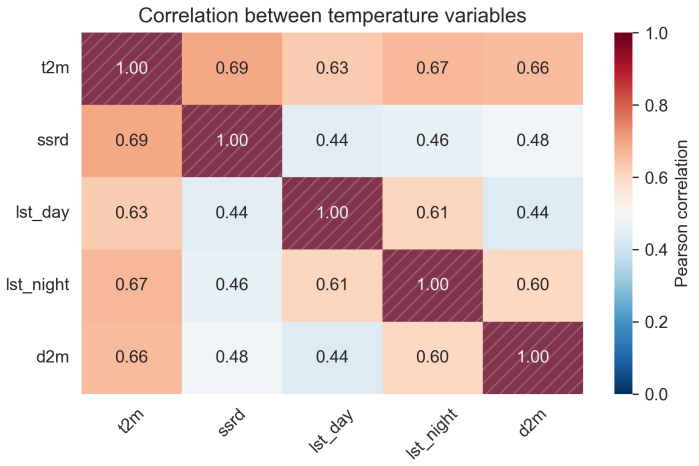

Figure 6.16.: **Correlation between Temperature-related Variables.** High correlation coefficients highlight the redundancy between different temperature-based predictors. Diagonal cells are shaded to indicate perfect self-correlation.

Despite the strong correlations observed among several temperature-related variables, model performance remained consistently high. In particular, experiments in which highly correlated variables such as *lst_day* were removed from the dataset and the model retrained resulted in only slightly reduced predictive skill. The test $F1-$Score decreased from 0.792 (with all variables) to 0.786 when excluding *lst_day*, and further to 0.780 when removing three correlated variables in total. Unlike linear models such as linear regression, which are sensitive to multicollinearity due to their reliance on matrix inversion, DL architectures can effectively ignore redundant inputs, with backpropagation remaining unaffected by feature correlations (Dormann et al., 2013).

However, the presence of highly correlated features can introduce significant challenges for interpretability, particularly when employing XAI techniques. In such cases, attribution methods may distribute importance across correlated inputs in arbitrary or unstable ways, as highly correlated features can effectively "substitute" for one another. This often leads to distorted or inconsistent importance estimates, making it difficult to discern the unique contribution of each variable (Das & Rad, 2020; Molnar, 2025; Salih, 2024). A detailed discussion of the interpretability challenges associated with the specific XAI methods applied in this study is provided in Section 6.8. To mitigate these interpretability challenges and avoid over-reliance on a single method, a diverse suite of XAI approaches was employed in our analysis of wildfire prediction.

6.8. Limitations

The models are trained on historical satellite-based input data and evaluated using fixed assumptions about fire occurrence, which introduces potential biases. Specifically, fire events were included only if the total burned area exceeded 30 hectares, assuming that such fires reflect true ignition and spread events (see Section 3.4). Furthermore, due to the absence of exact ignition coordinates, the centroid of each burned area polygon was used as a proxy for the ignition location. While this approximation enables a consistent mapping to the gridded input data, it introduces uncertainty, as actual ignitions may occur on the edge of the burned area and fire spread dynamics may be shaped by wind direction, topography, or fuel continuity. Moreover, satellite-based inputs are affected by missing values due to cloud cover or resolution limits; the imputation strategy (see Section 3.4) may suppress temporal variability and introduce artifacts.

Another limitation is the absence of key ignition-related factors such as lightning strikes, which can be critical drivers of fire occurrence but are not represented in the dataset. Additionally, the real causes of extreme fires are not always well-documented, meaning that the physical interpretation of model outputs is inherently constrained by the limited availability of robust and openly accessible post-fire reports.

The dataset itself spans only the period from 2006 to 2022, with a single validation year, raising questions about the temporal robustness of the models. Furthermore, because the Mesogeos dataset is region-specific, it remains uncertain whether the trained models and findings generalize well to other fire-prone regions worldwide.

Regarding explainability, IG depend heavily on the choice of baseline input, which is inherently arbitrary and can distort attributions when far from the data manifold. Moreover, IG is restricted to differentiable models restricting its use in comparing against non-differentiable baselines like RF. SHAP, assumes feature independence when estimating conditional expectations. This assumption is frequently violated in environmental datasets, where strong correlations among variables are common. As a result, SHAP may assign misleading importance scores when features are collinear, leading to spurious or fragmented attributions. Additionally, it is important to emphasize that SHAP values explain the model's internal reasoning, not the true underlying data-generating process. They indicate how much each feature contributes to a specific model prediction, not how important that feature is in the real world or to the target variable itself. Consequently, XAI can only be as reliable as the model it interprets. The goal must be to first build a well-performing and robust model, and then use XAI techniques to understand its behavior, without making claims about causality. Additionally, the default feature importance metric in *scikit-learn*'s *RandomForestClassifier*, based on Mean Decrease in Impurity (MDI), can be biased (Jérémie du Boisberranger & Loïc Estève, 2007). Specifically, MDI tends to inflate the importance of numerical or high-cardinality features, even when they are not predictive (Strobl et al., 2007). Moreover, when features are correlated, as exemplary *lst_day*, *lst_night*, *t2m*, the importance is often arbitrarily split among them. For instance, if two features are perfectly correlated, MDI typically assigns each about half the total importance instead of capturing their joint relevance (Altmann et al., 2010).

7. Conclusion & Outlook

This thesis evaluated temporal DL models for next-day wildfire danger forecasting in the Mediterranean, integrating XAI techniques to uncover what these models have learned. Our findings provide key insights into the potential of DL for operational wildfire forecasting.

First, all DL models achieved strong predictive performance ($F_1 > 0.8$), clearly outperforming the Random Forest baseline. Despite exploring diverse architectures, performance differences among DL models were minimal. Notably, complex attention-based variants such as TFT and GTN did not surpass a standard Transformer, suggesting that simpler attention mechanisms are sufficient to capture the relevant spatio-temporal dependencies.

Second, the temporal analysis confirmed that Transformers effectively exploit extended historical contexts, with performance steadily improving up to 30-day input windows. By contrast, recurrent models plateaued after shorter sequences, reflecting their inherent limitations in modeling long-term dependencies.

Third, interpretability analyses using SHAP and IG revealed strong consistency between model decisions and established fire science, with temperature, humidity, precipitation, and vegetation indices emerging as dominant drivers. Case studies showed that models reliably detected well-documented extreme fire events, while misclassifications could be linked to missing ignition-specific information (e.g., lightning) or geographic imbalances in the data. Further work is needed to improve explanations and to better understand why removing a single temperature variable (e.g., *lst_day*) can substantially alter model predictions, as discussed in Section 6.6.1.

Furthermore, the correlation analysis revealed substantial multicollinearity among temperature-related variables, which did not degrade model performance but introduced interpretability challenges. Attribution methods occasionally distributed importance inconsistently across correlated features, emphasizing the need for multi-method explainability to ensure robust conclusions about driver relevance.

Interestingly, while the Transformer achieved the highest predictive accuracy, the Random Forest benchmark captured more physically consistent relationships that no DL model represented correctly. This raises an important question, how we can define the "ideal" wildfire forecasting model. Should preference be given to models like Transformers with superior predictive accuracy, or to models like Random Forests that exhibit stronger physical interpretability but risk overfitting and under performing in predictive skill? The answer likely depends on the intended application, whether the goal is operational forecasting or scientific understandingand underscores the trade-off between accuracy and interpretability in wildfire modeling.

Nonetheless, several limitations remain as shown in Section 6.8. In particular, it is not possible to definitively rank variables by overall importance, as their influence varies across individual fire events and can differ slightly between models. These variations reflect the diversity of wildfire types and align with findings in Kondylatos, Prapas, Ronco, et al. (2022).

In conclusion, standard Transformer models deliver accurate, interpretable, and computationally efficient wildfire danger forecasts. However, aspects of their decision-making remain insufficiently understood, and require further investigation. Future work should focus on addressing these gaps and advancing our understanding of model decision-making, as outlined in Section 7.1.

7.1. Recommendations for Further Studies

Building upon the results presented in Chapter 6, several promising directions emerge for future research. One key aspect is a more systematic analysis of false negative predictions. As highlighted in the case studies in Section 6.6.1, certain fire events are missed due to missing ignition-specific variables such as lightning strike occurrence or potentially other currently unknown variables. A clustering-based analysis of misclassified samples could help uncover common patterns in false negatives. Based on such analyses, specific ignition-related features, such as lightning strike records, storm or weather forecast indicators, or other newly identified drivers, could then be systematically incorporated into future versions of the dataset.

Moreover, as shown in Section 6.2, there is a pronounced spatial imbalance in the performance of prediction, particularly between coastal and inland areas. Investigating this coastal–inland discrepancy in more depth could reveal structural weaknesses in current models and reduce the false negative rate in underrepresented inland regions.

Another avenue for future work is the generalization of the current framework to other geographic regions. Adapting the Mesogeos datacube concept for global datasets would allow testing whether the learned spatio-temporal patterns remain robust across different fire regimes, vegetation types, and climate zones.

Additionally, shifting the modeling objective from binary fire occurrence to the prediction of burned area size would provide more nuanced and actionable outputs. Instead of classifying the presence or absence of fire above a fixed threshold (e.g., 30 hectares), future models could estimate the final size of fires. This task can be formulated either as a regression or a multi-class classification problem, based on suggestions in Kondylatos, Prapas, Camps-Valls, and Papoutsis (2023).

Finally, there is strong potential in exploring physics-informed or hybrid modeling approaches that combine deep learning with fire science principles. Embedding known fire-weather interactions or physical constraints into the model architecture could enhance both predictive accuracy and interpretability. This hybrid modeling strategy may help bridge the current gap between high-performance, data-driven models and physically plausible forecasting tools, ultimately contributing to more trustworthy and actionable wildfire danger predictions.

Bibliography

Abatzoglou, J. T., Williams, A. P., & Barbero, R. (2019). Global Emergence of Anthropogenic Climate Change in Fire Weather Indices. *Geophysical Research Letters*, *46*(1), 326–336. https://doi.org/10.1029/2018GL080959

Abdollahi, A., & Pradhan, B. (2021). Urban Vegetation Mapping from Aerial Imagery Using Explainable AI (XAI) [Number: 14 Publisher: Multidisciplinary Digital Publishing Institute]. *Sensors*, *21*, 4738. https://doi.org/10.3390/s21144738

Abdollahi, A., & Pradhan, B. (2023). Explainable artificial intelligence (XAI) for interpreting the contributing factors feed into the wildfire susceptibility prediction model. *Science of The Total Environment*, *879*, 163004. https://doi.org/10.1016/j.scitotenv.2023.163004

Adams, H. D., Barron-Gafford, G. A., Minor, R. L., Gardea, A. A., Bentley, L. P., Law, D. J., Breshears, D. D., McDowell, N. G., & Huxman, T. E. (2017). Temperature response surfaces for mortality risk of tree species with future drought [Publisher: IOP Publishing]. *Environmental Research Letters*, *12*(11), 115014. https://doi.org/10.1088/1748-9326/aa93be

AEMET. (2022). Avance Climático Nacional del invierno 2021-2022. Retrieved July 26, 2025, from https://aemetblog.es/2022/03/16/avance-climatico-nacional-del-invierno-2021-2022/

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A Next-generation Hyperparameter Optimization Framework. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2623–2631. https://doi.org/10.1145/3292500.3330701

Alonso, L., Gans, F., Karasante, I., Ahuja, A., Prapas, I., Kondylatos, S., Papoutsis, I., Panagiotou, E., Mihail, D., Cremer, F., Weber, U., & Carvalhais, N. (2024). SeasFire Cube: A Global Dataset for Seasonal Fire Modeling in the Earth System. https://doi.org/10.5281/ZENODO.13834057

Altmann, A., Toloi, L., Sander, O., & Lengauer, T. (2010). Permutation importance: A corrected feature importance measure. *Bioinformatics*, *26*(10), 1340–1347. https://doi.org/10.1093/bioinformatics/btq134

Andela, N., & van der Werf, G. R. (2014). Recent trends in African fires driven by cropland expansion and El Niño to La Niña transition [Publisher: Nature Publishing Group]. *Nature Climate Change*, *4*(9), 791–795. https://doi.org/10.1038/nclimate2313

Apley, D. W., & Zhu, J. (2020). Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *82*(4), 1059–1086. https://doi.org/10.1111/rssb.12377

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Clarendon Press.

Bistinas, I., Oom, D., Sá, A. C. L., Harrison, S. P., Prentice, I. C., & Pereira, J. M. C. (2013). Relationships between Human Population Density and Burned Area at Continental and Global Scales [Publisher: Public Library of Science]. *PLOS ONE*, *8*(12), e81188. https://doi.org/10.1371/journal.pone.0081188

Bladon, K. D., Emelko, M. B., Silins, U., & Stone, M. (2014). Wildfire and the Future of Water Supply. *Environmental Science & Technology*, *48*(16), 8936–8943. https://doi.org/10.1021/es500130g

Bowman, D. M. J. S., Balch, J. K., Artaxo, P., Bond, W. J., Carlson, J. M., Cochrane, M. A., DAntonio, 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 [Publisher: American Association for the Advancement of Science]. *Science*, *324*(5926), 481–484. https://doi.org/10.1126/science.1163886

Bradshaw, L. S., Deeming, J. E., Burgan, R. E., & compilers.Cohen, J. D. (1984). The 1978 National Fire-Danger Rating System: Technical documentation. *169*, 44. https://doi.org/10.2737/INT-GTR-169

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*(2), 123–140. https://doi.org/10.1007/BF00058655

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Brown, T. J., Hall, B. L., & Westerling, A. L. (2004). The Impact of Twenty-First Century Climate Change on Wildland Fire Danger in the Western United States: An Applications Perspective. *Climatic Change*, *62*(1-3), 365–388. https://doi.org/10.1023/B:CLIM.0000013680.07783.de

Cammalleri, C., Vogt, J. V., Bisselink, B., & de Roo, A. (2017). Comparing soil moisture anomalies from multiple independent sources over different regions across the globe [Publisher: Copernicus GmbH]. *Hydrology and Earth System Sciences*, *21*(12), 6329–6343. https://doi.org/10.5194/hess-21-6329-2017

Cascio, W. E. (2018). Wildland fire smoke and human health. *Science of The Total Environment*, *624*, 586–595. https://doi.org/10.1016/j.scitotenv.2017.12.086

Cho, K., Merrienboer, B. v., Bahdanau, D., & Bengio, Y. (2014). On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. https://doi.org/10.48550/arXiv.1409.1259

Chuvieco, E., Yebra, M., Martino, S., Thonicke, K., Gómez-Giménez, M., San-Miguel, J., Oom, D., Velea, R., Mouillot, F., Molina, J. R., Miranda, A. I., Lopes, D., Salis, M., Bugaric, M., Sofiev, M., Kadantsev, E., Gitas, I. Z., Stavrakoudis, D., Eftychidis, G., … Viegas, D. (2023). Towards an Integrated Approach to Wildfire Risk Assessment: When, Where, What and How May the Landscapes Burn. *Fire*, *6*(5), 215. https://doi.org/10.3390/fire6050215

Cilli, R., Elia, M., DEste, M., Giannico, V., Amoroso, N., Lombardi, A., Pantaleo, E., Monaco, A., Sanesi, G., Tangaro, S., Bellotti, R., & Lafortezza, R. (2022). Explainable artificial intelligence (XAI) detects wildfire occurrence in the Mediterranean countries of Southern Europe [Publisher: Nature Publishing Group]. *Scientific Reports*, *12*(1), 16349. https://doi.org/10.1038/s41598-022-20347-9

Cochard, H. (2021). A new mechanism for tree mortality due to drought and heatwaves. *Peer Community Journal*, *1*. https://doi.org/10.24072/pcjournal.45

Conedera, M., Cesti, G., Pezzatti, G., Zumbrunnen, T., & Spinedi, F. (2006). Lightning-induced fires in the Alpine region: An increasing problem. *Forest Ecology and Management*, *234*, S68. https://doi.org/10.1016/j.foreco.2006.08.096

Copernicus Emergency Management Service data. (2022). Devastating wildfire in Sierra de la Culebra, Spain | Copernicus. Retrieved July 1, 2025, from https://www.copernicus.eu/en/media/image-day-gallery/devastating-wildfire-sierra-de-la-culebra-spain

Copernicus Sentinel-2 imagery. (2022). Spanish Province of Zamora, in Castilla y León, ravaged by wildfires | Copernicus. Retrieved July 1, 2025, from https://www.copernicus.eu/en/media/image-day-gallery/spanish-province-zamora-castilla-y-leon-ravaged-wildfires

Cramer, W., Guiot, J., Fader, M., Garrabou, J., Gattuso, J.-P., Iglesias, A., Lange, M. A., Lionello, P., Llasat, M. C., Paz, S., Peñuelas, J., Snoussi, M., Toreti, A., Tsimplis, M. N., & Xoplaki, E. (2018). Climate change and interconnected risks to sustainable development in the Mediterranean [Publisher: Nature Publishing Group]. *Nature Climate Change*, *8*(11), 972–980. https://doi.org/10.1038/s41558-018-0299-2

Cunningham, C. X., Williamson, G. J., & Bowman, D. M. J. S. (2024). Increasing frequency and intensity of the most extreme wildfires on Earth [Publisher: Nature Publishing Group]. *Nature Ecology & Evolution*, *8*(8), 1420–1425. https://doi.org/10.1038/s41559-024-02452-2

Das, A., & Rad, P. (2020). Opportunities and Challenges in Explainable Artificial Intelligence (XAI): A Survey. https://doi.org/10.48550/arXiv.2006.11371

Di Giuseppe, F., McNorton, J., Lombardi, A., & Wetterhall, F. (2025). Global data-driven prediction of fire activity [Publisher: Nature Publishing Group]. *Nature Communications*, *16*(1), 2918. https://doi.org/10.1038/s41467-025-58097-7

Di Virgilio, G., Evans, J. P., Blake, S. A. P., Armstrong, M., Dowdy, A. J., Sharples, J., & McRae, R. (2019). Climate Change Increases the Potential for Extreme Wildfires. *Geophysical Research Letters*, *46*(14), 8517–8526. https://doi.org/10.1029/2019GL083699

Didan, K. (2015). MOD13A2 MODIS/Terra Vegetation Indices 16-Day L3 Global 1km SIN Grid V006. https://doi.org/10.5067/MODIS/MOD13A2.006

Dormann, C. F., Elith, J., Bacher, S., Buchmann, C., Carl, G., Carré, G., Marquéz, J. R. G., Gruber, B., Lafourcade, B., Leitão, P. J., Münkemüller, T., McClean, C., Osborne, P. E., Reineking, B., Schröder, B., Skidmore, A. K., Zurell, D., & Lautenbach, S. (2013). Collinearity: A review of methods to deal with it and a simulation study evaluating their performance. *Ecography*, *36*(1), 27–46. https://doi.org/10.1111/j.1600-0587.2012.07348.x

Dorph, A., Marshall, E., Parkins, K. A., & Penman, T. D. (2022). Modelling ignition probability for human- and lightning-caused wildfires in Victoria, Australia [Publisher: Copernicus GmbH]. *Natural Hazards and Earth System Sciences*, *22*(10), 3487–3499. https://doi.org/10.5194/nhess-22-3487-2022

Dunn, O. J. (1964). Multiple Comparisons Using Rank Sums [Publisher: ASA Website _eprint: https://www.tandfonline.com/doi/pdf/10.1080/00401706.1964.10490181]. *Technometrics*, *6*(3), 241–252. https://doi.org/10.1080/00401706.1964.10490181

Dupuy, J.-l., Fargeon, H., Martin-StPaul, N., Pimont, F., Ruffault, J., Guijarro, M., Hernando, C., Madrigal, J., & Fernandes, P. (2020). Climate change impact on future wildfire danger and activity in southern Europe: A review. *Annals of Forest Science*, *77*(2), 35. https://doi.org/10.1007/s13595-020-00933-5

Eyring, V., Cox, P. M., Flato, G. M., Gleckler, P. J., Abramowitz, G., Caldwell, P., Collins, W. D., Gier, B. K., Hall, A. D., & Hoffman, F. M. (2019). Taking climate model evaluation to the next level [Publisher: Nature Publishing Group]. *Nature Climate Change*, *9*(2), 102–110. https://doi.org/10.1038/s41558-018-0355-y

Farid, A., Alam, M. K., Goli, V. S. N. S., Akin, I. D., Akinleye, T., Chen, X., Cheng, Q., Cleall, P., Cuomo, S., Foresta, V., Ge, S., Iervolino, L., Iradukunda, P., Luce, C. H., Koda, E., Mickovski, S. B., OKelly, B. C., Paleologos, E. K., & Peduto, D. (2024). A Review of the Occurrence and Causes for Wildfires and Their Impacts on the Geoenvironment. *Fire*, *7*, 295. https://doi.org/10.3390/fire7080295

Finney, M. A. (1998). *FARSITE, Fire Area Simulator–model Development and Evaluation* [Google-Books-ID: d2O0M1GePi0C]. The Station.

Francisco Seijo. (2017). European Fire Governance in the Era of Megafires. Retrieved July 25, 2025, from https://www.greeneuropeanjournal.eu/european-fire-governance-in-the-era-of-megafires/

Franks, S., & Rengarajan, R. (2023). Evaluation of Copernicus DEM and Comparison to the DEM Used for Landsat Collection-2 Processing [Number: 10 Publisher: Multidisciplinary Digital Publishing Institute]. *Remote Sensing*, *15*, 2509. https://doi.org/10.3390/rs15102509

Ganteaume, A., Camia, A., Jappiot, M., San-Miguel-Ayanz, J., Long-Fournel, M., & Lampin, C. (2013). A Review of the Main Driving Factors of Forest Fire Ignition Over Europe. *Environmental Management*, *51*(3), 651–662. https://doi.org/10.1007/s00267-012-9961-z

Gehring, J., Auli, M., Grangier, D., Yarats, D., & Dauphin, Y. N. (2017). Convolutional Sequence to Sequence Learning. *Proceedings of the 34th International Conference on Machine Learning*, 1243–1252. Retrieved June 10, 2025, from https://proceedings.mlr.press/v70/gehring17a.html

Gers, F., & Schmidhuber, E. (2001). LSTM recurrent networks learn simple context-free and context-sensitive languages. *IEEE Transactions on Neural Networks*, *12*(6), 1333–1340. https://doi.org/10.1109/72.963769

Giglio, L., Schroeder, W., & Justice, C. O. (2016). The collection 6 MODIS active fire detection algorithm and fire products. *Remote Sensing of Environment*, *178*, 31–41. https://doi.org/10.1016/j.rse.2016.02.054

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. (2016). The status and challenge of global fire modelling. *Biogeosciences*, *13*(11), 3359–3375. https://doi.org/10.5194/bg-13-3359-2016

Harris, L., & Grzes, M. (2019). Comparing Explanations between Random Forests and Artificial Neural Networks. *IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2978–2985. https://doi.org/10.1109/SMC.2019.8914321

Harrison, S. P., Bartlein, P. J., Brovkin, V., Houweling, S., Kloster, S., & Prentice, I. C. (2018). The biomass burning contribution to climatecarbon-cycle feedback. *Earth System Dynamics*, *9*(2), 663–677. https://doi.org/10.5194/esd-9-663-2018

Ho, T. K. (1995). Random decision forests. *Proceedings of 3rd International Conference on Document Analysis and Recognition*, *1*, 278–282 vol.1. https://doi.org/10.1109/ICDAR.1995.598994

Hochreiter, S. (1991). Untersuchungen zu dynamischen neuronalen Netzen. *Diploma, Technische Universität München*, *91*(1), 31.

Hochreiter, S., & Schmidhuber, J. (1996). LSTM can Solve Hard Long Time Lag Problems. *Advances in Neural Information Processing Systems*, *9*. Retrieved April 29, 2025, from https://proceedings.neurips.cc/paper/1996/hash/a4d2f0d23dcc84ce983ff9157f8b7f88-Abstract.html

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

Holsten, A., Dominic, A. R., Costa, L., & Kropp, J. P. (2013). Evaluation of the performance of meteorological forest fire indices for German federal states. *Forest Ecology and Management*, *287*, 123–131. https://doi.org/10.1016/j.foreco.2012.08.035

Huot, F., Hu, R. L., Ihme, M., Wang, Q., Burge, J., Lu, T., Hickey, J., Chen, Y.-F., & Anderson, J. (2021). Deep Learning Models for Predicting Wildfires from Historical Remote-Sensing Data. https://doi.org/10.48550/arXiv.2010.07445

Iban, M. C., & Sekertekin, A. (2022). Machine learning based wildfire susceptibility mapping using remotely sensed fire data and GIS: A case study of Adana and Mersin provinces, Turkey. *Ecological Informatics*, *69*, 101647. https://doi.org/10.1016/j.ecoinf.2022.101647

Jain, P., Coogan, S. C., Subramanian, S. G., Crowley, M., Taylor, S., & Flannigan, M. D. (2020). A review of machine learning applications in wildfire science and management. *Environmental Reviews*, *28*(4), 478–505. https://doi.org/10.1139/er-2020-0019

Jérémie du Boisberranger & Loïc Estève. (2007). Permutation Importance vs Random Forest Feature Importance (MDI). Retrieved July 1, 2025, from https://scikit-learn/stable/auto_examples/inspection/plot_permutation_importance.html

Ji, Y., Wang, D., Li, Q., Liu, T., & Bai, Y. (2024). Global Wildfire Danger Predictions Based on Deep Learning Taking into Account Static and Dynamic Variables. *Forests*, *15*(1), 216. https://doi.org/10.3390/f15010216

Jolly, W. M., Cochrane, M. A., Freeborn, P. H., Holden, Z. A., Brown, T. J., Williamson, G. J., & Bowman, D. M. J. S. (2015). Climate-induced variations in global wildfire danger from 1979 to 2013 [Publisher: Nature Publishing Group]. *Nature Communications*, *6*(1), 7537. https://doi.org/10.1038/ncomms8537

Jones, B. A. (2017). Are we underestimating the economic costs of wildfire smoke? An investigation using the life satisfaction approach. *Journal of Forest Economics*, *27*, 80–90. https://doi.org/10.1016/j.jfe.2017.03.004

Jones, M. W., Abatzoglou, J. T., Veraverbeke, S., Andela, N., Lasslop, G., Forkel, M., Smith, A. J. P., Burton, C., Betts, R. A., van der Werf, G. R., Sitch, S., Canadell, J. G., Santín, C., Kolden, C., Doerr, S. H., & Le Quéré, C. (2022). Global and Regional Trends and Drivers of Fire Under Climate Change. *Reviews of Geophysics*, *60*(3), e2020RG000726. https://doi.org/10.1029/2020RG000726

Kim, S. G., Ryu, S., Jin, K., & Kim, H. (2025). Quantitative comparison of explainable artificial intelligence methods for nuclear power plant accident diagnosis models. *Progress in Nuclear Energy*, *180*, 105605. https://doi.org/10.1016/j.pnucene.2025.105605

Kolen, J. F., & Kremer, S. C. (2001). Gradient Flow in Recurrent Nets: The Difficulty of Learning LongTerm Dependencies. In *A Field Guide to Dynamical Recurrent Networks* (pp. 237–243). IEEE. https://doi.org/10.1109/9780470544037.ch14

Kondylatos, S., Prapas, I., Camps-Valls, G., & Papoutsis, I. (2023). Mesogeos: A multi-purpose dataset for data-driven wildfire modeling in the Mediterranean. https://doi.org/10.48550/arXiv.2306.05144

Kondylatos, S., Prapas, I., Ronco, M., Papoutsis, I., Camps-Valls, G., Piles, M., Fernández-Torres, M.-Á., & Carvalhais, N. (2022). Wildfire Danger Prediction and Understanding With Deep Learning. *Geophysical Research Letters*, *49*(17). https://doi.org/10.1029/2022GL099368

Kong, B. (2024). A Comparative Analysis of Machine Learning Models for Wildfire Prediction. Retrieved June 30, 2025, from https://nhsjs.com/2024/a-comparative-analysis-of-machine-learning-models-for-wildfire-prediction/

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Commun. ACM*, *60*(6), 84–90. https://doi.org/10.1145/3065386

Kruskal, W. H., & Wallis, W. A. (1952). Use of Ranks in One-Criterion Variance Analysis. *Journal of the American Statistical Association*, *47*(260), 583–621. https://doi.org/10.1080/01621459.1952.10483441

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

Lever, J., Cheng, S., & Arcucci, R. (2023). Human-sensors & physics aware machine learning for wildfire detection and nowcasting. In J. Mikyka, C. de Mulatier, M. Paszynski, V. V. Krzhizhanovskaya, J. J. Dongarra, & P. M. Sloot (Eds.), *Computational science ICCS 2023* (pp. 422–429). Springer Nature Switzerland.

Li, F., Zhu, Q., Yuan, K., Ji, F., Paul, A., Lee, P., Radeloff, V. C., & Chen, M. (2024). Projecting large fires in the western US with a more trustworthy machine learning method. https://doi.org/10.22541/essoar.171623766.68002899/v1

Li, L.-M., Song, W.-G., Ma, J., & Satoh, K. (2009). Artificial neural network approach for modeling the impact of population density and weather parameters on forest fire risk [Publisher: CSIRO PUBLISHING]. *International Journal of Wildland Fire*, *18*(6), 640–647. https://doi.org/10.1071/WF07136

Liaw, A., & Wiener, M. (2002). Classication and Regression by randomForest. *R news*.

Lim, B., Ark, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, *37*(4), 1748–1764. https://doi.org/10.1016/j.ijforecast.2021.03.012

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. *Advances in Neural Information Processing Systems*, *30*. Retrieved April 25, 2025, from https://papers.nips.cc/paper_files/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html

Metz, C. (2016). An Infusion of AI Makes Google Translate More Powerful Than Ever. *Wired*. Retrieved April 30, 2025, from https://www.wired.com/2016/09/google-claims-ai-breakthrough-machine-translation/

Michail, D., Panagiotou, L.-I., Davalas, C., Prapas, I., Kondylatos, S., Bountos, N. I., & Papoutsis, I. (2024). Seasonal Fire Prediction using Spatio-Temporal Deep Neural Networks. https://doi.org/10.48550/arXiv.2404.06437

Molnar, C. (2025). *Interpretable Machine Learning - A Guide for Making Black Box Models Explainable* (3rd ed.). https://christophm.github.io/interpretable-ml-book

Moritz, M. A., Moody, T. J., Krawchuk, M. A., Hughes, M., & Hall, A. (2010). Spatial variation in extreme winds predicts large wildfire locations in chaparral ecosystems. *Geophysical Research Letters*, *37*(4). https://doi.org/10.1029/2009GL041735

Mukunga, T., Forkel, M., Forrest, M., Zotta, R.-M., Pande, N., Schlaffer, S., & Dorigo, W. (2023). Effect of Socioeconomic Variables in Predicting Global Fire Ignition Occurrence [Number: 5 Publisher: Multidisciplinary Digital Publishing Institute]. *Fire*, *6*(5), 197. https://doi.org/10.3390/fire6050197

Muñoz-Sabater, J., Dutra, E., Agustí-Panareda, A., Albergel, C., Arduini, G., Balsamo, G., Boussetta, S., Choulga, M., Harrigan, S., Hersbach, H., Martens, B., Miralles, D. G., Piles, M., Rodríguez-Fernández, N. J., Zsoter, E., Buontempo, C., & Thépaut, J.-N. (2021). ERA5-Land: A state-of-the-art global reanalysis dataset for land applications. *Earth System Science Data*, *13*, 4349–4383. https://doi.org/10.5194/essd-13-4349-2021

Myneni, R. B., Shabanov, N. V., Knyazikhin, Y., Yang, W., Dong, H., & Tan, B. (2002). MOD15A2: Global LAI and FPAR. *2002*, B61B–0719. Retrieved April 28, 2025, from https://ui.adsabs.harvard.edu/abs/2002AGUFM.B61B0719M

Nowack, P., Runge, J., Eyring, V., & Haigh, J. D. (2020). Causal networks for climate model evaluation and constrained projections [Publisher: Nature Publishing Group]. *Nature Communications*, *11*(1), 1415. https://doi.org/10.1038/s41467-020-15195-y

O'Shea, K., & Nash, R. (2015). An Introduction to Convolutional Neural Networks. https://doi.org/10.48550/arXiv.1511.08458

Palaiologou, P., Kalabokidis, K., Troumbis, A., Day, M. A., Nielsen-Pincus, M., & Ager, A. A. (2021). Socio-Ecological Perceptions of Wildfire Management and Effects in Greece [Number: 2 Publisher: Multidisciplinary Digital Publishing Institute]. *Fire*, *4*, 18. https://doi.org/10.3390/fire4020018

Park Williams, A., Allen, C. D., Macalady, A. K., Griffin, D., Woodhouse, C. A., Meko, D. M., Swetnam, T. W., Rauscher, S. A., Seager, R., Grissino-Mayer, H. D., Dean, J. S., Cook, E. R., Gangodagamage, C., Cai, M., & McDowell, N. G. (2013). Temperature as a potent driver of regional forest drought stress and tree mortality [Publisher: Nature Publishing Group]. *Nature Climate Change*, *3*, 292–297. https://doi.org/10.1038/nclimate1693

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., & Desmaison, A. (2017). Automatic differentiation in PyTorch.

Pimont, F., Dupuy, J.-L., & Linn, R. R. (2012). Coupled slope and wind effects on fire spread with influences of fire size: A numerical study using FIRETEC [Publisher: CSIRO PUBLISHING]. *International Journal of Wildland Fire*, *21*, 828–842. https://doi.org/10.1071/WF11122

Potapov, P., Hansen, M. C., Pickens, A., Hernandez-Serna, A., Tyukavina, A., Turubanova, S., Zalles, V., Li, X., Khan, A., Stolle, F., Harris, N., Song, X.-P., Baggett, A., Kommareddy, I., & Kommareddy, A. (2022). The Global 2000-2020 Land Cover and Land Use Change Dataset Derived From the Landsat Archive: First Results [Publisher: Frontiers]. *Frontiers in Remote Sensing*, *3*. https://doi.org/10.3389/frsen.2022.856903

Potter, S., Solvik, K., Erb, A., Goetz, S. J., Johnstone, J. F., Mack, M. C., Randerson, J. T., Román, M. O., Schaaf, C. L., Turetsky, M. R., Veraverbeke, S., Walker, X. J., Wang, Z., Massey, R., & Rogers, B. M. (2020). Climate change decreases the cooling effect from postfire albedo in boreal North America. *Global Change Biology*, *26*(3), 1592–1607. https://doi.org/10.1111/gcb.14888

Prapas, I., Ahuja, A., Kondylatos, S., Karasante, I., Panagiotou, E., Alonso, L., Davalas, C., Michail, D., Carvalhais, N., & Papoutsis, I. (2023). Deep Learning for Global Wildfire Forecasting. https://doi.org/10.48550/arXiv.2211.00534

Prapas, I., Bountos, N. I., Kondylatos, S., Michail, D., Camps-Valls, G., & Papoutsis, I. (2023). TeleViT: Teleconnection-driven Transformers Improve Subseasonal to Seasonal Wildfire Forecasting. https://doi.org/10.48550/arXiv.2306.10940

Prestemon, J. P., Hawbaker, T. J., Bowden, M., Carpenter, J., Brooks, M. T., Abt, K. L., Sutphen, R., & Scranton, S. (2013). *Wildfire Ignitions: A Review of the Science and Recommendations for Empirical Modeling* (tech. rep.). U.S. Department of Agriculture, Forest Service, Southern Research Station. Asheville, NC. https://doi.org/10.2737/srs-gtr-171

Pullabhotla, H. K., Zahid, M., Heft-Neal, S., Rathi, V., & Burke, M. (2023). Global biomass fires and infant mortality. *Proceedings of the National Academy of Sciences*, *120*(23), e2218210120. https://doi.org/10.1073/pnas.2218210120

Radeloff, V. C., Mockrin, M. H., Helmers, D., Carlson, A., Hawbaker, T. J., Martinuzzi, S., Schug, F., Alexandre, P. M., Kramer, H. A., & Pidgeon, A. M. (2023). Rising wildfire risk to houses in the United States, especially in grasslands and shrublands [Publisher: American Association for the Advancement of Science]. *Science*, *382*(6671), 702–707. https://doi.org/10.1126/science.ade9223

Raschka, S. (2014). An Overview of General Performance Metrics of Binary Classifier Systems. https://doi.org/10.48550/arXiv.1410.5330

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., & Prabhat. (2019). Deep learning and process understanding for data-driven Earth system science [Publisher: Nature Publishing Group]. *Nature*, *566*(7743), 195–204. https://doi.org/10.1038/s41586-019-0912-1

Roberts, D. R., Bahn, V., Ciuti, S., Boyce, M. S., Elith, J., Guillera-Arroita, G., Hauenstein, S., Lahoz-Monfort, J. J., Schröder, B., Thuiller, W., Warton, D. I., Wintle, B. A., Hartig, F., & Dormann, C. F. (2017). Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography*, *40*(8), 913–929. https://doi.org/10.1111/ecog.02881

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6), 386–408. https://doi.org/10.1037/h0042519

Rothermel, R. C. (1983). *How to Predict the Spread and Intensity of Forest and Range Fires* [Google-Books-ID: beRPolhCd8gC]. U.S. Department of Agriculture, Forest Service, Intermountain Forest; Range Experiment Station.

Ruffault, J., Curt, T., Martin-StPaul, N. K., Moron, V., & Trigo, R. M. (2018). Extreme wildfire events are linked to global-change-type droughts in the northern Mediterranean [Publisher: Copernicus GmbH]. *Natural Hazards and Earth System Sciences*, *18*(3), 847–856. https://doi.org/10.5194/nhess-18-847-2018

Ruffault, J., Curt, T., Moron, V., Trigo, R. M., Mouillot, F., Koutsias, N., Pimont, F., Martin-StPaul, N., Barbero, R., Dupuy, J.-L., Russo, A., & Belhadj-Khedher, C. (2020). Increased likelihood of heat-induced large wildfires in the Mediterranean Basin. *Scientific Reports*, *10*(1), 13790. https://doi.org/10.1038/s41598-020-70069-z

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors [Publisher: Nature Publishing Group]. *Nature*, *323*(6088), 533–536. https://doi.org/10.1038/323533a0

Salih, A. M. (2024). Explainable Artificial Intelligence and Multicollinearity : A Mini Review of Current Approaches. https://doi.org/10.48550/arXiv.2406.11524

San-Miguel-Ayanz, J., Durrant, T., Boca, R., Maianti, P., Libertá, G., Artés-Vivancos, T. A., Oom, D., Branco, A., Rigo, D. d., Ferrari, D., Pfeiffer, H., Grecchi, R., & Nuijten, D. (2022). *Advance report on wildfires in Europe, Middle East and North Africa 2021*. Publications Office of the European Union. Retrieved June 17, 2025, from https://data.europa.eu/doi/10.2760/039729

Sasaki, Y. (2007). The truth of the F-measure. *1*, 5. https://nicolasshu.com/assets/pdf/Sasaki_2007_The%20Truth%20of%20the%20F-measure.pdf

Shams Eddin, M. H., Roscher, R., & Gall, J. (2023). Location-Aware Adaptive Normalization: A Deep Learning Approach for Wildfire Danger Forecasting. *IEEE Transactions on Geoscience and Remote Sensing*, *61*, 1–18. https://doi.org/10.1109/TGRS.2023.3285401

Shapley & Lloyd, S. (1953). *A value for n-person games*. Princeton University Press Princeton.

Shaw, P., Uszkoreit, J., & Vaswani, A. (2018). Self-Attention with Relative Position Representations. https://doi.org/10.48550/arXiv.1803.02155

Shrikumar, A., Greenside, P., & Kundaje, A. (2019). Learning Important Features Through Propagating Activation Differences. https://doi.org/10.48550/arXiv.1704.02685

Singla, S., Mukhopadhyay, A., Wilbur, M., Diao, T., Gajjewar, V., Eldawy, A., Kochenderfer, M., Shachter, R., & Dubey, A. (2021). WildfireDB: An Open-Source Dataset Connecting Wildfire Spread with Relevant Determinants. https://doi.org/10.5281/zenodo.5636429

Strobl, C., Boulesteix, A.-L., Zeileis, A., & Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics*, *8*, 25. https://doi.org/10.1186/1471-2105-8-25

Sun, H., Wang, W. J., Liu, Z., Zou, X., Zhang, Z., Ying, H., Dong, Y., & Yang, R. (2021). The relative importance of driving factors of wildfire occurrence across climatic gradients in the Inner Mongolia, China. *Ecological Indicators*, *131*. https://doi.org/10.1016/j.ecolind.2021.108249

Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic Attribution for Deep Networks. *International Conference on Machine Learning*, 3319–3328.

Swain, D. L., Prein, A. F., Abatzoglou, J. T., Albano, C. M., Brunner, M., Diffenbaugh, N. S., Singh, D., Skinner, C. B., & Touma, D. (2025). Hydroclimate volatility on a warming Earth [Publisher: Nature Publishing Group]. *Nature Reviews Earth & Environment*, *6*(1), 35–50. https://doi.org/10.1038/s43017-024-00624-z

Tatem, A. J. (2017). WorldPop, open data for spatial demography [Publisher: Nature Publishing Group]. *Scientific Data*, *4*(1), 170004. https://doi.org/10.1038/sdata.2017.4

Taud, H., & Mas, J. (2018). Multilayer Perceptron (MLP). In M. T. Camacho Olmedo, M. Paegelow, J.-F. Mas, & F. Escobar (Eds.), *Geomatic Approaches for Modeling Land Change Scenarios* (pp. 451–455). Springer International Publishing. https://doi.org/10.1007/978-3-319-60801-3_27

Tavakkoli Piralilou, S., Einali, G., Ghorbanzadeh, O., Nachappa, T. G., Gholamnia, K., Blaschke, T., & Ghamisi, P. (2022). A Google Earth Engine Approach for Wildfire Susceptibility Prediction Fusion with Remote Sensing Data of Different Spatial Resolutions [Number: 3 Publisher: Multidisciplinary Digital Publishing Institute]. *Remote Sensing*, *14*(3), 672. https://doi.org/10.3390/rs14030672

Tibshirani, R., Friedman, J., & Trevor, H. (2009). The Elements of Statistical Learning: Data mining, inference, and prediction.

Tonini, M., DAndrea, M., Biondi, G., Degli Esposti, S., Trucchia, A., & Fiorucci, P. (2020). A Machine Learning-Based Approach for Wildfire Susceptibility Mapping. The Case Study of the Liguria Region in Italy [Number: 3 Publisher: Multidisciplinary Digital Publishing Institute]. *Geosciences*, *10*, 105. https://doi.org/10.3390/geosciences10030105

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

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser,. u., & Polosukhin, I. (2017). Attention is All you Need. *Advances in Neural Information Processing Systems*, *30*. Retrieved May 9, 2025, from https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

Vazquez, A., & Moreno, J. M. (1998). Patterns of Lightning-, and People-Caused Fires in Peninsular Spain [Publisher: CSIRO PUBLISHING]. *International Journal of Wildland Fire*, *8*(2), 103–115. https://doi.org/10.1071/wf9980103

Wan, Z., Hook, S., & Hulley, G. (2015). MOD11A1 MODIS/Terra Land Surface Temperature and the Emissivity Daily L3 Global 1km SIN Grid. *NASA LP DAAC*.

Wang, S. S.-C., & Wang, Y. (2020). Quantifying the effects of environmental factors on wildfire burned area in the south central US using integrated machine learning techniques [Publisher: Copernicus GmbH]. *Atmospheric Chemistry and Physics*, *20*(18), 11065–11087. https://doi.org/10.5194/acp-20-11065-2020

Weise, D. R., & Biging, G. S. (1997). A Qualitative Comparison of Fire Spread Models Incorporating Wind and Slope Effects. *Forest Science*, *43*(2), 170–180. https://doi.org/10.1093/forestscience/43.2.170

Wikipedia. (2022a). 2022 European and Mediterranean wildfires [Page Version ID: 1299055999]. Retrieved July 10, 2025, from https://en.wikipedia.org/w/index.php?title=2022_European_and_Mediterranean_wildfires&oldid=1299055999#Spain

Wikipedia. (2022b). Incendios de la sierra de la Culebra de 2022 [Page Version ID: 165315842]. Retrieved July 1, 2025, from https://es.wikipedia.org/w/index.php?title=Incendios_de_la_sierra_de_la_Culebra_de_2022&oldid=165315842

World Tourism Organisation. (2018). UNWTO Tourism Highlights: 2018 Edition | World Tourism Organization. Retrieved July 25, 2025, from https://www.e-unwto.org/doi/book/10.18111/9789284419876

Xanthopoulos, G. (2021). SUPPRESSION VERSUS PREVENTION-THE DISASTROUS FOREST FIRE SEASON OF 2021 IN GREECE. Retrieved July 1, 2025, from https://www.iawfonline.org/article/suppression-versus-prevention-the-disastrous-forest-fire-season-of-2021-in-greece/

Xu, Z., Li, J., Cheng, S., Rui, X., Zhao, Y., He, H., & Xu, L. (2024, October). Wildfire Risk Prediction: A Review. https://doi.org/10.48550/arXiv.2405.01607

Zhang, G., Wang, M., & Liu, K. (2019). Forest Fire Susceptibility Modeling Using a Convolutional Neural Network for Yunnan Province of China. *International Journal of Disaster Risk Science*, *10*(3), 386–403. https://doi.org/10.1007/s13753-019-00233-1

Zhang, Y., Tio, P., Leonardis, A., & Tang, K. (2021). A Survey on Neural Network Interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *5*(5), 726–742. https://doi.org/10.1109/TETCI.2021.3100641

Zhao, H., Zhang, Z., Wang, X., Zhen, S., Zhang, H., Bu, Z.-J., Zhao, J., Guo, X., Wei, K., & Dong, L. (2025). Future enhanced threshold effects of wildfire drivers could increase burned areas in northern mid- and high latitudes [Publisher: Nature Publishing Group]. *Communications Earth & Environment*, *6*(1), 224. https://doi.org/10.1038/s43247-025-02202-7

Appendix

A. Data

Abbreviation	Description
	Fuel
lai	Leaf Area Index (leaf surface area m^2/ground surface area m^2)
ndvi	Normalized Difference Vegetation Index [-1;1]
lc_agriculture	Land cover class: agriculture [0;1]
lc_forest	Land cover class: forest [0;1]
lc_grassland	Land cover class: grassland [0;1]
lc_shrubland	Land cover class: shrubland [0;1]
lc_sparse_vegetation	Land cover class: sparse vegetation [0;1]
lc_settlement	Land cover class: settlement [0;1]
lc_water_bodies	Land cover class: water bodies [0;1]
lc_wetland	Land cover class: wetland [0;1]
	Meterology
d2m	Dewpoint temperature at 2m (K)
lst_day	Day land surface temperature (K)
lst_night	Night land surface temperature (K)
rh	Relative Humidity [0;1]
smi	Soil Moisture Index [0;1]
sp	Surface Pressure (Pa)
ssrd	Surface Solar Radiation Downwards (J/m^2)
t2m	Temperature at 2m above the Earth's surface (K)
tp	Total Precipitation (m)
wind_speed	Wind Speed (m/s)
	Human Factors
population	Population
roads_distance	Distance from Roads
	Topographical Data
dem	Elevation (m)
slope	Slope of the Area

Table A.1.: **Categories of Wildfire Contributing Factors.** Overview of all explanatory variables used in the fire prediction models, grouped into four categories: Fuel, Meteorology, Human Factors and Topographical Data.

Variable	Sign	References
d2m	–	Chuvieco et al., 2023
lc_agriculture	+	Mukunga et al., 2023
lc_forest	+	Chuvieco et al. (2023) and Zhao et al. (2025)
lc_grassland	+	Radeloff et al. (2023) and Xu et al. (2024)
lc_settlement	–	Zhao et al. (2025)
lc_shrubland	+	Chuvieco et al. (2023), Radeloff et al. (2023), and Xu et al. (2024)
lc_sparse_vegetation	+	Chuvieco et al. (2023)
lc_water_bodies	–	Zhao et al. (2025)
lc_wetland	–	Chuvieco et al. (2023)
lst_day	+	Chuvieco et al. (2023) and Di Giuseppe et al. (2025)
lst_night	+	Di Giuseppe et al. (2025)
rh	–	Andela and van der Werf (2014) and Holsten et al. (2013)
roads_distance	+	Mukunga et al. (2023) and Zhao et al. (2025)
slope	+	Pimont et al. (2012) and Weise and Biging (1997)
smi	–	Andela and van der Werf (2014) and Chuvieco et al. (2023)
ssrd	+	Chuvieco et al. (2023) and Di Giuseppe et al. (2025)
tp	–	Andela and van der Werf (2014) and Zhao et al. (2025)
t2m	+	Chuvieco et al. (2023) and Di Giuseppe et al. (2025)
wind_speed	+	Pimont et al. (2012) and Weise and Biging (1997)

Table A.2.: **Physical Relationships between Wildfire Drivers and Fire Occurrence.** The table summarizes the expected direction of influence (positive or negative) of key environmental and anthropogenic variables on wildfire occurrence, based on findings from recent literature. A positive sign (+) indicates a promoting effect on fire occurrence, while a negative sign (–) suggests a suppressing effect.

Variable	Source	Sp. Res.	Temp. Res.	Units
Dynamic variables				
Max Temperature	ERA5-Land	9 km	hourly	K
Max Wind Speed	ERA5-Land	9 km	hourly	m/s
Max Wind Direction	ERA5-Land	9 km	hourly	ř
Max Dewpoint Temperature	ERA5-Land	9 km	hourly	K
Max Surface Pressure	ERA5-Land	9 km	hourly	Pa
Min Relative Humidity	ERA5-Land	9 km	hourly	%
Total Precipitation	ERA5-Land	9 km	hourly	m
Mean Surface Solar Radiation Downwards	ERA5-Land	9 km	hourly	J/m^2
Day Land Surface Temperature	MODIS	1 km	daily	K
Night Land Surface Temperature	MODIS	1 km	daily	K
Normalized Difference Vegetation Index (NDVI)	MODIS	500 m	16-days	-
Leaf Area Index (LAI)	MODIS	500 m	8-days	-
Soil Moisture	EDO	5 km	10-days	-
Burned Areas	EFFIS	1 km	vector	{0,1}
Ignition Points	MODIS	1 km	vector	hectares
Semi-static variables				
Population	Worldpop	1 km	yearly	$people/km^2$
Fraction of agriculture	Copernicus CCS	300 m	yearly	%
Fraction of forest	Copernicus CCS	300 m	yearly	%/
Fraction of grassland	Copernicus CCS	300 m	yearly	%
Fraction of settlements	Copernicus CCS	300 m	yearly	%
Fraction of shrubland	Copernicus CCS	300 m	yearly	%
Fraction of sparse vegetation	Copernicus CCS	300 m	yearly	%
Fraction of water bodies	Copernicus CCS	300 m	yearly	%
Fraction of wetland	Copernicus CCS	300 m	yearly	%
Static variables				
Roads distance	Worldpop	1 km	static	km
Elevation	COP-DEM	30 m	static	m
Slope	COP-DEM	30 m	static	rad
Aspect	COP-DEM	30 m	static	ř
Curvature	COP-DEM	30 m	static	rad

Table A.3.: **Overview of Variables in the Mesogeos Dataset.** The table derived from Kondylatos, Prapas, Camps-Valls, and Papoutsis (2023) lists all variables included in the Mesogeos Dataset, grouped by temporal scale into dynamic, semi-static, and static categories.

B. Results

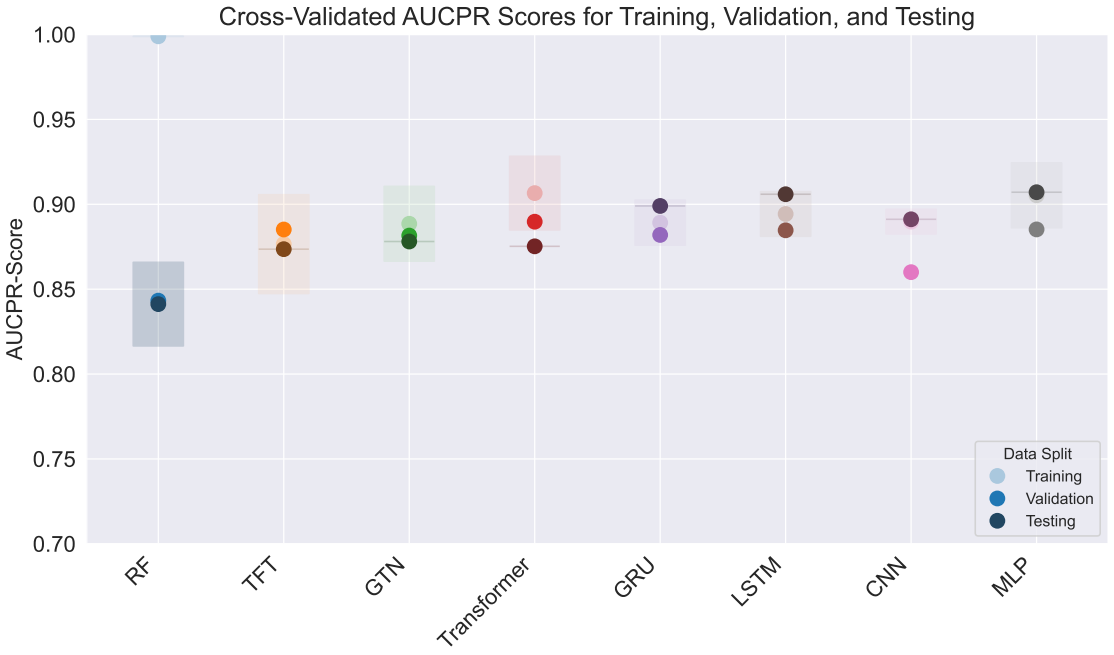

Figure B.1.: **AUPRC-Scores across Training, Validation, and Test Sets.** The figure shows three points per model with shaded boxes indicating the standard deviation across cross-validation splits for training and testing. The AUPRC results remain largely consistent with the overall F_1-score results, with the Random Forest model showing pronounced overfitting compared to the neural network architectures.

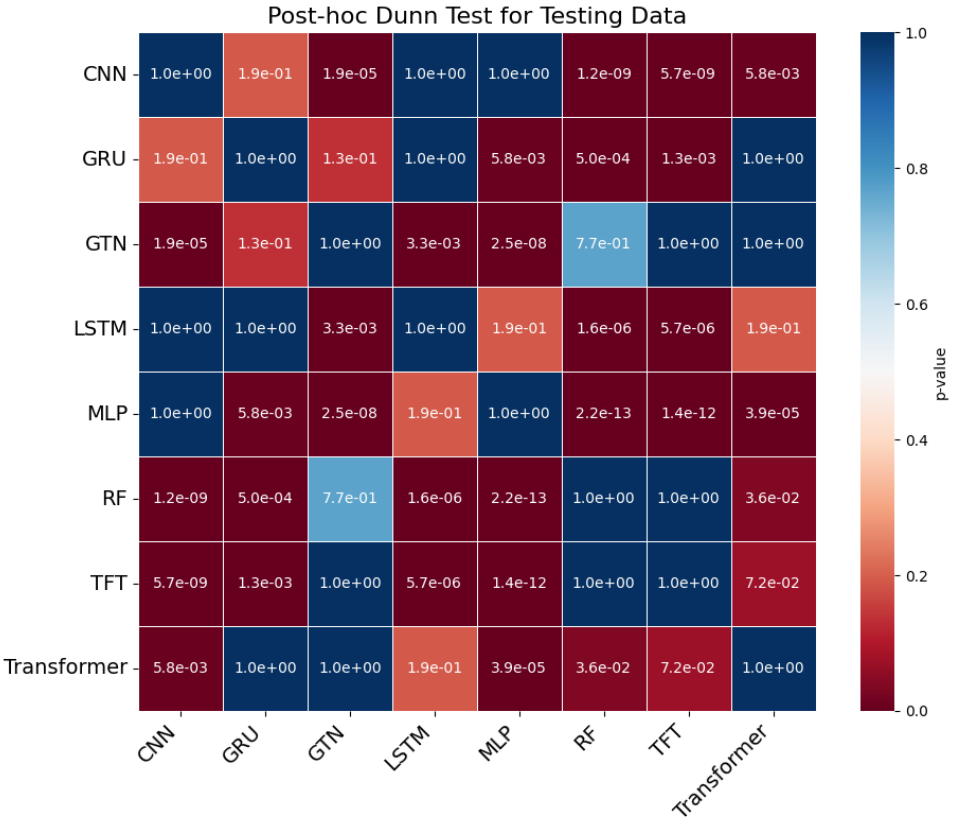

Figure B.2.: **Pairwise Dunn Test Heatmap of the Testing Data.** The heatmap displays Holm-adjusted p-values for all pairwise model comparisons based on cross-validated F_1-scores. Significant differences ($p < 0.05$) are marked by darker cells. Random Forest performs significantly worse than all neural architectures, whereas differences among deep learning models are mostly non-significant.

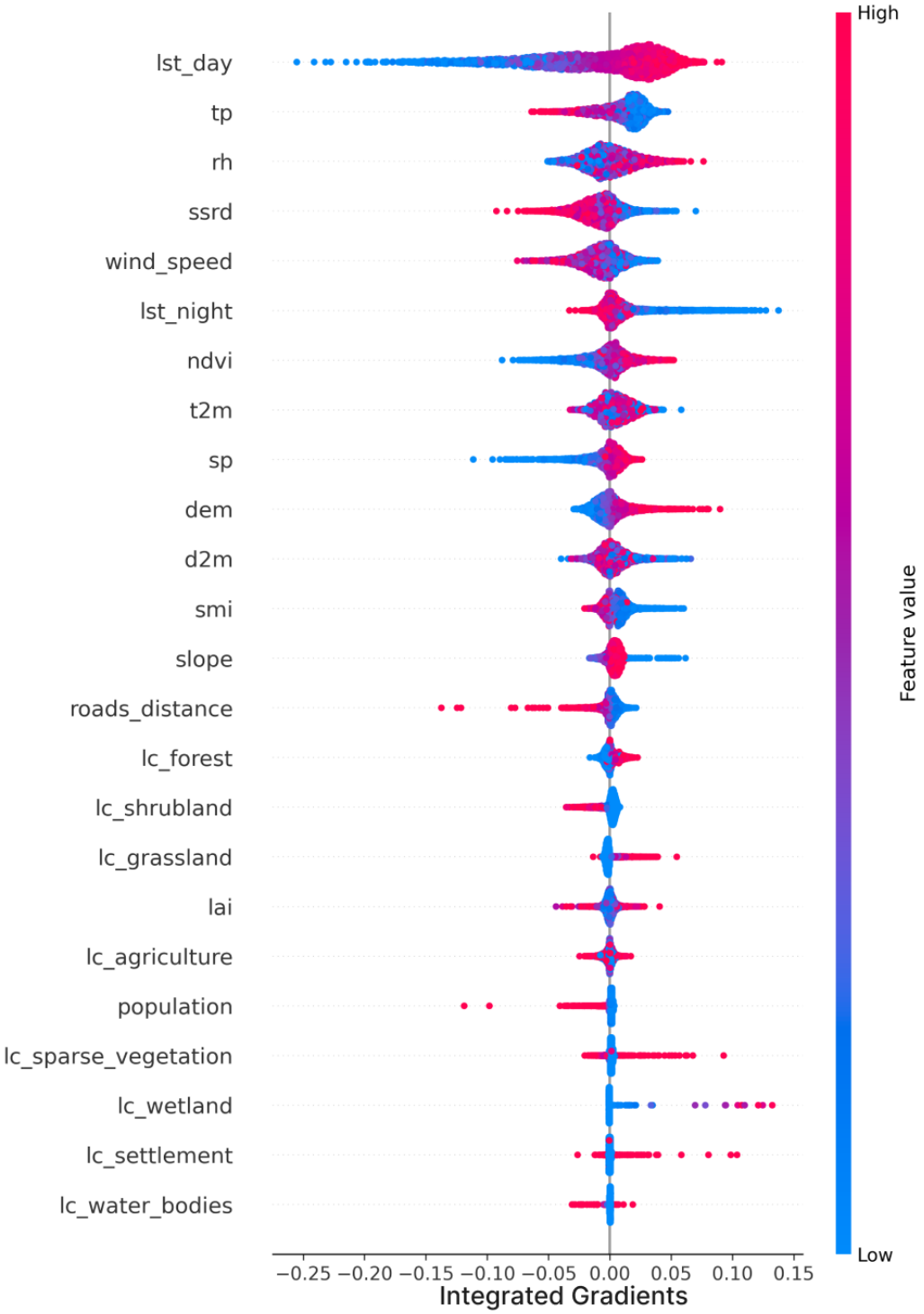

Figure B.3.: **IG Beeswarm Plots from the Transformer Model.** Integrated Gradient values are aggregated over time steps, and colored by feature value (red = high, blue = low). Features are sorted by mean importance.

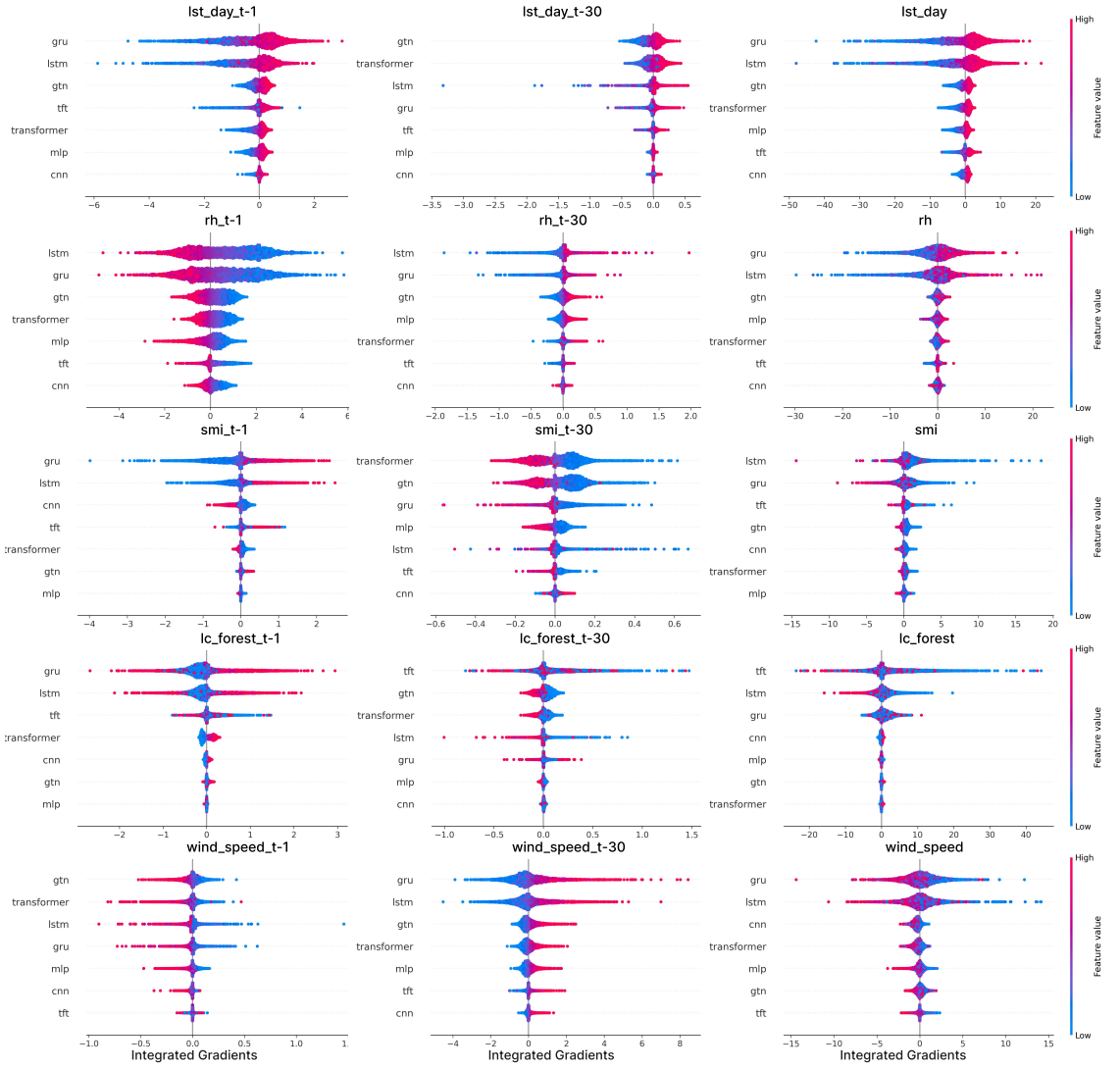

Figure B.4.: **IG Comparison Plots across all Models for selected Features.** Analogous to SHAP, the figure shows input feature attributions for t-1, t-30, and aggregated time series over 30 days prior to fire ignition. IG values represent the integrated feature contributions computed along the path from baseline to model output.

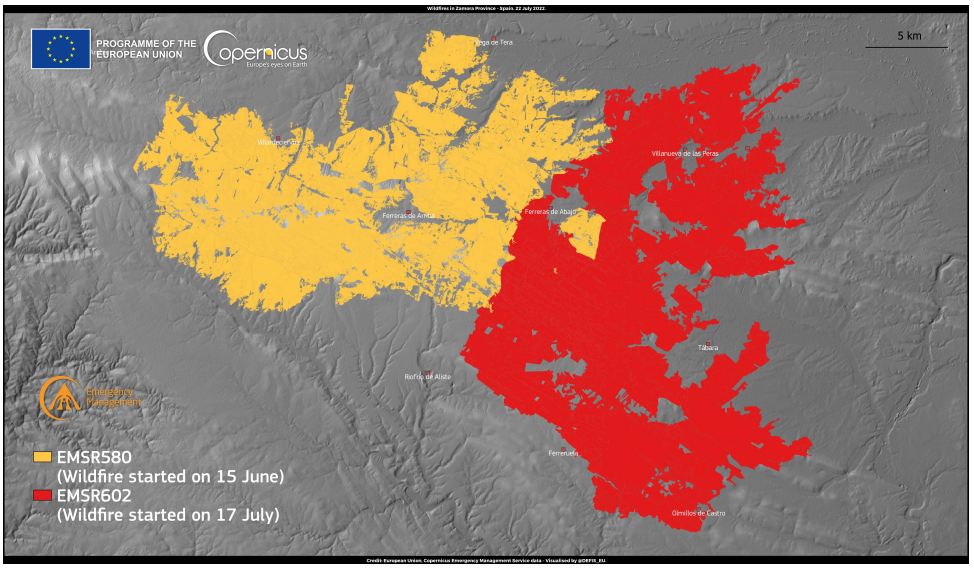

Figure B.5.: **Burned Areas of Two Big Wildfires in Zamora Province, Spain 2022.**: EMSR580 started on 15 June (yellow), EMSR602 on 17 July 2022 (red), Credit: European Union, Copernicus Emergency Management Service Data

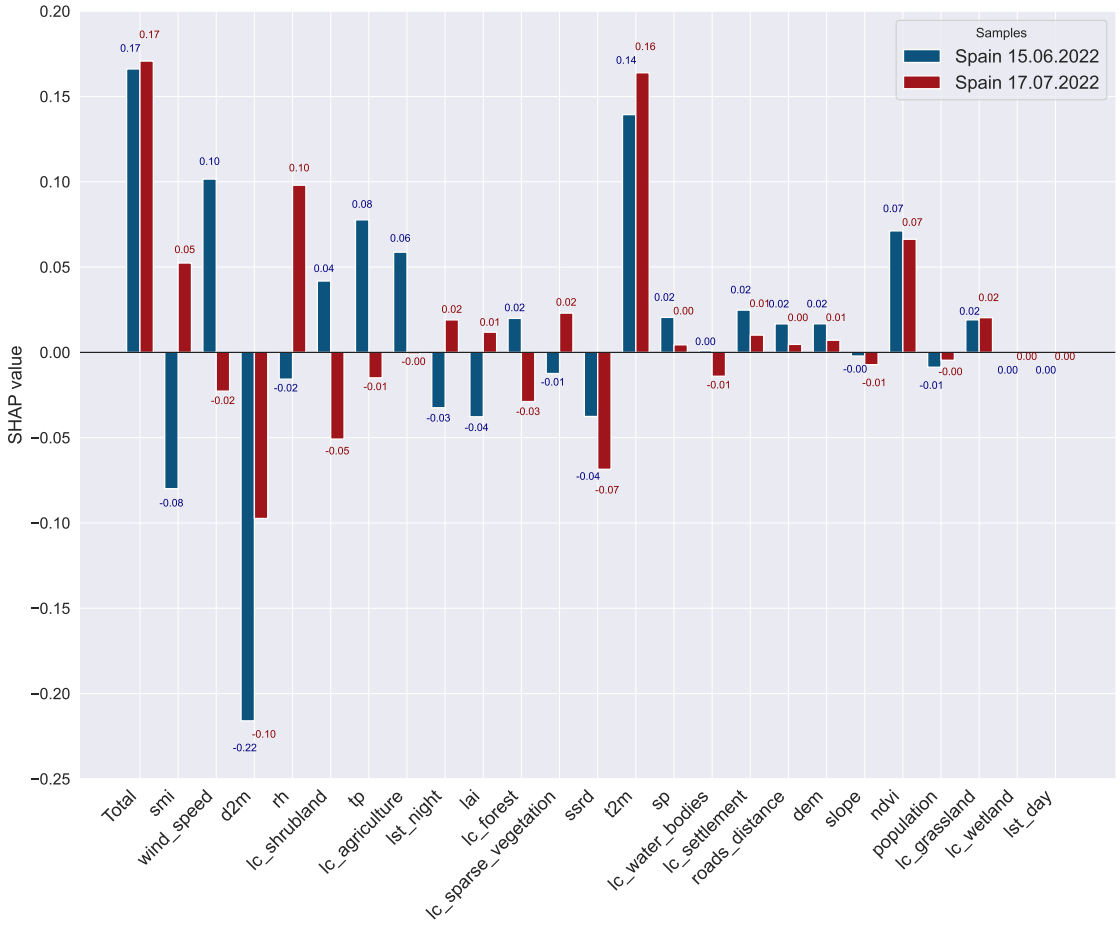

Figure B.6.: **SHAP values for the June and July Spain fire samples predicted by the model without *lst_day*.** In contrast to Figure 6.14, this version enables comparison of feature attributions in the absence of land surface temperature.

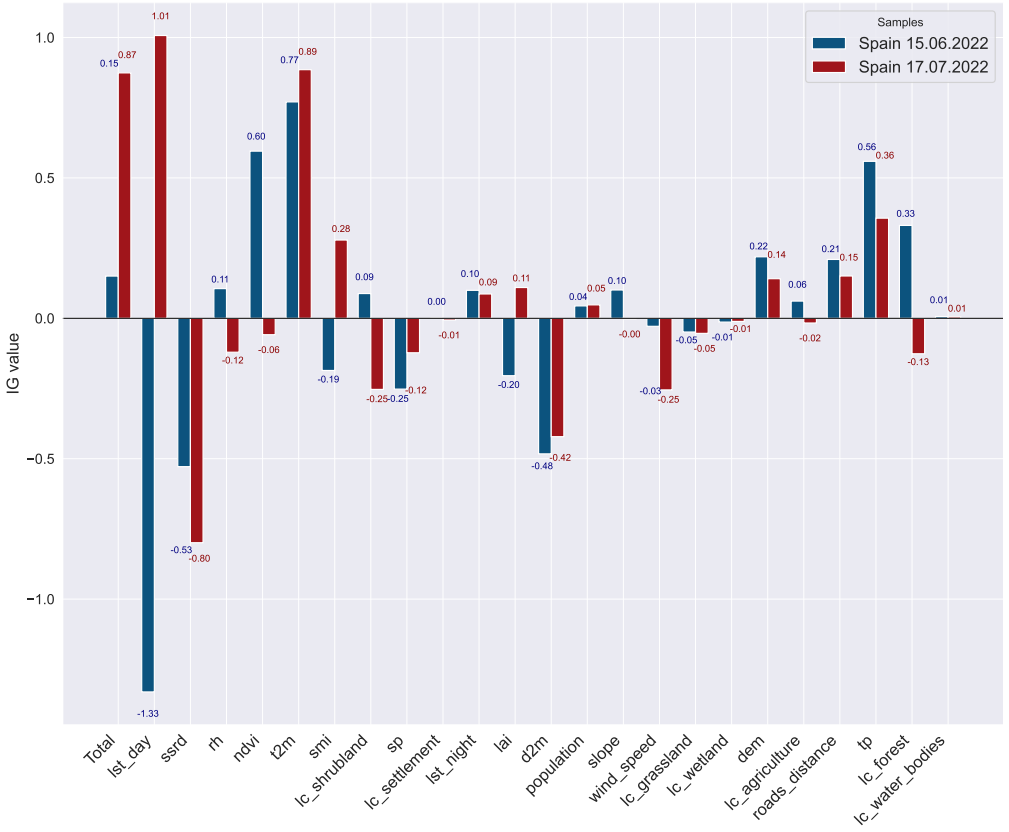

Figure B.7.: **IG Comparison of the two Spain Wildfire Events (June vs. July 2022).** Feature contributions to the final prediction are shown for both events. Colors indicate individual IG values, with red for the July event (True Positive) and blue for the June event (False Negative). To ensure comparability with Figure 6.14, features are ordered by descending absolute SHAP difference.

Affidavit

I declare that I have developed and written the enclosed thesis completely by myself, have not used sources or means without declaration in the text and that I have complied with the statutes of the Karlsruhe Institute of Technology for ensuring good scientific practice in their current version.

Karlsruhe, 07.08.2025 _____

Pauline Becker