Eidgenössische Technische Hochschule Zürich

Master thesis spring 2024

Master in Computational Science and Engineering

In-Season Crop Classification using Satellite Imagery

Author:
Anya-Aurore Mauron

Supervisors:
Dr. Gregor Perich
Dr. Lukas Graf
Dr. Michele Volpi
Prof. Simone Deparis
Prof. Fernando Perez-Cruz

Contents

Abstract

Crop classification is a crucial task in remote sensing, as it is the gateway technology for a more effective and sustainable supervision of arable land. For most applications, such as fertilization and water consumption, crop type classification need to be given in real-time, as early as possible. Russwurm et al. suggested the ELECTS as an early crop classification model. Based on their work, our study aims to trade between accuracy and earliness. Using their model as benchmark, our goal is to reach earlier predictions, as well as to align the variability of our model's predictions with the variability expected from phenology.

To do so, we develop the D-ELECTS model, which estimates a classification score and a number of timestamps left until the prediction is finalized. Our model mainly relies on a daily structure of satellite time series data. The D-ELECTS cost function optimizes the classification accuracy, the prediction earliness and the decreasing amount of timestamps left through the time series length.

Our experiments resulted in earlier predictions than the ELECTS model, while maintaining a reasonable accuracy. Our work suggests the great potential of predictions aligned with phenological variability. Additionally, it offers a countdown before final classification, which is useful for practical applications.

1 Introduction

In the next decade, human population is expected to severely raise [40], necessitating an estimated global yield increase of 25 to 70% [20]. In addition, numerous agricultural challenges are arising. Current climate change is alarming for agriculture; temperature and precipitation patterns are varying, the frequency and intensity of extreme weather events are increasing [38]. Due to a higher need for provisions and the impact of climate change on agricultural resources, worldwide food security is threatened [25]. At the same time, cropland extension and intensive use of agricultural areas, like deforestation, biodiversity loss and degradation of ecosystem, are often connected with negative ecological impacts [9, 17].

From there, crop classification is the basis for a more efficient and sustainable supervision of arable land. In fact, it is the gateway technology to many applications, as all use cases, such as fertilization, water consumption, and irrigation, are crop dependent [10, 43]. Crop classification is also demanded for farmer subsidies requests and planted area estimation, required by insurers and paying agencies.

Until now, farmers have declared the majority crop type of fields for public institutions statistics [42]. Nevertheless, this process is time-consuming, bureaucratically heavy, and not necessarily consistent.

Automated crop classification has been approached by different methods while predicting crops at the end of the year, using satellite imagery [24, 26, 31, 35, 36, 41]. Indeed, today, a large variety of remote data are accessible free-of-charge to researchers, agencies and the general public [32, 35, 37, 39]. For instance, in 2014, the Copernicus program of the European Space Agency (ESA) launched the first Sentinel satellite [28]. In particular, Sentinel-2 mission provides high resolution optical image for land monitoring.

However, for most use cases, crop type classification is usually needed as early as possible [42]. Early time series classification is often called *in-season* or *early* crop classification. A few studies [5, 16, 21, 22, 29, 33] found that a high classification accuracy can be achieved in specific regions. They rely on the crop-specific growing seasons. Crops can be categorised into two groups: annual and perennial. On the one hand, annual crops last for one growing season. They can be subdivided in winter crops and spring crops. Winter crops, like winter wheat and winter rapeseed, are sown in autumn and harvested in the summer of the following year. Spring crops, such as sunflowers and corn, are sown and harvested in the same year. On the other hand, perennial crops, also called permanent crops, last for more than two growing seasons, growing continuously (like fruit trees) or dying back after each season [11].

Based on the different growing seasons of the crops, Russwurm et al. focused on optimizing a Machine Learning (ML) model for the objective of an early classification for each field parcel [34]. They developed End-to-end Learned Early Classification of Time Series (ELECTS), which consists of a neural network with a novel cost function optimizing for both a misclassification cost and an earliness reward. ELECTS relies on ceasing the prediction depending on a stopping probability, output from the model.

In particular, ELECTS usually predicts rapeseed in Brittany, France, in May. This is surprising, considering that flowering in rapeseed, which typically occurs before May, results in a significant yellow reflectance peak. We would thus expect that an early classification model would pick up such a relevant signal and therefore make earlier predictions. Moreover, when examining relatively uniform spatial units like Brittany, we expect the variability in the occurrence of phenological events to be in the magnitude of days. However, the predictions made by the ELECTS model show variability on the order of weeks.

In this context, building on the spectral and temporal differences of crops, our work is dedicated to trading between the accuracy and earliness in in-season crop classification. More precisely, our first goal is to predict earlier than the benchmark model ELECTS, while maintaining a reasonable accuracy. From there, our second goal is to align the variability of our model's predictions with the variability expected from phenology. To do so, we suppose that each crop passes through a significant phenological event, which can be observed by optical satellite sensors. For example, it could be a blooming event, such as rapeseed blooming in Spring. Then we modified the ELECTS model by replacing the stopping probability by the number of timestamps left. At each day of the year, the model outputs the number of days left until the prediction is finalized. Consequently, the number of timestamps left decreases throughout the year. We name this new model Daily ELECTS (D-ELECTS). Moreover, we developed a new way of viewing satellite imagery for crop classification, by correcting the inconsistency of the BreizhCrops dataset used in the ELECTS study and by shaping it on a daily basis. Furthermore,we designed an algorithm based on the classification probability to determine a date per crop when a major phenological event happens. Finally, we developed an innovative cost function. It takes into account the classification accuracy, the prediction earliness, and the decreasing amount of timestamps left through time. We compare two types of misclassification penalties; one that penalize wrong prediction at the beginning of the year, and one that pushes late wrong prediction to happen earlier in the year.

The structure of the thesis is as follows. First, we present the related work to our classification problem. Second, the datasets are presented, followed by the methodology, introducing the mathematical background,

the models, and the cost functions. Afterwards, the results of our experiments leading to our best model are shown, as well as a comparison between our D-ELECTS model and the benchmark model, ELECTS. We finish the thesis with a discussion of the results and conclusion.

2 Related Work

In this section, we first present the related work in crop classification. Second, Early Classification of Multivariate Time Series (ECMTS) is introduced, giving examples of the different approaches for this task. Third, we display early crop classification, introducing the ELECTS paper and the BreizhCrops dataset.

2.1 Crop Classification

Using satellite imagery, crop classification consists of accurately identifying crop types for parcels. To complete this task, two important classification strategies have been studied [5]. The first one is to exclusively use the spectral features from a single satellite scene, sampled on an exact day during the year [7]. The second strategy uses both the spectral and temporal information during one or several growing seasons [12].

This first strategy relies on the assumption that different land covers have distinctive spectral features, which can then implies the label classification. However, some crops, such as wheat and barley, have similar spectral information during certain time of the year, which makes separation of crop types difficult.

The second strategy uses both the spectral and the temporal information. In other words, the input of the models are satellite data, viewed as time series of images. We define those time series of images as Multivariate Time Series (MTS), which is, more generally, a sequence of timestamps with multiple values on each timestamp [19]. The classification accuracy improves, as crops usually have different seasonal variations and sowing dates.

Leaning on the second strategy of observing the satellite data as time series of images, recent work uses deep neural networks to minimize classification error as cost function. For example, Russwurm et al. adapted an encoder structure with recurrent layers in order to classify crop types [31]. In [36], Sainte Fare Garnot et al. proposed the transformer architecture to embed time series. Another example is the use of temporal convolutional neural network, to automatically learn temporal and spectral features [26].

2.2 Early Classification of Multivariate Time Series (ECMTS)

ECMTS is the task of predicting the class label of a MTS by only using the starting subsequence of the time series. In other words, given the timestamps one by one, by increasing order, ECMTS aims to give the label of a MTS as soon as possible. Its goal is to classify an MTS given the least number of timestamps. The terms MTS and ECMTS will be defined later formally in Section 4.

Such a challenge of predicting the label of a time series, with the smallest starting subsequence, is ingrained in the balance of accuracy and earliness. On the one hand, the longer the subsequence, the more information the model receives, therefore, the higher the accuracy. However, the earliness is compromised. On the other hand, the shorter the subsequence, the earlier the classification is completed. Nevertheless, with less data, high accuracy is more complex to achieve.

To the best of our knowledge, studies on early classification on time series data started by Rodriguez et al. in 2001 [19, 30]. They focused on the situations when only a partial time series was available.

A straightforward method for early classification using random forests is the *incremental* approach [21]. It consists of performing a supervised classification every time a new image acquisition is available, while still using the previously available imagery.

Then, most time series classification algorithms require to transform the input sequence into a feature vector, from which the model could make predictions. Many research studies [4, 8, 44, 45] were proposed on the Nearest Neighbor (NN) algorithm. Later, to provide interpretable results for MTS, [13] proposed the multivariate shapelet detection, which could classify MTS.

Furthermore, multi-Domain Deep Neural Network was introduced in 2018 as an incorporation of a convolutional neural network and a long-short term memory, to learn feature representation and relationship embedding for early classification [19]. This framework can make predictions at any timestamp, which widens the feasibility in many real-world applications.

2.3 Early Crop Classification

One specific domain of ECMTS is *early* or *in-season* crop classification. It consists of predicting the crop type from a MTS viewed as a time series of images, with the fewest timestamps possible. In other words, it involves the prediction of the crop type of a parcel or pixel, the earliest possible in the year. Compared to simple crop classification, not only is the label wanted, but as soon as possible.

Studies have shown the potential of a high classification accuracy within the growing season [5, 16, 21, 22, 29, 33, 34]. For example, Inglada et al. used an incremental classification procedure, which consists

of performing a supervised classification every time that a new image acquisition is available, using all the previously available data. In that case the supervised classifier was the random forest algorithm and the performance could reach an accuracy of about 72% around May [21]. Furthermore, most recent studies on early crop classification rely on deep learning. For example, Kondmann et al. used a Multi-Scale ResNet model as a temporal encoder [22]. In this study, the deep learning model is applied on the Planet Fusion dataset, a cloud-free dataset. An accuracy of about 60% was reached around May and 70% around the beginning of September. In another study, a transformer model was used as a crop type classifier over several years [29]. Specifically, the authors focused on training a SITS-BERT model [46], such that the hidden features are robust and transferable in a spectral and temporal manner. An F1-score of about 70% was reached around May. Later 82.5% was reached at 210th day of the year, which corresponds to end of July.

Within our research, the most up-to-date and promising work we could find in early crop classification was the ELECTS paper [34], which is based on optimizing a model on a loss function which take both the accuracy and the earliness into account.

The ELECTS model is a in-season crop type mapping model [34]. The method relies on a deep learning feature extractor with two decision heads and a loss function that optimizes the dual objective of accuracy and earliness. The specificity of this model relies on its two decision heads; the first predicts the labels' probabilities, and the second outputs a scalar probability of stopping. They managed predicting the crop types on the BreizhCrops dataset around the first week of June on average, with an overall accuracy of 80%.

3 Datasets

In this section, we introduce the Sentinel-2 mission in Section 3.1. Then, we present the BreizhCrops dataset in Section 3.2, which is one of the four datasets used in the ELECTS paper [32, 34]. From there, we adapt the dataset to our task and create the Reduced BreizhCrops dataset, exhibited in Section 3.3.

3.1 Sentinel-2 Mission

Optical satellites are equipped with sensors and instruments that capture spectral bands within the visible spectrum. These satellites rely on the sunlight reflected off the Earth's surface and atmosphere, and therefore work best with minimal cloud cover.

The Copernicus programme, implemented by ESA, aims at providing precise, timely and easily accessible information to enhance the management of the environment. In 2015, ESA launched the first Sentinel-2 satellites, whose mission supports applications such as agricultural monitoring and land cover classification [28].

The Sentinel-2 mission includes two identical optical satellites operating simultaneously, phased at 180° to each other. They share a sun-synchronous orbit at an average altitude of 786 km. Each satellite completes one orbit in 10 days, allowing most areas on Earth to be revisited every five days. However, due to overlap between swaths from adjacent orbits, certain areas are visited more frequently, as detailed in Figure 1.

Sentinel-2 carries an optical instrument payload that samples 13 spectral bands: three bands at 60 m, six bands at 20 m, and four bands at 10 m spatial resolution, as summarized in Table 1. The available products for the general public are Level-1C (Top-of-atmosphere reflectances in cartographic geometry) and Level-2A (Atmospherically corrected Surface Reflectances in cartographic geometry).

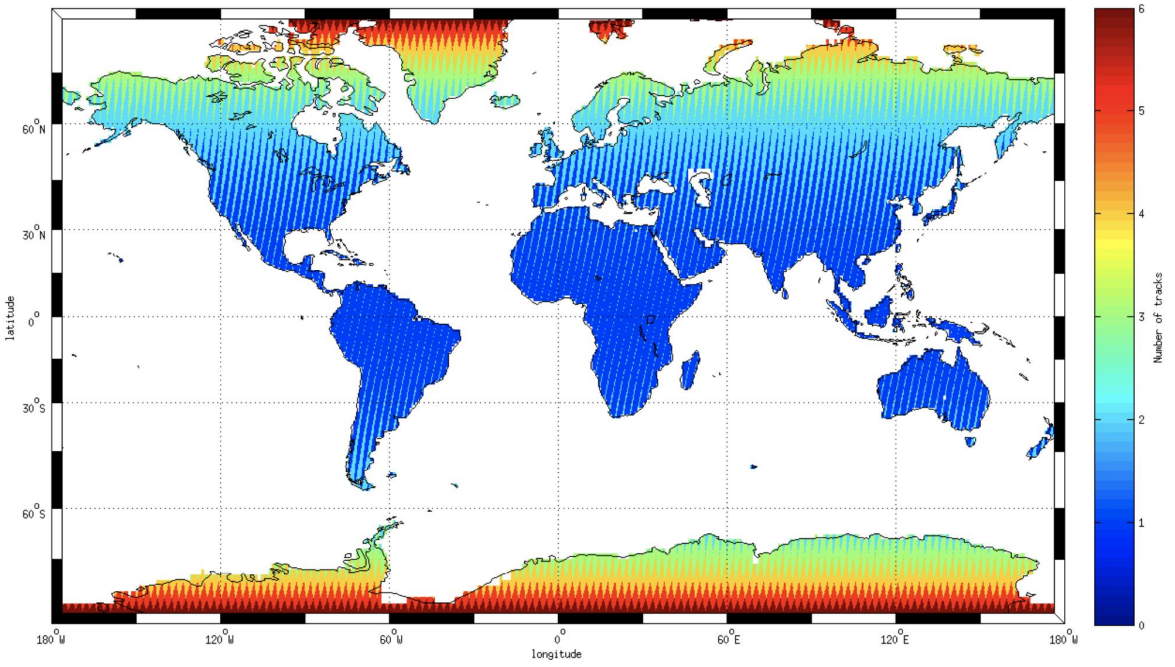


Figure 1: Revisit frequency, due to overlap area of adjacent orbits. Figure taken from Pascal Lacroix [28].

3.2 BreizhCrops

To evaluate our models, we choose a dataset that necessitates the least cleaning and preprocessing. We choose the BreizhCrops dataset [32], which contains time series of 608,263 field parcels in Brittany, France, from 2017. Time series are extracted from the Sentinel-2 satellites data.

In the following subsections, we present the dataset's properties in Section 3.2.1, regions in Section 3.2.2, labels in Section 3.2.3 and sequence lengths in Section 3.2.4, as they were set in the ELECTS paper [34].

Table 1: Sentinel-2 spectral bands [1, 27].

Band number	Band	Spatial Resolution (meters)
1	Coastal aerosol	60
2	Blue	10
3	Green	10
4	Red	10
5	Vegetation red edge 1	20
6	Vegetation red edge 2	20
7	Vegetation red edge 3	20
8	Near-Infra-Red (NIR)	10
8A	Narrow NIR	20
9	Water Vapour	60
10	Short Wave Infra-Red (SWIR) Cirrus	60
11	SWIR	20
12	SWIR	20

3.2.1 Dataset Properties

The dataset is composed of Sentinel-2 image time series extracted from January 1, 2017 to December 31, 2017. The satellite images are derived from the 13 spectral bands at level L1C, which contain the raw reflectances at the top-of-atmosphere processing level. Data are collected by the satellites every 2.5 to 5 days, due to overlap area of two orbits, which results to annual time series of lengths either 51 or 102.

Knowing the geometry of the fields, the reflectance values are averaged over the bounds of the field. That means that for each timestamp, over each field, each spectral band is averaged to obtain one feature vector.

Note that the data are biased in single observations by clouds, which create noise by causing outlier values in the satellite image sequences. For example, the spectral bands of a corn field are shown in Figure 2. The signal peaks are due to clouds.

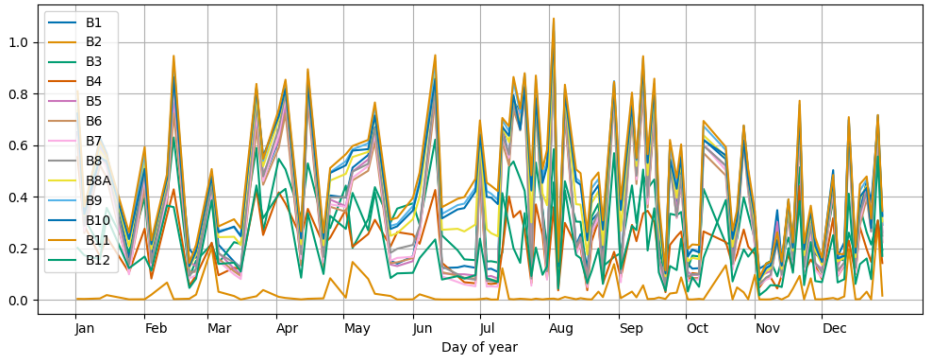
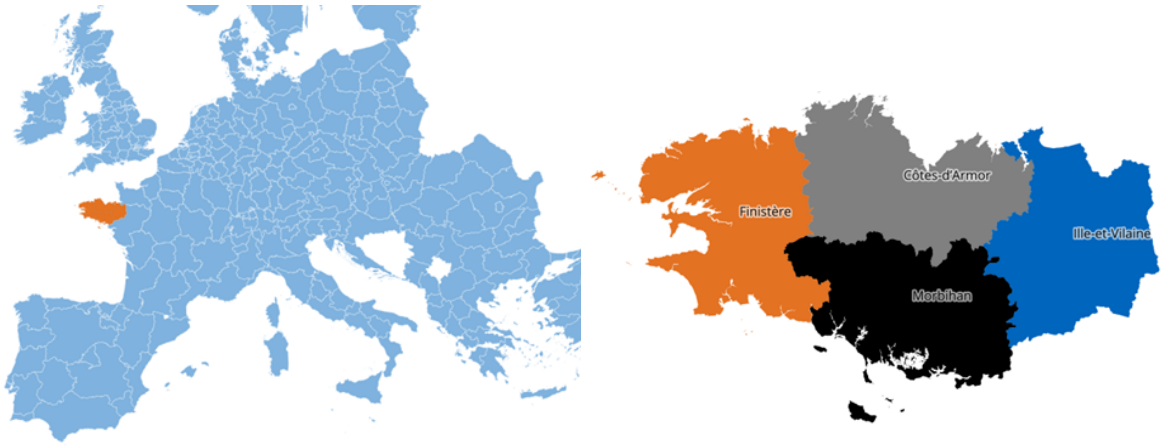


Figure 2: Spectral bands of a sample from test dataset, labelled with corn.

3.2.2 Regions

The satellite data sequences are organized at a regional level by the Nomenclature of Territorial Units for Statistics (NUTS), which forms the European standard for referencing authoritative districts. Brittany is highlighted in Figure 3a, as the NUTS-2 region FRH0. This region is then divided into the four NUTS-3 regions, as presented in Figure 3b: Côte-d'Armor (FRH01), Finistère (FRH02), Ille-et-Vilaine (FRH03), and Morbihan (FRH04).

From there, the train, validation and test datasets are obtained according to the NUTS-3 regions, as they are spatially distinct, see Table 2.



(a) Brittany as part of the NUTS-2 regions in Europe.

(b) Data partition of the NUTS-3 departments of Brittany.

Figure 3: NUTS-3 region FRH0, located in Brittany, France. Figures taken from [32].

Table 2: Spatially separated train, validation and test datasets from BreizhCrops.

Dataset	NUTS-3 region	Department	# parcels
Train	FRH01, FRH02	Côte-d'Armor, Finistère	319'258
Validation	FRH03	Ille-et-Vilaine	166'391
Test	FRH04	Morbihan	122'614

3.2.3 Labels

Only nine distinct crop categories are selected. The labels are quite varied, from frequent (barley, wheat) to rare classes (sunflower, nuts), as well as closely related groups (permanent and temporary meadows). On Figure 4, we see that the distributions of the labels are similar within train, validation and test datasets.

3.2.4 Sequence Lengths

In Section 3.2.1, it was described that the time series have sequence length of either 51 or 102, because of the overlap area of the acquisition stripes. We thus expect the sequence length distribution to be amassed at 51 and 102. Yet, in the original paper [34], the authors mentioned that the sequence lengths should be between 71 and 147. After contacting Dr. Russwurm about this confusing statement, he confirmed that the L1C data should either be around 50 or 100. However, Figure 5 shows that sequence lengths also take a non-negligible amount of values between 51 or 102. For example, in Ille-et-Vilaine department, about 98% of parcel samples have sequence length which is neither 51 nor 102. From the information we have, we can suppose that some processing or filtering was done on the data to have such sequence lengths. Nevertheless, this is only a hypothesis; we do not know the exact reasons.

Furthermore, a common sequence length is set in the data sets during training and testing. More precisely, when loading time series for model training, the authors of the ELECTS paper randomly choose sequences of 70 observations from the originally longer complete time series and arrange them in ascending time. If the time series has a shorter sequence length than 70, then it is padded with zeros. For testing, the common sequence length is set at 150, which results in all time series being complete and padded with zeros.

Consequently, during training, the time series' time stamps cannot be associated to a precise date. To be attuned to the temporal aspects of the time series, a classifier can just rely on the order of elements in the time series, and not on the absolute position of the elements in the time series.

Because of the lack of explanation of lengths between 51 and 102, and the absence of absolute temporal aspects in the time series during the training phase, we decide to create a derived dataset, the *Reduced BreizhCrops* dataset, which only has samples with sequence length 102 and with daily timestamps.

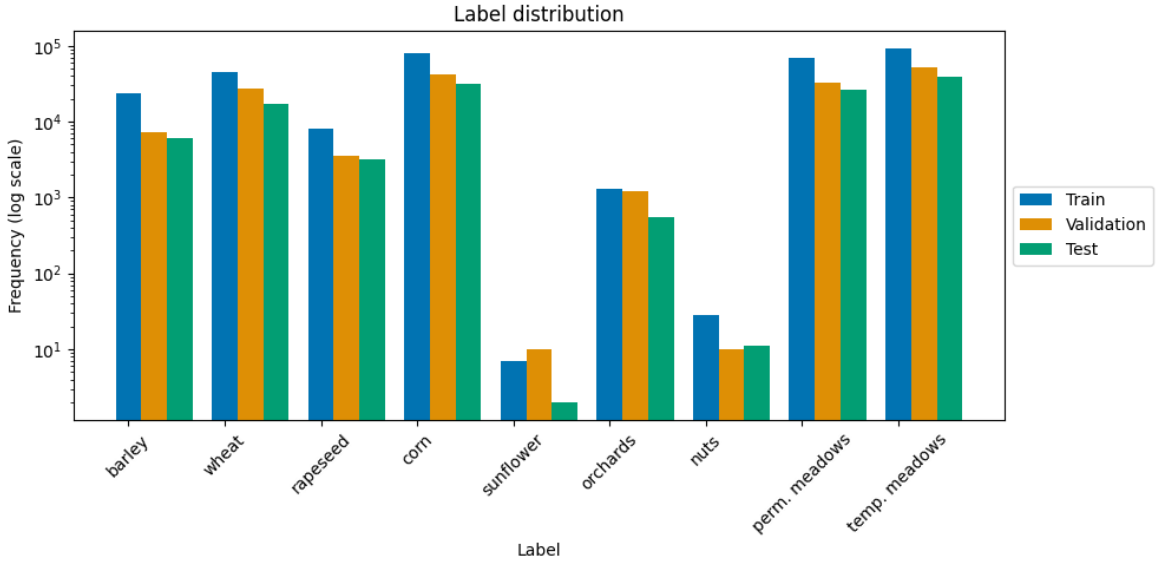
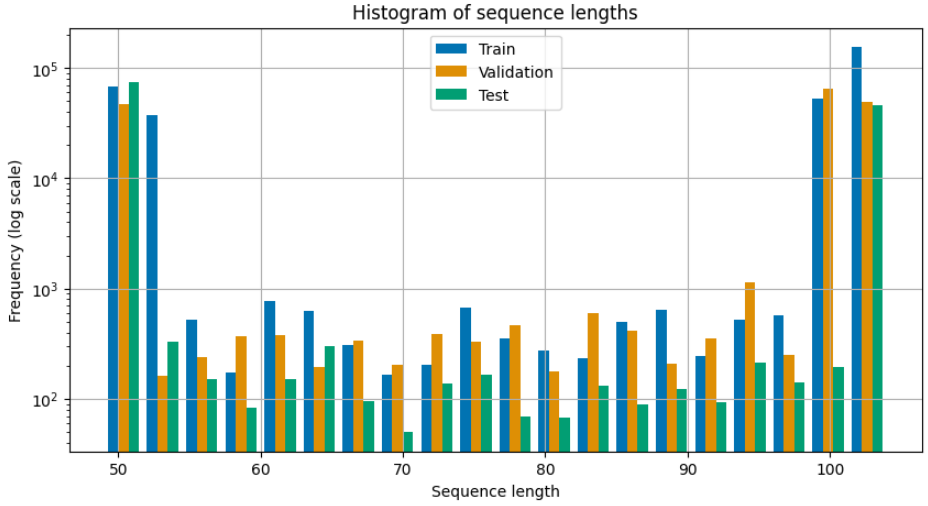
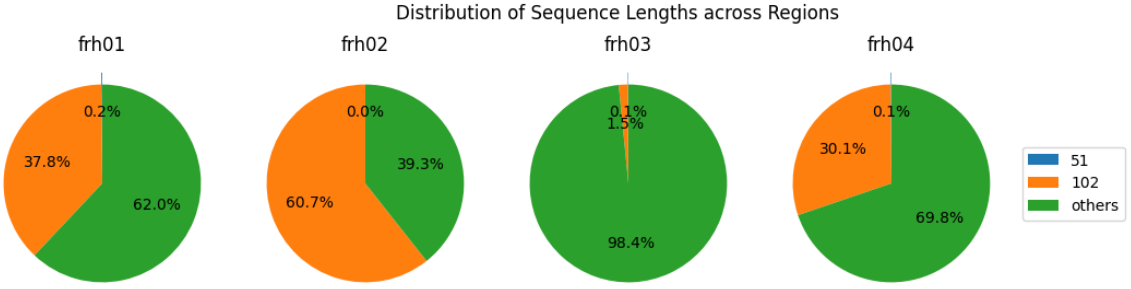


Figure 4: Label distributions of the BreizhCrops dataset.



(a) Sequence length distribution per datasets (train, validation, test).



(b) Sequence length distribution across regions. Each pie plot correspond to the distribution of the sequence lengths in one region, written above the plot. The relevant sequence lengths are indicated in the legend, i.e. 51, 102 and others.

Figure 5: Sequence length distribution of the BreizhCrops dataset, across train/valid/test datasets and regions.

3.3 Reduced BreizhCrops

In the following subsections, we describe the reduced dataset with time series of original length 102, and the re-organisation of the train, validation and test sets according to the regions and the label distributions. Then, we depict the daily timestamps, which consists of developing time series where each timestamp is equal to one day.

3.3.1 New Region Separations and Labels

Only the time series with sequence length 102 are kept, due to the small amount or absence of time series of length 51 in the region datasets, see in Figure 5b. Because of the change of the region dataset sizes, the train, validation, and test datasets need to be rearranged. We decide to keep the Finistère FRH02 department as the train set, move Côte-d'Armor FRH01 from train to validation set, and assign Ille-et-Vilaine FRH03 and Morbihan FRH04 as test sets. The new datasets are summarized in Table 3. This new arrangement is motivated by the regional separability, necessary to have independent datasets. In addition, it is also influenced by the classes distribution, as shown in Figure 6. Such combination of regional datasets allows similar label distributions for the train, validation and test datasets. Note that we remove the sunflower and nut classes because of their small size; the sunflower class is not present in each regional dataset, and the nuts class is only represented by less than 10 samples per dataset.

Table 3: Spatially separated train, validation and test datasets of Reduced BreizhCrops.

Dataset	NUTS-3 region	Department	# parcels
Train	FRH02	Finistère	85'310
Validation	FRH01	Côte-d'Armor	67'523
Test	FRH03, FRH04	Ille-et-Vilaine, Morbihan	39'378

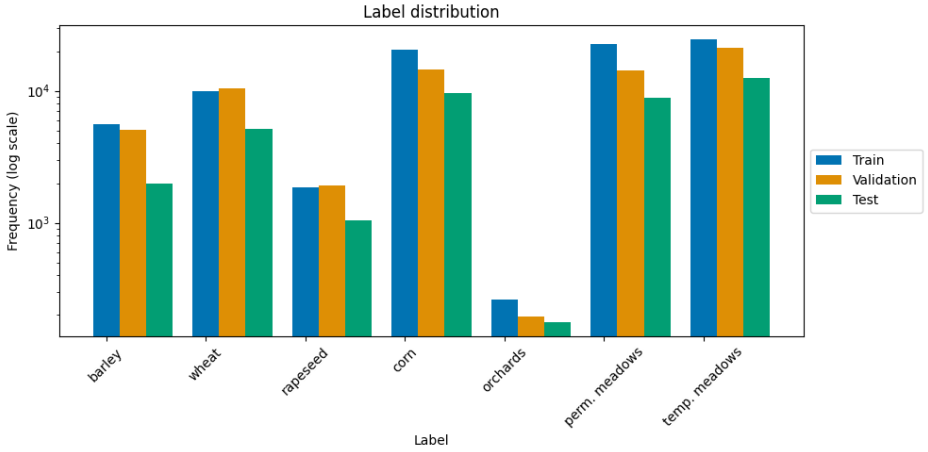


Figure 6: Label distributions of the Reduced BreizhCrops dataset.

Consequently, the classes in the Reduced BreizhCrops dataset are barley, wheat, rapeseed, corn, orchards, permanent and temporary meadows. We expect the classifying model to have difficulties at predicting the orchards class, as this crop type groups a mix of trees and grass. Moreover, permanent and temporary meadows are very similar on the spectral level. The difference between the two meadow types is that temporary meadows are planted, and permanent meadows are constantly present. Since our dataset contains data over the span of a year, we do not expect the model to discern them.

3.3.2 Daily Timestamps

As mentioned previously in Section 3.2.4, the BreizhCrops training set consists of time series with a common sequence length of 70, where the elements of the time series are randomly undersampled or padded with zeros. This results in timestamps that do not relate to a day of the year. However, we think that the model could learn from the date when the satellite data are captured. Therefore, for the Reduce BreizhCrops dataset, we decide to use *daily timestamps*, which means that each timestamp in a time series corresponds to a day of the year. In

short, time series contain satellite data on the days they were recorded. When no satellite data are available, the signal is null. An example of a temporary meadows signal is given in Figure 7.

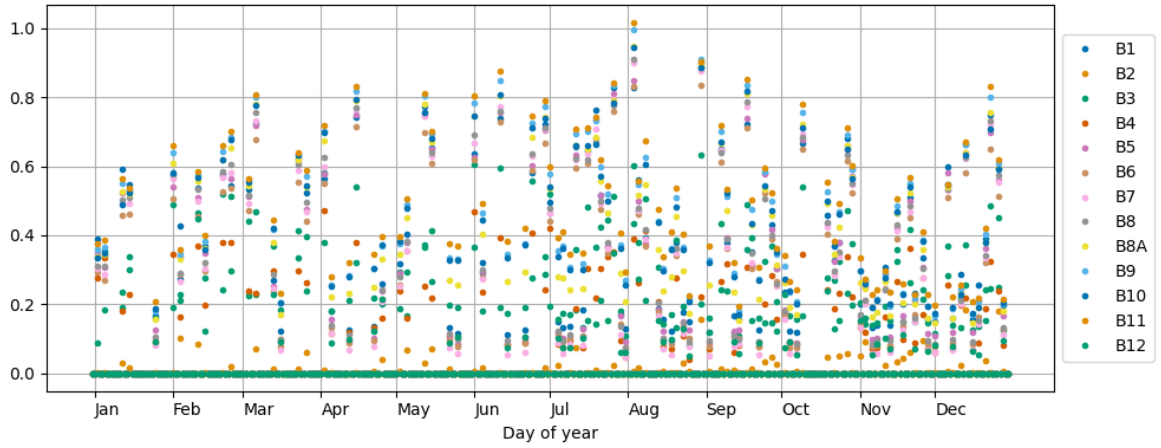


Figure 7: Spectral bands of a temporary meadows signal, processed for the Reduced BreizhCrops dataset. One timestamp in the time series corresponds to one day of the year. The signal is not null only when satellite data is available. This results in a sparse dataset.

4 Methodology

In this section, we introduce the mathematical foundation for earliness classification. We present formal definitions of time series classification, metrics, model architectures, and cost functions.

4.1 Mathematical background

4.1.1 Time Series

This part is mainly inspired by [19, 23], with some adaptations to fit the current task.

Definition 4.1 (Univariate time series)**.** A univariate time series $\boldsymbol{x} = \{x_1, x_2, \ldots, x_T\}$ is a sequence of T real values arranged in ascending order by timestamp $t = 1, 2, \ldots, T$. The value T is called the *sequence length* of $\boldsymbol{x}$.

Definition 4.2 (MTS)**.** An D-dimensional MTS of length T is defined as $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_T\}$, where $\boldsymbol{x}_i \in \mathbb{R}^D$, for $i = 1, \ldots, T$, are feature vectors that represent the spectral dimensions of $\boldsymbol{X}$. Each MTS is associated with a class label $y \in C$, where C is a finite set of class labels.

In this setting, let's call $\mathcal{X}_D^T$ the set of all D-dimensional MTS of length T.

Definition 4.3 (Dataset)**.** A dataset is a collection of N pairs $\{(\boldsymbol{X}_1, y_1), (\boldsymbol{X}_2, y_2), \ldots, (\boldsymbol{X}_N, y_N)\}$ where $\boldsymbol{X}_i$ is a multivariate time series and $y_i \in C$ its corresponding label for $i = 1, \ldots, N$.

In the BreizhCrops training dataset $N = 319'258$, and since there are 13 spectral bands, $D = 13$.

Definition 4.4 (Early subsequence)**.** For an D-dimensional MTS $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_T\}$ of length T, an early subsequence of $\boldsymbol{X}$ of length t is defined as $\boldsymbol{X}_{\to t} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_t\}$.

An early subsequence $\boldsymbol{X}_{\to t}$ is thus a time series starting at the same time as $\boldsymbol{X}$ but containing information until timestamp $t \leq T$. It finishes earlier than $\boldsymbol{X}$.

4.1.2 Classification

Definition 4.5 (Classification of MTS)**.** Given C a finite set of class labels and a dataset of MTS belonging to $\mathcal{X}_D^T$, the task of classifying MTS is to learn a classifier to assign labels to MTS, namely generating a function $f_\theta : \mathcal{X}_D^T \to C$, where θ are the parameters of the model.

In *early* classification, a model needs to classify MTS as soon as possible. In other words, it must predict the label of an early subsequence with the highest accuracy.

Definition 4.6 (Early Classification of Multivariate Time Series (ECMTS))**.** An early classification task of MTS consists of classifying MTS as soon as possible, i.e. generating a classifier $f_\theta : \mathcal{X}_D^T \to C$ such that f_θ can predict from early subsequence $\boldsymbol{X}_{\to t}$ with t as small as possible and with the highest accuracy possible.

The timestamp when the decision is made is called *stopping time*.

Definition 4.7 (Stopping time)**.** In a task of ECMTS, let $f_\theta : \mathcal{X}_D^T \to C$ be the classifier. Then, the stopping time is the timestamp when f_θ outputs the final label. In other words, for $\boldsymbol{X} \in \mathcal{X}_D^T$, the stopping time is defined as $t \in \{1, 2, \ldots, T\}$ such that $f_\theta(\boldsymbol{X}_{\to t})$ outputs the final classification.

To be able to compare models taking both the variables of accuracy and earliness into account, we define some metrics accordingly, as defined in the next section.

4.1.3 Metrics

Definition 4.8 (Earliness of a classification)**.** Let Y be the set of true labels. For each y_i in Y, where $i = 1, \ldots, |Y|$, let T_i be the sequence length of the associated MTS, and t_i the stopping time. The earliness of a classification [3] is defined as

$$Earliness = 1 - \frac{1}{|Y|} \sum_{y_i \in Y} \frac{t_i}{T_i}.$$

From there, we can define the Harmonic Mean between Accuracy and Earliness to measure the quality of a classification [2, 3].

Definition 4.9 (Harmonic Mean between Accuracy and Earliness (HM))**.** The harmonic mean of accuracy and earliness HM of a classification is defined as

$$HM = \frac{2 \cdot Earliness \cdot Accuracy}{Earliness + Accuracy} \tag{1}$$

where the accuracy is the fraction of correct labels over the total number of labels.

The measure HM is widely used to compare models performances [3, 6], though this measure lacks additivity.

Moreover, to compare the distribution of the classification earliness, we define a metric which measures how widely the stopping times are distributed. That is why we define the *Standard Deviation Score*.

Definition 4.10 (Standard Deviation Score)**.** The Standard Deviation Score is the mean over the classes earliness standard deviation. Let σ_c be the standard deviation of the stopping times associated with the true class $c \in C$. Then, the Standard Deviation Score $\tilde{\sigma}$ is defined as

$$\tilde{\sigma} = \frac{1}{|C|} \sum_{c \in C} \sigma_c.$$

4.2 Models

We present the derivation of the model we developed: the D-ELECTS model. First, the ELECTS model is presented in details as it is the basis on which we build our model. Then, our derived model, the daily early classification model D-ELECTS is presented.

4.2.1 ELECTS

The ELECTS model, published in 2023 [34], is the first main inspiration of this master thesis. In this work, Russwurm et al. presented an approach to explicitly optimize a model for the objective of an early classification in remote sensing. As predictions of crop categories from field parcels need to be as early and accurate as possible, they used a loss function that leverages both earliness and accuracy. They evaluate their model performance on four datasets, including BreizhCrops.

The ELECTS model consists of a deep learning feature extractor with two decision heads. The loss function is designed such that it optimizes both the accuracy and the earliness of the predictions. A simple diagram in Figure 8 shows how the model is functioning to output predictions.

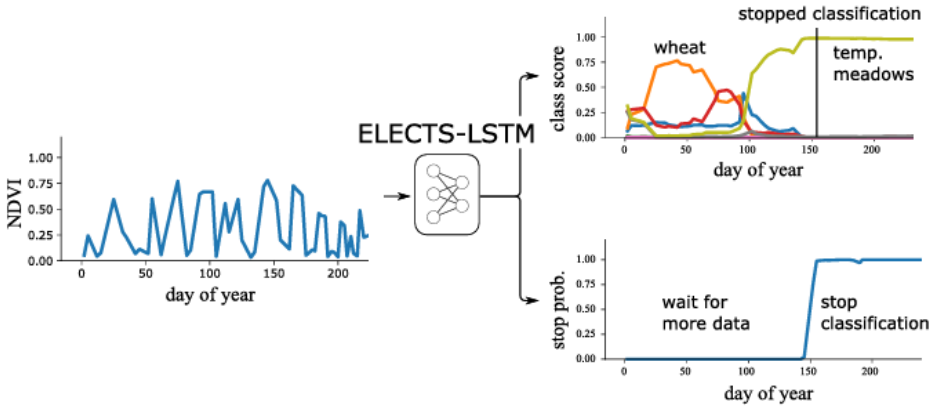


Figure 8: Output of an early classification model trained using ELECTS. This model processes a time series incrementally, incorporating one data point at a time. It calculates a probability distribution across different crop categories (displayed in the top right) and concurrently assesses the probability of stopping further data intake (shown at the bottom right). The model continues to intake and analyze more data points as long as the stopping probability is deemed insufficient, indicating the need for additional information to achieve precise classification. Figure taken from [34].

4.2.1.1 ELECTS model

In the ELECTS model, the architecture combines a deep learning feature extractor with two decision-making heads. The deep learning feature extractor, f_{θ_h}, ingests time series data one observation at a time, and outputs an embedding of the sequence. We will refer to this model as the *backbone model*.

By its recurrent nature, the chosen backbone model is a Recurrent Neural Network (RNN). More formally, at given time t and early subsequence $\boldsymbol{X}_{\to t} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_t\} \in \mathbb{R}^{t \times D}$, which are the observations up to the image acquisition at time t, the backbone model estimate the hidden representation $\boldsymbol{h}_t$, of dimension $N_{HiddenDims}$. As a recurrent model, it can process time series of different sequence lengths. It is defined as

$$\begin{aligned} f_{\theta_h} \colon \mathbb{R}^D \times \mathbb{R}^{N_{HiddenDims}} &\longrightarrow \mathbb{R}^{N_{HiddenDims}} \\ (\boldsymbol{x}_t, \boldsymbol{h}_{t-1}) &\longmapsto f_{\theta_h}(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}) = \boldsymbol{h}_t \end{aligned}$$

As a feature extractor, the RNN is a natural choice, as it projects a variable-length input sequence to a fixed-size representation. To avoid vanishing gradient, the Long Short-Term Memory (LSTM) [18] was chosen.

Further, two linear decision heads update the output $\boldsymbol{h}_t$. One head produces a classification probability for each class

$$\hat{\boldsymbol{y}}_t = \text{softmax}\left(f_{\theta_c}(\boldsymbol{h}_t)\right) \tag{2}$$

and the second one outputs a scalar probability of stopping the classification decision

$$\hat{d}_t = \sigma(f_{\theta_d}(\boldsymbol{h}_t)).$$

The symbol σ denotes the sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$, which scales the output of the stopping head to a value between 0 and 1. During training and testing, $\hat{d}_T = 1$ irrespectively of the model output, to make sure that the model has taken a stopping decision.

During test time, the stopping probability $\hat{d}_t$ works as follows. At iteration t, the classification is stopped with a stopping probability of $\hat{d}_t$. Respectively, the classification continues with probability $1 - \hat{d}_t$. In the latter case, the model predicts the next embedding $\boldsymbol{h}_{t+1}$ as well as the decision heads outputs $\hat{\boldsymbol{y}}_{t+1}$ and $\hat{d}_{t+1}$, and the same process is repeated.

From there, the stopping time is the timestamp when the decision is made, without any decision made in the past. Thus the probability that the model stops predicting at t_{stop} is:

$$P(t_{stop} = t) = \hat{d}_t \prod_{i=1}^{t-1} (1 - \hat{d}_i).$$

Overall, the ELECTS model can be defined as

$$\begin{aligned} f_\theta \colon \mathbb{R}^D \times \mathbb{R}^{N_{HiddenDims}} &\longrightarrow \mathbb{R}^t \times \mathbb{R}^{|C|} \\ (\boldsymbol{x}_t, \boldsymbol{h}_{t-1}) &\longmapsto (\hat{\boldsymbol{d}}_{\to t}, \hat{\boldsymbol{y}}_t) \end{aligned}$$

where $\theta = (\theta_h, \theta_c, \theta_d)$, $\hat{\boldsymbol{d}}_{\to t} = (\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_t)$ and the hidden state $\boldsymbol{h}_0$ is set as the null vector.

4.2.2 D-ELECTS Model

From the ELECTS model, we design the D-ELECTS model. The D-ELECTS model relies on the number of timestamps left until the final prediction is made, instead of the probability of stopping. This model architecture is motivated by the fact that the probability of stopping was not necessarily increasing with respect to the day of the year, which was an issue. Moreover, from a practical and agronomic point of view, knowing the number of timestamps/days left until the prediction is more convenient than knowing the probability of stopping.

In terms of the model architecture, the only difference is that the second linear decision head does not output a scalar probability of stopping the classification decision, but rather outputs the number of timestamps left until the final decision, which varies from 0 to T. More formally, the number of timestamps left z_t is computed as

$$z_t = T \cdot \sigma(f_{\theta_d}(\boldsymbol{h}_t)).$$

Thus the overall D-ELECTS model is defined as

$$\begin{aligned} f_\theta \colon \mathbb{R}^D \times \mathbb{R}^{N_{HiddenDims}} &\longrightarrow \mathbb{R}^t \times \mathbb{R}^{|C|} \\ (\boldsymbol{x}_t, \boldsymbol{h}_{t-1}) &\longmapsto (\boldsymbol{z}_{\to t}, \hat{\boldsymbol{y}}_t) \end{aligned}$$

where $\theta = (\theta_h, \theta_c, \theta_d)$, $\boldsymbol{z}_{\to t} = (z_1, z_2, \ldots, z_t)$ and the hidden state $\boldsymbol{h}_0$ is set as the null vector. A simple diagram explains the outputs of the D-ELECTS model, in Figure 9.

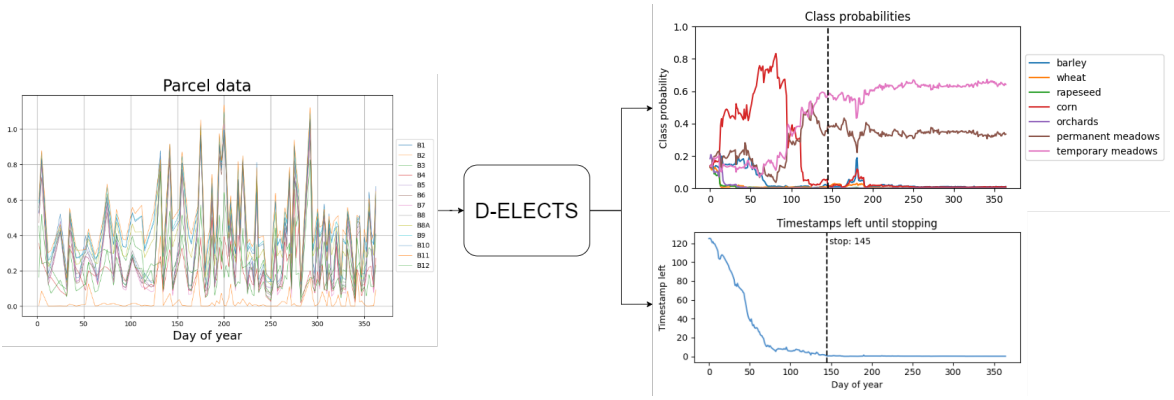


Figure 9: Input and output of the D-ELECTS model. The input is the average Sentinel-2 reflectance aggregated by parcel, with 13 spectral bands. The model processes incrementally the MTS. It outputs the probability distribution across the crop types (displayed in the top right) and the number of timestamps left until the model stops predicting (shown at the bottom right). The model continues to intake more data points as long as the timestamps left are above zero. The D-ELECTS model predicts the class temporary meadows at a stopping time of 145, indicated as a vertical black line.

4.3 Cost functions

In an early classification problem, the model needs to leverage accuracy and earliness. That is naturally reflected on the cost function. On the one hand, one part of the cost function should focus on pushing the model to give the correct label to the MTS, focusing on the accuracy. On the other hand, a second part should focus on rewarding the model when giving an early correct answer. This part focuses on the earliness of the classification.

Both of the previous costs are expressed in the same unit and convey characteristics of the application domain. They might take more variables as input.

Now, let's focus on more specific cost functions. The ELECTS loss function from [34] paper is first presented, followed by the D-ELECTS cost function that we developed.

4.3.1 ELECTS Cost Function

At each time stamp t, the *classification, earliness-rewarded loss* L_{CER} is calculated:

$$L_{CER}(\hat{\boldsymbol{y}}_t, \boldsymbol{y}, t) = \alpha L_C(\hat{\boldsymbol{y}}_t, \boldsymbol{y}) - (1-\alpha) R_e(\hat{\boldsymbol{y}}_t, \boldsymbol{y}, t).$$

The *misclassification loss* is the negative log-likelihood

$$L_C(\hat{\boldsymbol{y}}_t, \boldsymbol{y}) = -\sum_{c=1}^{C} y_c \log \hat{y}_{c,t}, \tag{3}$$

where $\hat{\boldsymbol{y}} \in \mathbb{R}^C$ is as introduced in equation (2) and $\boldsymbol{y} \in \mathbb{R}^C$ is a one-hot vector of $|C|$ classes, i.e. $y_c = 1$ if c is the true label, otherwise 0. This part of the cost function focuses on the accuracy of the model. The earliness reward, defined as

$$R_e(\hat{\boldsymbol{y}}_t, \boldsymbol{y}, t) = \sum_{c=1}^{C} y_c \hat{y}_{c,t} \left(\frac{T-t}{T} \right),$$

focuses on the earliness of the classification. It decreases linearly the closer t gets to T. Moreover, the linear term is scaled with the probability of the correct class $\sum_{c=1}^{C} y_c \hat{y}_{c,t}$. That way, this term applies the reward only if the probability of the correct class is large.

Both terms $L_C(\hat{\boldsymbol{y}}_t, \boldsymbol{y})$ and $R_e(\hat{\boldsymbol{y}}_t, \boldsymbol{y}, t)$ are weighted with $\alpha \in [0, 1]$ an hyper-parameter, which trades off accuracy and earliness reward.

For now, only the accuracy and the earliness reward are present, for an exact timestamp. Let's add a term that includes the probability of stopping. Let's define $D(\hat{\boldsymbol{d}}_{\rightarrow t})$, the joint probability of making a decision at time t and not having made a decision before:

$$D(\hat{\boldsymbol{d}}_{\rightarrow t}) = \hat{d}_t \prod_{i=1}^{t-1} (1 - \hat{d}_i) + \frac{\epsilon}{T}.$$

A small constant offset $\frac{\epsilon}{T}$ is added, as it was shown that without, the model tended to fall in local minima [34].

Now, for each time stamp t, a joint expression of accuracy, earliness reward L_{CER} and explicit earliness D is computed :

$$L_{ELECTS}(\hat{\boldsymbol{d}}_{\rightarrow t}, \hat{\boldsymbol{y}}_t, \boldsymbol{y}, t) = D(\hat{\boldsymbol{d}}_{\rightarrow t}) L_{CER}(\hat{\boldsymbol{y}}_t, \boldsymbol{y}, t).$$

Finally, the overall objective is defined over all pairs $(\boldsymbol{X}, \boldsymbol{y})$ in the dataset, over all time stamps, namely

$$\sum_{(\boldsymbol{X},\boldsymbol{y})} \sum_{t=1}^{T} L_{ELECTS}(\hat{\boldsymbol{d}}_{\rightarrow t}, \hat{\boldsymbol{y}}_t, \boldsymbol{y}, t). \tag{4}$$

The learnable parameters $\theta = (\theta_h, \theta_c, \theta_d)$ of the backbone model and the two decision heads are computed by minimizing the objective

$$\arg\min_{\theta} \sum_{(\boldsymbol{X},\boldsymbol{y})} \sum_{t=1}^{T} L_{ELECTS}(\underbrace{f_\theta(\boldsymbol{x}_t, \boldsymbol{h}_{t-1})}_{(\hat{\boldsymbol{d}}_{\rightarrow t}, \hat{\boldsymbol{y}}_t)}, \boldsymbol{y}, t).$$

A schematic view of the model and the loss is shown in Figure 10.

Another way to write the overall loss (4) is by rearranging the terms as follows

$$\sum_{(\boldsymbol{X},\boldsymbol{y})} \sum_{t=1}^{T} L_{ELECTS}(\hat{\boldsymbol{d}}_{\rightarrow t}, \hat{\boldsymbol{y}}_t, \boldsymbol{y}, t) = \sum_{(\boldsymbol{X},\boldsymbol{y})} \sum_{t=1}^{T} \alpha D(\hat{\boldsymbol{d}}_{\rightarrow t}) L_C(\hat{\boldsymbol{y}}_t, \boldsymbol{y}) - (1-\alpha) D(\hat{\boldsymbol{d}}_{\rightarrow t}) R_e(\hat{\boldsymbol{y}}_t, \boldsymbol{y}, t) \tag{5}$$

$$= \sum_{(\boldsymbol{X},\boldsymbol{y})} \underbrace{\alpha \sum_{t=1}^{T} D(\hat{\boldsymbol{d}}_{\rightarrow t}) L_C(\hat{\boldsymbol{y}}_t, \boldsymbol{y})}_{(1)} + \underbrace{(\alpha - 1) \sum_{t=1}^{T} D(\hat{\boldsymbol{d}}_{\rightarrow t}) R_e(\hat{\boldsymbol{y}}_t, \boldsymbol{y}, t)}_{(2)} \tag{6}$$

where (1) can be interpreted as the misclassification cost function, pushing the model to predict labels correctly and (2) the delay cost function, rewarding the earliness of correct prediction.

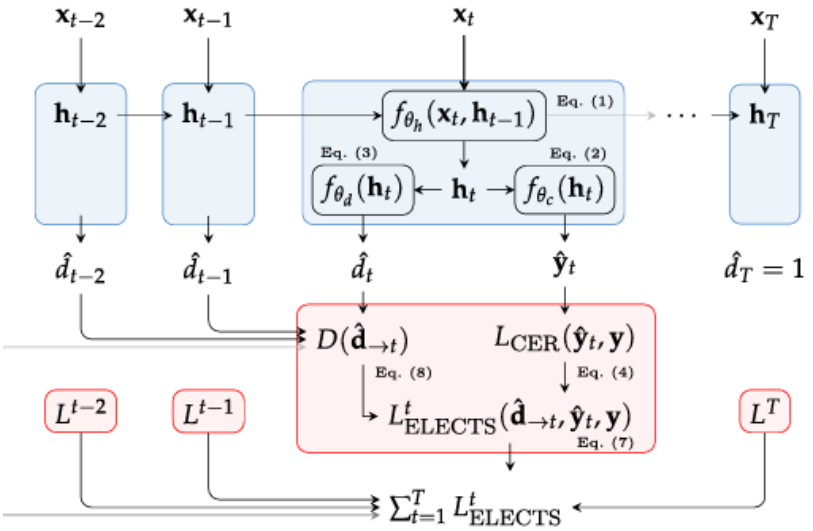


Figure 10: Schematic illustration of the model architecture and loss functions outlined. The diagram uses blue to represent the model components and red for the losses, with arrows showing the direction of data flow through the functions. The neural network is symbolized as f_θ, where θ represents the model parameters. During the inference phase, with θ set to their learned values, the model can analyze a time series data up to any given point t. For the training phase, loss calculations are performed over the entire duration of the time series, concluding at the final time step T. The figure is taken from [34], thus the equation references correspond to the ELECTS paper's equations numbering.

4.3.1.1 Implementation details

The implementation in Python can be found on the author's Github as well as on the master thesis Github repository. An initial layer projects the original input vector to a learned 64-dimensional feature representation at each time, followed by two mono-directional LSTM layers. The overall model has 67'108 trainable

parameters. The Adam optimizer was used, with a learning rate of 0.001 and a dropout of 20%. The model was trained with a batch size of $B = 256$. As explained in Section 3.2.4, a common sequence length of 70 was used during training; if the time series was initially shorter, it is padded with zeros, and if it was longer, it is randomly undersampled. At test time, the complete variable-length time series are given as input (the sequence length was 150). The offset parameter ϵ is set to 10 and the weight $\alpha = 0.5$. During training, if the test loss has not decreased in the last 30 epochs, the model stops. The final model is the one saved at the minimum of the test loss.

4.3.2 D-ELECTS Cost Function

Here we present the cost functions of the D-ELECTS model that we developed. As stated in Section 4.2.2, z_t is the number of timestamps left before stopping the prediction. This value replaces the "stopping probability" of the ELECTS model. We developed four parts of the D-ELECTS cost function. Let's depict them one by one.

4.3.2.1 Misclassification Cost

We set the misclassification cost to be the same as the ELECTS in (7), i.e.

$$C_m(\hat{\mathbf{y}}|\mathbf{y}) = -\sum_{t=1}^{T}\sum_{c=1}^{C} y_c \log \hat{y}_{c,t}. \tag{7}$$

This cost needs to be as small as possible. The predicted label is at each time t, because we would like the model to have the flexibility to learn the classification at each timestamp.

4.3.2.2 Earliness Reward

The earliness reward, opposed to the misclassification cost, depends on the predicted label at the final time $t + z_t$:

$$C_d(\mathbf{z}, \hat{\mathbf{y}}|\mathbf{y}) = -\sum_{t=1}^{T}\sum_{c=1}^{C} y_c \hat{y}_{c,t+z_t}\left(1 - \frac{z_t}{T}\right)\left(1 - \frac{t}{T}\right) \tag{8}$$

where $\mathbf{z} = (z_1, z_2, \ldots, z_T)$ the vector of the predicted timestamps left. The term $\sum_{c=1}^{C} y_c \hat{y}_{c,t+z_t}$ ensures that reward is given when the classification is correct at the final timestamp $t + z_t$. Moreover, the reward must decrease over time, thus the factor $\left(1 - \frac{t}{T}\right)$. Finally, a bigger reward is given when z_t gets closer to zero, thus the term $\left(1 - \frac{z_t}{T}\right)$. Overall, the earliness reward pushes z_t to be small.

4.3.2.3 Wrong-Prediction Penalty

The earliness reward was defined from the idea that earliness should be rewarded in the loss function. From there, we could also define the *wrong-prediction penalty*, which penalizes wrong predictions. Two wrong-prediction penalties were tested, with both the aim to penalize wrong predictions.

First Wrong-Prediction Penalty (v1)

The idea behind the first version is to penalize wrong predictions if they occur early in the year. That way, the timestamps left should become bigger if the classification is incorrect. More formally, the first version is given by

$$C_p(\mathbf{z}, \hat{\mathbf{y}}|\mathbf{y}) = \sum_{t=1}^{T}\sum_{c=1}^{C}(1 - y_c)\hat{y}_{c,t+z_t}\left(1 - \frac{z_t}{T}\right)\left(1 - \frac{t}{T}\right). \tag{9}$$

The term $(1 - y_c)$ selects the incorrect classes and $\hat{y}_{c,t+z_t}$ the final prediction of the model. Then, $\left(1 - \frac{t}{T}\right)$ makes the penalty bigger for t small, similar to $\left(1 - \frac{z_t}{T}\right)$ making the penalty bigger for z_t small. This cost function pushes z_t to be bigger for early wrong predictions.

Second Wrong-Prediction Penalty (v2)

The second version aims to penalize wrong predictions if they occur late in the year. That way, if a wrong prediction happens at the end of the year, better give it sooner so that the earliness of the model is better. Indeed, the model could be uncertain no matter how much information it receives. We expect the model to discriminate some classes like orchards, permanent or temporary meadows, as mentioned in section 3.3. We want this ambiguity to be reflected as early as possible.

Thus the second version of wrong-prediction penalty is given by

$$C_p(\boldsymbol{z}, \hat{\boldsymbol{y}}|\boldsymbol{y}) = \sum_{t=1}^{T}\sum_{c=1}^{C}(1-y_c)\hat{y}_{c,t+z_t}\left(\frac{t+z_t}{T}\right). \tag{10}$$

The same motivations as for the first version are behind the terms $(1-y_c)$ and $\hat{y}_{c,t+z_t}$. Concerning the term $\left(\frac{t+z_t}{T}\right)$, it aims to enforce the penalty to be bigger for $t+z_t$ big. In other words, the penalty becomes bigger as the prediction is close to the end of the year.

4.3.2.4 Piece-wise linear regression for z_t

The timestamps left z_t take values between 0 and T. Moreover, suppose that the classification should be terminated at a certain date common for each sample sharing the same label. This supposition is motivated by the phenological events happening at the same time for specific kind of crops in the same area. For example, rapeseed blooms around May and exposes its particular yellow flower during that time. We then expect the model to recognize this crop around the 130 day of the year, when this phenological event happens.

Thus, the final classification should be given around a *targeted stopping time* $\mu_c \in \{0, 1, \dots, T\}$, for each class $c \in C$. In the case of rapeseed, μ_c should have a value around 130. In other words, z_t should reach the value 0 around μ_c. If we moreover suppose that z_t should decrease linearly until then, it would only make sense to push z_t to be piece-wise linear:

$$z_t = \begin{cases} \mu_c - t & \text{if } t \leq \mu_c \\ 0 & \text{otherwise} \end{cases}$$

From there, we formulate the *piece-wise linear regression cost* for z_t

$$C_{lr}(\boldsymbol{z}, \boldsymbol{y}) = \sum_{c=1}^{C} y_c \left(\sum_{t=1}^{\mu_c} \left(\frac{\mu_c - t - z_t}{T} \right)^2 + \sum_{t=\mu_c+1}^{T} \left(\frac{z_t}{T} \right)^2 \right). \tag{11}$$

Let's break down this formula. The term $\sum_{c=1}^{C} y_c$ ensures that we are looking one class at a time. Then, the first part of the sum $\sum_{t=1}^{\mu_c} \left(\frac{\mu_c - t - z_t}{T}\right)^2$ pushes z_t to decrease linearly and reach the value zero at $t = \mu_c$. The second part of the sum $\sum_{t=\mu_c+1}^{T} \left(\frac{z_t}{T}\right)^2$ pushes z_t to stay at zero afterwards.

Note that there is no term with $\hat{\boldsymbol{y}}_{c,t}$, the predicted label at time t. The reason is that we want z_t to be piece-wise linear whether or not the prediction is correct. Not including this term also enforces the standard deviation score to be smaller, as we assume a common stopping time for each sample of the same class.

Computing the Targeted Stopping Time μ_c

First, we would like our model to be able to predict crops between May and July. That is why we set $\mu_c = T * 150/365$ for $c \in C$, where 150 corresponds to the day of the year beginning of June. Then, through the training, the value μ_c is determined looking at the overall classification probability of the classes.

Let's define $p_{thresh} \in (0, 1)$ the probability threshold. After first training the classification only for T_c epochs, we have the probabilities of classification over time, per class. Then, looking at the classification probabilities of a certain true label, we have probabilities over time for each class. If the mean of the correct class c is bigger than p_{thresh} at timestamp t, then μ_c is set at this timestamp. In Figure 11, the class probabilities are shown over time, with μ_c plotted as a vertical red line, when the probability is going over p_{thresh}.

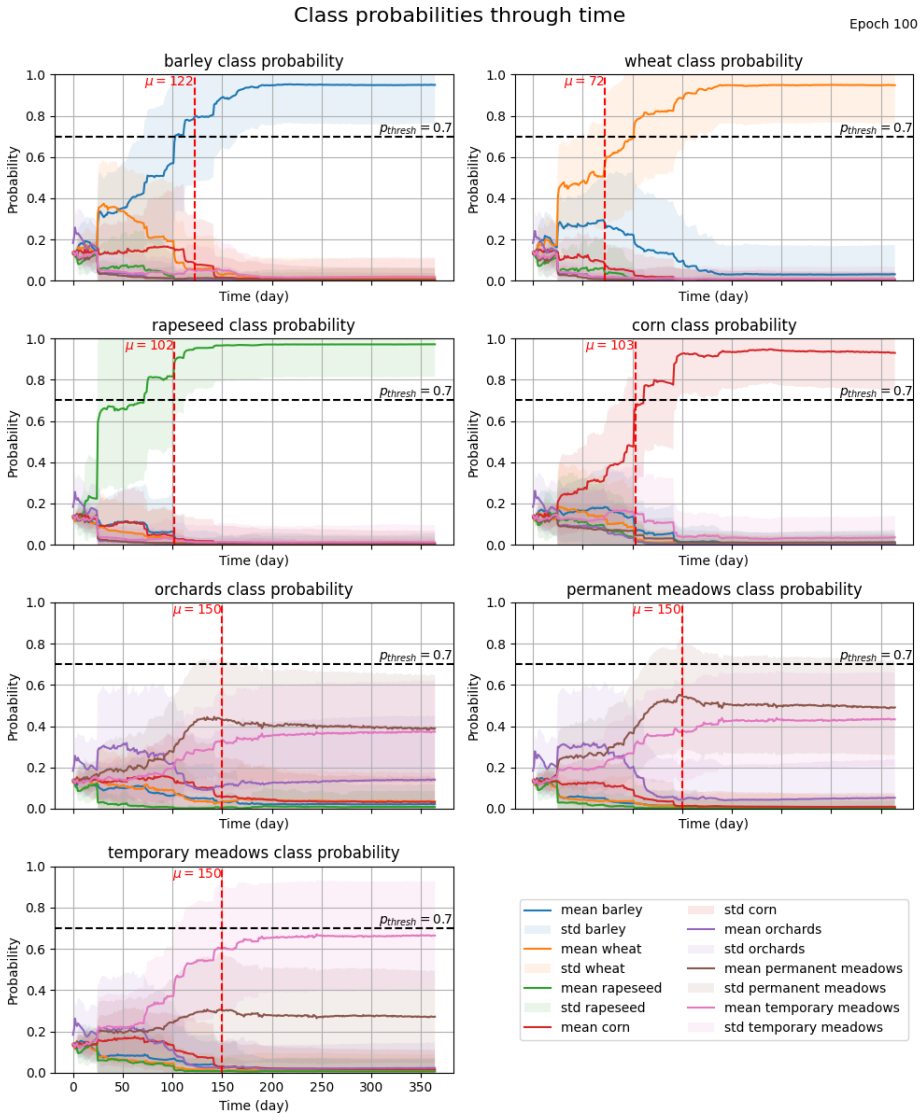


Figure 11: Class probabilities through time, with $p_{thresh} = 0.7$ and μ_c indicated by vertical red lines, for $c \in C$.

4.3.2.5 General D-ELECTS Cost Function

Finally, from the four parts above, the general D-ELECTS cost function is given by the linear combination

$$\frac{1}{|Y|} \sum_{y_i \in Y} \alpha_1 C_m(\hat{\boldsymbol{y}}_i | \boldsymbol{y}_i) + \alpha_2 C_d(\boldsymbol{z}_i, \hat{\boldsymbol{y}}_i | \boldsymbol{y}_i) + \alpha_3 C_p(\boldsymbol{z}_i, \hat{\boldsymbol{y}}_i) + \alpha_4 C_{lr}(\boldsymbol{z}_i, \boldsymbol{y}_i) \tag{12}$$

where $\boldsymbol{z}_i = \{z_{i,1}, z_{i,2}, \ldots, z_{i,T}\}$ is the predicted timestamp left vector associated to the true label $\boldsymbol{y}_i$, and α_j for $j = 1, \ldots, 4$ are hyper-parameters such that $\sum_{j=1}^{4} \alpha_j = 1$. In short the general D-ELECTS cost function can be summarized as the linear combination of

1. the misclassification cost $C_m(\hat{\boldsymbol{y}}_i | \boldsymbol{y}_i)$ from equation (7), which assures the accuracy of the classification,
2. the earliness reward $C_d(\boldsymbol{z}_i, \hat{\boldsymbol{y}}_i | \boldsymbol{y}_i)$ from equation (8), which guarantees the earliness of the prediction by pushing the timestamps left to be smaller for correct predictions,
3. the wrong-prediction penalty $C_p(\boldsymbol{z}_i, \hat{\boldsymbol{y}}_i)$, which penalizes wrong predictions. It has two versions:
   (a) from equation (9) which pushes the timestamps left to be bigger for early wrong predictions,
   (b) from equation (10) which assures the earliness of predictions, by penalizing wrong predictions when happening late in the year,
4. the piece-wise linear regression cost $C_{lr}(\boldsymbol{z}_i, \boldsymbol{y}_i)$ from equation (11), which ensures the timestamps left to be piece-wise linear with respect to time.

General D-ELECTS Absolute Cost Function

To have a better idea of the weights of each term in the linear combination (12), we define the General D-ELECTS Absolute Cost Function, which is the sum of the linear combination of the absolute costs:

$$\frac{1}{|Y|}\sum_{y_i \in Y} |\alpha_1 C_m(\hat{\boldsymbol{y}}_i|\boldsymbol{y}_i)| + |\alpha_2 C_d(\boldsymbol{z}_i, \hat{\boldsymbol{y}}_i|\boldsymbol{y}_i)| + |\alpha_3 C_p(\boldsymbol{z}_i, \hat{\boldsymbol{y}}_i)| + |\alpha_4 C_{lr}(\boldsymbol{z}_i, \boldsymbol{y}_i)|. \tag{13}$$

This quantity will help analyze the importance of each term in the general cost. Each term $|\alpha_j C_j|$, where C_j is the cost and α_j its associated hyper-parameter, can be plotted and compared to the other terms. For example, see Figure 20.

Implementation Details

The D-ELECTS model was trained on the dataset with original sequence length of 102, the reduced BreizhCrops dataset, see Section 3.3. The training is done on the full time series. Furthermore, since the classes are unbalanced in this dataset, weights which are inversely proportional to the classes size were added to each costs in the general D-ELECTS cost function (12).

The training starts only with $\alpha_1 = 1$, for T_c epochs. When the parameter T_c is reached, then the parameter α_1 decreases linearly towards a value $\alpha_{min} < 1$. The hyper-parameters α_2, α_3 and α_4 are computed as a certain percentage of $1 - \alpha_1$. The intuition behind this procedure is that the model first learns how to classify the classes, until T_C epoch. Then, it learns how to predict earlier, as the hyper-parameters in front of the earliness reward, wrong-prediction penalty and piece-wise linear regression grow bigger.

Moreover, μ_c is updated every 5 epochs, as explained in Equation 4.3.2.4. That way, the model, which is pushed to learn earlier by the earliness reward, can adjust its linear regression.

In short, the parameters that were optimized are $\alpha_{min} \in \{0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$, the percentages associated to α_2, α_3 and α_4, the batch size in $\{128, 256, 512\}$, the total number of epochs in $\{100, 200\}$, the number of hidden dimensions $N_{HiddenDims} \in \{16, 32, 64, 128\}$, the probability threshold $p_{thresh} \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$ and the number of epochs after which the classifier starts to learn the earliness $T_c \in \{2, 10, 20, 40\}$. Finally the initial value of the bias of the decision head which outputs the timestamp left could take the default value, 1 or 5. This last parameter was varied because we wanted a big value for the timestamp left at the beginning of the training.

The fixed parameters are the learning rate at 0.001 and the sequence length at 365.

Figure 12 depicts the training phase schematically.

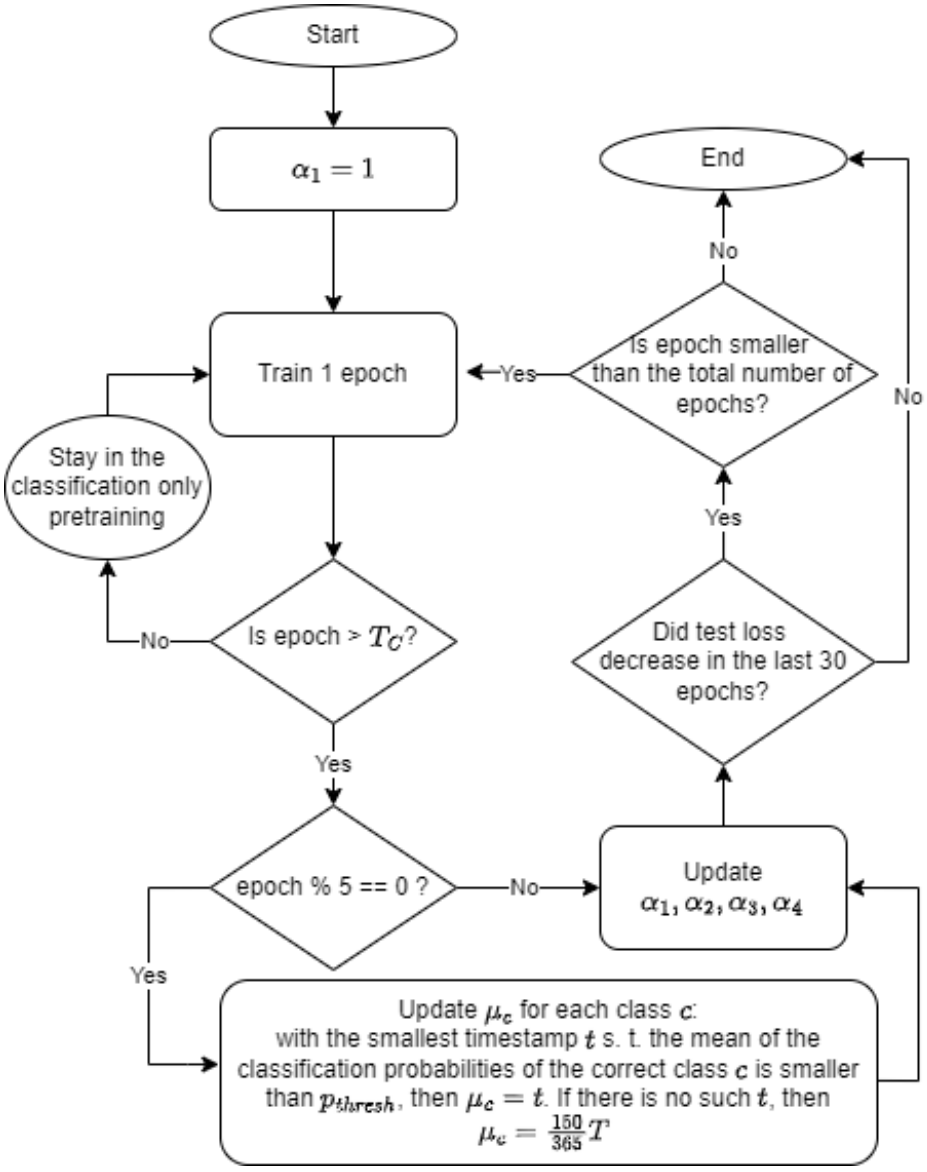


Figure 12: Chart-flow of the training phase of the D-ELECTS model.

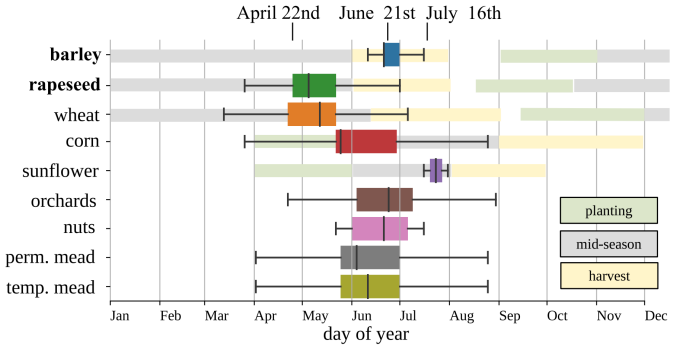
5 Results

This section presents the results obtained by the ELECTS and D-ELECTS models described in Section 4. The results are separated in three parts. First, we present the ELECTS results and a reproduction of them in Section 5.1. This part will serve as benchmark to compare the D-ELECTS model with. Secondly, the D-ELECTS results are depicted in Section 5.2 through some development steps. We select the best D-ELECTS model at the end of this section, with the help of the metrics introduced in Section 4.1.3.

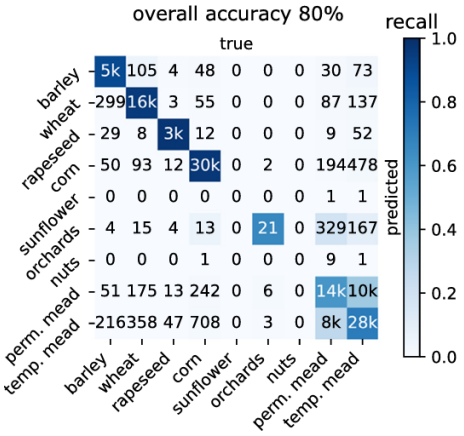
For the last part of the results, Section 5.3 compares the performances of the original ELECTS model with the D-ELECTS model that we developed.

5.1 ELECTS Results

From the results of ELECTS paper, we keep the important graph shown in Figure 13a, representing the predicted stopping times, and the confusion matrix in Figure 13b, which shows the overall accuracy of the model. The ELECTS model confuses the permanent and temporary meadows because they are both meadows, which was expected by the nature of the crop type. For small classes such as sunflowers and nuts, there are no true labels plotted.



(a) Quantitative assessment of stopping times for different crop types in Brittany, France, combined with data on planting, mid-season, and harvesting periods from the USDA Foreign Agricultural Service's crop calendar for France [34].



(b) Accuracy and earliness of predictions for each class on on BreizhCrops are evaluated. The ELECTS model delivers highly accurate predictions for most classes, using only a portion of the full time series length in both datasets. [34].

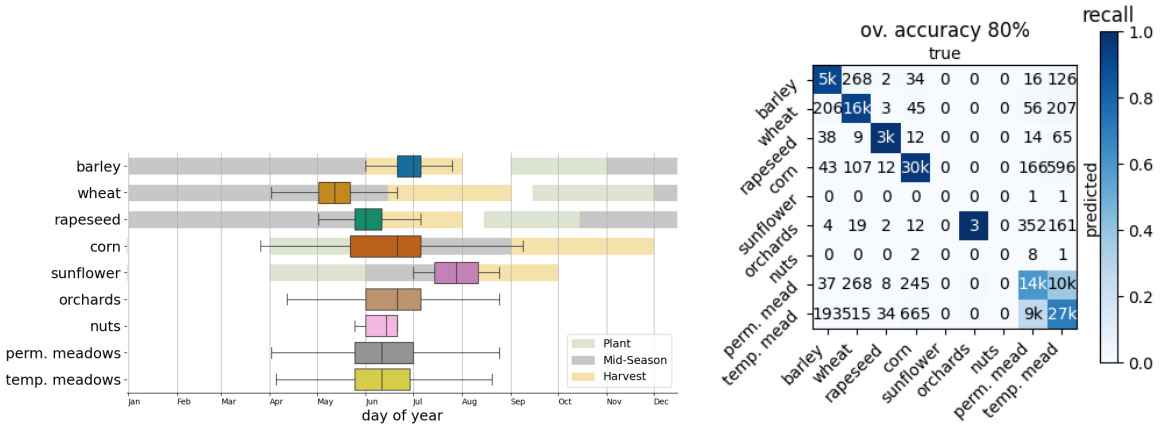
Figure 13: Results of ELECTS paper [34], directly taken from the original paper.

5.1.1 Reproduction of ELECTS Results

We reproduce the results from ELECTS paper to use the performance of the model as a benchmark. We train the model with the same parameters as presented in the paper, with a different random seed. Moreover the stopping times might differ due to the binomial decision taken at each timestamp. The different decisions, which depend on the decision probabilities, can especially be observed on smaller classes like sunflower and nuts. We obtain the boxplots of stopping times and the confusion matrix as shown in Figure 14. We compute the performance metrics and summarize them in Table 4. The earliness is about 85%, which is pretty high. This means that on average, with only 15% of the time series, the classification stops. In addition, the standard deviation score is about 7.05, which is small compared to the common sequence length being 150. However, this small quantity can be explained by the sequence length variation between 51 and 102.

Table 4: Performances of the reproduced ELECTS model on the test set.

Model	Accuracy	Earliness	Harmonic Mean	STD score
ELECTS	0.80	0.85	0.83	7.05



(a) Quantitative assessment of stopping times for different crop types in Brittany, France. Planting, mid-season and harvesting periods according to the USDA Foreign Agricultural Services crop calendar for France are also plotted.

(b) Confusion matrix of BreizhCrops dataset.

Figure 14: Reproduction of the results of ELECTS paper.

5.2 D-ELECTS Results

In this section, the development stages of the D-ELECTS model are presented. First, in Section 5.2.1 we focus only on classification to check that the D-ELECTS model can potentially classify labels correctly, without taking earliness into account. Then, Section 5.2.2 introduces all four parts of the general cost function into training and a comparison of the performances of the D-ELECTS model with the two versions of the wrong-prediction penalties are presented. From there, the best model we could develop was selected according to the comparison presented in paragraph 5.2.2.1. The selected model's performances are depicted in Section 5.2.3 with graphs.

5.2.1 Classification Only Pretraining

Before training the model with the general loss introduced in equation (12), we only trained the D-ELECTS model with the misclassification cost (7). That way, we ensured that the model can predict accurate results on the Reduced BreizhCrops dataset, before focusing on the earliness. If the model can actually learn classification only, then it will go through a pretraining of T_C epochs and learn earliness afterwards (see paragraph 4.3.2.5).

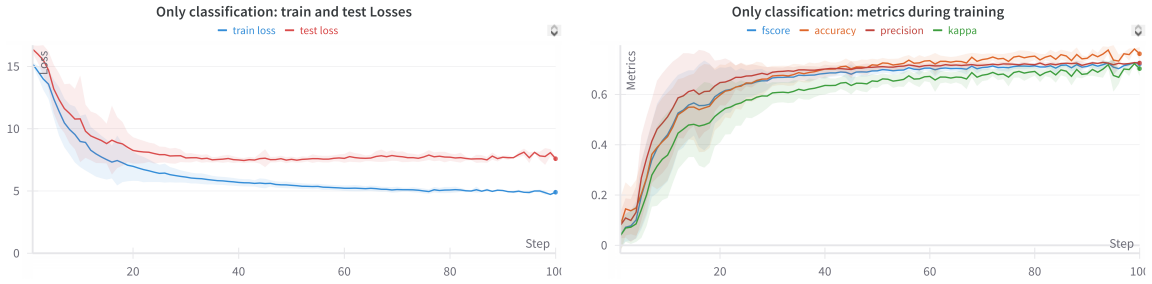
The backbone model and the classification head are inspired by the ELECTS model, which could achieve early classification in the BreizhCrops dataset. Thus, the ELECTS model should easily achieve end-to-end classification on the original dataset. The uncertainty of the classification only pretraining is the model's performance on the Reduced BreizhCrops dataset, which is derived from the BreizhCrops dataset.

Let's fix $\alpha_1 = 1$ and $\alpha_2 = \alpha_3 = \alpha_4 = 0$ through the whole training. Concerning the hyper-parameters, only the bias of the decision head predicting the timestamps left is varied. Figure 15 presents the train and test losses through the training, as well as the metrics measuring the quality of the classification. The train loss decreases through the training.

Now, let's select the trained model with the highest accuracy and depict its performances on the test dataset. The model obtained an accuracy of 68.8% and a f1-score of 68.9% on the test dataset. The normalized confusion matrix is presented in Figure 16. On one hand, the normalized scores for the classes barley, wheat, rapeseed, and corn are over 93%. On the other hand, orchards, permanent and temporary meadows have scores of 62.7%, 33.4% and 56.4% respectively. The model mostly mixed these three crop categories together.

To study how the model recognize the classes through time, the class probabilities are plotted through time in Figure 17. More precisely, each subplot corresponds to a specific class. Then, the mean and the standard deviation of the predicted classification probabilities are computed for each class and timestamp. The result is a subplot with $|C|$ means and standard deviations plotted over time.

Again we see that for crops barley, wheat, rapeseed, and corn, the mean probability of the same class goes above 0.8 percent at some point during the year. For classes orchards, permanent and temporary meadows, the mean probability does not go over 0.6. On the subplot of permanent meadows class probability, we clearly see the mean probabilities of these three classes being close to each other through time.



(a) Mean and standard deviation of the train and test losses during training.

(b) Mean and standard deviation of the metrics on the validation dataset during training.

Figure 15: D-ELECTS model trained on classification task only. The losses and metrics are plotted with respect to the epoch (step). The full lines are the means, whereas the shaded parts represent the standard deviation.

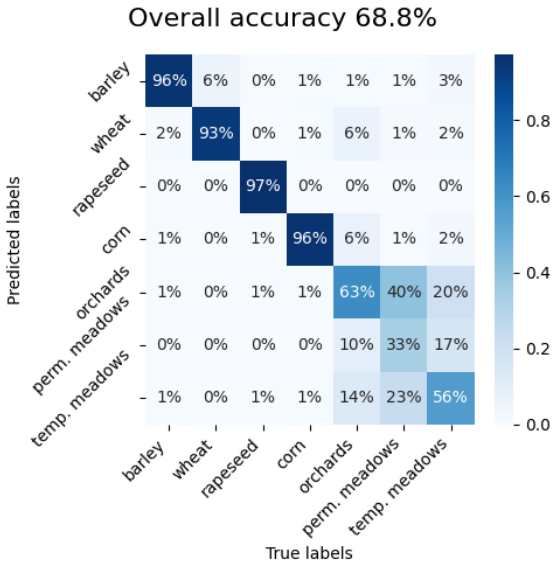


Figure 16: Confusion matrix of the prediction of the model trained only on classification, on the test dataset.

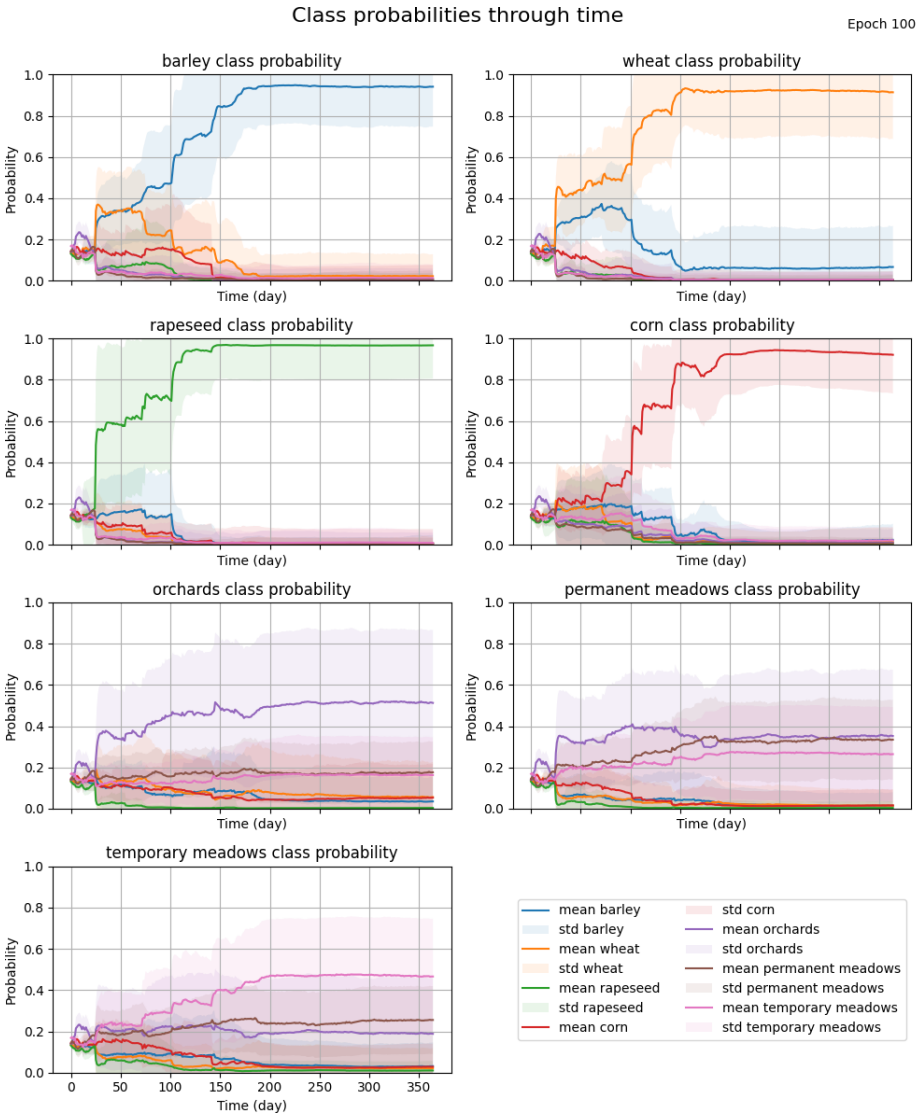


Figure 17: Class probabilities through time, for each class $c \in C$. Each subplot represents the mean and standard deviation of the classification probabilities for a certain true label, written on top of the subplot. The legends of the classes are written at the bottom right of the figure.

5.2.2 Comparison between the Two Wrong-Prediction Penalties

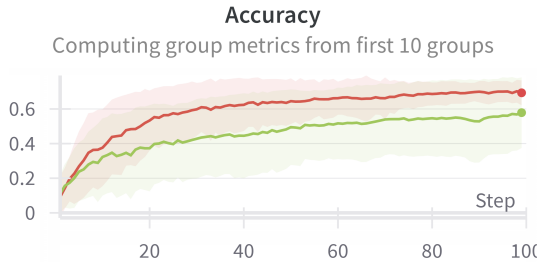
Let's now train the D-ELECTS model on the whole general D-ELECTS cost function from equation (12) and compare the performance of the models depending of the wrong-prediction penalties from paragraph 4.3.2.3. As a reminder, the First Wrong-Prediction Penalty (v1) pushes the timestamps left to be bigger for early wrong predictions. The v2 aims to penalize wrong predictions when happening late in the year, and pushes them to be predicted earlier. Models are trained as explained in the implementation details in paragraph 4.3.2.5.

In Figure 18, the mean and the standard deviation of some relevant metrics are plotted with respect to the training epoch. The metrics are computed with the performances of the model on the validation dataset. Moreover, the distribution of the accuracy, earliness, harmonic mean and standard deviation score at the end of training are plotted, with respect to the wrong-prediction penalty. When observing the evolution of the metrics with respect to the epochs, we notice in Figure 18a that the second version leads to a lower average accuracy of the model, compared to the first version. In contrast, the second version leads to lower earliness average than the first version, as depicted in Figure 18b. This trade off can be observed on the harmonic mean evolution in Figure 18c; the two versions reach similar mean through training. The standard deviation score in Figure 18d also has similar behaviour for the two versions, though we notice a smaller standard deviation of the standard deviation score at the end of training for the v1 in red.

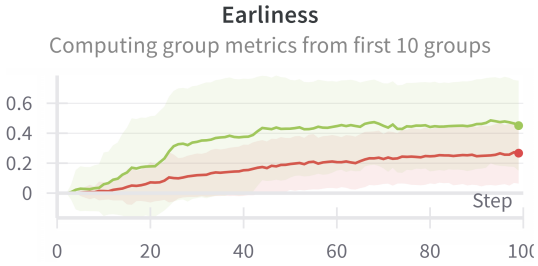
Concerning the metrics distribution, Figure 18e shows that the two versions share the same extremes in the accuracy distribution, but the first quartile of v1 is close to the third quartile of v2. Moreover, the interquartile range is much smaller for v1 than for v2. In addition, Figure 18f depicts a smaller range of earliness for v1 than for v2. The first version has a median around 0.1, whereas the second version has it around 0.4.

Regarding the distribution of the harmonic mean at the end of the training presented in Figure 18g, the extremes are approximately the same for each version v1 and v2. However the distribution is quite different, as the median is significantly lower for v1 than for v2.

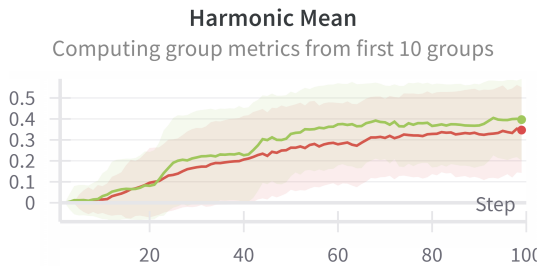
Figure 18h presents the distribution of the standard deviation score at the end of the training and shows that the distributions are quite similar. One difference is the maximum of the standard deviation score of v1 being smaller than the one of v2.



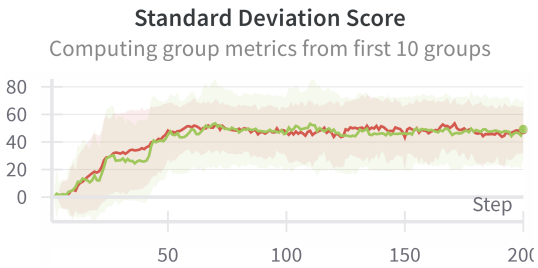
(a) Mean and standard deviation of the accuracy.



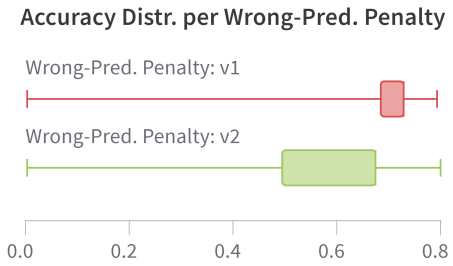
(b) Mean and standard deviation of the earliness as defined in definition 4.8.



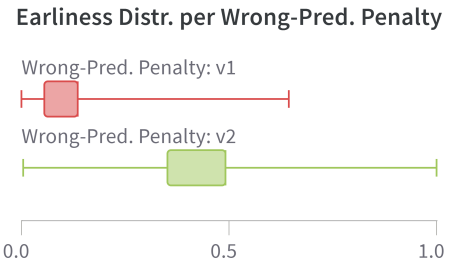
(c) Mean and standard deviation of the harmonic mean as defined in definition 4.9.



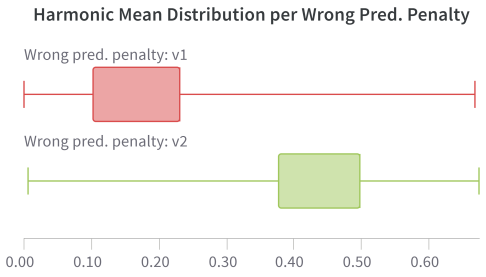
(d) Mean and standard deviation of the standard deviation score as defined in definition 4.10.



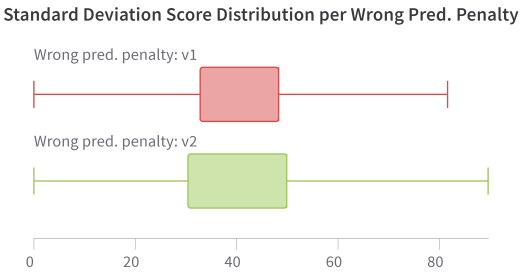
(e) Accuracy distribution on the validation set, at the end of training.



(f) Earliness distribution on the validation set, at the end of training.



(g) Harmonic mean distribution on the validation set, at the end of training.

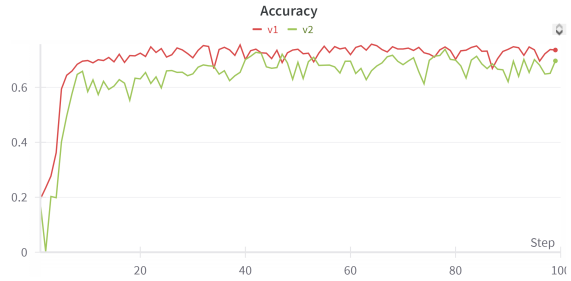


(h) Standard deviation score distribution on the validation set, at the end of training.

Figure 18: Performances of all models during training, separated by wrong-prediction penalty (red for the first version v1, green for the second v2). The metrics are measured on the performances of the models on the validation dataset. On the first four subplots, the mean and standard deviation of accuracy, earliness, harmonic mean and standard deviation score are plotted with respect to the epoch. The total number of epochs is truncated at 100 epochs. On the two last rows, the four subplots depict the distribution of the accuracy, earliness, harmonic mean and the standard deviation score at the end of the training.

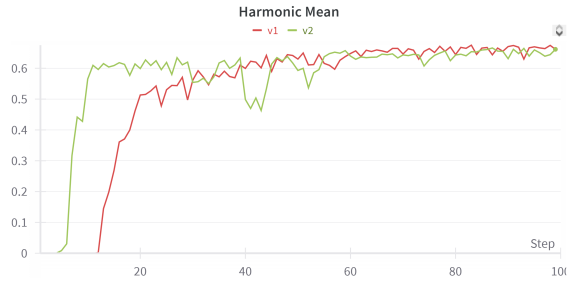
5.2.2.1 Best model for Each Wrong-Prediction Penalty

For each wrong-prediction penalty, let's get the model with the the highest harmonic score on the validation dataset and compare their performances to select the best model. Figure 19 shows metrics which were computed from the model's predictions during training. The accuracy, as shown in Figure 19a, depicts the performance of the best model related to wrong-prediction penalty v1 to be slightly better than the best model related to v2. Regarding earliness, Figure 19b shows that both earliness increase during training. The T_C value is 10 for the v1 model and 2 for the v2 model, which explains why the earliness is equal to zero until then. Then, Figure 19c depicts the harmonic mean evolution through epochs. Both versions reach very close results. Finally, Figure 19d shows the standard deviation score through training. The best model related to v1 has a decreasing behaviour from epoch 20, whereas the best model related to v2 oscillates more.

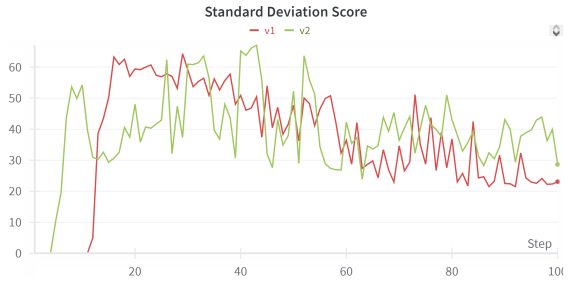


(a) Accuracy

(b) Earliness as defined in definition 4.8.



(c) Harmonic mean as defined in definition 4.9.



(d) standard deviation score as defined in definition 4.10.

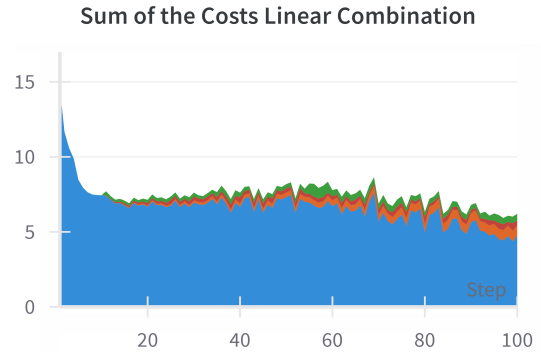
Figure 19: Metrics with respect to epochs. The performances belong to the models which have the highest harmonic mean on the validation set. The red line corresponds to the best model with the first wrong-prediction penalty, which penalizes wrong prediction if they happen early in the year. The green line corresponds to the best model of the second wrong-prediction penalty, which penalizes late wrong predictions.

Furthermore, the first row of Figure 20 presents the general D-ELECTS absolute cost function with respect to the epoch, as defined in equation (13). This figure allows to analyze the weights of each part of the linear combination of the costs. The bigger the epoch, the smaller α_1 (as it decreases linearly through training), thus the bigger the other alphas. This behaviour is presented in the second row of the same figure.

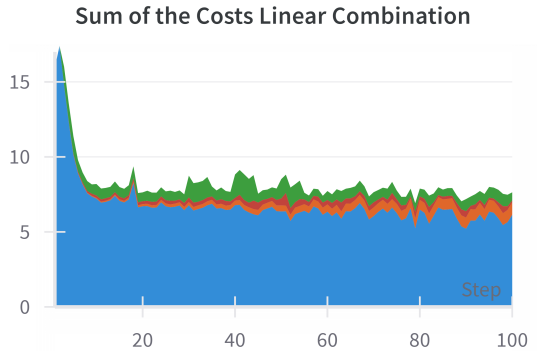
To have more precise measurements of the performances of the models, the metrics obtained on the validation set are given in Table 5. The model related to v1 has a higher accuracy than the v2 model, but a lower earliness. Taking both metrics into account, the first version model obtains a harmonic mean of 0.66 whereas the second version model gets 0.63. Regarding the standard deviation score, the v1 model has a lower score than the v2 model.

Table 5: Performances on the validation set of the models with the highest harmonic mean, with v1 and v2.

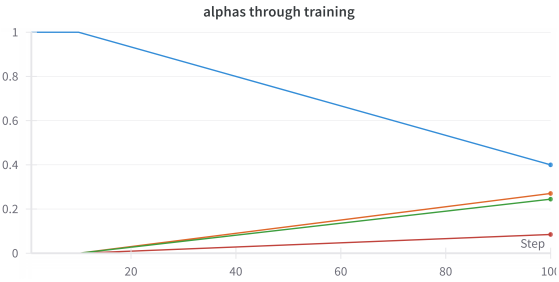
Wrong pred. penalty	Accuracy	Earliness	Harmonic Mean	STD score
v1	0.72	0.61	0.66	25.93
v2	0.62	0.64	0.63	46.45



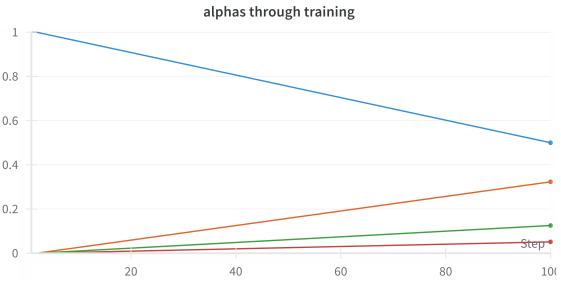
(a) Sum of linear combination of the absolute costs: v1.



(b) Sum of linear combination of the absolute costs: v2.



(c) Hyper-parameters α_j for $j = 1, 2, 3, 4$: v1.

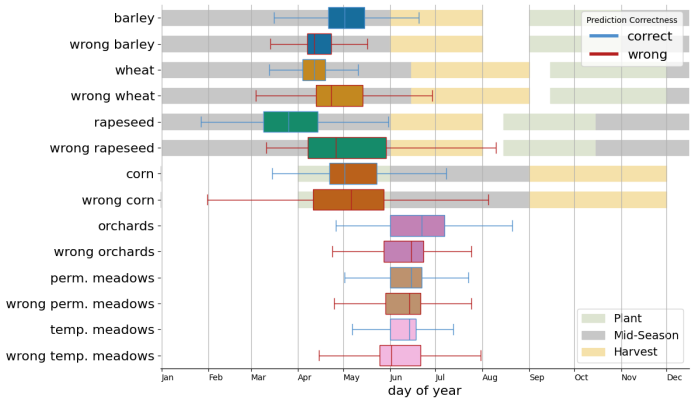


(d) Hyper-parameters α_j, $j = 1, 2, 3, 4$: v2.

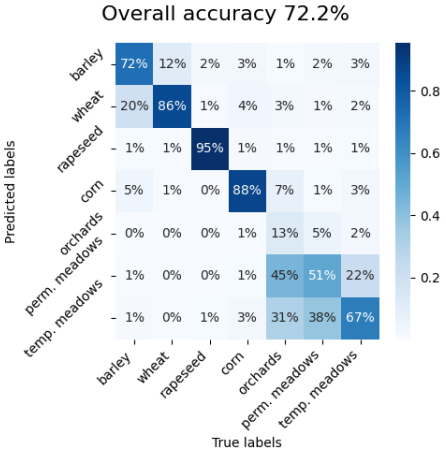
Figure 20: The first row depicts the sum of linear combination of the absolute costs, as written formally in equation (13). The performance was measured during training of the best models with v1 and v2. The metrics were computed on the validation dataset. Legends: blue $|\alpha_1 C_m(\hat{\boldsymbol{y}}_i|\boldsymbol{y}_i)|$, orange $|\alpha_2 C_d(\boldsymbol{z}_i, \hat{\boldsymbol{y}}_i|\boldsymbol{y}_i)|$, pink $|\alpha_3 C_p(\boldsymbol{z}_i, \hat{\boldsymbol{y}}_i)|$, green $|\alpha_4 C_{lr}(\boldsymbol{z}_i, \boldsymbol{y}_i)|$. The second row depicts the values of the hyper-parameters α_1 (blue), α_2 (orange), α_3 (pink), and α_4 (green) during training of the best model associated to one of the two versions of the wrong-prediction penalty.

To analyze further the earliness distribution, Figure 21 shows the normalized confusion matrix, as well as the stopping times box plots, with respect to the crop type and the correctness of the prediction.

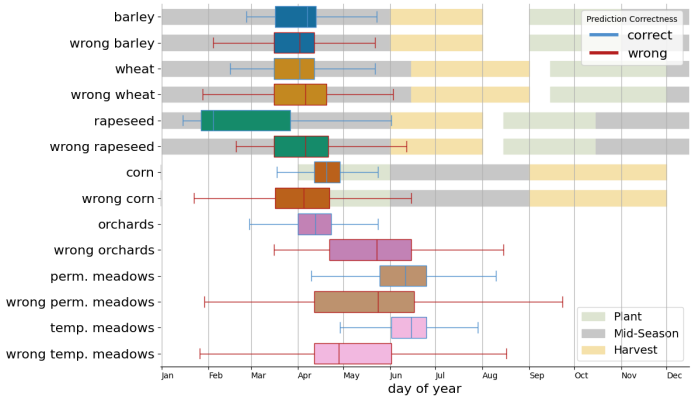
Regarding the stopping times, we see that for the first version of the wrong-prediction penalty, the orchards and meadows classes have more similar distributions between correct/incorrect, compared to the second version. Moreover, with the second version, not only the wrong predictions get classified sooner, but also the correct ones. The incorrect predictions are spread.



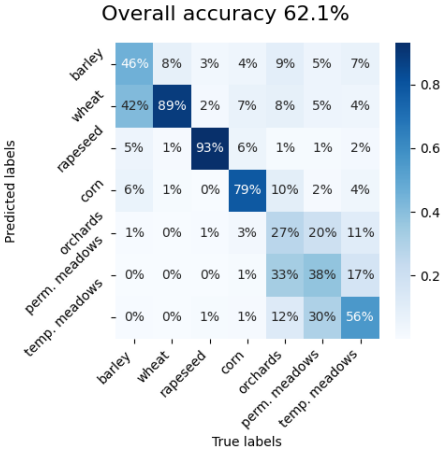
(a) Stopping times assessment of the best model with v1.



(b) Confusion matrix, normalized over the true labels (columns) of the best model with v1y.



(c) Stopping times assessment of the best model with v2.



(d) Confusion matrix, normalized over the true labels (columns) of the best model with v2.

Figure 21: Stopping times and confusion matrix performances of models with v1 and v2, on the validation set. The first row shows the performances of the first version, and the second row the second version. Stopping times box plots, with respect to the crop type and the correctness of the prediction. Blue, respectively red, whiskers, indicates the stopping times of correct, resp. incorrect, predictions. A wrong crop type box plot depicts the stopping time distribution of samples that should have been predicted as that crop type, but were predicted wrongly. The French crop calendar from the USDA Foreign Agricultural Services is also plotted.

5.2.3 Best Model Performance and Graphs

We select the best run Daily ELECTS with the first version of the wrong-prediction penalty (D-ELECTS-v1) according to the previous section. As seen in Table 5, this model has the highest harmonic mean on the validation set. In addition, the standard deviation score is smaller for this model too. These are the main reasons why we keep this model as the best one. For results for the second version model, see the Appendix, Section A.1.

The selected model has parameters $N_{HiddenDims} = 128$, the total number of epochs equal to 100, $p_{thresh} = 0.7$, $T_C = 10$, $\alpha_{min} = 0.4$ and the percentages corresponding to each hyper-parameters α_2, α_3 and α_4 are 45.1%, 14.1% and 40.8% respectively.

The selected model performance is computed on the test datasets. The metrics are given in Table 6, compared to the original ELECTS model on the BreizhCrops dataset. A more thorough comparison is done in Section 5.3. Performances of the selected model are depicted through several graphs.

Table 6: Performances of the selected D-ELECTS model and the ELECTS model. The models are trained on the respective train datasets. The metrics are results from the prediction of the models on the respective test dataset. The standard deviation score depends greatly on the dataset structure (daily timestamps in the Reduced BreizhCrops or not).

Model	Accuracy	Earliness	Harmonic Mean	STD score
D-ELECTS-v1	0.74	0.60	0.67	27.42
ELECTS	0.80	0.85	0.83	7.05

In Figure 22, the distribution of the stopping times and the confusion matrix are plotted. The interquartile are of the stopping times are during the mid-season, before the harvest period. Regarding the normalized confusion matrix in Figure 22b, the model manages to predict 80% of correct labels for the crop types barley, wheat, rapeseed and corn, see the values in the main diagonal. Concerning the last three categories, the model has more struggles. For orchards, it only classifies 10.7% of orchards correctly. The model mixes it up with permanent meadows (50.8%) and temporary meadows (26.0%). Then, the model performs between 62 and 64% for permanent and temporary meadows, because it confuses them.

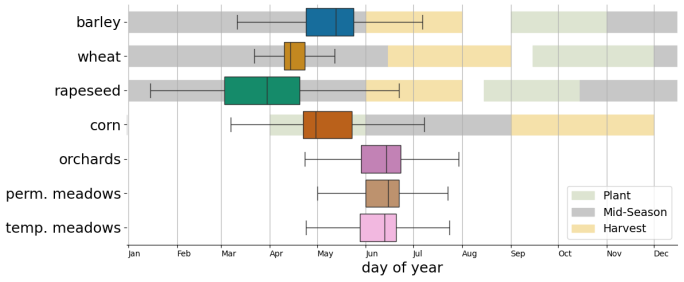
Moreover, stopping time distributions are plotted with regard to the label and correctness of the prediction in Figure 22c. We see some differences in between the correct and incorrect predictions stopping time distributions. For example, for classes barley, wheat, corn, orchards, and temporary meadows, the distributions of correct predictions has a smaller range than for incorrect predictions.

Furthermore, Figure 23 presents the class probabilities per true class. This plot gives also the probability threshold as well as the value μ_c per class c. We notice that the mean probabilities of orchards, permanent and temporary meadows never go above the threshold.

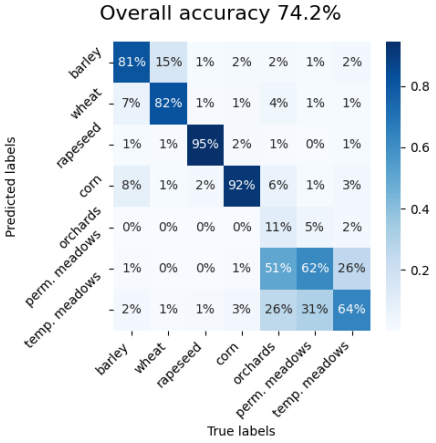
In Figure 24, the mean and standard deviation of the timestamps left per class are plotted. In addition, the values μ_c for each class c are given. This plot shows if the timestamps left indeed follows a piece-wise linear shape.

Finally, in Figure 25, the D-ELECTS model predictions from an area of interest are presented over the span of 3 months in the forms of maps. The parcels are selected from a rectangle area of 3.7 km $\times$ 3.9 km in the test dataset. The figure illustrates the model's prediction process through time. The first three columns correspond to three dates: April 12, May 22 and June 21 2017. The last column contains the ground truth, the final predictions of the model, and the day of stopping per parcels. The first row presents Sentinel-2 satellite images of the area of interest. Then, the second row shows the crop classification of the model. Each color correspond to one crop category, as explained in the legend. The third row shows the incorrect (red) versus correct (blue) predictions. For the second and third row, the translucent parcels do not have their final predictions yet. Finally the last row shows the active (white) versus stopped (black) predictions.

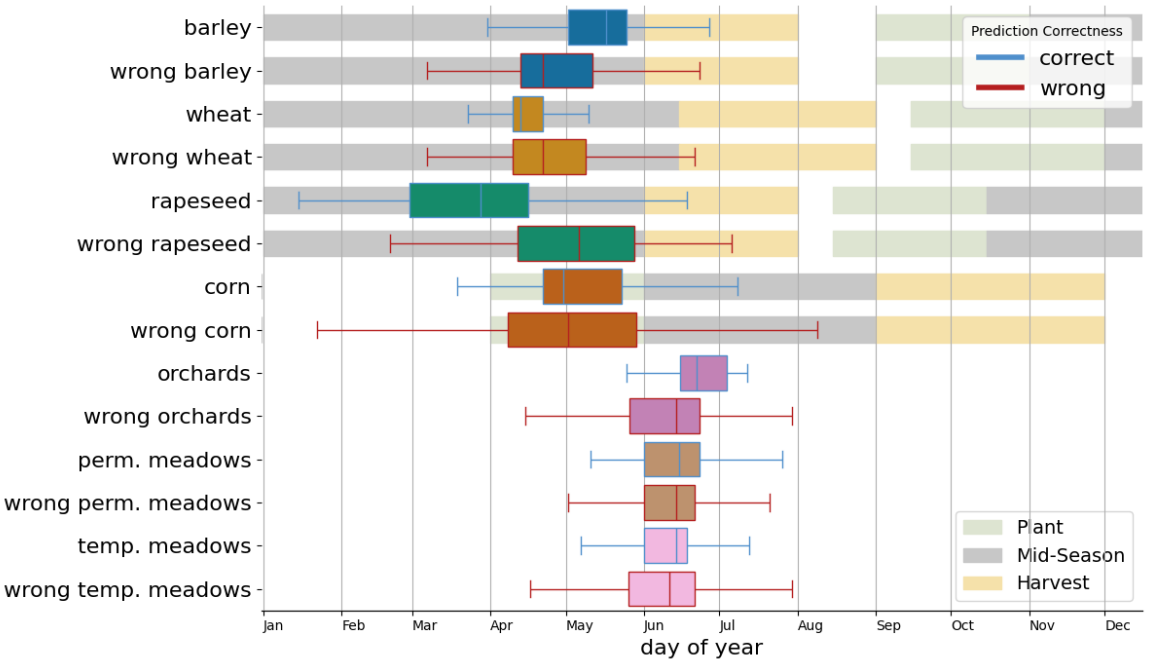
Over time, more and more parcels are eventually predicted by the selected model. This can be seen as in the first column, there is almost no stopped prediction, whereas most predictions are finished at the end of June, in the third column.



(a) Stopping times assessment of the best performing model. Planting, mid-season and harvesting periods according to the USDA Foreign Agricultural Services crop calendar for France are also plotted.



(b) Confusion matrix, normalized over the true labels (columns).



(c) Stopping times box plots, with respect to the crop type and the correctness of the prediction. Blue, respectively red, whiskers, indicates the stopping times of correct, resp. incorrect, predictions. A wrong crop type box plot depicts the stopping time distribution of samples that should have been predicted as that crop type, but were predicted wrongly. The French crop calendar from the USDA Foreign Agricultural Services is also plotted.

Figure 22: Best model performances assessment on the test dataset.

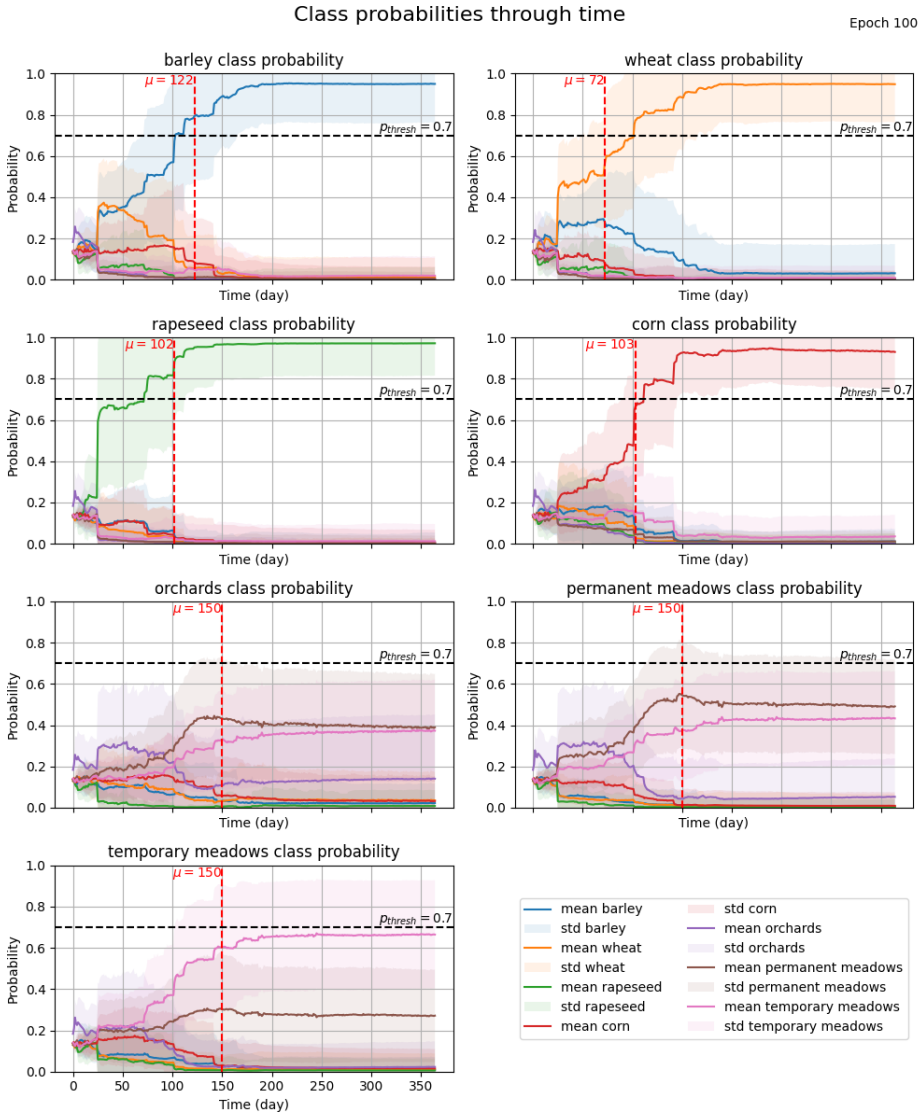


Figure 23: Class probabilities with respect to time. Each subplot corresponds to the probabilities associated to one true label. The mean and the standard deviation are plotted in colors relating to the true crop types. Moreover, the probability threshold $p_{thresh} = 0.7$ is indicated as a black horizontal line. For each subplot, the corresponding μ_c is drawn as a red dashed vertical line.

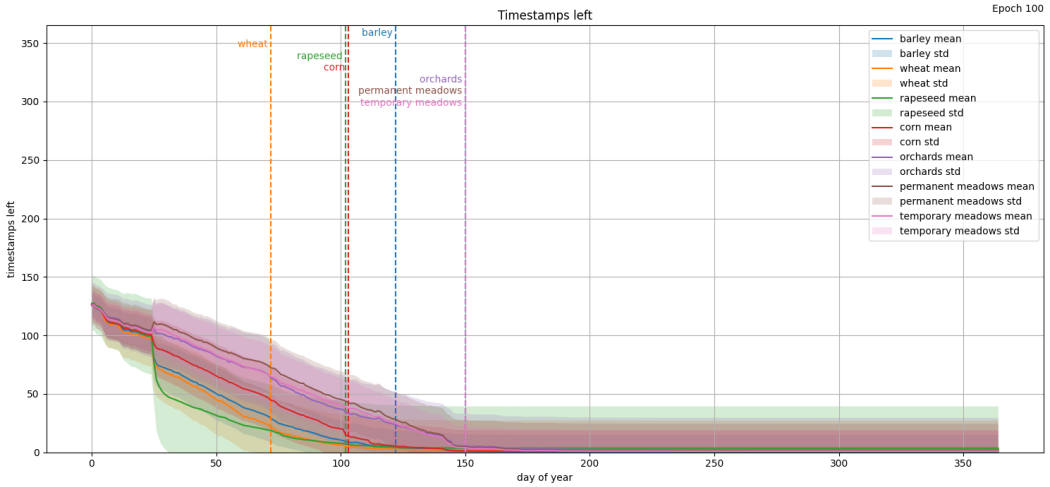


Figure 24: Timestamps left per class, with respect to time. Each full colored line corresponds to the mean of the timestamps left of a certain crop type. The standard deviation is plotted around it in clear. The values μ_c are indicated as vertical lines for each label $c \in C$.

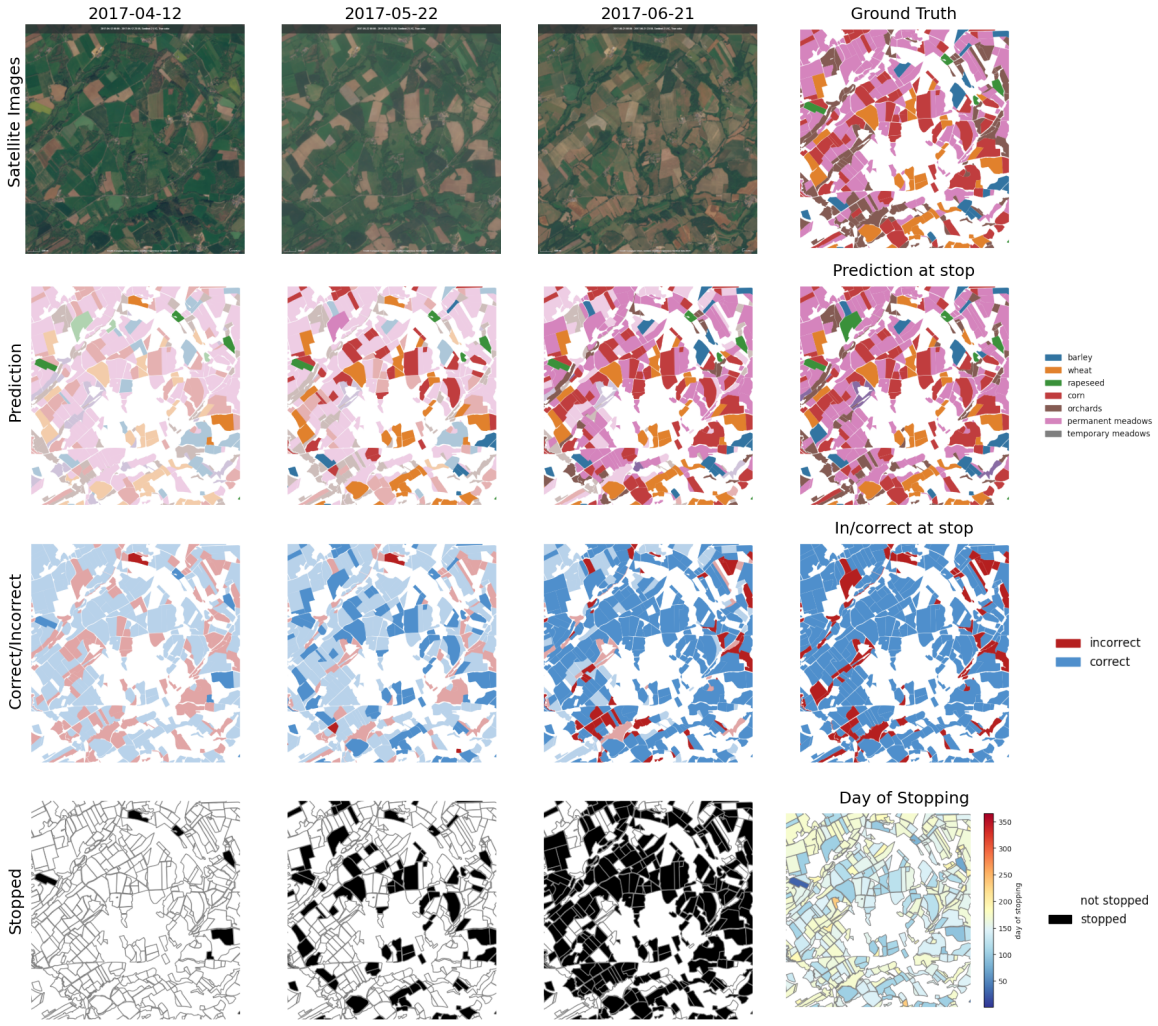


Figure 25: The D-ELECTS prediction process is shown over the span of 3 months. The model predicts class labels and stops the predictions of individual fields. The focus here is on a 3.7 km × 3.9 km site in the Brittany test region. The three first columns present the predictions at three dates. The last column presents the final results. Concerning the rows, first row shows the satellite images of the L1C layer and the ground truth. The second row presents the predictions through the year, whereas the third row present the correct/incorrect map. On both rows, transparency indicates that the model has not decided to stop yet. The last row shows a binary score to indicate which classifications have been stopped (black) or not (white). Finally, the bottom right image shows the stopping day for all parcels.

5.3 Comparison of D-ELECTS with ELECTS

A reproduction of the original ELECTS model results was presented in Section 5.1.1. Now, we use the ELECTS model as benchmark to compare the D-ELECTS model we developed and selected in Section 5.2.3.

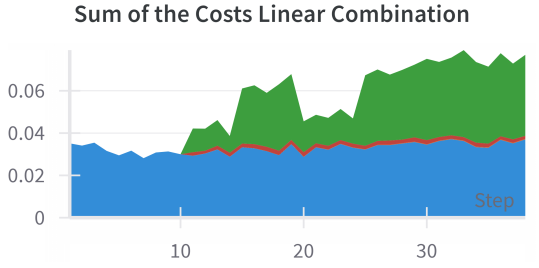
To compare the models fairly, we need to train and test them on the same datasets. We choose the datasets of the original ELECTS paper [34], i.e. the dataset presented in Section 3.2. Note that the training and validation datasets contain MTS of length 70, whereas the test dataset has MTS of length 150.

Figure 26 presents the general D-ELECTS absolute cost function, the associated α_j, the training and test losses, as well as some performance metrics. Figure 26a presents the train and test losses with respect to the epoch. The training stopped at epoch 38 because of early stopping (see paragraph 4.3.1.1). The final model is saved at the lowest point of the test loss, which takes place before $T_C = 10$. Note that the losses have a range under 0.1, which is 100 times smaller compared to the range of the losses when the model is trained on the adapted dataset in Section 3.3.

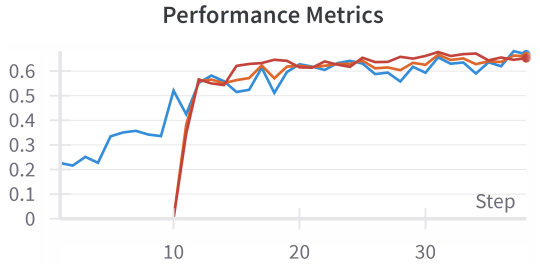
Furthermore, Figure 26b depicts the general D-ELECTS absolute cost function and shows how much each term weights during training. The term associated to the piecewise learning regression cost $|\alpha_4 C_{lr}(\boldsymbol{z}_i, \boldsymbol{y}_i)|$ (from paragraph 4.3.2.4) in green is quite significant, eventhough its associated hyper-parameter is under 0.1.

Concerning the performance of the selected model on the original dataset, Figure 26c depicts the accuracy, earliness and harmonic mean. The metrics have a normal behavior; the accuracy increases through the epochs, and the earliness and harmonic mean start at T_C and then increase until they reach a plateau.

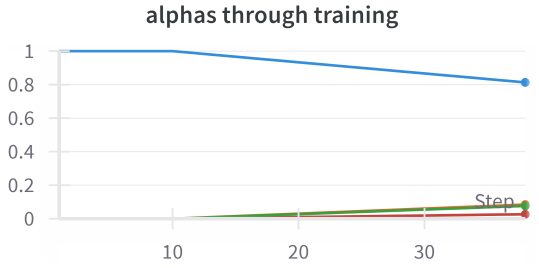
(a) Train and test losses during the training of the selected D-ELECTS model on the original ELECTS dataset.



(b) Sum of linear combination of the absolute costs, as defined in equation (13). Legends: blue $|\alpha_1 C_m(\hat{\boldsymbol{y}}_i|\boldsymbol{y}_i)|$, orange $|\alpha_2 C_d(\boldsymbol{z}_i, \hat{\boldsymbol{y}}_i|\boldsymbol{y}_i)|$, pink $|\alpha_3 C_p(\boldsymbol{z}_i, \hat{\boldsymbol{y}}_i)|$, green $|\alpha_4 C_{lr}(\boldsymbol{z}_i, \boldsymbol{y}_i)|$.



(c) Performance metrics during training. Legend: accuracy in blue, earliness in red, harmonic mean in orange.



(d) Hyper-parameters α_j, $j = 1, 2, 3, 4$. Legend: α_1 in blue, α_2 in orange, α_3 in red, and α_4 in green.

Figure 26: Measurements of the D-ELECTS model performance on the original ELECTS train and validation datasets. Going by column: the train and test losses are plotted with respect to the epoch in Figure 26a. Below it, Figure 26c shows the accuracy, earliness and harmonic mean of the model during training. In the second column, Figure 26b depicts the general D-ELECTS absolute cost function as a sum of each term. The corresponding hyper-parameters α_j ($j = 1, 2, 3, 4$) are given in Figure 26d.

To study the performance of the selected model on the original test dataset of BreizhCrops in Section 3.2, Figure 27 presents the normalized confusion matrix. The overall accuracy of the model on the test dataset is 36.4%. Moreover, the earliness is at 0; the model outputs the number of timestamps left to always be above zero and therefore does not stop the classification. This causes a harmonic mean of 0.

Figure 28 presents the class probabilities per crop category. Vertical red lines indicate the value μ_c per class c.

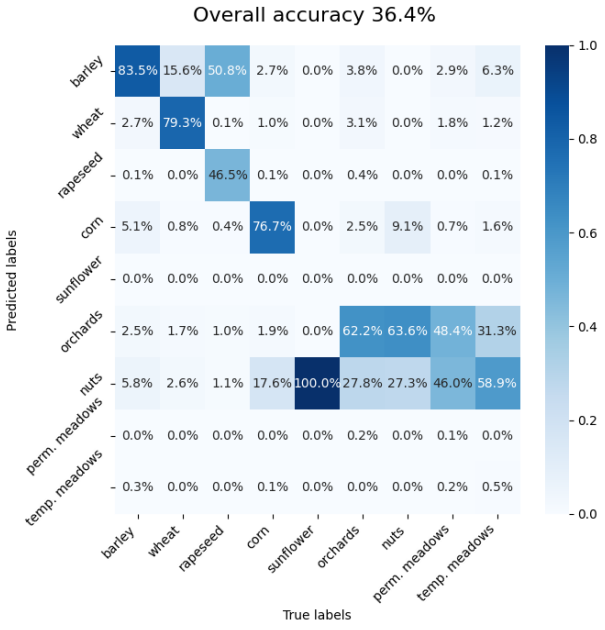


Figure 27: Confusion matrix on the original test dataset of BreizhCrops, with a sequence length of 150. The matrix is normalized per column. The overall accuracy is 36.4%.

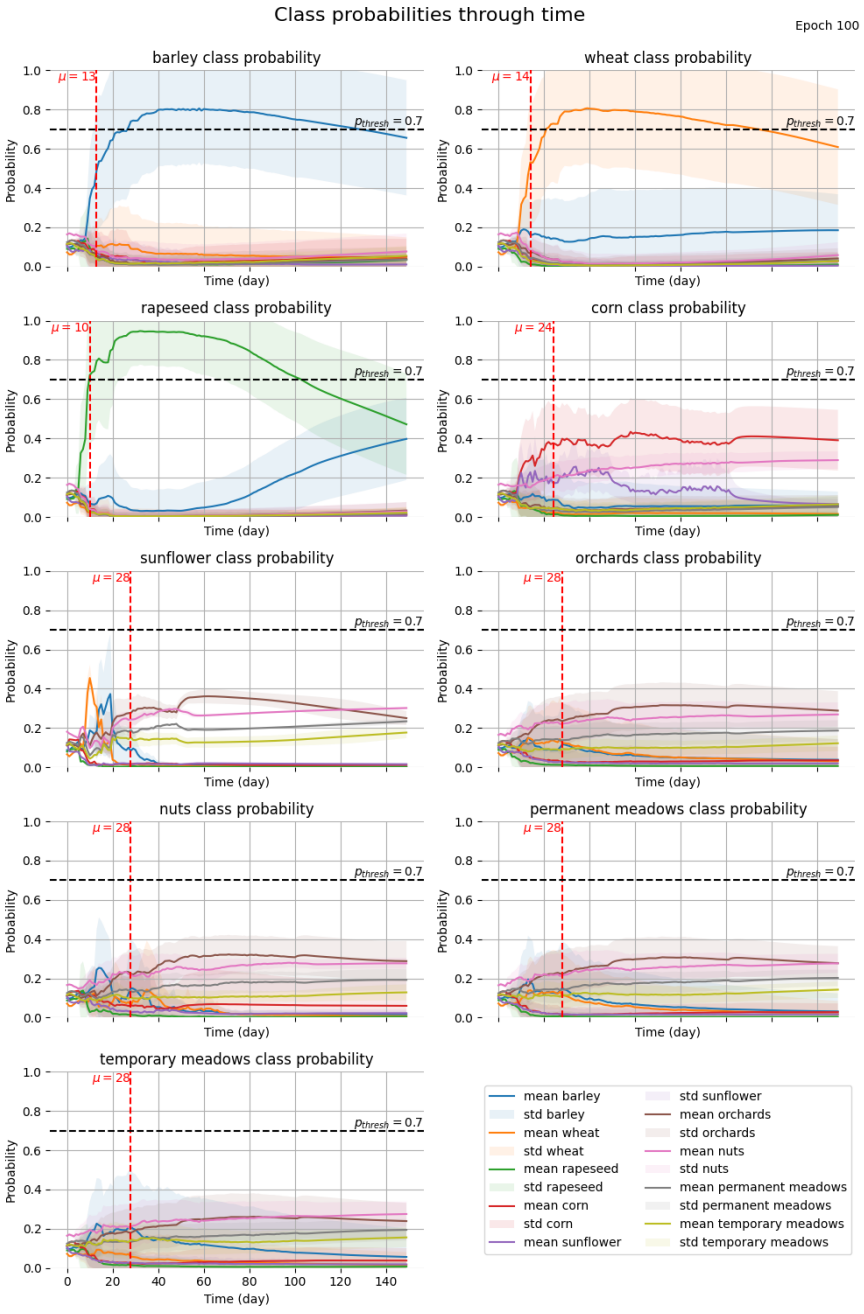


Figure 28: class probabilities are extracted from the selected D-ELECTS model on the original ELECTS test dataset. The test dataset contains time series of length 150. Each subplot corresponds to the probabilities associated to one true label. The mean and the standard deviation are plotted in colors relating to the true crop types. Moreover, the probability threshold $p_{thresh} = 0.7$ is indicated as a black horizontal line. For each subplot, the corresponding μ_c is drawn as a red dashed vertical line.

6 Discussion

This section follows the structure of the Results in Section 5 and presents potential further research. First, in Section 6.1, we comment shortly on the ELECTS results and the reproduced results. Second, we analyze the D-ELECTS model's performance in Section 6.2. Third, in Section 6.3, we review the comparison between D-ELECTS and ELECTS models' performances. Finally, we show further research in Section 6.4.

6.1 ELECTS Discussion

The ELECTS model relies on the probability of stopping to cease classification. The cost function aims to enforce the probability of stopping to become bigger over time, but when testing the model on the Breizhcrop test dataset, the stopping time was sometimes increasing and then decreasing. This shows the limitation of the ELECTS cost function.

Figure 13a shows that stopping time distributions are different for each crop type. Most crops, i.e. wheat, rapeseed, corn and sunflowers, are classified during their corresponding mid-season. Indeed, wheat, rapeseed, corn are mostly predicted in May, and sunflowers are labelled in July. They are classified before the harvest season, which starts in June for rapeseed and wheat, in August for sunflower, and in September for corn. The harvest season is considered by domain experts as the end-of-series date for accuracy only classifiers. The barley class is mostly predicted during the harvest season, in June; it seems that the model stops predicting when recognizing harvest operation. Classes like orchards, permanent meadows and temporary meadows have larger whiskers as the model encounters more struggle to tell them apart.

In Figure 13b, the model mixes up the orchards, permanent and temporary meadows. The reason for this might be that orchards incorporate both trees and grass, which makes the class inhomogeneous. As for meadows, the two classes are sprectrally similar, which makes them hard to discern.

The standard deviation score is about 7.05, which is small compared to the common sequence length being 150. However, this small quantity can be explained by the original sequence length varying between 51 and 102.

Concerning the BreizhCrops dataset, several shortcomings need to be addressed. The first issues concern crop labels in general. For example, only the main cultivar is reported in the parcel data. Indeed, in practice, there is often more than one crop per parcel, due to crop rotation and intercropping. In addition, an unknown percentage of labels is incorrect. The reasons are last-moment change in crop types, which highly depends on weather conditions, or unreported field geometries. Additionally, specifically regarding the BreizhCrops dataset, we do not know how the authors obtain data with lengths between 51 and 102. The dataset is not reproducible from raw data. Moreover, the value of the common sequence length, set at 150, appears arbitrary by the authors. When testing with another sequence length, we achieve a different outcomes, see the Appendix in Section B.1. This challenges the interpretation of the results, and will also make the comparison between the ELECTS and D-ELECTS models more difficult to interpret.

6.2 D-ELECTS Discussion

In this part, each subsection of D-ELECTS results in Section 5.2 will be analyzed and commented. We start by examining the classification only pretraining in Section 5.2.1, then we explore the comparison between the two wrong-prediction penalties in Section 5.2.2, and finally we investigate the best model performances in Section 5.2.3.

Note that all models are trained on the training dataset. Their performance could have been improved by training them on both the training and validation datasets, after finding the best combination of parameters.

6.2.1 Classification Only Pretraining

The model is trained with the misclassification cost only. It allows us to assess the performance of a model trained only on classification. Nevertheless the presented metrics could have been improved by tuning the hyper-parameters.

In Figure 15, we see that the model reaches a plateau in about 30 epochs. We see that the model architecture can adapt to a dataset structured differently. However, in Figure 16, a misclassification between orchards, temporary and permanent meadows is present.

In Figure 17, the model's confusion over the three crop types orchards, temporary and permanent meadows is again notable. Orchards mean probability does stand out a bit more, while still being under 0.6 and having a big standard deviation. The high standard deviation can be explained by the nature of orchards being a mix of

trees and grass, with different spectral signatures and seasonality. The model seems to be the most confused for the permanent meadow class.

This results analysis confirms our expectations in Section 3.3 concerning the model's performance on the orchards and meadows classes. Consequently, we do not expect the D-ELECTS model we developed to discern between the three crop types orchards, temporary and permanent meadows; it is probably not possible with the information we input to the model.

Lastly, because of the uncertainty of the model about orchards and meadows classes, the value μ_c is set to an arbitrary value, fixed at $\frac{150T}{365}$. However, this term might not be suited for datasets structured differently, like the original BreizhCrops dataset. This is one of the main limitation of our model.

6.2.2 Comparison between the Two Wrong-Prediction Penalties

The wrong-prediction penalties aim to penalize wrong predictions of the model for each class. Two versions were presented in paragraph 4.3.2.3. The First Wrong-Prediction Penalty (v1) consists of penalizing wrong predictions if they happen early in the year, with a term $(1 - z_t/T)(1 - t/T)$, whereas the Second Wrong-Prediction Penalty (v2) focuses to penalize wrong predictions if they occur late in the year, with a term $(t + z_t)/T$.

On average over all the runs in Figure 18, v1 favors higher accuracy, in contrast with v2 favoring higher earliness. This is because of the definitions of the two versions. A model with v1 penalty cost pushes wrong predictions to happen later so that the model can look at more data to give a more precise prediction. For the penalty to be given, z_t and t must be small. A model with the v2 cost pushes late wrong prediction to occur earlier, especially when $t + z_t$ is close to T, i.e. when the prediction happens later in the year.

Regarding the standard deviation score average over all the runs in Figure 18d, the behaviour of the two versions might be similar because the standard deviation score is noisy once the classification only pretraining is done. This would explain the increasing and then stabilizing averages.

Concerning the distribution of the metrics at the end of the training in the last two rows of Figure 18, v1 has a much lower median of the harmonic mean in Figure 18g because the earliness is on average around 0.2, compared to v2 which has an earliness average around 0.5. The accuracy cannot compensate for such a low earliness, over all the runs.

However, when looking at the highest performing model for each version, v1 seems like a more appropriate approach for the early classification task.

6.2.2.1 Best Model for each Wrong-Prediction Penalty

In Figure 19, the accuracy, earliness and harmonic mean have approximately the same increasing and converging behaviour for both versions. We can notice though some instability for v2 in earliness, which is then reflected in the harmonic mean. The peaks between the 30th and 50th epochs in earliness coincide with some instability in the piecewise-linear regression cost and the wrong-prediction penalty, see the Appendix in Section B.2, in Figure 34. The fluctuation suggests v2 competes against the piecewise-linear cost. That would then explain the irregularity in the standard deviation score in Figure 19d. The standard deviation score gets better with the piecewise linear regression cost, but gets worse with v2 pushing the wrong predictions earlier.

Moreover, the opposite goals of the piecewise linear cost and v2 penalty can also be observed in Figure 20b. The sum of linear combination of absolute costs slowly decreases. It stagnates more than the same quantity for v1, as shown in Figure 20a.

To finish comparing the two versions of the wrong-prediction penalty, we analyze Figure 21. In the first column, it shows that the incorrect predictions stopping times are generally more spread for v2, compared to v1. This is particularly striking for classes like orchards and meadows.

The idea behind v2 was to push late wrong predictions to be predicted earlier, which would have resulted with a shift to the left of the wrong predictions in Figure 21c. Unfortunately the model did not behave as expected. One potential reason is the instability it creates with the piecewise linear regression cost. The model cannot find a minimum which satisfies both criteria. In addition, the stopping times distribution are a bit more satisfying for bigger classes such as barley, wheat, rapeseed and corn. We observe an overall shift to the left. For the corn class, the model with v2 penalty shows even better results than with v1 penalty. Therefore v2 might be more sensitive to small classes, compared to v1.

In addition, we compare the normalized confusion matrices in Figure 21b and Figure 21d with the confusion matrix obtained in the classification only pretraining, in Figure 16. First, the overall accuracy obtained with the v1 penalty (72.2%) is higher than with classification only (68.8%). This results from the better performance of the model on the permanent (51% instead of 33%) and temporary (67% instead of 56%) meadows,

which are big classes. This phenomenon can also be seen on the class probabilities, see the Appendix in Section B.2, in Figure 35. Nevertheless, the model trained with v1 penalty has smaller values for other classes, i.e. barley, wheat, rapeseed corn and orchards. The performance fall is striking for the orchards class, from 63% to 13%. This might come from pushing the model to predict as early as possible. Indeed, the orchard class probability stays stable for the classification only pretraining, whereas it decreases for the model trained with v1, see in Figure 17 and in Appendix Section B.2. The general accuracy improvement can be explained by the optimized parameters of the model trained with v1 penalty, and the focus of the v1 penalty on the accuracy of the model.

Concerning the confusion matrix from the performances of the model trained with the v2 penalty, the overall accuracy (62.1%) is lower than the one achieved on the classification only pretraining (68.8%). This is surely due from the penalty focusing on earliness rather than accuracy.

6.2.3 Best Model Performance and Graphs

We analyze the performances in Table 6 of the best selected model, with v1. The model D-ELECTS-v1 encourages a higher accuracy with the first version of the wrong-prediction penalty, which explains why the accuracy (0.74) is higher than the earliness (0.60). The standard deviation score is at 27.42, which is equivalent to about a month. This is an average over the standard deviation of the stopping times over all classes, as defined in definition 4.10. Thus each class standard deviation has the same weight in the score. Nevertheless, we can assume from Figure 22a that some classes, such as wheat and permanent meadows, have smaller standard deviation than others, like barley and rapeseed.

According to [15], rapeseed usually flowers around May in France. Thus our model manages to predict before that notable phenological event. Moreover, we note that the D-ELECTS-v1 model manages to predict the crops barley, wheat, rapeseed and corn before the harvest season, which is an exploit. We also observe that corn is mainly recognized during its planting period, during April and May.

Looking at Figure 22b and Figure 22c, we see that barley samples which were wrongly predicted mainly get assigned the wheat (8%) or corn (7%) label. Barley and wheat are two close plant species, which explains the confusion. This uncertainty is reflected in the stopping times distribution of the incorrect labels: the wrong barley get predicted sooner than the correct ones, closer to the prediction dates of the corn and wheat classes.

Then, the model mixes up wheat for barley (15%) and we see again that the stopping time distribution of the wrong wheat looks like the distribution of barley.

Regarding the classes orchards and meadows, which are harder to classify, the D-ELECTS-v1 model struggles to tell them apart. In addition, the medians of the boxplots are almost all aligned. This coordination is thanks to the piecewise linear regression cost, as μ_c is fixed at 150 day of the year, which corresponds to beginning of June.

This leads us to Figure 23. The values μ_c for classes $c \in C$ are updated every 5 epochs during training, after T_C is reached. Grid search was done to find the best parameter p_{thresh}. We have seen that this protocol gives satisfactory results, however the value μ_c when the threshold is never reached is arbitrary. Moreover, the average might not be the best representative measure to grasp the behaviour of the class probabilities through time. We see on the figure that the standard deviation can be about 0.2, which is non-negligible in a probability scale.

In Figure 24, the timestamps left averages and standard deviations are shown. Their aspect mostly look as piecewise linear. We notice though a drop or small increase around the 25 day of the year, which could also be seen in the class probabilities in Figure 23. It seems that the model starts to get an intuition of the class end of January, which coincides with the moment D-ELECTS-v1 model starts to predict rapeseed.

Furthermore, Figure 25 shows how the selected model gradually predicts the crop type in each parcel. On April 12th, most of the fields are green on the RGB satellite images. The D-ELECTS-v1 model predicted some rapeseed parcels, in green on the second row. They appear in fact light green on the satellite image. About half of the parcels are correctly predicted at that time, as shown on the third row. Because the model does not have enough information, it waits for more input and most parcels stay active (in white in the last row). As time passes, more colors appear on the satellite images, more predictions are becoming correct (less red on third row) and are ceased (black in last row).

Finally, compared to other studies on early crop classification, our model reaches an accuracy which is in the range of performances attained in May, that were cited in Section 2.3. Regarding earliness, studies on early crop classification did not use this metric [21, 22, 29]. The reason is that they did not explicitly provide a stopping criterion and merely showed how their model performance evolved with respect to time. A stopping criterion is one of the strengths of our model, as it could easily be applied to real-world case scenarios. Specifically, the number of timestamps left is a concept that is easily explained to the general public.

The earliness metric was only given by the ELECTS paper [34]. A thorough discussion about the comparison of the D-ELECTS and the ELECTS model follows.

6.3 Comparison of D-ELECTS and ELECTS Models Discussion

In Figure 26, the training of the D-ELECTS-v1 on the BreizhCrops training dataset is presented. The losses do not decrease for several reasons. First, the D-ELECTS model relies on the daily timestamps to choose the right time to cease the classification. Our model expects sequence length of 365 timestamps. With the original sequence lengths of the BreizhCrops time series varying between 51 and 102, the model cannot learn. The approximation of stopping time per class μ_c does not make sense anymore, and so does the linear regression cost. It can be seen in Figure 26b. Second, the scale of the losses are very small; the general D-ELECTS cost function from is less than 0.1. The scale change comes from the sequence length change; the cost is a sum over the timestamps. The tiny costs might results with rounding errors, and more importantly, the optimizer learning rate is surely too big for a minimum to be found.

Due to early stopping, the saved model is the one trained before the 10th epoch, thus during the classification only pretraining. To add one more difficulty, the BreizhCrops test dataset has a common sequence length of 150, which gives the values μ_c even less meaning than before. The values are too early for the test dataset. From there we obtain a low accuracy.

Even though the datasets and the models are different, we can still compare the task results of the two models on their respective datasets on a high level, analyzing the metrics in Table 6. The ELECTS model has a higher accuracy, earliness and harmonic mean compared to the D-ELECTS-v1 model. The standard deviation score is also smaller, but that metric is harder to compare since the time series sequence lengths vary for unknown reasons.

When comparing the boxplots distribution next to each other in Appendix Section B.3, the D-ELECTS-v1 model has earlier medians for classes barley, wheat, rapeseed and corn. For the rest of the classes, i.e. orchards, permanent and temporary meadows, the medians are pretty close, in the month of June. It raises the question if the earliness as defined in definition 4.8 does accurately reflect earliness as we want. Maybe a more fair metric would be to compute the number of timestamps with non-zero data before the stopping time, over the number of total number of timestamps with non-zero data, instead of t_{stop}/T.

Moreover, the whiskers are smaller for the classes wheat, corn, orchards, permanent and temporary meadows, so 5 classes over 7. Only the boxplots of barley and rapeseed classes have bigger whiskers.

Finally, the ELECTS model relies on a stopping probability to cease the prediction, whereas the D-ELECTS model leans on on the number of timestamps left before the final prediction. We saw that the stopping probability might not always be increasing, while the timestamps seem to have an average decreasing behavior. This makes our stopping criterion more robust on the dataset. An advantage of the stopping criterion of the D-ELECTS model is that it gives an approximate time when the model will stop the prediction. The ELECTS might give a low or high probability of stopping, but it does not give an idea of the number of data points it needs to complete the classification with high probability. That is why our approach might be more relevant for real-world scenarios, when a real-time prediction is needed.

6.4 Future Research

Regarding the structure of the D-ELECTS model, the method to compute the targeted stopping times μ_c could be improved. For example, we suggest other approaches relying on the prediction time rather than on the classification probabilities. The values μ_c could be the median of the stopping times, for each class c. Another approach would be to predict the value μ_c via an additional decision head. Furthermore, as suggested by Dr. Russwurm, replacing the backbone model of the ELECTS model by a transformer could improve its performance. It would also be interesting to study the D-ELECTS model with a transformer as its backbone model, instead of the current LSTM.

In order to improve D-ELECTS performance, the model should be trained on a bigger dataset. Indeed, we briefly examined the behavior of the model trained on both the training and validation datasets, instead of the training dataset only. We could observe a significant improvement in terms of earliness and shortened period of predictions, while still maintaining the same overal accuracy, see Section B.4 in the Appendix. The variability of our model's prediction seem to better match with the variability expected from phenology. For example, rapeseed parcels are predicted in April, and most of the predictions occur within about 10 days.

Moreover, data fusion could also be used to enhance the performance of the D-ELECTS model. For instance, Graf et al. suggested using thermal time, or accumulated growing degree day, instead of calendar dates [14]. Additionally, Inglada et al. demonstrated that incorporating high temporal resolution Synthetic-

Aperture Radar (SAR) satellite imagery alongside high temporal and spatial resolution optical imagery could improve accuracy in early crop classification [21].

Concerning models comparison, we encountered limitations because of the datasets differences and earliness definition. In fact, other studies on early crop classification are trained on a different number of labels, different classes sizes, and different crop hierarchy. This makes the comparison harder to interpret. Furthermore, as mentioned in Section 6.3, a new definition of earliness could dismiss the zero data in the Reduced BreizhCrops dataset, in order to have a more fair comparison of earliness.

Lastly, it would be interesting to test the D-ELECTS model robustness on different regions and years. For instance, Russwurm et al. tested the ELECTS applicability across datasets [34]. Their model was tested on large-scale datasets in Europe, on field parcels in France (BreizhCrops) and Germany (BavarianCrops). It was also evaluated in small-scale datasets in Africa. The datasets contained four crop types, extracted from data from South Sudan and Ghana. Additionally, Račič et al. trained their model on Sentinel-2 satellite imagery with crop labels from Slovenia, over years 2019, 2020, and 2021. They tested their model on a target year, trained only with the complementary years, and achieved an average F1-score of 82.5% [29].

7 Conclusion

We introduced a training framework for in-season crop classification, called Daily ELECTS (D-ELECTS), based on ELECTS paper from Russwurm et al. [34]. Specifically, the name of the model derived from the model's input, which consisted of parcel-averaged satellite image time series, where each timestamp corresponds to one day. The first main contribution is a modification of the second decision head of the ELECTS model, which was updated to output the number of timestamps left before attributing a final crop type to a parcel. This architecture offered a stopping criterion that approximated the time needed to have a final prediction, providing a more practical alternative than the stopping probability of the ELECTS model.

The second core contribution was the development of the general D-ELECTS cost function. Indeed, this loss took into account several factors: a misclassification cost, an earliness reward, a regularizer for the decreasing property of the timestamps left, and a wrong-prediction penalty. The regularizer enforced that all plants from the same crop type were predicted in the same period. In particular, it aimed that each crop type received their final classification around a targeted stopping time, which was computed using an algorithm relying on class probabilities. In addition, regarding the wrong-prediction penalty, we tested two versions. The first version penalized incorrect predictions if they happened early in the year, whereas the second one penalized wrong predictions if they occurred late in the year.

As a result, we concluded that the best wrong-prediction penalty is the first version. The associated model predicted crop types before the harvesting season, obtained a higher harmonic score, as well as predicted plants of the same type on shorter periods. The second version has a slightly better earliness, however the first version had a significantly higher classification accuracy, due to pushing the model to wait for more information for early wrong predictions. We reached an overall accuracy of 74% and an earliness of 60% on the Reduced BreizhCrops dataset.

Then, to compare the ELECTS and the D-ELECTS models performances, we analyzed the medians of prediction times on their respective datasets. Overall, the D-ELECTS model predicted earlier than the ELECTS model. Thus, by predicting earlier, our model has a slightly lower accuracy than the ELECTS model, while still being in the range of the classification accuracy of current early crop classification models.

Furthermore, we could unfortunately not conclude that the D-ELECTS model had significantly shorter prediction periods compared to the ELECTS model.

One of the limitations of the model is the computation of the targeted stopping time per class. Instead of relying on the class probabilities, we could consider the median of the stopping times. An alternative approach could lean on an extra decision head whose output would be the targeted stopping times per class.

Furthermore, the D-ELECTS model faced the limitations of the daily timestamps. The model is restricted to work with a specific dataset structure, making performance comparison difficult with other models.

To conclude, our promising study leads to potential work with a bigger training dataset. Moreover, we suggest to focus on data fusion with thermal time or SAR satellite image time series, in order to improve classification accuracy [14, 21] . Upcoming research could also study the robustness of the model in time and space, by evaluating the model on several years as in Račič et al. study, and on datasets covering different regions and crop types, as in Russwurm et al. paper [29, 34].

Appendix A Results

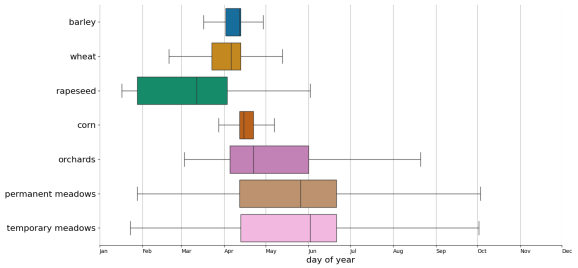
A.1 Performance and Graphs of D-ELECTS with v2

The best harmonic score is obtained by the model with the second version of wrong prediction penalty, $N_{HiddenDims} = 128$, the total number of epochs equal to 100, $p_{thresh} = 0.5$, $T_C = 2$, $\alpha_{min} = 0.5$ and the percentages corresponding to each hyper-parameters α_2, α_3 and α_4 are 64.6%, 10.2% and 25.1% respectively. Figures of section 5.2.3 are reproduced here by the model with v2.

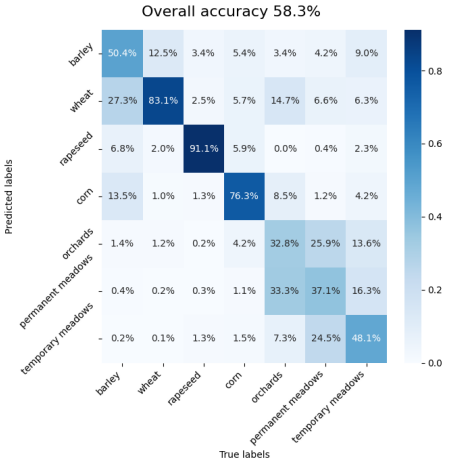
Let's now get a look of the performance of the models on the test datasets. The final performances of the two models are presented in Table 7.

Table 7: Performances of the models with the highest harmonic mean, for each wrong prediction penalty (first version: v1, second version: v2). The metrics are results from the prediction of the models on the test dataset.

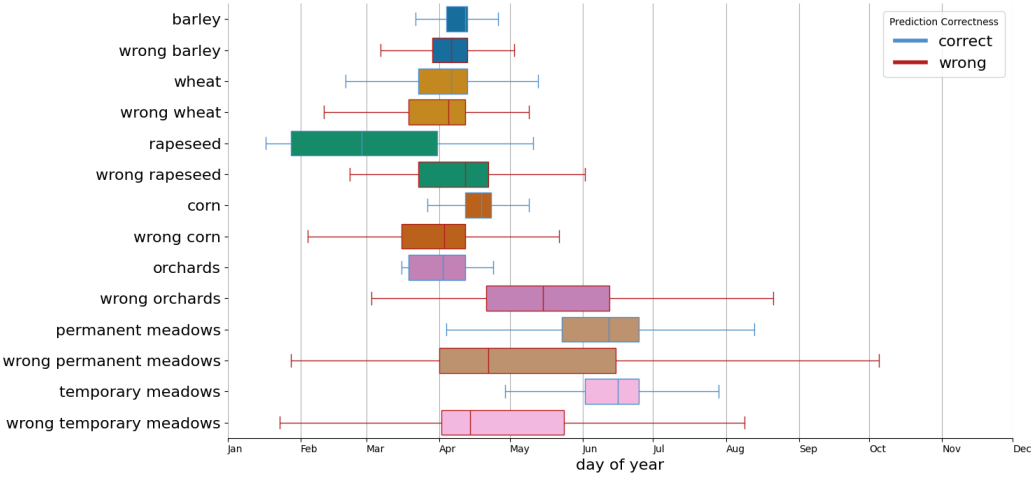
Wrong pred. penalty	Accuracy	Earliness	Harmonic Mean	STD score
v1	0.74	0.60	0.67	27.42
v2	0.58	0.65	0.61	46.98



(a) Stopping times assessment of the best performing model.



(b) Confusion matrix, normalized over the true labels (columns).



(c) Stopping times box plots, with respect to the crop type and the correctness of the prediction. Blue, respectively red, whiskers, indicates the stopping times of correct, resp. incorrect, predictions. A wrong crop type box plot depicts the stopping time distribution of samples that should have been predicted as that crop type, but were predicted wrongly.

Figure 29: Performances assessment of the best model with v2 on the test dataset.

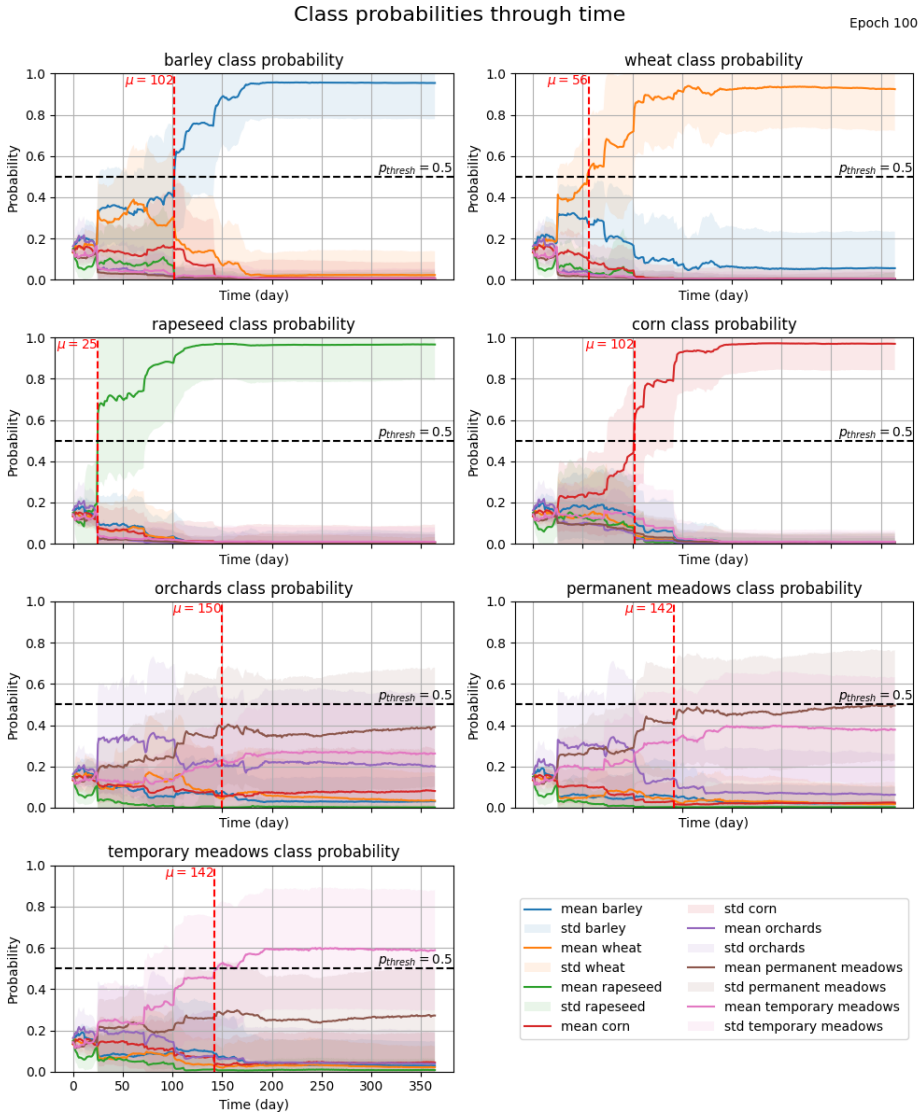


Figure 30: Class probabilities with respect to time. Each subplot corresponds to the probabilities associated to one true label. The mean and the standard deviation are plotted in colors relating to the true crop types. Moreover, the probability threshold $p_{thresh} = 0.5$ is indicated as a black horizontal line. For each subplot, the corresponding μ_c is drawn as a red dashed vertical line.

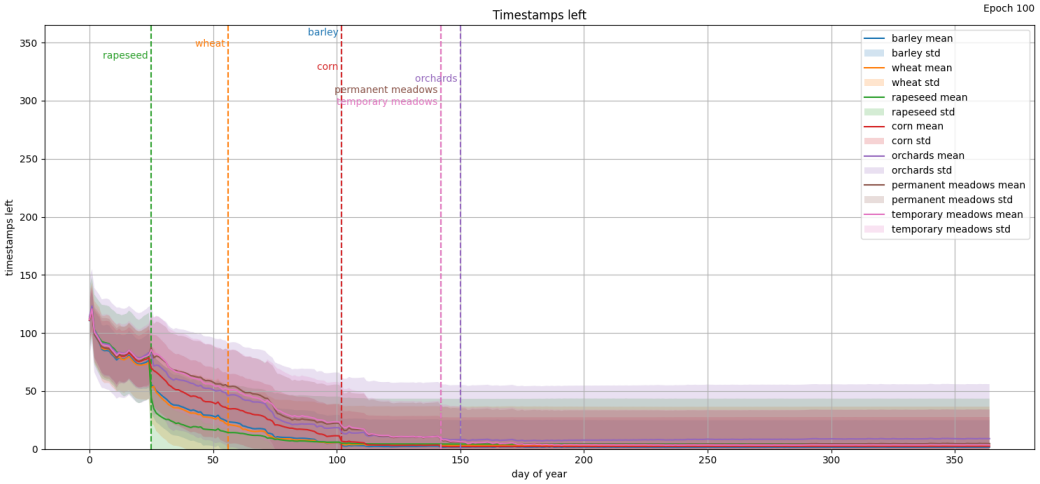


Figure 31: Timestamps left per class, with respect to time. Each full colored line corresponds to the mean of the timestamps left of a certain crop type. The standard deviation is plotted around it in clear. The values μ_c are indicated as vertical lines for each label $c \in C$.

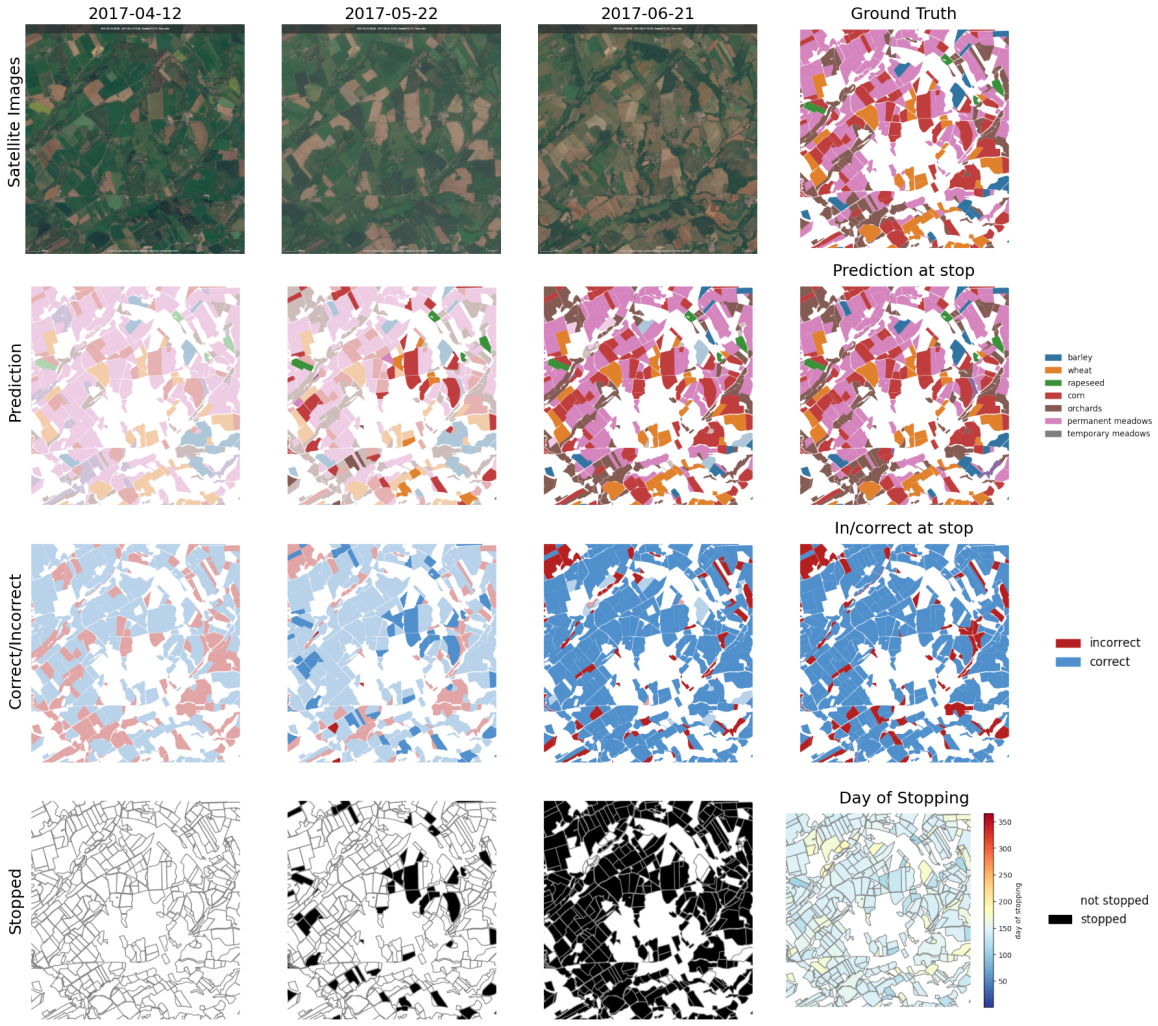
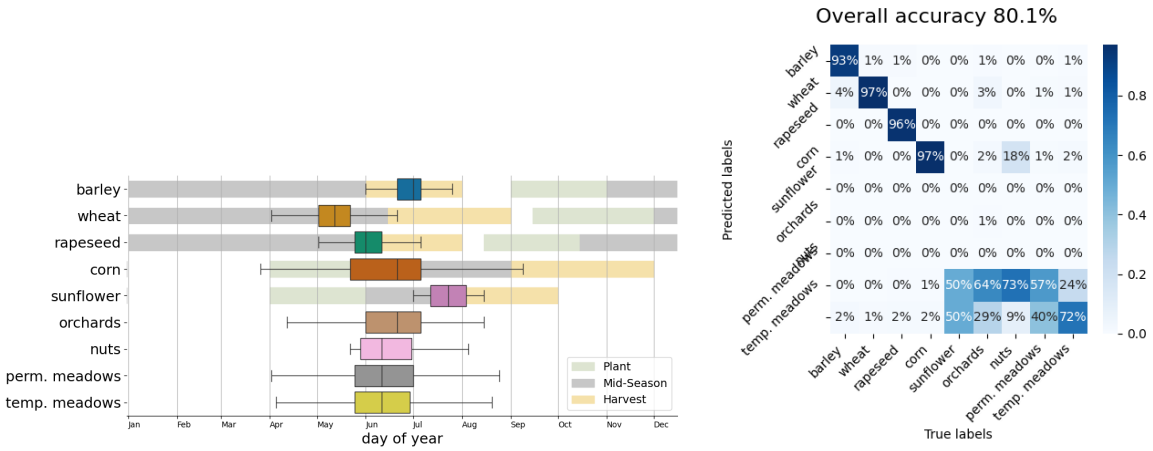


Figure 32: The D-ELECTS prediction process is shown over the span of 3 months. The model predicts class labels and stops the predictions of individual fields. The focus here is on a 3.7 km × 3.9 km site in the Brittany test region. The three first columns present the predictions at three dates. The last column presents the final results. Concerning the rows, first row shows the satellite images of the L1C layer and the ground truth. The second row presents the predictions through the year, whereas the third row present the correct/incorrect map. On both rows, transparency indicates that the model has not decided to stop yet. The last row shows a binary score to indicate which classifications have been stopped (black) or not (white). Finally, the bottom right image shows the stopping day for all parcels.

Appendix B Discussion

B.1 ELECTS Discussion

We present the results with a common sequence length of 110 on the test dataset.



(a) Quantitative assessment of stopping times for different crop types in Brittany, France. Planting, mid-season and harvesting periods according to the USDA Foreign Agricultural Services crop calendar for France are also plotted.

(b) Confusion matrix of BreizhCrops dataset.

Figure 33: Results of the ELECTS model on the Breizhcrops dataset with common sequence length of 110.

Table 8: Performances of the reproduced ELECTS model on the test set, with common sequence length .

Model	Common Sequence Length	Accuracy	Earliness	Harmonic Mean	STD score
ELECTS	150	0.80	0.85	0.83	7.05
ELECTS	110	0.80	0.80	0.80	6.86

B.2 Best Model for each Wrong-Prediction Penalty

We present two figures about the performances of the model with v2, in order to complete the discussion about the best model for each wrong-prediction penalty.

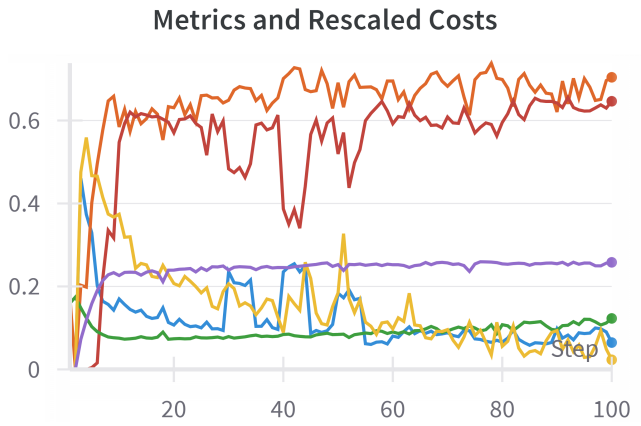


Figure 34: Metrics and rescaled costs of the performance of the best model with v2. Legend: accuracy (orange), earliness (red), wrong-prediction penalty $C_p \cdot 10^{-2}$ (yellow), earliness reward $C_d \cdot 10^{-1}$ (purple), piece-wise linear regression cost $C_{lr} \cdot 10^{-1}$ (blue), misclassification cost $C_m \cdot 10^{-2}$ (green).

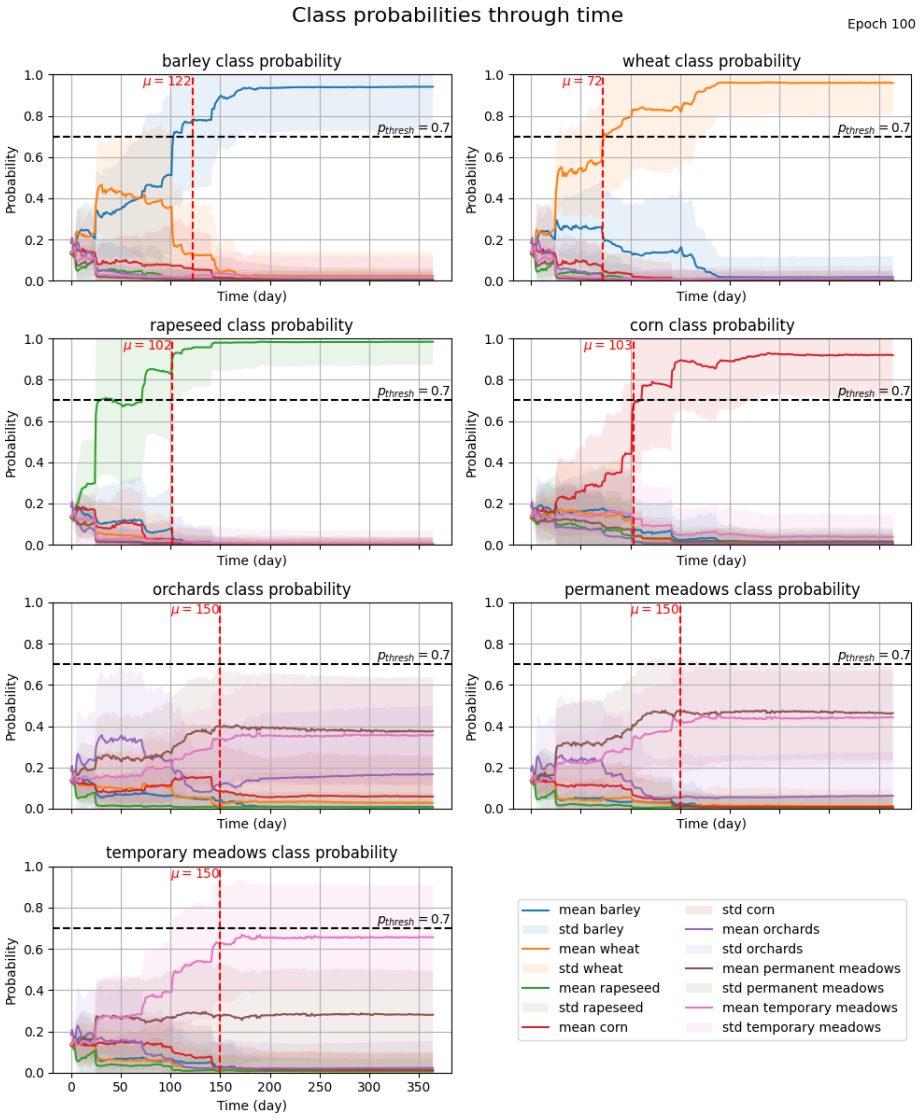
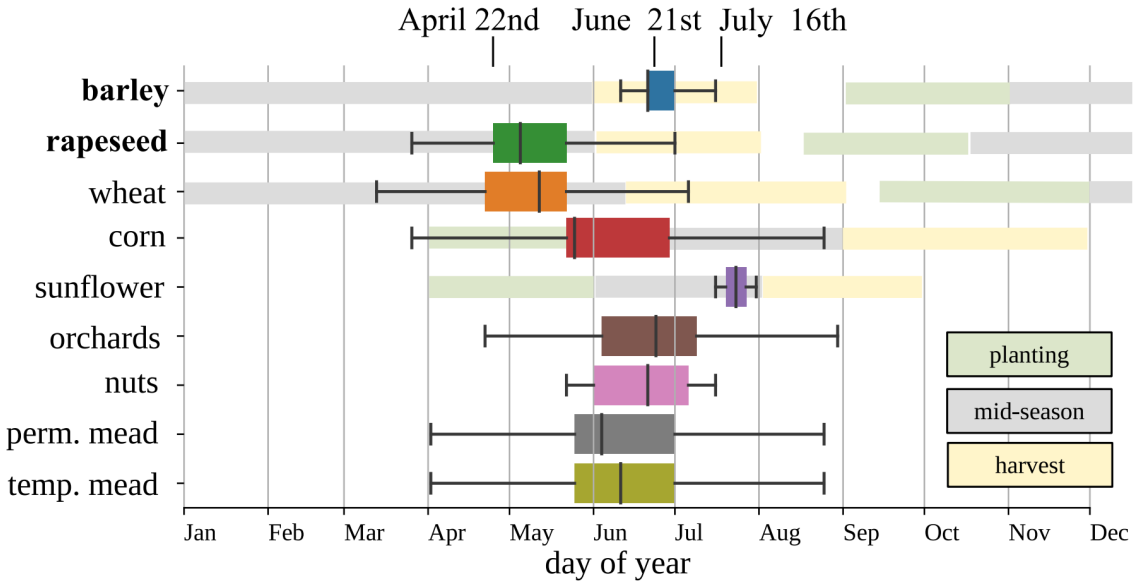


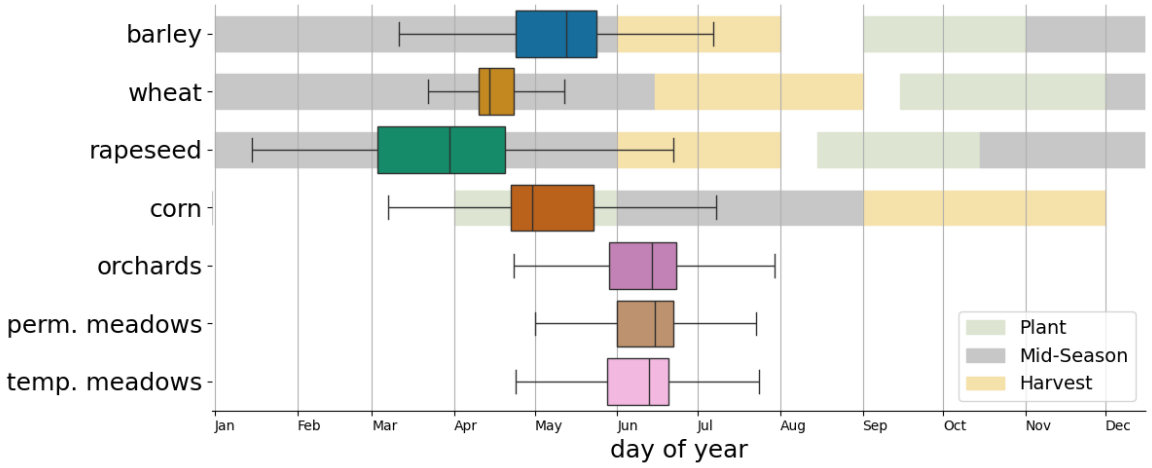
Figure 35: Class probabilities on the validation set, with v2 penalty.

B.3 Comparison of D-ELECTS and ELECTS Discussion

A comparison of the stopping time boxplots is given in Figure 36.



(a) Stopping time boxplots of the ELECTS original paper.

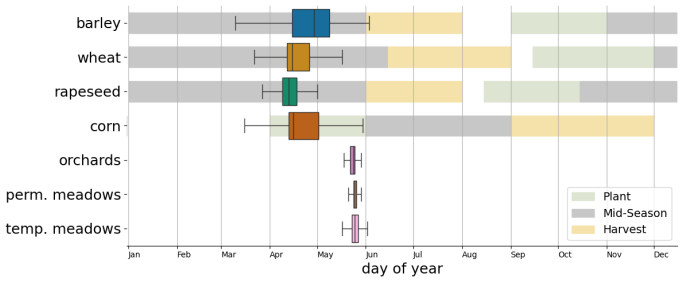


(b) Stopping times assessment of our best performing model. Planting, mid-season and harvesting periods according to the USDA Foreign Agricultural Services crop calendar for France are also plotted.

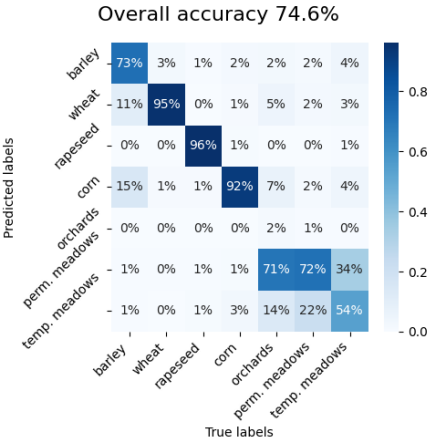
Figure 36: Comparison of the stopping times boxplots. We put the two figures presented in results (Figure 13a and Figure 22a) next to each other to facilitate the comparison.

B.4 Future Research

The results of the D-ELECTS model trained on both the training and validation datasets of Reduced BreizhCrops are presented in Figure 37.



(a) Quantitative assessment of stopping times for different crop types in Brittany, France. Planting, mid-season and harvesting periods according to the USDA Foreign Agricultural Services crop calendar for France are also plotted.



(b) Confusion matrix of Reduced BreizhCrops test dataset.

Figure 37: Results of the D-ELECTS model on the Reduced Breizhcrops test dataset. The model was trained on both the training and validation sets.

Acknowledgement

First, I would like to thank Dr. Lukas Valentin Graf, Dr. Gregor Perich, and Dr. Michele Volpi for their extraordinary support throughout this Master thesis. I am grateful for Gregor's and Lukas' agricultural expertise and supportive input, and especially to Lukas guidance towards the specific topic of the thesis. I am also thankful for Michele's machine learning expertise, enthusiasm, and his supportive words through the weekly meetings. Second, I would like to express my deepest appreciation for Prof. Simone Deparis for his encouragement and guidance, as well as for supporting me since my Bachelor studies at École Polytechnique Fédérale de Lausanne. Third, I would like to thank Juraj for our enriching exchanges about our theses and his friendliness. Finally, I would like to express my gratitude to my family, fiancé, and friends, for their encouragement and support throughout my studies.

Acronyms

HM Harmonic Mean between Accuracy and Earliness. 12, 13

D-ELECTS Daily ELECTS. 1–3, 13–15, 17, 19–24, 26, 28, 31, 36, 38–40, 42–45, 53, 54

D-ELECTS-v1 Daily ELECTS with the first version of the wrong-prediction penalty. 31, 41, 42

ECMTS Early Classification of Multivariate Time Series. 1, 4, 12

ELECTS End-to-end Learned Early Classification of Time Series. 1–5, 13–17, 22, 23, 31, 36, 38, 39, 42–44, 50, 53

ESA the European Space Agency. 2, 6

LSTM Long Short-Term Memory. 14, 16, 42

ML Machine Learning. 2

MTS Multivariate Time Series. 4, 12, 15, 36

NIR Near-Infra-Red. 7

NN Nearest Neighbor. 4

NUTS the Nomenclature of Territorial Units for Statistics. 7, 8, 10

RNN Recurrent Neural Network. 14

SAR Synthetic-Aperture Radar. 42, 44

SWIR Short Wave Infra-Red. 7

v1 First Wrong-Prediction Penalty. 17, 26–30, 40, 41

v2 Second Wrong-Prediction Penalty. 1, 18, 26–30, 40, 41, 45, 46, 51

References

[1] S2 mission - sentiwiki. https://sentiwiki.copernicus.eu/web/s2-mission. Accessed: 2024-04-04.

[2] Charilaos Akasiadis, Evgenios Kladis, Evangelos Michelioudakis, Elias Alevizos, and A. Artikis. Early time-series classification algorithms: An empirical comparison. *ArXiv*, abs/2203.01628, 2022.

[3] Jakub Michał Bilski and Agnieszka Jastrzebska. Calimera: A new early time series classification method. *Information Processing Management*, 60(5):103465, 2023.

[4] Aníbal Bregón, María Aránzazu Simón Hurtado, Juan José Rodríguez Diez, Carlos J. Alonso, Belarmino Pulido Junquera, and Q. Isaac Moro. Early fault classification in dynamic systems using case-based reasoning. In *Conferencia de la Asociación Española para la Inteligencia Artificial*, 2005.

[5] Yaping Cai, Kaiyu Guan, Jian Peng, Shaowen Wang, Christopher Seifert, Brian Wardlow, and Zhan Li. A high-performance and in-season classification system of field-level crop types using time-series landsat data and a machine learning approach. *Remote Sensing of Environment*, 210:35–47, 2018.

[6] Huiling Chen, Aosheng Tian, Ye Zhang, and Yuzi Liu. Early time series classification using tcn-transformer. In *2022 IEEE 4th International Conference on Civil Aviation Safety and Information Technology (ICCASIT)*, pages 1079–1082, 2022.

[7] Rick Mueller Claire Boryan, Zhengwei Yang and Mike Craig. Monitoring us agriculture: the us department of agriculture, national agricultural statistics service, cropland data layer program. *Geocarto International*, 26(5):341–358, 2011.

[8] Hui Ding, Goce Trajcevski, Peter Scheuermann, Xiaoyue Wang, and Eamonn J. Keogh. Querying and mining of time series data: experimental comparison of representations and distance measures. *Proc. VLDB Endow.*, 1:1542–1552, 2008.

[9] NB Dise, M Ashmore, S Belyazid, A Bleeker, R Bobbink, W De Vries, JW Erisman, T Spranger, CJ Stevens, and LJL Van den Berg. 20-nitrogen as a threat to european terrestrial biodiversity. 2011.

[10] Loubna Elmansouri. Multiple classifier combination for crop types phenology based mapping. In *2017 International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, pages 1–6, 2017.

[11] Eurostat. Statistics explained, agricultural production - crops. https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Agricultural_production_-_crops, 2023. Accessed: 2024-07-17.

[12] Saskia Foerster, Klaus Kaden, Michael Foerster, and Sibylle Itzerott. Crop type mapping using spectral–temporal profiles and phenological information. *Computers and Electronics in Agriculture*, 89:30–40, 2012.

[13] Mohamed F Ghalwash, Dušan Ramljak, and Zoran Obradović. Early classification of multivariate time series using a hybrid hmm/svm model. In *2012 IEEE International Conference on Bioinformatics and Biomedicine*, pages 1–6. IEEE, 2012.

[14] Lukas Valentin Graf, Quirina Noëmi Merz, Achim Walter, and Helge Aasen. Insights from field phenotyping improve satellite remote sensing based in-season estimation of winter wheat growth and phenology. *Remote Sensing of Environment*, 299:113860, 2023.

[15] J. Han, Z. Zhang, Y. Luo, J. Cao, L. Zhang, J. Zhang, and Z. Li. The rapeseedmap10 database: annual maps of rapeseed at a spatial resolution of 10 m based on multi-source data. *Earth System Science Data*, 13(6):2857–2874, 2021.

[16] Pengyu Hao, Yulin Zhan, Li Wang, Zheng Niu, and Muhammad Shakir. Feature selection of time series modis data for early crop classification using random forest: A case study in kansas, usa. *Remote Sensing*, 7(5):5347–5369, 2015.

[17] Felix Herzog, Volker Prasuhn, Ernst Spiess, and Walter Richner. Environmental cross-compliance mitigates nitrogen and phosphorus pollution from swiss agriculture. *Environmental Science Policy*, 11(7):655–668, 2008.

[18] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 11 1997.

[19] Huai-Shuo Huang, Chien-Liang Liu, and Vincent S. Tseng. Multivariate time series early classification using multi-domain deep neural network. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 90–98, 2018.

[20] Mitchell C. Hunter, Richard G. Smith, Meagan E. Schipanski, Lesley W. Atwood, and David A. Mortensen. Agriculture in 2050: Recalibrating targets for sustainable intensification. *BioScience*, 67:386–391, 2017.

[21] Jordi Inglada, Arthur Vincent, Marcela Arias, and Claire Marais-Sicre. Improved early crop type identification by joint use of high temporal resolution sar and optical image time series. *Remote Sensing*, 8(5), 2016.

[22] Lukas Kondmann, Sebastian Boeck, Rogerio Bonifacio, and Xiao Xiang Zhu. Early crop type classification with satellite imagery-an empirical analysis. 2022.

[23] Junwei Lv, Xuegang Hu, Lei Li, and Peipei Li. An effective confidence-based early classification of time series. *IEEE Access*, 7:96113–96124, 2019.

[24] Nando Metzger, Mehmet Ozgur Turkoglu, Stefano D'Aronco, Jan Dirk Wegner, and Konrad Schindler. Crop classification under varying cloud cover with neural ordinary differential equations. *CoRR*, abs/2012.02542, 2020.

[25] yasmina imani Omar Bourja Ouiam Lahlou Yahya Zennayi François Bourzeix Ismaguil Hanadé Houmma Mouad Alami Machichi, loubna El mansouri and Rachid Hadria. Crop mapping using supervised machine learning and deep learning: a systematic literature review. *International Journal of Remote Sensing*, 44(8):2717–2753, 2023.

[26] Charlotte Pelletier, Geoffrey I Webb, and François Petitjean. Temporal convolutional neural network for the classification of satellite image time series. *Remote Sensing*, 11(5):523, 2019.

[27] Gregor Perich. *Satellite-based estimation of crop nitrogen and yield in Switzerland*. PhD thesis, ETH Zurich, 2023.

[28] Copernicus Programme. Copernicus programme, 2024.

[29] Matej Račič, Krištof Oštir, Anže Zupanc, and Luka Čehovin Zajc. Multi-year time series transfer learning: Application of early crop classification. *Remote Sensing*, 16(2):270, 2024.

[30] Juan J. Rodríguez, Carlos J. Alonso, and Henrik Boström. Boosting interval based literals. *Intell. Data Anal.*, 5(3):245–262, aug 2001.

[31] Marc Rußwurm and Marco Körner. Multi-temporal land cover classification with sequential recurrent encoders. *ISPRS International Journal of Geo-Information*, 7(4):129, 2018.

[32] Marc Rußwurm, Charlotte Pelletier, Maximilian Zollner, Sébastien Lefèvre, and Marco Körner. Breizhcrops: A time series dataset for crop type mapping. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences ISPRS (2020)*, 2020.

[33] Marc Rußwurm, Romain Tavenard, Sébastien Lefèvre, and Marco Körner. Early classification for agricultural monitoring from satellite time series. *arXiv preprint arXiv:1908.10283*, 2019.

[34] Marc Rußwurm, Nicolas Courty, Rémi Emonet, Sébastien Lefèvre, Devis Tuia, and Romain Tavenard. End-to-end learned early classification of time series for in-season crop type mapping. *ISPRS Journal of Photogrammetry and Remote Sensing*, 196:445–456, 2023.

[35] Marc Rußwurm and Marco Körner. Self-attention for raw optical satellite time series classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 169:421–435, 2020.

[36] Vivien Sainte Fare Garnot, Loic Landrieu, Sebastien Giordano, and Nesrine Chehata. Satellite image time series classification with pixel-set encoders and temporal self-attention. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12322–12331, 2020.

[37] Dimitrios Sykas, Maria Sdraka, Dimitrios Zografakis, and Ioannis Papoutsis. A sentinel-2 multiyear, multicountry benchmark dataset for crop classification and segmentation with deep learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:3323–3339, 2022.

[38] M.C. Tirado, R. Clarke, L.A. Jaykus, A. McQuatters-Gollop, and J.M. Frank. Climate change and food safety: A review. *Food Research International*, 43(7):1745–1765, 2010. Climate Change and Food Science.

[39] Devis Tuia, Konrad Schindler, Begüm Demir, Gustau Camps-Valls, Xiao Xiang Zhu, Mrinalini Kochupillai, Sašo Džeroski, Jan N van Rijn, Holger H Hoos, Fabio Del Frate, et al. Artificial intelligence to advance earth observation: a perspective. *arXiv preprint arXiv:2305.08413*, 2023.

[40] United Nations, Department of Economic and Social Affairs, Population Division. World population prospects 2022. UN DESA/POP/2022/TR/NO. 3, 2022. Summary of Results.

[41] Ziqiao Wang, Hongyan Zhang, Wei He, and Liangpei Zhang. Phenology alignment network: A novel framework for cross-regional time series crop classification. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2934–2943, 2021.

[42] Frank Weilandt, Robert Behling, Romulo Goncalves, Arash Madadi, Lorenz Richter, Tiago Sanona, Daniel Spengler, and Jona Welsch. Early crop classification via multi-modal satellite data fusion and temporal attention. *Remote Sensing*, 15(3), 2023.

[43] M. Weiss, F. Jacob, and G. Duveiller. Remote sensing for agricultural applications: A meta-review. *Remote Sensing of Environment*, 236:111402, 2020.

[44] Xiaopeng Xi, Eamonn Keogh, Christian Shelton, Li Wei, and Chotirat Ann Ratanamahatana. Fast time series classification using numerosity reduction. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 1033–1040, New York, NY, USA, 2006. Association for Computing Machinery.

[45] Zheng zheng Xing, Jian Pei, and Philip Yu. Early prediction on time series: A nearest neighbor approach. pages 1297–1302, 01 2009.

[46] Yuan Yuan and Lei Lin. Self-supervised pretraining of transformers for satellite image time series classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:474–487, 2021.