INDUSTRIAL INTERNSHIP REPORT ON

Species Composition Prediction with High Spatial Resolution at Continental Scale Using Remote Sensing

A Major Project Report

Submitted to the Central University of Haryana

in partial fulfillment of requirements for the award of degree

Bachelor of Technology

in

Computer Science and Engineering

by

Yash Raj (202134)

Under the guidance of
Dr. Rakesh Kumar
Associate prof. & HOD
Dept. of Computer Science & Engineering

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF ENGINEERING & TECHNOLOGY
CENTRAL UNIVERSITY OF HARYANA, MAHENDRAGARH
HARYANA - 123031, IND

June 2024

DECLARATION

I hereby declare that the major project report entitled **Species Composition Prediction with High Spatial Resolution at Continental Scale Using Remote Sensing** by me Yash Raj (202134), is submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology of the Central University, Haryana Submitted in the department of Computer Science and Engineering and is a bonafide work done by us under the supervision of Dr. Rakesh Kumar. This submission represents our ideas in our own words and where ideas or words of others have been included, we have adequately and accurately cited and referenced the original sources.

Yash Raj

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Dr. Rakesh Kumar
(Head of Dept.)
Department of Computer Science & Engineering.
Central University of Haryana

Mr. Afelio Padilla
(Mentor and supervisor)
COO Ignitus Worldwide

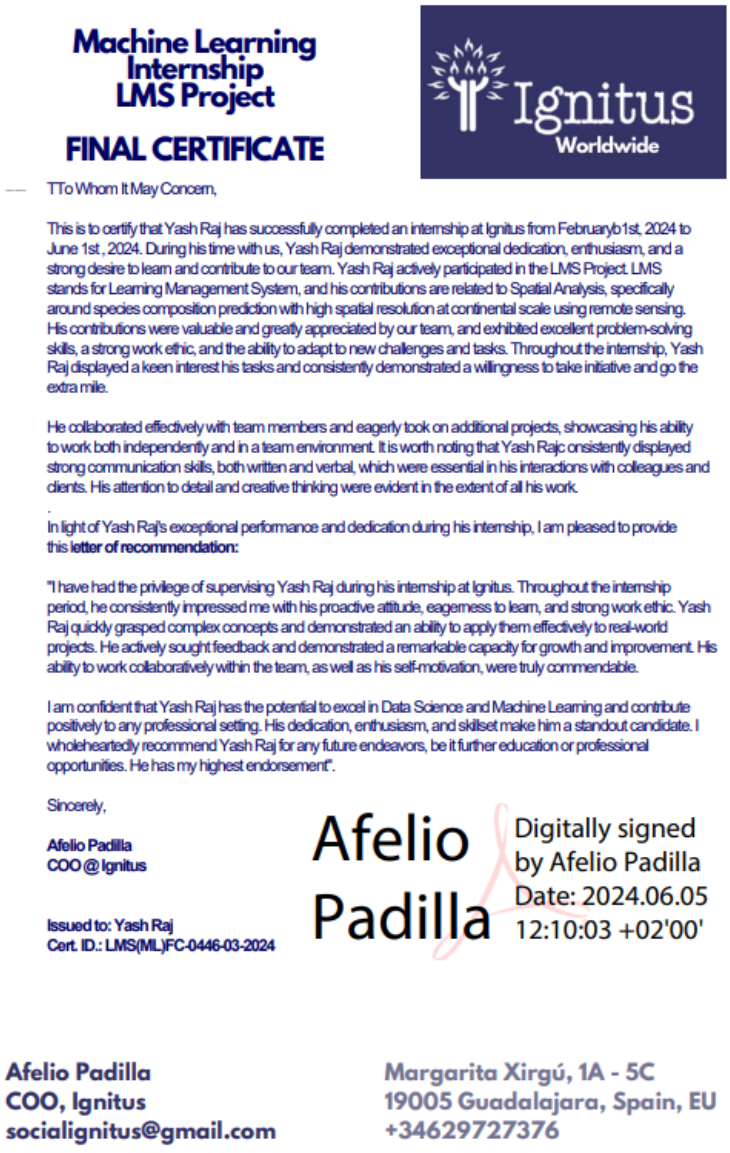
Acknowledgement

I take this opportunity to express my deepest sense of gratitude and sincere thanks to everyone who contributed to the successful completion of this work. The heartfelt appreciation goes to Dr. Rakesh kumar, the Head of the Department (HOD) of Computer Science and Engineering at Central University of Haryana, Mahendergarh, for their unwavering support and provision of all necessary facilities, which were instrumental in the realization of this project, done under their guidance.
I would also like to thank my Mentor and supervisor Mr. Afellio Padilla, COO Ignitus Worldwide for his valuable insights and mentorship throughout the internship process.
I would also like to extend my thanks to the entire faculty of the Computer Science and Engineering department for their constant encouragement, valuable insights, and academic guidance. The collective efforts of the teachers have significantly shaped our understanding and enriched our learning experience.

Furthermore, I express gratitude to myself for the dedication and hard work invested in the project. Additionally, I extend my thanks to all my friends and family members who, through their contributions, played a crucial role in the successful fulfillment of this project work.

Yash Raj (202134)

Certificate of Completion



Machine Learning Internship LMS Project

FINAL CERTIFICATE

Ignitus Worldwide

TTo Whom It May Concern,

This is to certify that Yash Raj has successfully completed an internship at Ignitus from Februaryb1st, 2024 to June 1st , 2024. During his time with us, Yash Raj demonstrated exceptional dedication, enthusiasm, and a strong desire to learn and contribute to our team. Yash Raj actively participated in the LMS Project. LMS stands for Learning Management System, and his contributions are related to Spatial Analysis, specifically around species composition prediction with high spatial resolution at continental scale using remote sensing. His contributions were valuable and greatly appreciated by our team, and exhibited excellent problem-solving skills, a strong work ethic, and the ability to adapt to new challenges and tasks. Throughout the internship, Yash Raj displayed a keen interest his tasks and consistently demonstrated a willingness to take initiative and go the extra mile.

He collaborated effectively with team members and eagerly took on additional projects, showcasing his ability to work both independently and in a team environment. It is worth noting that Yash Rajc onsistently displayed strong communication skills, both written and verbal, which were essential in his interactions with colleagues and clients. His attention to detail and creative thinking were evident in the extent of all his work.

.

In light of Yash Raj's exceptional performance and dedication during his internship, I am pleased to provide this **letter of recommendation:**

"I have had the privilege of supervising Yash Raj during his internship at Ignitus. Throughout the internship period, he consistently impressed me with his proactive attitude, eagerness to learn, and strong work ethic. Yash Raj quickly grasped complex concepts and demonstrated an ability to apply them effectively to real-world projects. He actively sought feedback and demonstrated a remarkable capacity for growth and improvement. His ability to work collaboratively within the team, as well as his self-motivation, were truly commendable.

I am confident that Yash Raj has the potential to excel in Data Science and Machine Learning and contribute positively to any professional setting. His dedication, enthusiasm, and skillset make him a standout candidate. I wholeheartedly recommend Yash Raj for any future endeavors, be it further education or professional opportunities. He has my highest endorsement".

Sincerely,

Afelio Padilla
COO @ Ignitus

Afelio Padilla
Digitally signed by Afelio Padilla
Date: 2024.06.05 12:10:03 +02'00'

Issued to: Yash Raj
Cert. ID.: LMS(ML)FC-0446-03-2024

Afelio Padilla
COO, Ignitus
socialignitus@gmail.com

Margarita Xirgú, 1A - 5C
19005 Guadalajara, Spain, EU
+34629727376

Abstract

Understanding the spatio-temporal distribution of species is a cornerstone of ecology and conservation. This project, titled "Location-based Species Presence Prediction," aims to advance species composition prediction using deep learning models and remote sensing data. By pairing 5 million plant species observations across Europe with high-resolution remote sensing imagery, land cover, elevation data, and coarse-resolution climate, soil, and human footprint variables, we developed models to predict species presence in 22,000 small plots. This project leverages a large-scale training set (single-label, presence-only data) and a test set (multi-label, presence-absence data) to enhance biodiversity management, conservation efforts, and species identification tools.

Our approach addresses the challenges in ecological modeling, particularly the biases associated with single positive label methods in multi-label evaluations. We introduced a novel learning strategy that combines single and multi-label data, leading to significant improvements in prediction accuracy. The outcomes of this project not only contribute to the scientific understanding of species distribution but also provide practical tools for conservation planning, policy-making, and educational purposes. By improving species composition prediction at fine spatial resolutions, this initiative supports proactive biodiversity management and helps mitigate the impacts of environmental changes on ecosystems.

Keywords

Species distribution modeling, multimodal data, neural network architecture, landsat, sentinel, bioclimatic data, environmental data, satellite imagery, model ensemble, class imbalance, biodiversity, model interpretation, prediction.

Contents

List of Figures

Chapter 1

Introduction

Understanding the spatio-temporal distribution of plant species is fundamental to ecology, conservation, and biodiversity management. The ability to accurately predict species presence at specific locations is vital for several applications, including the creation of high-resolution maps of species composition, biodiversity indicators, and the management of endangered or invasive species. This capability falls under the domain of Species Distribution Modelling (SDM), a well-established scientific practice that pairs species observations with geographic and environmental predictors to model the relationship between an environment and the species it supports.

Recent advancements in remote sensing and machine learning have opened new avenues for enhancing SDM. High-resolution satellite imagery, climate data, soil information, and other environmental variables can now be integrated into sophisticated models to provide more accurate and detailed predictions. This integration is particularly valuable for addressing ecological challenges posed by climate change, habitat destruction, and biodiversity loss. I presented an overview of my approach, the data utilized, and the modeling techniques employed. I also synthesize the strategies used by participating teams in the GeoLifeCLEF 2023 challenge and analyze the main results. Notably, I address the biases encountered with methods trained on single positive labels when applied to multi-label evaluation and introduce a new, effective learning strategy that combines single and multi-label data in training. The findings demonstrate significant improvements in prediction accuracy and provide valuable insights for future research in ecological modeling and conservation planning.

Furthermore, we delve into the intricacies of model evaluation and validation, highlighting the challenges associated with assessing model performance in multi-label prediction scenarios. Through a rigorous analysis of biases and limitations inherent in traditional evaluation metrics, we propose novel methodologies for mitigating these challenges and enhancing the robustness of SDM frameworks.

In addition to presenting our own methodology, we provide a synthesis of approaches adopted by leading research teams in the GeoLifeCLEF 2023 challenge. By analyzing the strategies and techniques employed by participants, we gain valuable insights into the state-of-the-art in SDM and identify emerging trends and best practices in the field.

Ultimately, our study aims to not only advance the theoretical foundations of species distribution modelling but also provide actionable insights for conservation practitioners and policymakers. By elucidating the complex relationships between species and their environment, we strive to empower stakeholders with the knowledge and tools necessary to safeguard biodiversity in the face of global environmental change. Our methodology also emphasizes the importance of integrating spatial and temporal dimensions into species distribution models. By accounting for the dynamic nature of ecosystems and the temporal variability of environmental conditions, our models offer more robust predictions of species distributions over time. This spatio-temporal perspective is particularly relevant in the context of ongoing environmental changes, where understanding how species respond to shifting environmental conditions is paramount for effective conservation planning.

By synthesizing insights from both our own methodology and the broader landscape of SDM research, our study seeks to push the boundaries of predictive ecology and pave the way for more effective conservation interventions. Through a combination of cutting-edge techniques and interdisciplinary collaboration, we aspire to unlock new avenues for understanding and conserving the rich tapestry of life on our planet.

1.1 About the Organisation

The ignitus worldwide is a europe based scientific non-profitable organisation aims in promoting the development of the people's intellectual potential.They address college students who want to boost their career and universities, professionals, researchers and startups looking for collaborations hiring students, and also address whoever wanting to step-up their careers as a researcher, serving as catalysts and as bridges between the different actors of the educational and industrial universe.

Website: http://ignitus.org

Industry: Education Administration Programs

Company size :2-10 employees

Headquarters :Pittsburgh, Pennsylvania

Founded in 2018

Specialties: researchOpportunities, Internships, and BuildingCareer

1.2 Collaborations

Figure 1.1: Collaborators

This project has received funding from the European Union's Horizon Research and Innovation program under grant agreements No. 101060639 (MAMBO project) and No. 101060693 (GUARDEN project).

Organizers and contributors

Lukas Picek, INRIA, LIRMM, Montpellier

Christophe Botella, INRIA, LIRMM, Montpellier

Diego Marcos, INRIA , Montpellier

Théo Larcher, INRIA, LIRMM, Montpellier

Joachim Estopinan, INRIA, LIRMM, Montpellier

César Leblanc, INRIA, LIRMM, Montpellier

Maximilien Servajean, Université Paul Valéry, LIRMM, Montpellier

Alexis Joly, INRIA, LIRMM, Montpellier

The project report explores impactful initiatives of the organisation, ranging from a data analysis to awards recognizing contributions in educational purposes through biodiversity exploration applications with features such as quests or contextualized educational pathways. By highlighting the unique contributions of these endeavors, the report underscores their broader impact on biodiversity management and conservation scenarios..

Chapter 2

Objective

The primary objective of the ”Location-based Species Presence Prediction” project is to develop and fine-tune advanced machine learning models capable of accurately predicting plant species presence at specific locations and times, utilizing a diverse array of predictors such as satellite imagery, climatic time series, land cover, human footprint, bioclimatic, and soil variables. This initiative aims to generate high-resolution maps of species composition and related biodiversity indicators, including species diversity, endangered species, and invasive species across Europe. By enhancing the accuracy of species identification tools like Pl@ntNet, the project seeks to refine the list of potential species observable at given sites, thereby improving existing biodiversity management and conservation tools. Additionally, the project aims to facilitate biodiversity inventories by developing location-based recommendation services that encourage citizen scientist participation and accelerate the annotation and validation of species observations. Addressing the challenges of multi-label learning from single positive labels, handling strong class imbalances, and integrating multi-modal data are key technical objectives, ensuring the robustness and accuracy of the predictive models. Moreover, the project endeavors to create educational resources that engage users in biodiversity exploration through features such as quests and contextualized educational pathways. Supporting conservation planning and biodiversity management through actionable insights derived from species presence predictions is another critical goal. Lastly, by participating in the GeoLifeCLEF 2024 challenge, the project aims to advance the state-of-the-art in species distribution mod-

eling and demonstrate the effectiveness of the developed models through competitive benchmarking. Together, these objectives aim to significantly contribute to the fields of ecology, conservation, and biodiversity management, while also providing valuable tools and resources for education and citizen science.

Motivation

Predicting the plant species present at a given location is helpful for many biodiversity management and conservation scenarios.

First, it allows for building high-resolution maps of species composition and related biodiversity indicators such as species diversity, endangered species, and invasive species. In scientific ecology, the problem is known as Species Distribution Modelling.

Moreover, it could significantly improve the accuracy of species identification tools - such as Pl@ntNet - by reducing the list of candidate species observable at a given site.

More generally, it could facilitate biodiversity inventories by developing location-based recommendation services (e.g., on mobile phones), encouraging citizen scientist observers' involvement, and accelerating the annotation and validation of species observations to produce large, high-quality data sets.

Finally, this could be used for educational purposes through biodiversity exploration applications with features such as quests or contextualized educational pathways.

Chapter 3

Methodology

3.1 Siamese neural network

3.1.1 Introduction

The Siamese approach, commonly used in machine learning and computer vision. The Siamese neural network, named after the famous ”Siamese twins” who are identical, represents a powerful architecture for tasks involving similarity measurement, matching, or verification. It was first introduced by Bromley et al. in the early 1990s for signature verification but has since found widespread applications in various fields, including computer vision, natural language processing, and recommendation systems.refers to a specific type of neural network architecture designed for tasks that involve comparing or finding the similarity between two inputs.A Siamese network consists of two or more identical subnetworks (neural networks) that share the same parameters and weights. These subnetworks process different inputs and their outputs are then compared to determine the similarity between the inputs. The fundamental concept of the Siamese architecture lies in its use of twin networks, which share the same architecture and parameters. These twin networks process two different inputs, typically referred to as the ”anchor” and the ”comparison” or ”positive” and ”negative” examples in the context of similarity measurement or verification tasks. The Siamese neural network architecture offers a powerful and flexible framework for tasks involving similarity measurement, matching, and verification, with applications across numerous domains. Its ability to learn robust representations from pairs of

inputs makes it well-suited for a wide range of machine learning tasks.

3.1.2 Key Features

Often one of the output vectors is precomputed, thus forming a baseline against which the other output vector is compared. This is similar to comparing fingerprints but can be described more technically as a distance function for locality-sensitive hashing

- **Parameter Sharing:** The subnetworks are identical in structure and share weights. This means that the same weights are used to process both inputs, ensuring that both are subjected to the same transformations.
- **Distance Metric:** After processing the inputs through the subnetworks, a distance metric (e.g., Euclidean distance, cosine similarity) is used to measure the similarity or dissimilarity between the outputs. The network is trained to minimize or maximize this distance depending on the task.

How Siamese Network is useful?

The Siamese approach adapted for processing multiple inputs with different modalities and formats, where each modality is processed with a different backbone also called as encoder.
Multimodal Siamese Approach includes:

1. **Multiple Modalities:** Different types of data (e.g., text, image, audio) are considered as different modalities.Each modality is processed by a distinct backbone network (encoder) tailored to handle its specific type of data.
2. **Backbone Networks:** Text: Processed using models like LSTM, GRU, or Transformer-based architectures (e.g., BERT).Image: Processed using Convolutional Neural Networks (CNNs) like ResNet, VGG, or EfficientNet.Audio: Processed using Recurrent Neural Networks (RNNs), CNNs, or specialized architectures like WaveNet.
3. **Encoding to 1D Vectors:** Each backbone network encodes its respective input into a fixed-length 1D feature vector.This encoding captures the essential

characteristics of the input data in a compact form.

4. **Concatenation of Feature Vectors:** The 1D vectors produced by each backbone are concatenated to form a single, unified feature vector.This combined vector represents the fused information from all modalities.

5. **Classification:** The concatenated feature vector is passed through a simple fully connected neural network (FCNN).The FCNN performs the final classification based on the combined features.

Process Overview

1. **Input:** Multiple inputs of different modalities (e.g., text, image, audio).

2. **Backbone Networks (Encoders):**
   Text Encoder → Text Feature Vector
   Image Encoder → Image Feature Vector
   Audio Encoder → Audio Feature Vector

3. **Feature Concatenation:**Concatenate the feature vectors from all encoders into a single vector.

4. **Fully Connected Neural Network (FCNN):**The concatenated vector is fed into an FCNN for classification.

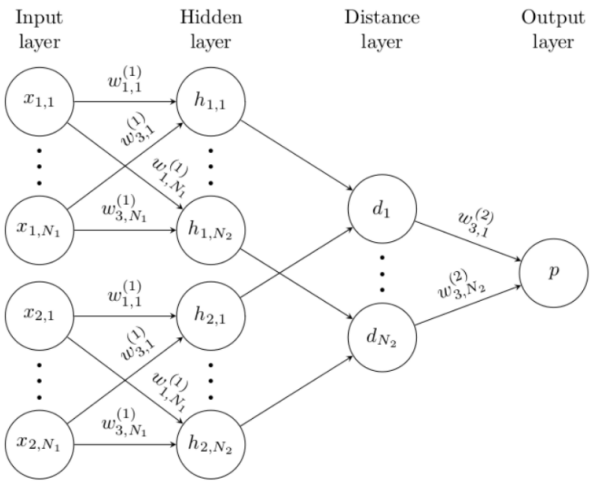


Figure 3.1: A simple 2 hidden layer siamese network

A straightforward Siamese network for binary classification featuring two hidden layers and a logistic prediction p. The network's structure is duplicated in both the top and bottom sections to create twin networks, utilizing shared weight matrices at each layer. Datasets where very few examples exist for some classes, offering a flexible and continuous method for incorporating inter-class information into the model.

3.1.3 Advantages of the Siamese Approach

1. **Parameter Efficiency and Consistency:**

    - **Shared Weights:** Siamese networks utilize shared weights between identical subnetworks, significantly reducing the number of parameters to be trained. This not only enhances training efficiency but also ensures that both inputs are processed consistently, extracting comparable features without introducing variability from different subnetworks. This consistency is crucial for tasks that rely on accurate comparison or similarity measurement, leading to more reliable and robust performance in applications such as face verification, signature verification, and image similarity.
    - **Robust Feature Extraction:** The consistency provided by identical subnetworks with shared weights leads to robust feature extraction. Each subnetwork extracts features from its respective input in the same way, producing comparable feature vectors. This reliability is crucial for applications like face verification, where the goal is to determine if two face images belong to the same person. The consistent processing ensures that similar inputs yield similar feature vectors, improving the accuracy and reliability of the similarity measurements. This approach also helps in regularizing the model, as the shared weights act as a form of regularization, reducing the risk of overfitting, especially when training data is limited..

2. **Flexibility and Enhanced Generalization:**

    - **Multiple Modalities and Inter-Class Information:** The Siamese ap-

proach is highly adaptable, allowing for the integration of various types of data (e.g., text, image, audio) by using specialized sub networks for each modality. This flexibility enables the fusion of diverse information sources, improving the model's capability to handle complex, multimodal data. Additionally, by learning to differentiate between pairs of inputs, Siamese networks effectively incorporate inter-class information, which enhances their ability to generalize to new, unseen classes. This makes them particularly useful for few-shot learning tasks, where they can generalize from very few examples, and for handling imbalanced datasets, as balanced pairing during training helps mitigate class imbalance issues.

3.1.4 Applications of the Siamese Approach

The Siamese approach has diverse applications across various domains. In face verification, it determines if two face images belong to the same person, enhancing security systems. In signature verification, it authenticates the legitimacy of signatures, crucial for banking and legal industries. One-shot learning leverages the approach to classify new instances based on a single example, useful in object recognition. It facilitates image similarity and retrieval, aiding search engines and e-commerce platforms. Textual similarity measures help in plagiarism detection, document clustering, and semantic search. Audio verification and matching are used in voice recognition and music recommendation services. In medical diagnosis, it compares medical images or patient records to detect diseases. Robotics and autonomous systems utilize it to integrate sensory data for better decision-making. Handwriting recognition systems apply it to digitize handwritten documents, while e-commerce platforms use it for product matching, identifying duplicate listings and finding similar products. The Siamese approach's capability to measure and learn similarities makes it highly adaptable for comparison, verification, and matching tasks across different data types.

3.1.5 Approach Overview

In my approach, I've outlined the use of a Multimodal Model, which leverages the Siamese approach to handle multiple inputs with different modalities and formats. Each modality is processed separately with a distinct backbone, or encoder, tailored to its specific data type. These encoders transform the data into 1D vectors, which are then concatenated. The concatenated vector undergoes classification using a simple fully connected neural network. To recap, this approach allows for the integration of various data sources, such as Landsat cubes, Bioclimatic cubes, and Sentinel Image Patches, each with its unique shape and characteristics. For instance, Landsat cubes are structured by bands, quarters, and years, while Bioclimatic cubes follow a raster-type, year, and month format. Sentinel Image Patches, on the other hand, are defined by red, green, blue, and near-infrared bands. By utilizing the Siamese approach, my proposed model can effectively process and extract meaningful information from these diverse data sources, contributing to the success of my project.

The Multimodal Ensemble Model is designed to process inputs from multiple modalities, including tabular data, Landsat cubes, Bioclimatic cubes, and Sentinel Image Patches. This approach adopts the Siamese architecture, where each modality is processed by a different backbone (encoder), tailored to its specific format and characteristics.

By integrating information from multiple modalities, the model enhances the understanding and prediction of target classes, making it suitable for a wide range of applications, from remote sensing to environmental monitoring and beyond. The flow diagram outlines the processing steps of the Multimodal Ensemble Model. It begins with the normalization and feature extraction of tabular data through a fully connected neural network. Simultaneously, Landsat and Bioclimatic cubes undergo normalization and are processed by modified ResNet18 models, adapted to accommodate their specific structures. Sentinel Image Patches are processed by a modified Swin Transformer. Extracted features from all modalities are concatenated into a unified representation, facilitating the fusion of diverse information sources. Finally, the concatenated features are fed through fully connected layers for further processing and classification, resulting in the prediction of target classes. This

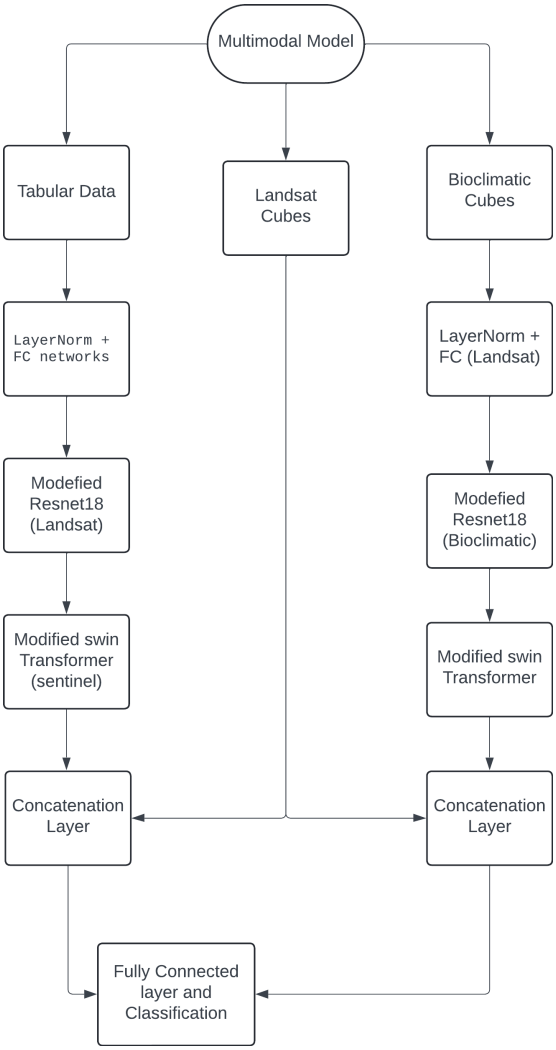


Figure 3.2: The flow of data through the Multimodal Model, including processing steps for each modality and the final classification stage

integrated approach leverages the strengths of each modality to enhance the model's performance in handling multiple data types effectively.

3.2 Algorithm for Multimodal Siamese Network for Species Distribution Prediction

3.2.1 Data Preprocessing

1. Load Landsat Cubes, Bioclimatic Cubes, Sentinel Image Patches.

2. Normalize each dataset.

3.2.2 Model Definition

1. Initialize `MultimodalSiameseNetwork` model with:
    (a) Normalization layers for each modality.
    (b) Modified ResNet18 for Landsat Cubes.
    (c) Modified ResNet18 for Bioclimatic Cubes.
    (d) Modified Swin-v2-t for Sentinel Image Patches.
    (e) Fully connected layers for classification.

3.2.3 Model Initialization

1. Define optimizer (AdamW).
2. Define loss function (BCEWithLogitsLoss).

3.2.4 Training Phase

1. For each epoch:
    (a) Training loop:
        i. For each batch of training data:
            - Apply mixup augmentation.
            - Forward pass through the model.
            - Compute loss.
            - Backward pass and update weights.
    (b) Validation loop (every few epochs):
        i. For each batch of validation data:
            - Forward pass through the model.
            - Compute validation loss.
        ii. Track best model based on validation loss.

3.2.5 Evaluation Phase

1. Load best model.
2. For each batch of test data:
   (a) Forward pass through the model.
   (b) Compute predictions.
3. Post-process predictions:
   (a) Sort predictions and select top-k.
   (b) Compute F1 score.

3.2.6 Result Analysis

1. Plot training and validation loss curves.
2. Plot F1 score vs. top-k.

Chapter 4

Work Done

The following sections describes the work done carried out during my internship period and all the necessary technologies used in this project along with the source code (mention in the appendix) of this project

4.1 Dataset and Evaluation Protocol

Standardized biodiversity observation data, such as presence-absence surveys conducted in small plots, have limitations in coverage due to their restricted spatial extent and the high cost associated with their renewal. To supplement this, new biodiversity monitoring initiatives, such as crowdsourcing programs (e.g., Pl@ntNet, iNaturalist, Observation.org), serve as valuable complementary data sources by offering millions of presence-only (PO) species records annually, each precisely geolocated. However, PO records alone cannot indicate the absence of unobserved species in local ecosystems, depict only a fraction of species communities in under-sampled regions, exhibit biases toward certain species, and consequently introduce biases into species distribution models. This issue was underscored by the preceding GeoLifeCLEF campaign, emphasizing the necessity for an evaluation approach based on comprehensive sampling of species communities at high spatial resolutions. Additionally, incorporating a small subset of standardized species observations, such as presence-absence (PA) plots, can mitigate many sampling biases inherent in PO data when integrated into the calibration of species distribution models, thereby leveraging the wealth of information within the extensive PO dataset. Nevertheless, even with

comprehensive PA data, modeling and mapping biological groups with vast taxonomic diversity, such as plants with over ten thousand species in Europe, remain challenging, given the prevalence of a few common species and a multitude of rare ones. This challenge is known as strong class imbalance in the field of machine learning.

The environmental data is invaluable for enriching the broader environmental context at larger spatial scales, defined by climatic, soil, or land cover attributes. However, the varying spatial resolutions pose challenges when integrating it into traditional deep learning frameworks.

The species observations and environmental predictors utilized for model training, along with the evaluation protocol involving standardized presence-absence (PA) data and evaluation metrics, were carefully selected. The training species observation data consisted of over 5 million plant species presence-only (PO) records, including species name, geo-location, and time, as well as approximately 5.9 thousand presence-absence (PA) surveys collected between 2017 and 2021, with a geo-location uncertainty of less than 100 meters. The PO data, sourced from the Global Biodiversity Information Facility (GBIF), integrated 13 trusted datasets, encompassing international citizen science programs such as Pl@ntNet, iNaturalist RG, Observation.org, and regional datasets, ensuring extensive spatial coverage across Europe (38 countries). Each PA survey entailed a comprehensive inventory of all plant species within small plots ranging from 10 to 400 square meters. The PA data originated from four source datasets covering France and Great Britain, including the ”Données de l’inventaire forestier national de l’IGN,” the ”National Plant Monitoring Scheme (Great Britain),” the ”Conservatoire Botanique National Mediterranéen,” and the ”Conservatoire Botanique National Alpin.” Together, the PO and PA data encompassed 10,038 species, representing the majority of the European flora. While acknowledging that the PO data only partially represented local species composition and was susceptible to various sampling biases, it was utilized to inform model outputs. Notably, a single PO record for one species does not imply the absence of other species, as observers may have overlooked or omitted certain species due to detection challenges, identification difficulties, or lack of interest. The PA surveys were included to mitigate sampling biases during model calibration.

GeoLifeCLEF sought to create and assess models capable of forecasting plant

species composition with high spatial resolution (approximately 10 meters) using various environmental predictors. These models were calibrated based on two distinct types of species observations: opportunistic presence-only records and standardized presence-absence surveys.

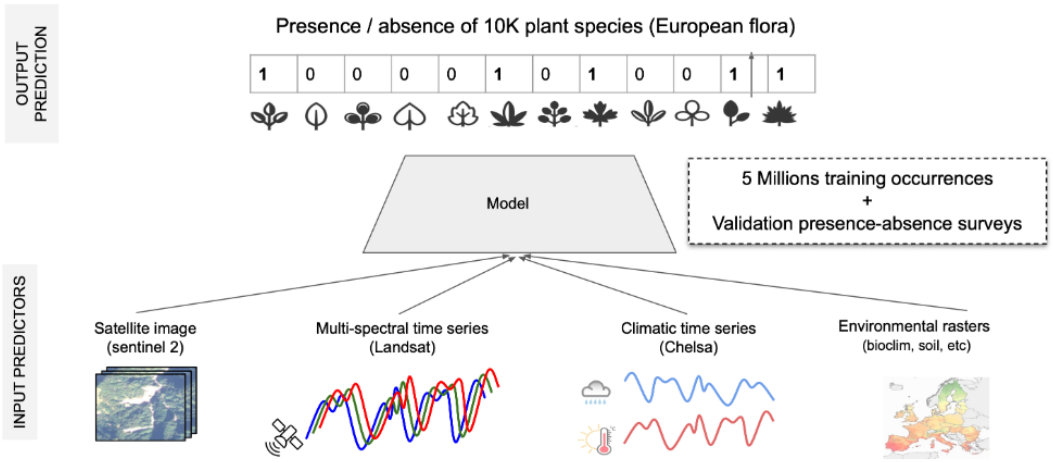


Figure 4.1: Developing and assessing models for predicting the composition of species

Data Descriptions

The obtained dataset used in this project is mainly divided into 2 category Species Observation data and Environmental data.

Species Observation data

The species related training data comprises Presence-Absence (PA) surveys and Presence-Only (PO) occurrences.

- **(PA) surveys:** Presence-absence (PA) surveys consist of approximately 90 thousand surveys covering roughly 10,000 species of the European flora. This presence-absence data (PA) is utilized to address false-absences in presence-only (PO) data and to calibrate models, thereby mitigating associated biases.
- **(PO) occurrences:** Presence-only (PO) occurrences encompass approximately five million observations sourced from multiple datasets acquired from the

Global Biodiversity Information Facility (GBIF, www.gbif.org). This dataset constitutes the predominant portion of the training data and spans all countries within our study area. However, its collection involved opportunistic sampling, lacking standardized protocols, resulting in various sampling biases. Notably, the absence of a species in local PO data does not necessarily indicate its true absence. Observers may have refrained from reporting it due to difficulties in detection during certain times of the year, misidentification as a non-target species, or lack of interest.

Environmental data

Furnished spatialized geographic and environmental data for utilization as predictors, serving as inputs for model predictions. For each species observation (in both PO and PA datasets), a four-band 128x128 satellite image at a resolution of 10 meters was supplied around the occurrence location, along with quarterly time series data spanning over 20 years for six spectral bands at that location. Additionally, a range of environmental raster datasets at the European scale, encompassing climatic, soil, land cover, human footprint, and elevation variables, was made available. Furthermore, monthly rasters of four climatic variables were provided, enabling the extraction of time series data for any observation.

Train-test split

A spatial block hold-out technique was applied to a grid with a width and height of 50 kilometers to partition the PA surveys. This process resulted in a validation set comprising 5,948 surveys (approximately 20% of the total), which was designated for model training, and a test set containing 22,404 surveys (approximately 80% of the total).

Evaluation metric

The technique used in this project was proposed as a multi-label classification task.The main evaluation metric for the project is the micro F1-score computed on the PA test set. The F1-score is a measure of overlap between the predicted and actual set of

species present, averaged over the test PA surveys. Each survey i is associated with a set of ground-truth labels Y_i, i.e., the set of plant species present at i. For each survey,provided a list of predicted labels $\hat{Y}_{i,1}, \hat{Y}_{i,2}, ..., \hat{Y}_{i,R_i}$.

$$F1 = \frac{1}{N} \sum_{i=1}^{N} \frac{TP_i}{TP_i + \frac{(FP_i + FN_i)}{2}}$$

Where:

$$\begin{cases} TP_i = \text{Number of predicted labels truly present, i.e.} |\hat{Y}_i \cap Y_i| \\ FP_i = \text{Number of labels predicted but absent, i.e.} |\hat{Y}_i \setminus Y_i| \\ FN_i = \text{Number of labels not predicted but present, i.e.} |Y_i \setminus \hat{Y}_i| \end{cases}$$

Satellite image patches

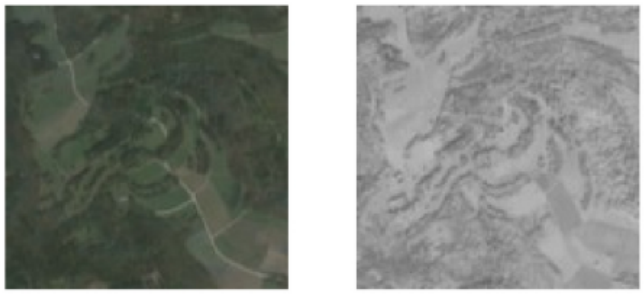


Figure 4.2: The 1280x1280m satellite image patches sampled in 2021 around the observation.

Satellite time series

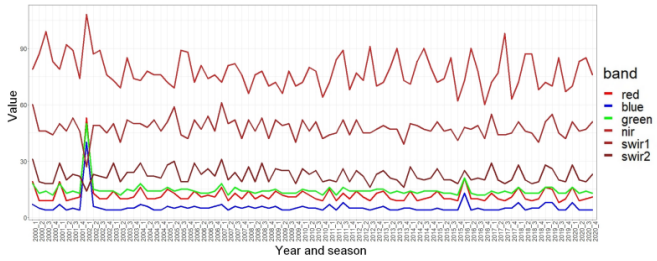


Figure 4.3: Quarterly time series of six satellite bands at the point location since winter 1999-2000

Environmental rasters

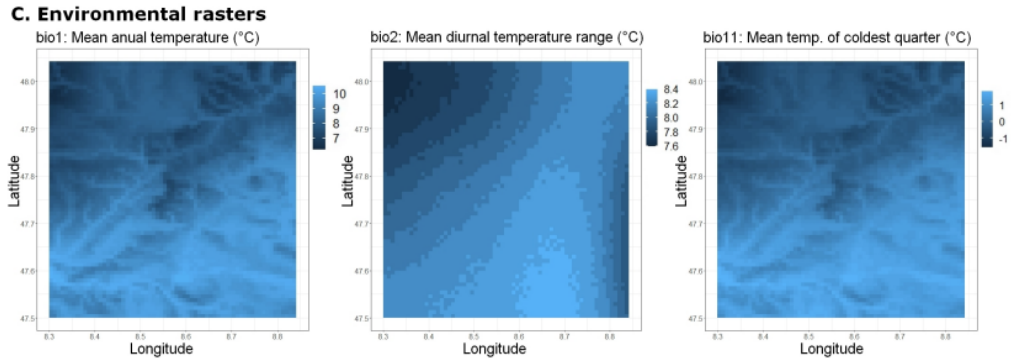


Figure 4.4: Three example bioclimatic images (65x65km) around the observation, extracted from the provided environmental rasters.

Environmental predictors

Name	Description	Source	Resolution
Climate	19 rasters of historical bioclimatic variables (1981-2010) traditionally used in SDMs	CHELSA	~1 km
Montly Climate	4 variables from January 2000 to december 2019	CHELSA	~1 km
Soil	9 pedological rasters	Soilgrids	~ 1 km
Elevation	Elevation above sea level	ASTER	~ 30 m
Land cover	According to IGBP classification (17 classes)	MODIS 500 m	~ 500 m
Human footprint	7 pressures on the environment for 1993 and 2009	Venter et al., 2016	~ 1km
Satellite imagery	RGB and NIR patches centered on each observation and taken the same year	Sentinel-2	10 m
Satellite time series	Time series of six quarterly satellite bands values since winter 1999	Landsat	30 m

Figure 4.5: Summary of the environmental predictors associated with the species observations, source and spatial resolution.

4.2 System Requirements

The following are the system requirements used for the model training and validation and same is required for running the software (recommended):

- **Operating System:** Windows 10, macOS Big Sur, or Ubuntu 20.04
- **Processor:** Intel Core i5 or equivalent
- **Memory (RAM):** 15 GB (minimum)
- **Disk Space:** 50 GB available space (minimum)
- **Graphics:** NVIDIA GeForce GTX 1050 or NVIDIA Tesla P100 or AMD Radeon RX 560 or equivalent local or cloud gpu's
- **Internet Connection:** Required for resources downloading, initial installation and updates

Note: These requirements are subject to change as the software evolves. It is recommended to check for the latest system requirements on the official documentation.

4.3 Tools and technologies Used

Apart from the basic machine learning and deep learning frameworks the project uses a custom framework designed by Plantnet (a services provided plant identification API & apps as well as research tools for everyone to use) The involves the use of various tools and technologies to analyze and extract insights from data as well as development and fine tuining the model. Below are some commonly used tools in the field:

Programming Language:

Python: Widely used for data analysis, machine learning, deep learning, and statistical modeling.

Deep learning Framework:

PyTorch: PyTorch is a machine learning library based on the Torch library, used for applications such as computer vision and natural language processing, originally developed by Meta AI

Visualization tool:

Matplotlib: A Python library for creating static, animated, and interactive visualizations.

Machine Learning:

Scikit-learn: A comprehensive library for classical machine learning algorithms in Python.

GPU Acceleration:

CUDA: A parallel computing platform and application programming interface model created by Nvidia. It allows software developers to use a CUDA-enabled graphics processing unit (GPU) for general-purpose processing. Tesla P100 or nvidia rtx 1050

Integrated Development Environments (IDE):

Jupyter: An open-source web application that allows you to create and share documents containing live code, equations, visualizations, and narrative text.

Version Control:

Git and Github: Software and a platform that allows developers to create, store, manage and share their code.

These tools and technologies collectively empower data scientists to explore, analyze, and derive valuable insights from diverse datasets.

4.4 Objective

The day-to-day tasks of the internship include both minor and major objectives such as:

- Understanding the data and deriving meaningful insights from it.
- Conducting initial development along with research experiments.
- Fine-tuning the developed model to maximize accuracy, minimize loss, or optimize the process.

- Implementing feedback from supervisors and peers to improve the model and processes.

4.4.1 Objective 1 : Comprehending the Dataset

The dataset used in this project is primarily divided into two categories: Species Observation Data and Environmental Data. Detailed descriptions can be found in the ”Data Description” subsection under the ”Dataset and Evaluation Protocol” section, as well as in Appendix A. Understanding the dataset is critical for effectively training and fine-tuning the model.

The entire dataset exceeds 90 GB of disk space, making it challenging to handle and process all at once. Therefore, a subset of the data is selected to create a customized, manageable dataset. This approach not only facilitates easier handling but also ensures that the most relevant data is used for training the model.

Moreover, comprehending the intricacies of the dataset allows for better preprocessing, which is crucial for optimizing model performance. By carefully selecting and preparing the data, we can focus on improving accuracy and reducing computational overhead, ultimately leading to more efficient and effective model development.

4.4.2 Objective 2 : Conducting Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a critical step in understanding the structure, patterns, and relationships within the dataset. It involves summarizing the main characteristics of the data, often using visual methods, to uncover insights that can guide subsequent stages of data preprocessing and model development.

Metadata Analysis

Test and train metadata file contains the geographic features such as latitude ,longitude, country and region etc. along with the surveyId Plotting the geo-locations provides valuable insights into the distribution of species across specific regions or countries. I merged the training and test metadata samples to obtain a comprehensive dataset of latitude and longitude information.

	lon	lat	year	geoUncertaintyInM	areaInM2	region	country	surveyId
0	10.033550	57.12081	2019	10.0	707.0	CONTINENTAL	Denmark	642
1	7.333000	46.22997	2019	NaN	10.0	ALPINE	Switzerland	1792
2	1.843658	42.58006	2018	0.0	-inf	ALPINE	France	3256
3	11.720090	46.26149	2021	10.0	35.0	ALPINE	Italy	3855
4	9.361870	55.90245	2017	10.0	79.0	CONTINENTAL	Denmark	4889

Figure 4.6: test metadata sample

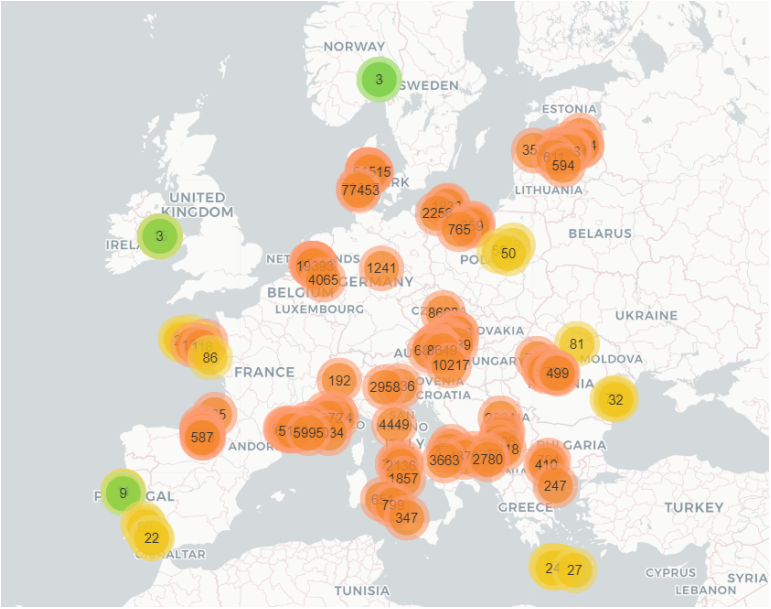


Figure 4.7: A map object centered at the mean location and FastMarkerCluster to the map

Elevation

Combining the elevation data from environmental rasters with the latitude and longitude coordinates to obtain a comprehensive 3D interpretation of the geographical locations with respect to species distributions.

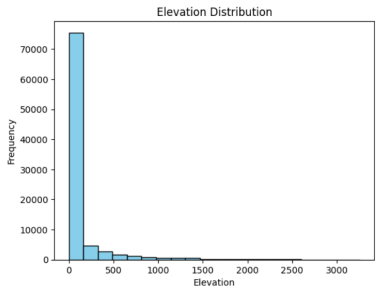


Figure 4.8: Elevation distribution

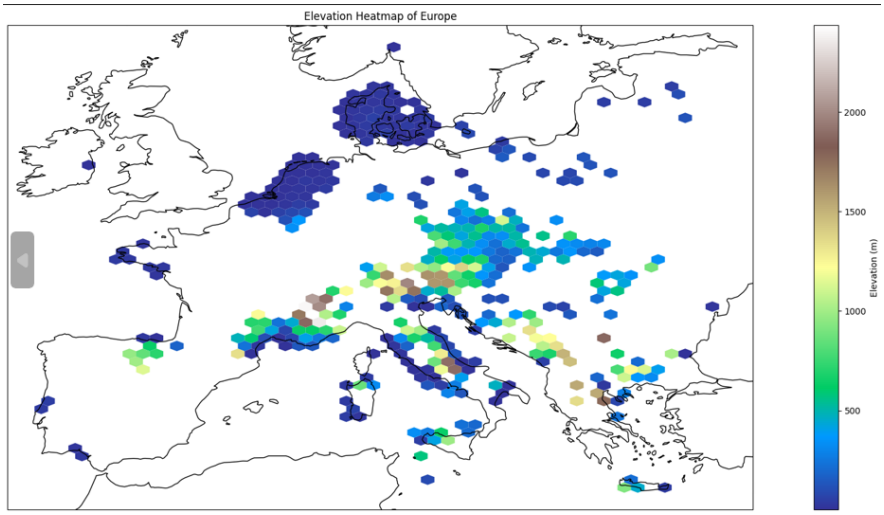


Figure 4.9: Elevation heatmap

Landcover Soil-grid Footprints

Now let's combine other environmental rasters data like land cover , soil grids ,human footprints from environmental rasters with the latitude and longitude coordinates with respect to species distributions

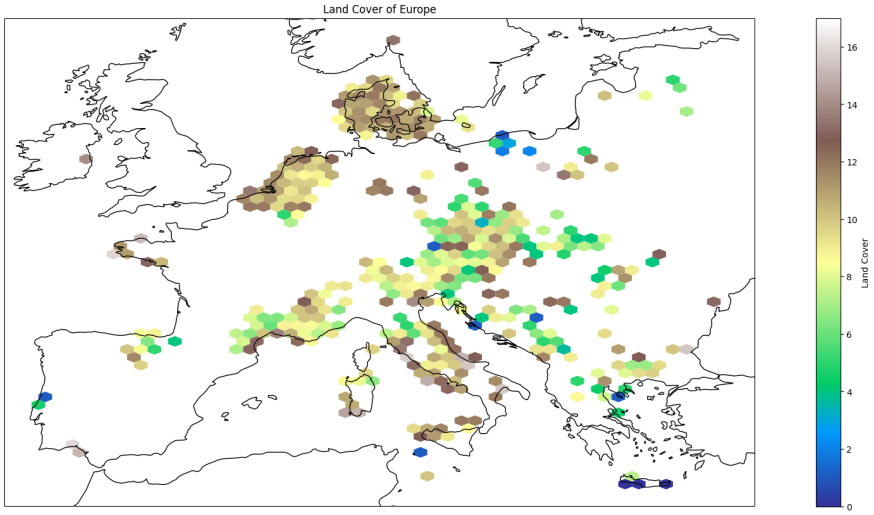


Figure 4.10: Landcover heatmap

Montly climatic rasters and bioclimatic rasters

Four climatic variables (mean, minimum, and maximum temperature, and total precipitation) were computed monthly from January 2000 to December 2019, resulting in 960 low-resolution rasters covering Europe.

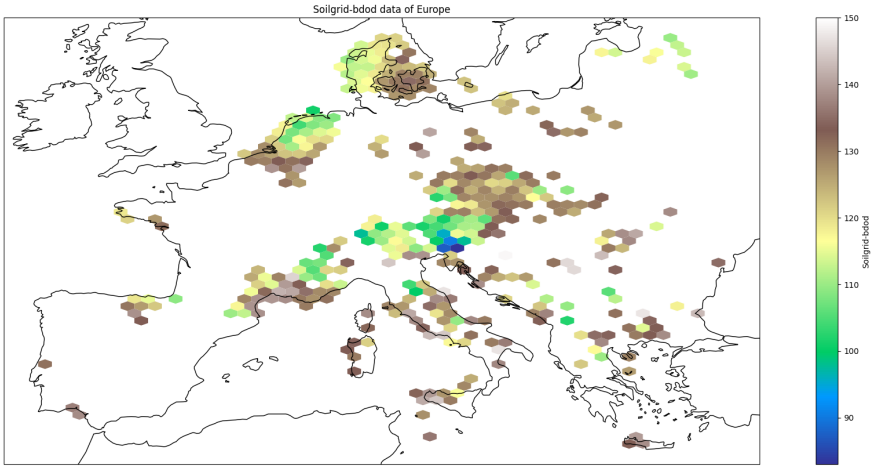


Figure 4.11: Soil-grid heatmap

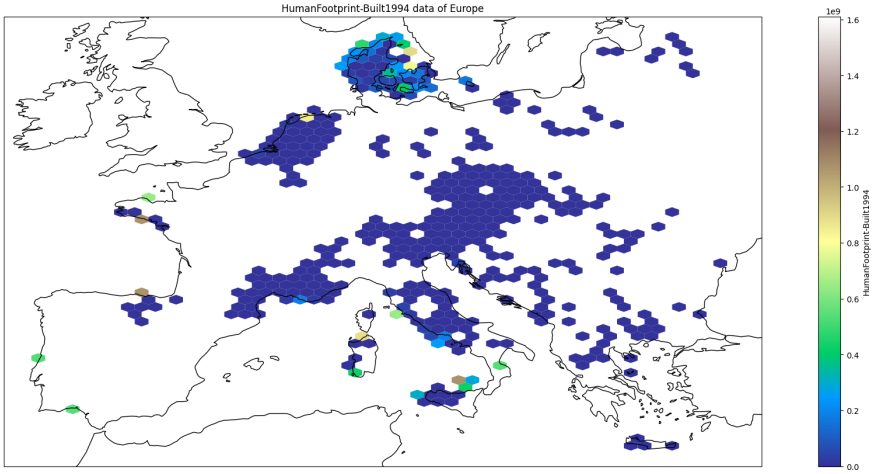


Figure 4.12: Human footprint heatmap

Nineteen low-resolution rasters covering Europe, commonly used in species distribution modeling, were provided in longitude/latitude coordinates (WGS84) see appendix A

Together, these datasets provide a rich source of environmental data that can be used to inform and enhance species distribution models. The historical climatic data helps in understanding past trends and patterns, while the additional rasters contribute to a more nuanced and detailed analysis of species distribution across Europe.

Train satellite patches RGB

The directory structure comprises several layers, starting with the root directory named ”PA_Train_SatellitePatches_RGB.” Within this directory, there are multiple subdirec-

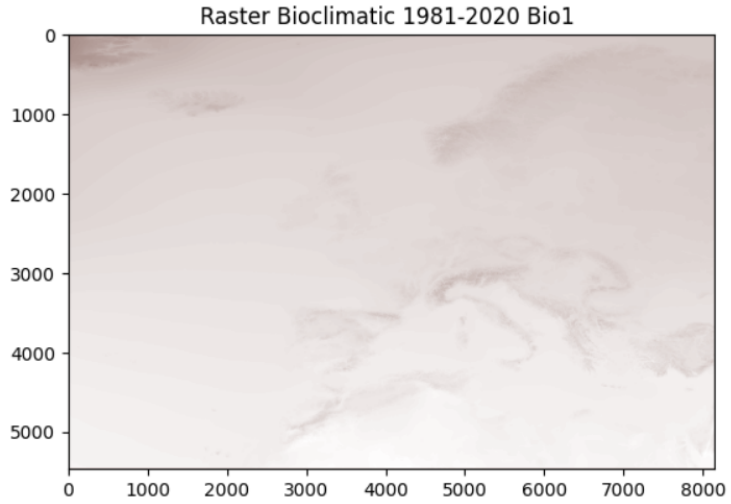


Figure 4.13: Bioclimatic raster 1

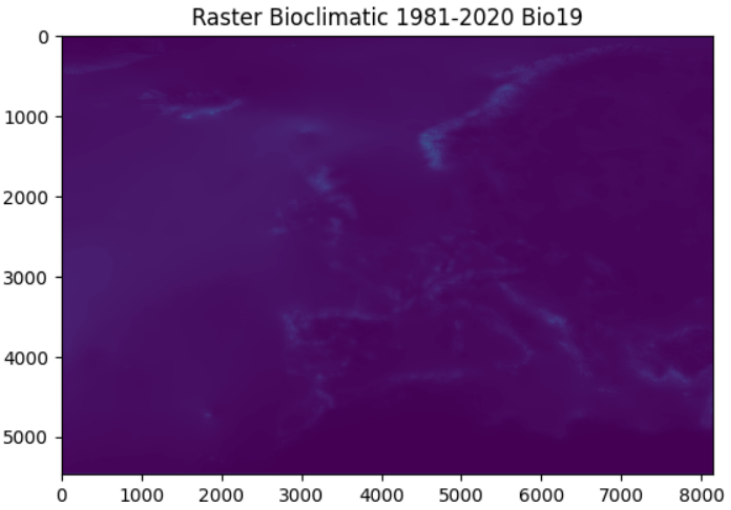


Figure 4.14: Bioclimatic raster 2

tories, including ”pa_train_patches_rgb” and several numbered directories, such as ”00.” The ”pa_train_patches_rgb” directory contains 100 subdirectories, presumably representing different categories or classes of data. Each of these subdirectories, labeled numerically from ”00” to ”99,” likely contains specific subsets of data related to the respective category. Furthermore, within each of these numerical subdirectories, there are additional subdirectories labeled ”04.” These ”04” subdirectories contain a total of 16 files each, which may represent individual images or data samples utilized for training or analysis purposes. Overall, this hierarchical directory structure facilitates the organization and access of satellite patches in RGB format for training and potentially other applications.

Figure 4.15: NIR

European Flora

Let's deep dive into europen flora by regions.

	lon	lat	year	geoUncertaintyInM	areaInM2	region	country	speciesId	surveyId
696020	5.762573	52.101347	2019	5.0	3.75	ATLANTIC	Netherlands	6903.0	1852742
1026966	2.493380	45.770600	2017	1.0	NaN	CONTINENTAL	France	6331.0	2730440
1451366	9.779140	57.350200	2017	10.0	79.00	CONTINENTAL	Denmark	10073.0	3837748

Figure 4.16: Train metadata

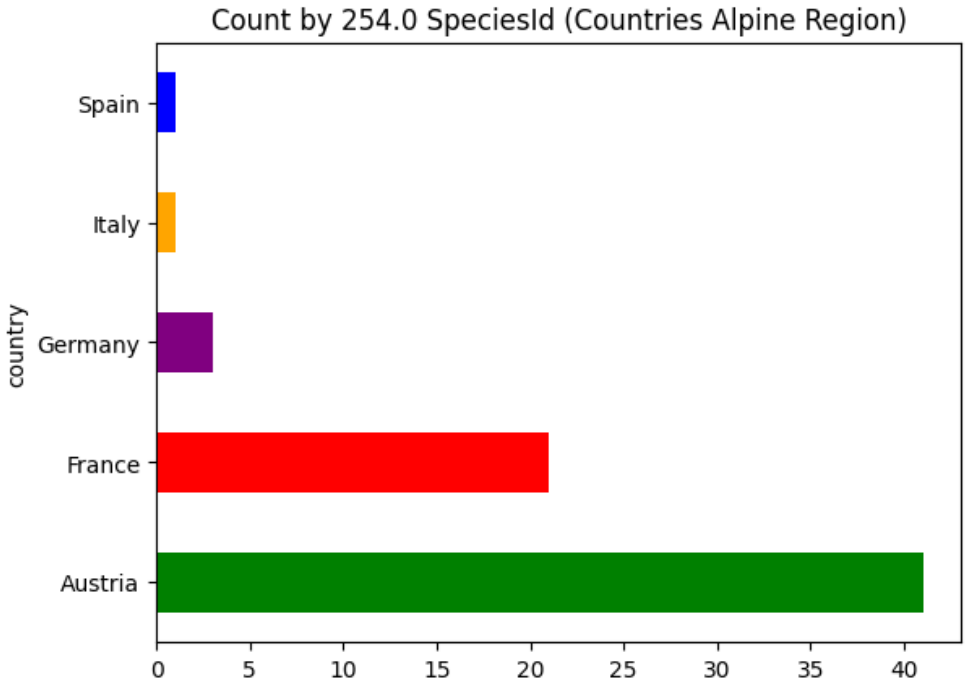


Figure 4.17: Species count in Alpine region

Countries have Alpine region (species 254.0)

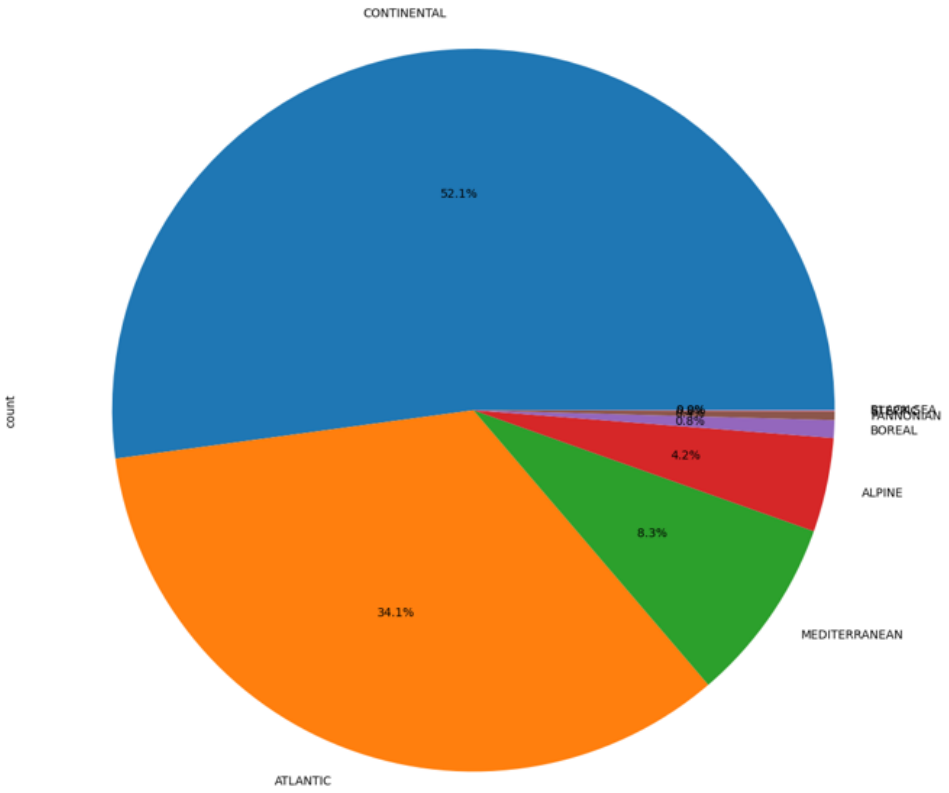


Figure 4.18: Species distributions in all regions

Insights and Observations

The EDA phase culminates in the extraction of insights and observations. These findings inform subsequent steps in the project, guiding feature engineering, model selection, and the overall approach to feature importances.

Exploratory Data Analysis is a critically important component of the project, offering a comprehensive understanding of the dataset and laying the groundwork for subsequent modeling and analysis.

In the next section, the process of extracting important features by combining all the necessary rasters and NIR (Near-Infrared) data using appropriate techniques is described.

During the exploratory data analysis (EDA), several key observations emerged. Firstly, the distributions of numerical variables such as temperature and precipitation appeared (climatic rasters) to follow approximate normal distributions, albeit with some regional and temporal variability. Additionally, outliers were identified in certain

Figure 4.19: Distributions across countries

environmental variables, suggesting potential anomalies or extreme climatic events during specific periods. Moreover, strong correlations were observed between certain pairs of variables, such as temperature and precipitation, indicating interdependencies within the dataset. Temporal analysis revealed distinct seasonal patterns and long-term fluctuations in climatic variables over the study period. Geospatial visualization further elucidated spatial patterns, revealing unique climatic regions and gradients across Europe. However, some data quality issues, such as missing values or inconsistencies, were also detected, underscoring the need for further investigation and preprocessing before model development. These observations collectively provide valuable insights into the characteristics and patterns inherent in the environmental data, thereby guiding subsequent modeling efforts and data-driven decision-making processes.

4.4.3 Objective 3 : Preprocessing Data

The preprocessing of data in this project involved meticulous steps aimed at ensuring data accuracy, completeness, and suitability for subsequent analysis. Initially, missing values within the dataset were meticulously handled using appropriate techniques such

as imputation or removal to avoid any distortions in the analysis. Subsequently, diverse environmental datasets spanning bioclimatic variables, elevation, human footprint, land cover, soil grids, and Landsat time series data were merged based on a common identifier, 'surveyId'. This consolidation facilitated the creation of a comprehensive dataset containing all necessary environmental variables essential for subsequent analysis. Additionally, the climatic timeseries data within a specified time window were merged with the main dataset to provide temporal context to the environmental features. Following the data merging process, an XGBoost classifier model was trained using the prepared dataset to predict the presence or absence of target species based on the environmental features. Furthermore, model evaluation was conducted using a validation set to assess its predictive accuracy and generalization capability.

The initial phase of data preprocessing involved several essential steps:

Handling Missing Values:

Before proceeding with any analysis, missing values within the dataset were addressed using appropriate techniques such as imputation or removal, ensuring data completeness and accuracy.

Data Merging:

Several environmental datasets, including bioclimatic, elevation, human footprint, land cover, soil grids, and Landsat time series data, were merged based on a common identifier, 'surveyId'. This consolidation facilitated the creation of a comprehensive dataset containing all necessary environmental variables for subsequent analysis.

Feature Engineering:

Merging Climatic Timeseries Data: Monthly climatic timeseries data within a 10-year time window (40 time points) was merged with the main dataset. This involved shifting and slicing the original data to create a window dataset. The resulting dataset was then merged with the main dataset based on the 'surveyId' key.

XGBoost Model Training: An XGBoost classifier model was trained using the prepared dataset. The model was configured to predict the presence or absence of

a target species (identified by 'speciesId') based on the environmental features.

Prediction and Mapping

Utilizing the trained XGBoost model, predictions were made on a test dataset to determine the probability of species presence at various geographic locations.The predicted probabilities were spatially visualized by mapping them onto geographic coordinates. This process involved creating point vector data, with each data point representing a specific geographic location where species presence was predicted.

Feature Importance Analysis

Feature importance analysis was conducted using SHAP (SHapley Additive exPlanations) values, employing the following approach:

SHAP Values Calculation: SHAP values were computed for each feature in the dataset to quantify their importance in influencing model predictions. These values provided insights into the relative contribution of each environmental variable to the model's predictive outcomes.

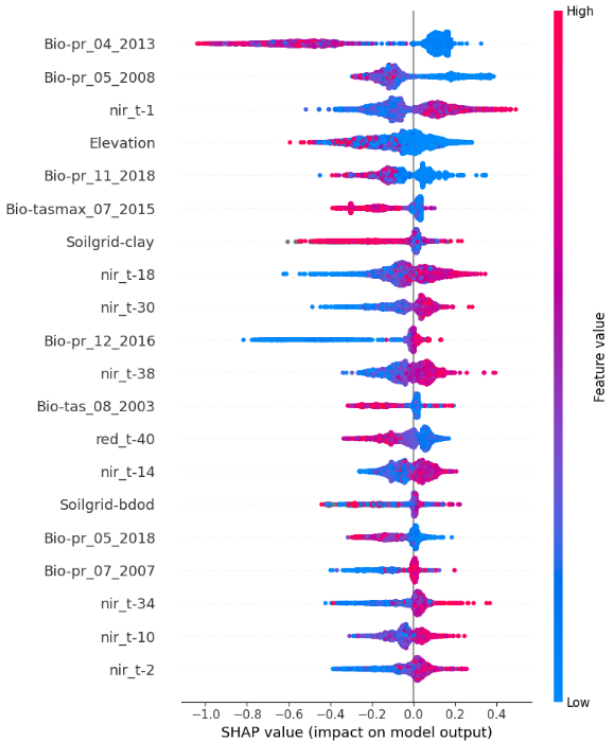


Figure 4.20: SHAP value

Feature Importance Visualization: SHAP summary plots were generated to visually represent the impact of each feature on model predictions. These plots facilitated the identification of key environmental factors driving species distribution patterns.

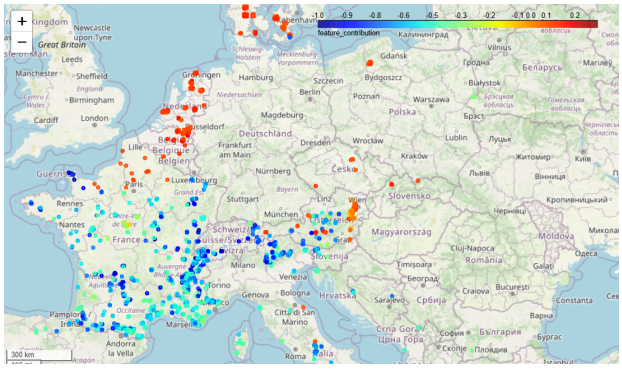


Figure 4.21: Elevation feature importance

Interactive Feature Importance Maps: Feature importance maps were created to visualize the contribution of individual features to model predictions spatially. These interactive maps allowed for a detailed exploration of how specific environmental variables influenced species distribution patterns across different geographic locations.

4.4.4 Objective 4 : Baseline Experiments

There are some baseline experiments conducted during the research work carried out under the supervison of my mentor.The supervision ensured that the experiments were conducted effectively, and the results were interpreted accurately.

Baseline Experiment with Bioclimatic Cubes

Overview:

The baseline experiment with bioclimatic cubes employs a modified ResNet18 architecture combined with Binary Cross Entropy loss function to predict species composition from climatic data. Despite its simplicity, this baseline model yields competitive results on the metric, showcasing the potential of utilizing bioclimatic data in species distribution modeling.

Methodology:

The experiment utilizes bioclimatic cubes, which provide detailed climatic information for specific locations. The modified ResNet18 architecture is employed as the

backbone neural network for feature extraction and classification. Binary Cross Entropy loss function is utilized as the optimization criterion to train the model. By learning the intricate relationship between the climatic history of a location and its species composition, the model aims to accurately predict species presence or absence based on environmental variables.

Conclusion:

The baseline experiment with bioclimatic cubes lays a solid foundation for further exploration and experimentation in species distribution modeling. By leveraging climatic data, the model demonstrates the potential to accurately predict species composition, highlighting the importance of environmental variables in ecological modeling tasks.

Baseline Experiment with LandSat Cubes

Overview:

The baseline experiment with Landsat cubes utilizes a modified ResNet18 architecture combined with Binary Cross Entropy loss function to predict species composition based on satellite imagery data. Despite its simplicity, this baseline model achieves competitive performance on the defined metric, showcasing the potential of leveraging Landsat data in species distribution modeling.

Methodology:

In this experiment, Landsat cubes, consisting of spectral bands including Red (R), Green (G), Blue (B), Near-Infrared (NIR), Short-Wave Infrared 1 (SWIR1), and Short-Wave Infrared 2 (SWIR2), are utilized as input data. The modified ResNet18 architecture is employed as the neural network backbone for feature extraction and classification. Binary Cross Entropy loss function is used as the optimization criterion to train the model. By learning the relationship between spectral values at specific locations and species composition, the model aims to accurately predict the presence or absence of species.

Conclusion:

The baseline experiment with Landsat cubes provides valuable insights into the predictive capabilities of satellite imagery data in species distribution modeling. By

leveraging spectral information from Landsat imagery, the model demonstrates the potential to accurately predict species composition, highlighting the importance of remote sensing data in ecological research. Moving forward, continued experimentation and refinement of modeling techniques will contribute to advancing our understanding of species-environment relationships.

Baseline Experiment with Sentinel Images

Overview:

The baseline experiment with Sentinel Image Patches utilizes a modified Swin-v2-t architecture combined with Binary Cross Entropy loss function to predict species composition based on satellite image data. This baseline model offers promising results and ranks highly on the leaderboard, showcasing the potential of leveraging image data for species distribution modeling.

Methodology:

In this experiment, Sentinel Image Patches, which provide image-like representations of habitats and localities, are utilized as input data. The modified Swin-v2-t architecture is employed as the neural network backbone for feature extraction and classification. Binary Cross Entropy loss function is used as the optimization criterion to train the model. By analyzing image features captured by Sentinel satellites, the model aims to accurately predict the presence or absence of species based on habitat characteristics.

Conclusion:

The baseline experiment with Sentinel Image Patches highlights the importance of leveraging image data for species distribution modeling. By analyzing habitat characteristics captured by satellite imagery, the model demonstrates the potential to accurately predict species composition. Moving forward, continued experimentation and refinement of modeling techniques will contribute to advancing our understanding of species-environment relationships and supporting conservation efforts.

Baseline Experiment Combined Bioclim+LandSat+Sentinel

Overview:

In this experiment, we present a multimodal approach that combines Landsat and Bioclimatic Cubes with Sentinel images using a siamese network architecture. By leveraging multiple modalities of environmental data, including satellite imagery and climatic variables, this approach aims to capture comprehensive information about habitat characteristics and localities. The model architecture employs a siamese network with shared weights across different modalities, facilitating effective feature extraction and integration.

Methodology:

The multimodal approach utilizes Landsat and Bioclimatic Cubes as well as Sentinel images as input data. A siamese network architecture is employed, where each modality is processed separately by distinct encoders before being concatenated and passed through a shared decoder. Binary Cross Entropy loss function is used as the optimization criterion to train the model. By integrating information from multiple modalities, the model aims to capture diverse aspects of habitat characteristics and enhance prediction accuracy.

Conclusion:

The multimodal approach combining Landsat and Bioclimatic Cubes with Sentinel images offers a promising framework for species distribution modeling. By integrating information from diverse environmental data sources, the model demonstrates the potential to capture comprehensive habitat characteristics and improve prediction accuracy. Moving forward, continued refinement and experimentation with model architectures and techniques will contribute to advancing our understanding of species-environment relationships and supporting conservation efforts.

4.4.5 Objective 5 : Assessing Experiment Results

Baseline Experiment with Bioclimatic Cubes

Results:

The baseline experiment achieves a performance metric of [0.25784], demonstrating

the effectiveness of utilizing bioclimatic data in species distribution modeling.

Recommendations:

While the baseline experiment provides promising results, there is ample room for improvement. Experimentation with various techniques such as different neural network architectures, loss functions, and data augmentation strategies is encouraged to enhance the model's performance further. Additionally, incorporating additional environmental variables and fine-tuning hyperparameters could potentially lead to improved predictive accuracy.

Baseline Experiment with LandSat Cubes

Results:

The baseline experiment achieves a performance metric of [0.26424], indicating the effectiveness of utilizing Landsat data in species distribution modeling.

Recommendations:

While the baseline experiment yields promising results, there is ample scope for further improvement. Experimentation with various techniques such as data augmentation, transfer learning, and model architectures is encouraged to enhance the model's performance further. Additionally, exploring additional spectral bands or incorporating temporal data could provide valuable insights into species distribution dynamics.

Baseline Experiment with Sentinel Images

Results:

The baseline experiment achieves a performance metric of [0.23555], demonstrating the effectiveness of utilizing Sentinel Image Patches in species distribution modeling.

Recommendations:

While the baseline experiment yields promising results, there is room for further enhancement. Experimentation with advanced techniques such as data augmentation, fine-tuning model architectures, and incorporating additional contextual information could potentially improve the model's performance. Additionally, exploring ensemble methods or transfer learning approaches may provide valuable insights into optimizing model predictions.

Baseline Experiment Combined Bioclim+LandSat+Sentinel

Results:

The baseline experiment achieves a performance metric of [0.31626], demonstrating the effectiveness of the multimodal approach in species distribution modeling. By combining Landsat and Bioclimatic Cubes with Sentinel images, the model captures complementary information from different data sources, resulting in improved predictive performance compared to single-modality approaches.

Recommendations:

While the baseline experiment yields promising results, there is room for further improvement. Experimentation with advanced techniques such as attention mechanisms, feature fusion strategies, and ensemble learning may enhance the model's performance further. Additionally, exploring the incorporation of additional environmental variables or incorporating temporal information could provide valuable insights into species-environment relationships.

Baseline Experiments Summary

- **Baseline with Bioclimatic Cubes:**
  - Methodology: ResNet18 + Binary Cross Entropy
  - Performance Metric: [0.25784]
  - Insight: Utilizing climatic history data
- **Baseline with Landsat Cubes:**
  - Methodology: ResNet18 + Binary Cross Entropy
  - Performance Metric: [0.26424]
  - Insight: Relationship between location values and species
- **Baseline with Sentinel Image Patches:**
  - Methodology: Swin-v2-t + Binary Cross Entropy

- Performance Metric: [0.23555]
- Insight: Potential of satellite imagery for habitat characteristics

- **Baseline with Landsat and Bioclimatic Cubes + Sentinel images:**
  - Methodology: Siamese Network
  - Performance Metric: [0.31626]
  - Insight: Integrating multiple data sources for improved performance

These experiments demonstrate the potential of different data modalities and methodologies in predicting species composition. Further experimentation with various techniques, architectures, and losses is encouraged for enhanced performance in future scope.

4.4.6 Objective 6 : Transitioning to Modular Programming for Implementation

As the complexity of the project grows, transitioning to a modular programming approach becomes imperative for maintaining code readability, scalability, and reusability. This objective focuses on restructuring the existing codebase into modular components, encapsulating related functionality within individual modules or packages. By breaking down the implementation into smaller, cohesive units, we aim to enhance code organization and facilitate collaboration among team members. The transition to modular programming involves several key steps. Firstly, we identify common functionalities or tasks within the codebase and isolate them into standalone modules. These modules encapsulate specific tasks or functionalities, such as data preprocessing, model training, evaluation, and visualization. Each module should have well-defined inputs, outputs, and interfaces to promote modularity and ease of integration.

Furthermore, we leverage the principles of abstraction and encapsulation to hide the internal implementation details of each module, exposing only essential interfaces or APIs for interaction. This promotes loose coupling between modules, allowing changes to be made to one module without affecting others. Additionally, we strive to

adhere to the single responsibility principle, ensuring that each module is responsible for a single, well-defined task.

To facilitate seamless integration and testing, we employ standardized interfaces and protocols for communication between modules. This enables interoperability between different components of the system and promotes code reuse across multiple projects or teams. Moreover, we adopt a systematic approach to version control and dependency management, ensuring that changes to one module do not inadvertently break dependencies or introduce regressions in other parts of the system. Overall,

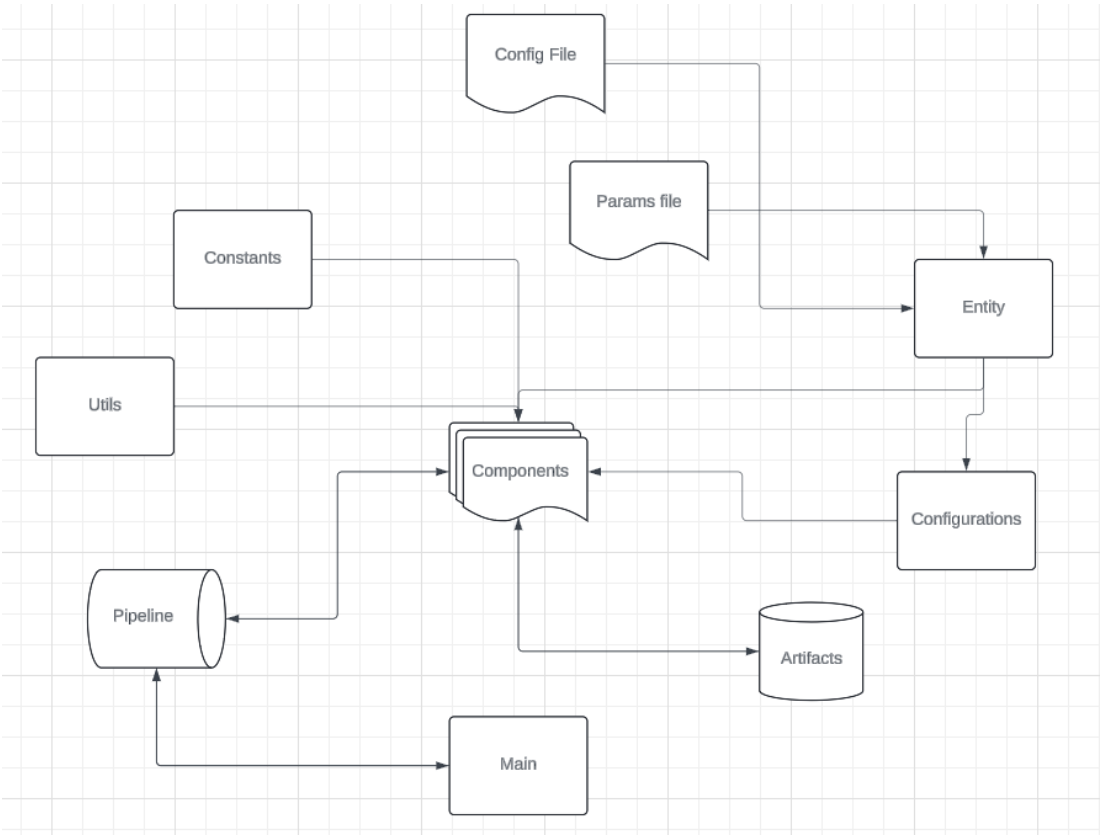


Figure 4.22: Modular approach design

transitioning to modular programming not only improves code maintainability and extensibility but also fosters collaboration and innovation within the development team. By embracing modular programming principles, we aim to streamline the development process and deliver robust, scalable solutions that meet the evolving requirements of the project.

4.5 Model building training and evaluation

4.5.1 Model building

To handle the diverse inputs consisting of tabular data, Landsat data, bioclimatic data, and Sentinel images, we employ a multimodal ensemble approach. Each data modality is processed using a specialized encoder, and their outputs are fused for classification.

Multimodal Model Design

The multimodal model integrates multiple data sources to enhance species classification accuracy. The components of the model include: Tabular Data Encoder,Processes tabular features through a feed-forward neural network, applying Layer Normalization for consistent scaling. Landsat Data Encoder,Utilizes a modified ResNet18 architecture to encode Landsat cubes, considering different spectral bands, time quarters, and years. Bioclimatic Data Encoder,Adopts another modified ResNet18 to handle bioclimatic data cubes, representing various raster types, years, and months. Sentinel Image Encoder,Employs a modified Swin Transformer (Swin-v2-t) to process Sentinel Image Patches, integrating RGB and NIR channels. Fusion Layer,Concatenates outputs from all encoders and passes them through fully connected layers to produce the final classification.

Model Initialization

The multimodal ensemble model is initialized, and the training device is set to CUDA if available to leverage GPU acceleration. (Note : Cuda device is necessary for this implementation otherwise training will stuck indefinitely)

4.5.2 Training the Model

The training process involves optimizing the model using the AdamW optimizer and Binary Cross Entropy (BCE) loss function. The key steps include data preprocessing, forward propagation, loss calculation, backpropagation, and optimization.

Hyperparameters

The primary hyperparameters for training are:

- Learning Rate: Set to 3e-4 to balance convergence speed and stability.
- Number of Epochs: 3 epochs for debugging and 15 epochs for full training.
- Optimizer: AdamW is chosen for its effective weight decay and optimization properties.
- Loss Function: BCEWithLogitsLoss is employed to handle the binary classification task.

Training Loop

The training loop iterates over epochs, processing data batches, applying Mixup augmentation to enhance generalization, computing loss, and updating model weights. Regular validation checks are performed to monitor model performance and prevent overfitting.

4.5.3 Model Evaluation

Evaluating the model's performance involves using various metrics and visualizations to ensure its effectiveness.

Loss Curves

Training and validation loss curves are plotted to monitor the model's learning progress over epochs. This helps in identifying overfitting or underfitting trends.

Performance Metrics

The model is evaluated using the F1-score, a harmonic mean of precision and recall, which is crucial for imbalanced datasets like ours. Additionally, F1-score versus top-K predictions is plotted to determine the optimal number of top predictions for best performance.

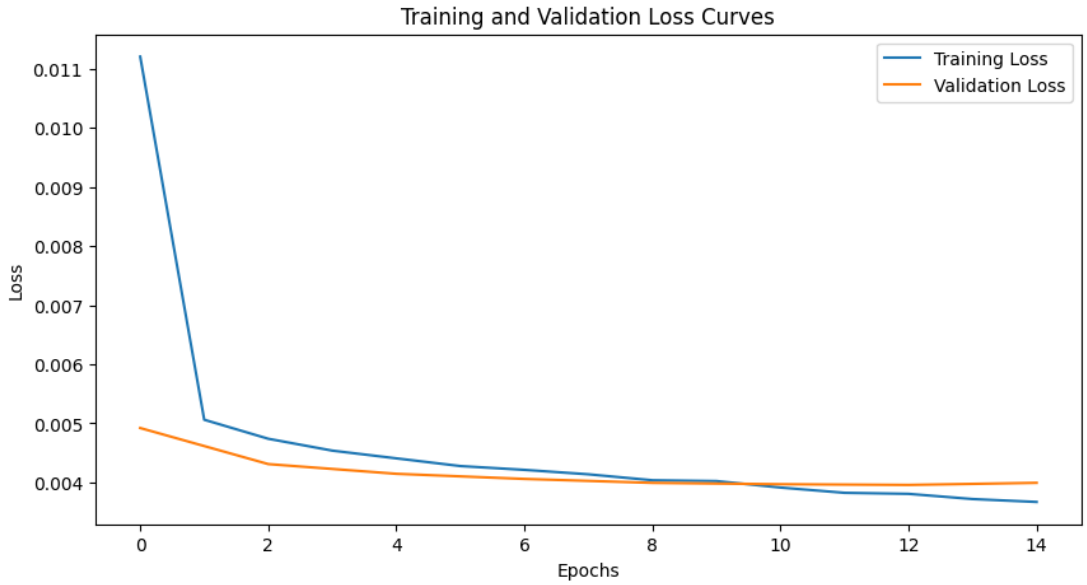


Figure 4.23: Training and vaidation loss

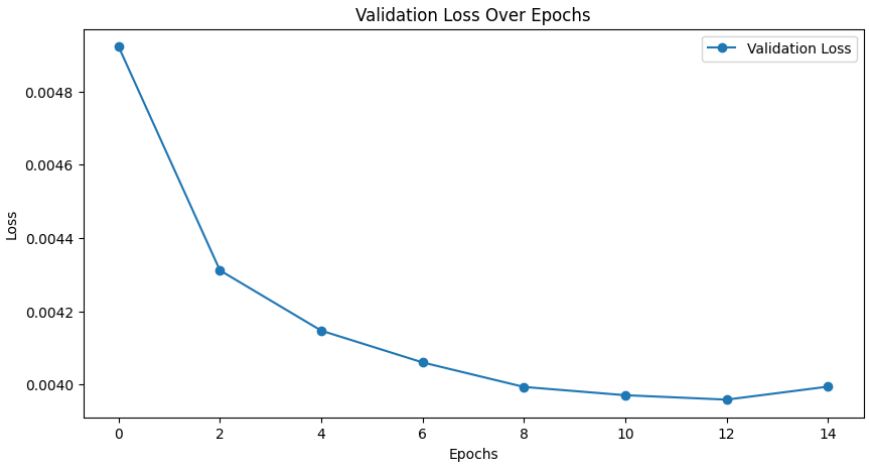


Figure 4.24: Validation loss

Visualization of Results

Visualizations include:

F1 Score vs. Top-K: This plot helps in understanding how the F1 score varies with the number of top predictions, aiding in fine-tuning the prediction threshold. Validation Loss Over Epochs: Shows how the validation loss changes over epochs, providing insights into the model's generalization capabilities.

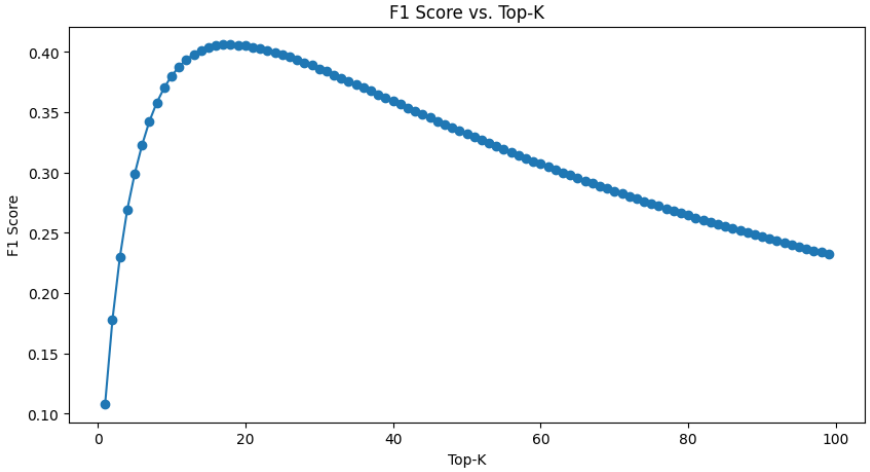


Figure 4.25: F1 score vs top k

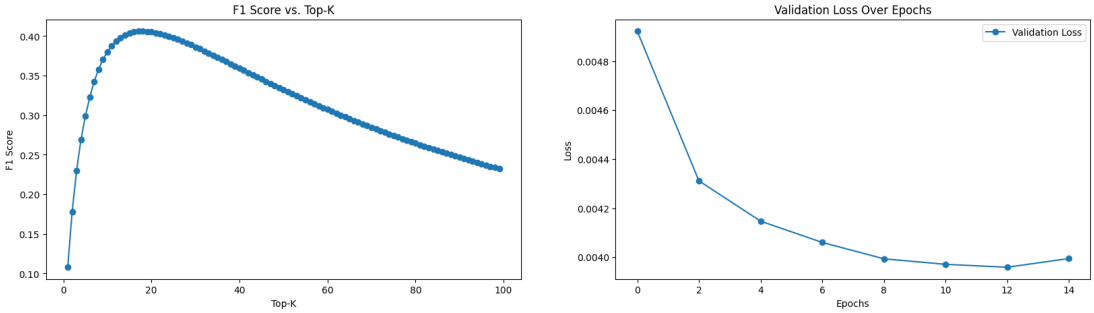


Figure 4.26: F1 score vs top k and validation loss

Generating Predictions on Test Data

The trained model is used to generate predictions on test data. The predictions are saved in a CSV file for submission or further analysis. This step ensures the model's practical applicability to new, unseen data.

4.5.4 Overall Model Evaluation

The overall model evaluation provides a detailed assessment of the multimodal model's strengths and weaknesses. By combining quantitative metrics with qualitative insights, we gain a comprehensive understanding of the model's performance and identify areas for future improvement. This holistic evaluation ensures that the model is well-equipped to handle real-world challenges and make accurate species predictions based on diverse data modalities.

4.6 Final Outcome

The culmination of our research and development efforts has led to the creation of a robust multimodal model capable of accurately predicting species compositions based on diverse data inputs. This final outcome section summarizes the key results obtained from our experiments and highlights the implications of our findings.

4.6.1 Result obtained

Baseline Experiments

Our initial experiments established several baselines using different modalities of data. The performance metrics from these baselines provided a foundation for further improvements. Here are the results from the baseline models:

- Bioclimatic Cubes (ResNet18 + BCE): Achieved a validation loss of 0.25784.
- Landsat Cubes (ResNet18 + BCE): Achieved a validation loss of 0.26424.
- Sentinel Image Patches (Swin-v2-t + BCE): Achieved a validation loss of 0.23555.

Multimodal Model Performance

Building on the insights from the baseline experiments, we developed a multimodal model that integrates tabular data, Landsat cubes, bioclimatic cubes, and Sentinel image patches. The model architecture employs a Siamese network approach with individual backbones for each modality, followed by a fully connected neural network for final classification. The combined approach yielded significant improvements:

- Multimodal Model (Combined): Achieved a validation loss of 0.31626, demonstrating the superior performance of integrating multiple data sources.

Chapter 5

Challenges

Overview

The project aimed to develop a robust model capable of predicting species distribution using multimodal data sources, including tabular data, Landsat cubes, bioclimatic cubes, and Sentinel image patches. This ambitious objective necessitated addressing various technical and methodological challenges to ensure accurate and reliable model performance.

During the course of this project, several significant challenges were encountered that impacted the development and performance of the multimodal model. These challenges are outlined as follows:

5.0.1 Multi-Label Learning from Single Positive Labels

One of the primary difficulties in this project was the need to perform multi-label learning from datasets where only single positive labels were available. This type of learning scenario complicates the model training process, as the model must learn to predict multiple possible labels for each input despite only being provided with single-label training examples. This can lead to incomplete learning and requires sophisticated techniques to generalize well.

5.0.2 Strong Class Imbalance

The dataset used for training exhibited strong class imbalance, with some species being underrepresented. This imbalance can severely affect model performance, causing it to be biased towards the more frequently occurring classes and resulting in poor predictive accuracy for the less common species. Addressing class imbalance often necessitates the use of specialized loss functions, resampling techniques, or augmentation strategies to ensure the model does not overlook rare classes.

5.0.3 Multi-Modal Learning

Incorporating multiple data modalities (tabular data, Landsat cubes, bioclimatic cubes, and Sentinel image patches) into a single predictive model introduced additional complexity. Each modality has distinct characteristics and requires different preprocessing steps and feature extraction techniques. Ensuring that the model effectively integrates these diverse data types to make coherent predictions posed a significant technical challenge.

5.0.4 Large-Scale Data Handling

Handling and processing large-scale datasets, especially those involving high-dimensional satellite imagery and time-series data, required substantial computational resources and efficient data management strategies. The need for high-capacity storage, memory management, and parallel processing capabilities was crucial to ensure smooth and efficient model training and evaluation.

Overall, addressing these challenges was crucial for the successful development and deployment of a robust multimodal model. Future work will continue to focus on overcoming these obstacles to further enhance model performance and applicability.

Chapter 6

Conclusion

This project aimed to develop a sophisticated multimodal model capable of predicting species distribution across diverse geographical regions using a variety of environmental data sources. The model leveraged tabular data, Landsat cubes, bioclimatic cubes, and Sentinel image patches, integrating these disparate modalities through a carefully designed neural network architecture. The journey from data preprocessing to model evaluation highlighted numerous challenges and provided valuable insights into the complexities of species distribution modeling.

Initially, the project focused on preprocessing the various data sources to ensure they were compatible with the neural network architecture. This involved meticulous data cleaning, normalization, and transformation processes, which were critical for achieving reliable model performance. The preprocessing stage also included handling missing values and merging multiple datasets to create a comprehensive feature set that accurately represented the environmental conditions influencing species distribution.

The model-building phase was characterized by the implementation of a multimodal ensemble approach, using a ”siamese” network architecture with distinct backbones for each data modality. The ResNet18 model was adapted for processing Landsat and bioclimatic cubes, while the Swin-v2-t model was utilized for handling Sentinel image patches. This architecture allowed the model to effectively learn from the unique features of each data modality and integrate these learnings to make accurate predictions about species presence.

Training and evaluation of the model involved rigorous experimentation with

various hyperparameters, loss functions, and optimization techniques. The model's performance was assessed through a series of baseline experiments, each focusing on a different data modality. The results demonstrated that the multimodal approach, which combined Landsat and bioclimatic cubes with Sentinel images, achieved the highest predictive accuracy, outperforming models that relied on a single data modality. This underscores the importance of integrating multiple data sources to capture the complex environmental factors influencing species distribution.

Despite the promising results, the project faced several challenges. These included the difficulty of multi-label learning from single positive labels, strong class imbalance, and the complexity of multimodal learning. Addressing these challenges required innovative solutions, such as specialized loss functions and data augmentation techniques, to ensure the model generalized well across different species and regions.

The project also highlighted the importance of modular programming in managing the complexity of the model and facilitating future enhancements. By structuring the code in a modular fashion, the project ensured that individual components could be easily modified and extended, paving the way for ongoing improvements and experimentation with new techniques and architectures.

In conclusion, this project successfully demonstrated the potential of a multimodal approach for species distribution modeling, integrating diverse environmental data sources to achieve high predictive accuracy. The insights gained from addressing the various challenges encountered along the way provide a solid foundation for future research in this area. Continued advancements in data preprocessing, model architecture, and training techniques will be crucial for further improving the performance and reliability of species distribution models, ultimately contributing to better conservation and management of biodiversity.

Chapter 7

Future Scope

The success of this project in predicting species distribution using a multimodal model paves the way for numerous future research and development opportunities. The potential for enhancing model performance and expanding its applications is vast, and several avenues can be explored to achieve these goals. Below are some key areas for future work:

7.1 Improvement in Model Architecture and Techniques

7.1.1 Advanced Deep Learning Architectures

Future research could explore more advanced deep learning architectures, such as transformer-based models or convolutional neural networks with deeper layers. These architectures may capture complex patterns and interactions between different environmental factors more effectively.

7.1.2 Incorporation of Attention Mechanisms

Attention mechanisms could be integrated into the model to allow it to focus on the most relevant features within each data modality. This could enhance the model's ability to learn from heterogeneous data sources and improve predictive accuracy.

7.1.3 Ensemble Methods

Further experimentation with ensemble methods, combining predictions from multiple models trained on different subsets of the data, could improve overall performance and robustness. Techniques like stacking, bagging, or boosting could be beneficial in this context.

7.2 Enhanced Data Utilization

7.2.1 Inclusion of Additional Data Modalities

Incorporating additional data modalities, such as soil data, hydrological information, or more detailed meteorological data, could provide a richer context for predicting species distribution. This could improve the model's ability to generalize across diverse ecological conditions.

7.2.2 Longitudinal Data Analysis

Analyzing longitudinal data to capture temporal changes in species distribution patterns could be another fruitful direction. This could involve modeling trends over time and understanding how species distributions shift in response to climatic changes, human activities, or other factors.

7.2.3 Higher Resolution Data

Utilizing higher resolution satellite imagery and finer granularity bioclimatic data could improve the model's ability to detect subtle environmental changes and their impact on species distribution.

7.3 Scalability and Real-Time Applications

7.3.1 Scalability to Larger Datasets

Future work should focus on scaling the model to handle larger datasets efficiently. This includes optimizing data preprocessing pipelines, leveraging cloud computing resources, and improving model training and inference speeds.

7.3.2 Real-Time Species Distribution Monitoring

Developing a system for real-time monitoring and prediction of species distribution changes could be highly valuable for conservation efforts. This would involve integrating the model into a real-time data pipeline and ensuring it can process and analyze new data promptly.

7.4 Improved Model Interpretability and User Interfaces

7.4.1 Model Interpretability

Improving the interpretability of the model's predictions will be crucial for gaining insights into the factors driving species distributions. Techniques such as SHAP values, feature importance analysis, and visualization of learned representations could help in understanding the model's decision-making process.

7.4.2 User-Friendly Interfaces

Creating user-friendly interfaces and visualization tools to present the model's predictions to ecologists, conservationists, and policymakers could enhance the practical utility of the model. These interfaces should allow users to interact with the data, explore predictions, and derive actionable insights.

7.5 Application to Conservation and Management

7.5.1 Conservation Planning and Policy Making

The model could be further developed to assist in conservation planning and policy making. By predicting how species distributions are likely to change under various scenarios, the model could help identify priority areas for conservation and inform strategies to mitigate the impacts of climate change and habitat loss.

7.5.2 Collaboration with Ecologists and Conservationists

Collaborating with ecologists and conservationists to validate the model's predictions in the field and refine its algorithms based on empirical observations will be essential. Such collaborations can ensure the model's predictions are grounded in real-world data and its applications are aligned with conservation goals.

In summary, the future scope of this project is broad and promising. By leveraging advanced techniques, expanding data sources, enhancing model interpretability, and applying the model to real-world conservation challenges, we can significantly advance our understanding and management of species distributions. These efforts will contribute to more effective biodiversity conservation and help mitigate the impacts of environmental changes on ecosystems worldwide.

References

1 E. S. Brondizio, J. Settele, S. Diaz, H. T. Ngo, Global assessment report on biodiversity and ecosystem services of the intergovernmental science-policy platform on biodiversity and ecosystem services, IPBES secretariat, Bonn, Germany.(2019).

2 Geolifeclef 2024 @ LifeCLEF & CVPR-FGVC (no date) Kaggle. Available at: https://www.kaggle.com/competitions/geolifeclef-2024/overview (Accessed: 05 June 2024).

3 C. Botella, A. Joly, P. Bonnet, P. Monestiez, F. Munoz, A deep learning approach to species distribution modelling, Multimedia Tools and Applications for Environmental & Biodiversity Informatics (2018) 169–199

4 B. Deneu, A. Joly, P. Bonnet, M. Servajean, F. Munoz, Very high resolution species distribution modeling based on remote sensing imagery: how to capture fine-grained and large-scale vegetation ecology with convolutional neural networks?, Frontiers in plant science 13 (2022) 839279.

5 Bengio, Yoshua. Learning deep architectures for ai. Foundations and Trends in Machine Learning, 2(1):1–127, 2009.

6 Analytics Vidhya. (n.d.). Analytics Vidhya — The ultimate place for Generative AI, Data Science and Data Engineering. https://www.analyticsvidhya.com/terms

7 J. Estopinan, M. Servajean, P. Bonnet, F. Munoz, A. Joly, Deep species distribution modeling from sentinel-2 image time-series: a global scale analysis on the orchid family, Frontiers in Plant Science 13 (2022) 839327

8 T. Hastie, W. Fithian, Inference from presence-only data; the ongoing controversy, Ecography 36 (2013) 864–867

9 T. Mesaglio, C. T. Callaghan, An overview of the history, current contributions and future outlook of inaturalist in australia, Wildlife Research 48 (2021) 289–303

10 GBIF.Org User, Occurrence download, 2022. URL: https://www.gbif.org/occurrence/download/0144742-220831081235567. doi:10.15468/DL.8WVZQF

11 C. Botella, D. Benjamin, M. Diego Gonzalez, S. Maximilien, L. Théo, E. Joaquim, L. César, P. Bonnet, A. Joly, The GeoLifeCLEF 2023 Dataset to evaluate plant species distribution models at high spatial resolution across Europe, 2023. URL: https://hal.science/hal-04152362, the full dataset is freely available at the link below (perennial repository) for academic use or other non-commercial use: https://lab.plantnet.org/seafile/d/936fe4298a5a4f4c8dbd/

12 D. Chen, Y. Xue, S. Chen, D. Fink, C. Gomes, Deep multi-species embedding, arXiv preprint arXiv:1609.09353 (2016)

Appendix A

Dataset

A.1 Satellite image patches

1280mx1280m RGB and NIR patches (four bands) centered at the observation geolocation and taken the same year. The patches are compressed in two zip files (patchs$_rgb.zip$, $patchs_nir.zip$)$accessibleinfolder/SatelliteImages/$.

Format: 128x128 JPEG images, a color JPEG file for RGB data and a grayscale one for Near-Infrared.

Resolution: 10 meters per pixel

Source: Sentinel2 remote sensing data pre-processed by the Ecodatacube platform

Access: First, one must download and decompress the provided zip files. Each JPEG file corresponds to a unique observation location (via ”surveyId”). To load the RGB or NIR patch for a selected observation, take the ”surveyId” from any occurrence CSV and load it following this rule –¿ ’. . . /CD/AB/XXXXABCD.jpeg’. For example, the image location for the surveyId 3018575 is ”./75/85/3018575.jpeg”. For all ”surveyId” with less than four digits, you can use a similar rule. For a ”surveyId” 1 is ”./1/1.jpeg”.

A.2 Satellite time series

Each observation is associated with the time series of the satellite median point values over each season since the winter of 1999 for six satellite bands (R, G, B, NIR, SWIR1, and SWIR2). This data carries a high-resolution local signature of the past 20 years' succession of seasonal vegetation changes, potential extreme natural events (fires), or land use changes.

1. Format1: Six CSV files, one per band. The corresponds to the "surveyId," and the columns are the 84 seasons from winter 2000 until autumn 2020.
2. Format2: TimeSeries-Cubes - The above-mentioned CSV aggregated into 3d tensors with axes as BAND, QUARTER, and YEAR.
3. Resolution: The original satellite data has a resolution of 30m per pixel
4. Source: Landsat remote sensing data pre-processed by the Ecodatacube platform
5. Access:/SatelliteTimeSeries/

A.3 Monthly climatic rasters

Four climatic variables computed monthly (mean, minimum and maximum temperature, and total precipitation) from January 2000 to December 2019, yielding 960 low-resolution rasters covering Europe.

1. Format1: CSV files, one per raster referenced through the "surveyId".
2. Format2: TimeSeries-Cubes - Above mentioned CSV aggregated into 3d tensors with axis as RASTER-TYPE, YEAR, and MONTH.
3. Resolution: 1 kilometer
4. Source: Chelsa
5. Access:/EnvironmentalRasters/Climate/Climatic$_{M}onthly_{2}$000 – 2019

A.4 Environmental rasters

For each observation, we provide additional environmental data such as GeoTIFF rasters and scalar values already extracted from the rasters. We provide CSV files, one per band raster type, i.e., Climate, Elevation, Human Footprint, LandCover, and SoilGrids.

1. Bioclimatic rasters: 19 low-resolution rasters covering Europe; commonly used in species distribution modeling. Provided in longitude/latitude coordinates (WGS84).

2. Soil rasters: Nine pedologic low-resolution rasters covering Europe. Provided variables describe the soil properties from 5 to 15cm depth and are determinant of plant species distributions. Check the definition.txt file about the provided variables (e.g., pH, clay, organic carbon and nitrogen contents, etc.).

3. Elevation: High-resolution raster covering Europe.

4. Land Cover: A medium-resolution multi-band land cover raster covering Europe. Each band describes either the land cover class prediction or its confidence under various classifications. We recommend the use of IGBP (17 classes) or LCCS (43 classes) layers, often used in species distribution modeling.

5. Humand footprint: Several low-resolution rasters describing human footprint, encapsulating seven pressures on the environment (e.g., nighlight level, population density) induced by human presence and activity, are provided for two time periods, the early 90's (1993) and late 2000' (2009). We provide two summary rasters combining all human pressures and two detailed rasters per pressure, which avoid an arbitrary degradation of the original data.

Appendix B

Codes

All the source code of this project implementation is present in the git hub repository: https://github.com/yash-raj202134/GeoLifeCLEF-LifeCLEF-CVPR-FGVC

B.1 Prepare custom dataset loader

```
class TrainDataset(Dataset):
    def __init__(self, tab, bioclim_data_dir, landsat_data_dir, sentinel_data_dir, metadata, transform=None):
        self.tab = tab
        self.transform = transform
        self.sentinel_transform = transform = transforms.Compose([
            transforms.ToTensor(),
            transforms.Normalize(mean=(0.5, 0.5, 0.5, 0.5), std=(0.5, 0.5, 0.5, 0.5)),
        ])

        self.bioclim_data_dir = bioclim_data_dir
        self.landsat_data_dir = landsat_data_dir
        self.sentinel_data_dir = sentinel_data_dir
        self.metadata = metadata
        self.metadata = self.metadata.dropna(subset="speciesId").reset_index(drop=True)
        self.metadata['speciesId'] = self.metadata['speciesId'].astype(int)
        self.label_dict = self.metadata.groupby('surveyId')['speciesId'].apply(list).to_dict()

        self.metadata = self.metadata.drop_duplicates(subset="surveyId").reset_index(drop=True)

    def __len__(self):
        return len(self.metadata)

    def __getitem__(self, idx):

        survey_id = self.metadata.surveyId[idx]
        tab = torch.Tensor(self.tab[self.tab["surveyId"]==survey_id][features].values[0])
        landsat_sample = torch.nan_to_num(torch.load(os.path.join(self.landsat_data_dir,
                                                                  f"GLC24-PA-train-landsat-time-series_{survey_id}_cube.pt")))
        bioclim_sample = torch.nan_to_num(torch.load(os.path.join(self.bioclim_data_dir,
                                                                  f"GLC24-PA-train-bioclimatic_monthly_{survey_id}_cube.pt")))

        rgb_sample = np.array(Image.open(construct_patch_path(self.sentinel_data_dir, survey_id)))
        nir_sample = np.array(Image.open(construct_patch_path(self.sentinel_data_dir.replace("rgb", "nir").replace("RGB", "NIR"), survey_id)))
```

```
        nir_sample = np.array(Image.open(construct_patch_path(self.sentinel_data_dir.replace("rgb", "nir").replace("RGB", "NIR"), survey_id)))
        sentinel_sample = np.concatenate((rgb_sample, nir_sample[...,None]), axis=2)

        species_ids = self.label_dict.get(survey_id, [])  # Get list of species IDs for the survey ID
        label = torch.zeros(num_classes)  # Initialize label tensor
        for species_id in species_ids:
            label_id = species_id
            label[label_id] = 1  # Set the corresponding class index to 1 for each species

        if isinstance(landsat_sample, torch.Tensor):
            landsat_sample = landsat_sample.permute(1, 2, 0)  # Change tensor shape from (C, H, W) to (H, W, C)
            landsat_sample = landsat_sample.numpy()  # Convert tensor to numpy array

        if isinstance(bioclim_sample, torch.Tensor):
            bioclim_sample = bioclim_sample.permute(1, 2, 0)  # Change tensor shape from (C, H, W) to (H, W, C)
            bioclim_sample = bioclim_sample.numpy()  # Convert tensor to numpy array

        if self.transform:
            landsat_sample = self.transform(landsat_sample)
            bioclim_sample = self.transform(bioclim_sample)
            sentinel_sample = self.sentinel_transform(sentinel_sample)

        return tab, landsat_sample, bioclim_sample, sentinel_sample, label, survey_id

class TestDataset(TrainDataset):
    def __init__(self, tab, bioclim_data_dir, landsat_data_dir, sentinel_data_dir, metadata, transform=None):
        self.tab = tab
        self.transform = transform
        self.sentinel_transform = transform = transforms.Compose([
            transforms.ToTensor(),
            transforms.Normalize(mean=(0.5, 0.5, 0.5, 0.5), std=(0.5, 0.5, 0.5, 0.5)),
        ])
```

```
        self.bioclim_data_dir = bioclim_data_dir
        self.landsat_data_dir = landsat_data_dir
        self.sentinel_data_dir = sentinel_data_dir
        self.metadata = metadata

    def __getitem__(self, idx):

        survey_id = self.metadata.surveyId[idx]
        tab = torch.Tensor(self.tab[self.tab["surveyId"]==survey_id][features].values[0])
        landsat_sample = torch.nan_to_num(torch.load(os.path.join(self.landsat_data_dir,
                                                                  f"GLC24-PA-test-landsat_time_series_{survey_id}_cube.pt")))
        bioclim_sample = torch.nan_to_num(torch.load(os.path.join(self.bioclim_data_dir,
                                                                  f"GLC24-PA-test-bioclimatic_monthly_{survey_id}_cube.pt")))

        rgb_sample = np.array(Image.open(construct_patch_path(self.sentinel_data_dir, survey_id)))
        nir_sample = np.array(Image.open(construct_patch_path(self.sentinel_data_dir.replace("rgb", "nir").replace("RGB", "NIR"), survey_id)))
        sentinel_sample = np.concatenate((rgb_sample, nir_sample[...,None]), axis=2)

        if isinstance(landsat_sample, torch.Tensor):
            landsat_sample = landsat_sample.permute(1, 2, 0)  # Change tensor shape from (C, H, W) to (H, W, C)
            landsat_sample = landsat_sample.numpy()  # Convert tensor to numpy array

        if isinstance(bioclim_sample, torch.Tensor):
            bioclim_sample = bioclim_sample.permute(1, 2, 0)  # Change tensor shape from (C, H, W) to (H, W, C)
            bioclim_sample = bioclim_sample.numpy()  # Convert tensor to numpy array

        if self.transform:
            landsat_sample = self.transform(landsat_sample)
            bioclim_sample = self.transform(bioclim_sample)
            sentinel_sample = self.sentinel_transform(sentinel_sample)

        return tab, landsat_sample, bioclim_sample, sentinel_sample, survey_id
```

B.2 Load metadata and prepare data loaders

```
train_landcover = pd.read_csv("/Dataset/geolifeclef-2024/EnvironmentalRasters/EnvironmentalRasters/LandCover/GLC24-PA-train-landcover.csv")
train_solidgrids = pd.read_csv("/Dataset/geolifeclef-2024/EnvironmentalRasters/EnvironmentalRasters/SoilGrids/GLC24-PA-train-soilgrids.csv")
train_humanfootprint = pd.read_csv("/Dataset/geolifeclef-2024/EnvironmentalRasters/EnvironmentalRasters/Human Footprint/GLC24-PA-train-human_footprint.cs
train_elevation = pd.read_csv("/Dataset/geolifeclef-2024/EnvironmentalRasters/EnvironmentalRasters/Elevation/GLC24-PA-train-elevation.csv")
train_climate = pd.read_csv("/Dataset/geolifeclef-2024/EnvironmentalRasters/EnvironmentalRasters/Climate/Average 1981-2010/GLC24-PA-train-bioclimatic.csv

train_tab = train_climate.merge(train_elevation,on="surveyId").merge(train_humanfootprint,on="surveyId").merge(train_solidgrids,on="surveyId").merge(trai
features = list(train_tab.columns)[1:]
train_tab = train_tab.fillna(-1).replace(np.inf, -1).replace(-np.inf, -1)

test_landcover = pd.read_csv("/Dataset/geolifeclef-2024/EnvironmentalRasters/EnvironmentalRasters/LandCover/GLC24-PA-test-landcover.csv")
test_solidgrids = pd.read_csv("/Dataset/geolifeclef-2024/EnvironmentalRasters/EnvironmentalRasters/SoilGrids/GLC24-PA-test-soilgrids.csv")
test_humanfootprint = pd.read_csv("/Dataset/geolifeclef-2024/EnvironmentalRasters/EnvironmentalRasters/Human Footprint/GLC24-PA-test-human_footprint.csv"
test_elevation = pd.read_csv("/Dataset/geolifeclef-2024/EnvironmentalRasters/EnvironmentalRasters/Elevation/GLC24-PA-test-elevation.csv")
test_climate = pd.read_csv("/Dataset/geolifeclef-2024/EnvironmentalRasters/EnvironmentalRasters/Climate/Average 1981-2010/GLC24-PA-test-bioclimatic.csv")

test_tab = test_climate.merge(test_elevation,on="surveyId").merge(test_humanfootprint,on="surveyId").merge(test_solidgrids,on="surveyId").merge(test_land
test_tab = test_tab.fillna(-1).replace(np.inf, -1).replace(-np.inf, -1)
test_tab.head()
```

B.3 Define and initialize a Multimodal Model

```
class MultimodalEnsemble(nn.Module):
    def __init__(self, num_classes):
        super(MultimodalEnsemble, self).__init__()
        self.tab_norm = nn.LayerNorm([len(features)])
        self.tab_model = nn.Sequential(nn.Linear(len(features),128),
                                       nn.ReLU(),
                                       nn.Linear(128,128),
                                       nn.ReLU(),
                                       nn.Linear(128,32),
                                      )

        self.landsat_norm = nn.LayerNorm([6,4,21])
        self.landsat_model = models.resnet18(weights=None)
        # Modify the first convolutional layer to accept 6 channels instead of 3
        self.landsat_model.conv1 = nn.Conv2d(6, 64, kernel_size=3, stride=1, padding=1, bias=False)
        self.landsat_model.maxpool = nn.Identity()

        self.bioclim_norm = nn.LayerNorm([4,19,12])
        self.bioclim_model = models.resnet18(weights=None)
        # Modify the first convolutional layer to accept 4 channels instead of 3
        self.bioclim_model.conv1 = nn.Conv2d(4, 64, kernel_size=3, stride=1, padding=1, bias=False)
        self.bioclim_model.maxpool = nn.Identity()

        self.sentinel_model = models.swin_t(weights="IMAGENET1K_V1")
        # Modify the first layer to accept 4 channels instead of 3
        self.sentinel_model.features[0][0] = nn.Conv2d(4, 96, kernel_size=(4, 4), stride=(4, 4))
        self.sentinel_model.head = nn.Identity()

        self.ln0 = nn.LayerNorm(32)
        self.ln1 = nn.LayerNorm(1000)
        self.ln2 = nn.LayerNorm(1000)
        self.ln3 = nn.LayerNorm(768)
```

```
        self.fc1 = nn.Linear(2768+32, 1024)
        self.fc2 = nn.Linear(1024, num_classes)

        self.dropout = nn.Dropout(p=0.15)

    def forward(self, t, x, y, z):
        t = self.tab_norm(t)
        t = self.tab_model(t)
        t = self.ln0(t)
        t = self.dropout(t)

        x = self.landsat_norm(x)
        x = self.landsat_model(x)
        x = self.ln1(x)
        x = self.dropout(x)

        y = self.bioclim_norm(y)
        y = self.bioclim_model(y)
        y = self.ln2(y)
        y = self.dropout(y)

        z = self.sentinel_model(z)
        z = self.ln3(z)
        z = self.dropout(z)

        txyz = torch.cat((t, x, y, z), dim=1)

        txyz = self.fc1(txyz).relu()
        txyz = self.dropout(txyz)

        out = self.fc2(txyz)
        return out
```

B.4 Training , Val & thresholding

```
best_val = None
best_model = deepcopy(model)

for epoch in range(num_epochs):
    # training
    total = 0
    total_loss = 0
    model.train()
    for batch_idx, (data0, data1, data2, data3, targets, _) in enumerate(tqdm(train_loader)):
        data0 = data0.to(device)
        data1 = data1.to(device)
        data2 = data2.to(device)
        data3 = data3.to(device)
        targets = targets.to(device)

        # Mixup
        if np.random.rand() < 0.4:
            lam = torch.tensor(np.random.beta(0.4, 0.4)).to(device)
            rand_index = torch.randperm(data0.size()[0]).to(device)
            mixed_data0 = lam * data0 + (1 - lam) * data0[rand_index]
            mixed_data1 = lam * data1 + (1 - lam) * data1[rand_index]
            mixed_data2 = lam * data2 + (1 - lam) * data2[rand_index]
            mixed_data3 = lam * data3 + (1 - lam) * data3[rand_index]
            targets_a, targets_b = targets, targets[rand_index]
            mixed_targets = lam * targets_a + (1 - lam) * targets_b
            outputs = model(mixed_data0, mixed_data1, mixed_data2, mixed_data3)
            loss = criterion(outputs, mixed_targets)
        else:
            outputs = model(data0, data1, data2, data3)
            loss = criterion(outputs, targets)

        optimizer.zero_grad()
        loss.backward()
```

```
        optimizer.step()

        total += data1.shape[0]
        total_loss += loss.item() * data1.shape[0]

        if DEBUG and batch_idx > 50:
            break

    total_loss /= total
    train_losses.append(total_loss)  # Record the training loss

    if epoch % 2 == 0:
        # validation
        vtotal = 0
        vtotal_loss = 0
        model.eval()
        with torch.no_grad():
            for batch_idx, (data0, data1, data2, data3, targets, _) in enumerate(val_loader):
                data0 = data0.to(device)
                data1 = data1.to(device)
                data2 = data2.to(device)
                data3 = data3.to(device)
                targets = targets.to(device)

                outputs = model(data0, data1, data2, data3)

                loss = criterion(outputs, targets)

                vtotal += data1.shape[0]
                vtotal_loss += loss.item() * data1.shape[0]

                if DEBUG and batch_idx > 50:
                    break
```

```
        vtotal_loss /= vtotal
        val_losses.append(vtotal_loss)  # Record the validation loss

        print(f'Epoch {epoch} : train_loss {total_loss:.5f} | val_loss {vtotal_loss:.5f}')

        if best_val is None or vtotal_loss < best_val:
            best_val = vtotal_loss
            best_model = deepcopy(model)

# Save the trained model
model.eval()
torch.save(model.state_dict(), "models/baseline/main/last.pth")
best_model.eval()
torch.save(best_model.state_dict(), "models/baseline/main/best.pth")
```

Appendix C

Additional documents

C.1 Appointment letter

Machine Learning Internship LMS Project POSITION OFFER

Ignitus Worldwide

March 8th, 2024

Dear Yash,

We are thrilled to extend our heartfelt congratulations to you on being selected as the new "ML-er" (LMS project) Intern at Ignitus. We believe that you have the right set of skills and experience required to excel in this position, and we look forward to working with you.

In the coming days, we will be focusing on wrapping up a few more formalities related to the remote working environment. Our goal is to ensure a smooth transition into your new role starting from February 1, 2024, to July 31 2024. We encourage you to read on and learn more about this opportunity, and we are confident that we can answer any lingering questions you may have.

We understand that starting your internship period with us will be an unpaid position. However, we want to assure you that this is not an obstacle, as we are confident that our actions towards the monetization of our services and products will be fruitful. As a result, we are hopeful that you may receive a stipend that will be negotiated once it is feasible.

As the Machine Learning Intern, you will be reporting to Afelio Padilla, Ignitus' COO. Your responsibilities will be focused on maintaining a creative and proactive spirit within the work team. Our ultimate goal is for you to become a leader within a team that is working on the Learning Management System (LMS) project. You will have all the possible help from Ignitus as you work towards increasing the learner's engagement through the most curated content, which will be widely different from standard MOOC platforms.

It is important to note that this employment offer is in no way a legally binding contract. As an at-will employee, both you and Ignitus can terminate employment for any reason at any time. However, we do not anticipate any such events as we are confident in your abilities to succeed in this position.

We at Ignitus are excited about the prospect of bringing you onboard and look forward to working with you. If you have any questions or concerns, please do not hesitate to reach out to us at any time. We are more than happy to help you in any way we can.

Best regards,

Afelio Padilla,
COO @ Ignitus

Candidate Name: Yash Raj

ID: LMS(ML)-OL-0416-01-2024

Afelio Padilla

Digitally signed by Afelio Padilla
Date: 2024.03.08 12:25:22 +01'00'

Afelio Padilla
COO, Ignitus
socialignitus@gmail.com

Margarita Xirgú, 1A - 5C
19005 Guadalajara, Spain, EU
+34629727376

C.2 Others

Presentation slides (summary of the report) can be accessed using :

https://docs.google.com/presentation/d/1MnskxAzVpnZ1f6M3Xvu1W2TN0jhR9bro/edit?usp=sharing&ouid=111864939534658425912&rtpof=true&sd=true

Non Disclosure Agreement

https://docs.google.com/document/d/1_QG_Ru29I2SruPcMTUehLThMlG4o2wOx/edit?usp=sharing&ouid=111864939534658425912&rtpof=true&sd=true

Certificate

https://drive.google.com/file/d/1Y-eCzqiebIMNNwJd_j4I4yhWoc6JUPke/view?usp=sharing

End of the document.