Performance Assessment of Within-Season Crop Type Mapping Methods Using Satellite Image Time Series

Rens Brouwer (s4059255)

Supervisors:
Dr. S. Mohammadi & Dr. A. Saxena

Bachelor Thesis– Data Science and Artificial Intelligence

Leiden Institute of Advanced Computer Science (LIACS)
liacs.leidenuniv.nl

July 16, 2026

Abstract

Accurate within-season crop type mapping is essential for proactive agricultural management, early warning of agricultural production shortfalls, and food security assessment. While deep learning architectures have achieved state-of-the-art performance on full-season Satellite Image Time Series (SITS), their ability to produce accurate and timely predictions under early-season data constraints remains underexplored. This thesis evaluates the performance of different crop-mapping models under early-season data constraints, with unavailable future timesteps addressed through different data-handling strategies. Three architectural paradigms, being Traditional Non-Sequential Machine Learning Classifiers, Sequential Deep Learning and Spatio-Temporal Models were benchmarked using multiple data-handling strategies to determine how early in a season they reach 90% of their full-season accuracy.

The baseline full-season evaluations revealed an interesting trend: traditional static algorithms (specifically the Support Vector Machine and Multi-Layer Perceptron) compared favorably to complex Spatio-Temporal Models when the complete sequence was available. For within-season classification, the results highlighted a trade-off. The Multi-Layer Perceptron (MLP) marginally achieved the best absolute early-season operational utility over the 3D Convolutional Neural Network (3D CNN). However, when within-season performance was normalized relative to each model's own full-season score, the 3D CNN paired with Last Value Imputation ranked slightly above the MLP, achieving 90% of its full-season performance ceiling by Week 12 of the complete 24-week growing season compared to week 13 for the MLP. The margins between these two models are small and neither should be framed as outright superior to the other. Furthermore, this study demonstrates that the early-season performance of architectures requiring fixed-length temporal inputs is highly dependent on the chosen imputation strategy. Mean Imputation delayed convergence by up to six weeks compared to Last Value Imputation due to signal dilution, while feeding the Attention-LSTM truncated sequences caused its performance to collapse.

This research concludes that analyzing spatial neighborhoods is especially useful for making up for missing future data. This research also indicates that a model's early-season success is very dependent on how missing data is handled, meaning architectures and imputation strategies must be evaluated as a combined framework.

Contents

1 Introduction

Crop type mapping is used for predicting crop yields at scale, which is essential for assessing available agricultural resources and ensuring food security. Additionally, it is used to monitor if agricultural production is meeting targets, predict and help prevent disasters [BYMC11], and manage agricultural supply chains. This process commonly relies on Satellite Image Time Series (SITS), which consist of repeated satellite observations of a geographic area over time, allowing researchers to follow a crop's specific phenology throughout a season [LCC+22]. SITS are commonly generated from data acquired by Earth observation missions like Sentinel-1, Sentinel-2 [SSL+12], and Landsat [RWL+14].

For many years, the state-of-the-art in this field relied on traditional machine learning algorithms, specifically architectures like the Support Vector Machine (SVM) and the Random Forest (RF) [PVI+16]. These methods were primarily used for mapping crops at the end of the season usually with manually engineered features, but they failed to take full advantage of the chronological, temporal nature of SITS data. Recently, deep-learning methods have become increasingly important in crop type mapping, particularly sequence-based architectures such as Long Short-Term Memory (LSTM) networks [RK18] and Transformers [VSP+17]. These architectures are capable of automatically extracting high-level spatial and temporal features directly from raw SITS.

This architectural advancement has given rise to within-season crop mapping, which uses only early-season data to make predictions, which is critical for quick responses to extreme weather events and estimating the impacts of climate shifts [KKH+20].

Although numerous models have been proposed for crop type mapping, comparative evaluations of widely used approaches focusing specifically on within-season prediction performance remain limited. Furthermore, for within-season mapping to be practically useful, models must reach a high enough accuracy as early in the season as possible. To address this gap in the literature, this thesis compares a wide array of methods (Including traditional non-sequential machine learning classifiers, sequential deep learning models and spatio-temporal models) on their within-season mapping capabilities to answer the following research question:

> *How early in the growing season do traditional machine-learning and deep-learning crop-mapping methods reach 90% of their full-season performance?*

The remainder of this thesis is structured as follows. Section 2 establishes the theoretical background, detailing the nature of SITS data, classification architectures, and the mechanics of missing data. Section 3 reviews related work, tracing the evolution of crop mapping and the shift toward in-season classification. Section 4 outlines the methodology, including dataset pre-processing, model architectures, and the experimental setup. Section 5 presents the empirical results, evaluating baseline performances, the impact of varying imputation strategies, and absolute and relative earliness rankings. Finally, Section 6 summarizes the conclusions and proposes directions for further research.

2 Preliminaries

This section provides the theoretical foundation necessary to understand the architectures and methodologies utilized in this thesis. It begins by defining the physical nature of the remote sensing data and the biological principles of crop phenology, followed by a mathematical breakdown of the deep learning architectures capable of processing this data. Finally, it explores the computer science mechanics of how neural networks handle incomplete temporal sequences.

2.1 Satellite Image Time Series and Crop Phenology

The foundation of modern large-scale agricultural monitoring relies on Earth Observation (EO) data. While a single, static satellite image captures a geographic area at a specific moment, it is often insufficient for distinguishing between vegetation types. For example, in the early spring, fields of corn and fields of soybean may both appear as identical patches of bare soil.

To resolve this ambiguity, researchers utilize Satellite Image Time Series (SITS) data. A SITS is generated by stacking multiple satellite observations of the exact same geographical location in chronological order. Optical satellite missions, such as the NASA/USGS Landsat program [RWL+14] and the European Space Agency's Sentinel-2 [SSL+12], are specifically designed for this purpose. The dataset utilized in this research is derived specifically from the Landsat 8 Operational Land Imager (OLI) and Landsat 7 Enhanced Thematic Mapper Plus (ETM+) sensors. These instruments capture non-spatially overlapping, high-quality multi-spectral imagery at a 30-meter spatial resolution. For the purpose of crop mapping, the temporal sequences are constructed using six distinct optical bands: blue, green, red, near-infrared (NIR), shortwave infrared 1 (SWIR-1), and shortwave infrared 2 (SWIR-2). By recording electromagnetic reflectance across these specific bands with a frequent revisit time (typically 8 to 16 days depending on satellite constellation overlap), Landsat allows for the continuous monitoring of the Earth's surface throughout a growing season.

The primary objective of compiling a SITS for crop mapping is to capture the phenology of the vegetation. Phenology refers to the study of cyclic and seasonal natural phenomena, especially in relation to climate and plant life. In an agricultural context, a crop's phenological cycle includes several distinct stages: planting (emergence from bare soil), vegetative growth (increasing leaf area and chlorophyll content), reproductive maturity (peaking greenness and flowering), and finally, senescence (browning) and harvest [Mei18].

Each of these biological stages interacts differently with the electromagnetic spectrum. As a crop grows, its chlorophyll absorbs visible red light while its cellular structure highly reflects near-infrared light. By tracking these spectral reflectance values over time, a SITS generates a unique "temporal signature" or growth curve for each pixel [ZWX+20].

Because different crop types possess distinct genetic traits, their phenological trajectories vary. Winter wheat, for instance, peaks in spectral reflectance much earlier in the year than late-summer crops like maize or sunflower [GZ21]. Therefore, by utilizing the temporal dimension provided by a SITS, machine learning models can differentiate between crops not just by what they look like, but by *how they change* over time. Understanding this dynamic temporal evolution forms the basis for in-season, early crop classification.

2.2 Architectures in Time Series Classification

The fundamental challenge in crop type mapping from a Satellite Image Time Series (SITS) lies in effectively extracting patterns from multi-dimensional data. As established in [Sec. 2.1], the spectral reflectance of a crop changes dynamically over the growing season. Therefore, a classification algorithm must be able to interpret not just the isolated spectral values, but the chronological progression of those values.

The architectures relevant to this study are categorized into three groups based on how they process this temporal dimension: Traditional Non-Sequential Machine Learning Classifiers (representing established static baselines), Sequential Deep Learning Models (capable of tracking chronological phenology), and Spatio-Temporal Models (which integrate temporal dynamics with physical neighborhood context).

2.2.1 Traditional Non-Sequential Machine Learning Classifiers

Historically, traditional machine learning algorithms have served as the standard baselines for remote sensing classification tasks. These architectures share a defining structural limitation: they are inherently "static". They do not possess an internal mathematical mechanism to process chronological order, memory, or sequential dependencies.

Despite this limitation, they function via robust mathematical principles that make them highly effective for general classification. This study evaluates the following models

- **Random Forest (RF):** Introduced by Breiman [Bre01], the RF is an ensemble learning method that constructs a multitude of decision trees during training. It utilizes "bagging" (bootstrap aggregating) to train each tree on a random subset of the data and a random subset of features. The final classification is determined by a majority vote across all trees, making the model more resistant to overfitting and noise.
- **Support Vector Machine (SVM):** The SVM operates by finding the optimal geometric hyperplane that maximizes the margin between different classes [CV95]. For non-linearly separable data, it employs the "kernel trick" to map the input features into a higher-dimensional space where a linear decision boundary can be drawn.
- **Multi-Layer Perceptron (MLP):** The MLP is a class of feed-forward artificial neural network. It consists of an input layer, one or more hidden layers, and an output layer, with each node fully connected to the next. It learns complex non-linear mappings by adjusting the weights of these connections through backpropagation [RHW86].

While powerful, these functional mechanics dictate how data must be fed to the algorithm. To utilize a SITS within a static architecture, the sequential data must undergo a structural transformation. For a given physical location, a time series is naturally represented as a 2D matrix of shape $T \times F$, where T represents the number of sequential temporal observations and F represents the number of spectral features per observation.

As static models require a standard one-dimensional input array for their decision nodes, hyperplanes, or fully connected layers, this temporal matrix must be mathematically flattened into a single, static vector containing exactly $T \times F$ elements. By concatenating the sequence into a 1D vector, the temporal dimension is lost. The model evaluates the spectral bands from the first observation and the final observation simultaneously, treating them as unordered variables rather than a continuous biological progression.

Consequently, these architectures cannot explicitly track the trajectory of a crop's phenological curve: they rely entirely on the flattened values. This structural inability to organically understand the passage of time justifies the field's more recent transition toward sequence modeling.

2.2.2 Sequential Deep Learning Models

To overcome the structural limitations of static architectures, the field of time-series analysis increasingly relies on Sequential Deep Learning models. Instead of flattening the temporal dimension into a static vector, these models are explicitly designed to ingest the sequential 2D matrix ($T \times F$) in its native chronological order. By doing so, they can isolate specific phenological events such as rapid spring green-up or gradual autumn senescence without seeing them as simultaneous data points.

Four primary Sequential Deep Learning Models are relevant to this study:

- **1D Convolutional Neural Network (1D CNN):** While traditional 2D CNNs are used for spatial image processing, 1D CNNs apply convolutional filters exclusively across the temporal axis. By sliding a kernel of a fixed size over the sequential timesteps, the network extracts localized temporal features. This makes the 1D CNN highly effective at identifying rapid, short-term spectral changes within the growing season.

- **Long Short-Term Memory (LSTM):** Proposed by Hochreiter and Schmidhuber [HS97], the Attention-LSTM is a specialized Recurrent Neural Network (RNN) designed to process entire sequences of data while mitigating the vanishing gradient problem. An LSTM processes the time series step-by-step. At each timestep t, the network takes the current spectral observation (x_t) and combines it with the "hidden state" (h_{t-1}) and "cell state" (c_{t-1}) from the previous timestep [Fig. 1].

  This recurrent mechanism allows the network to maintain a continuous chronological memory. It can learn dependencies between early-season and later-season spectral patterns. Because the hidden state relies strictly on the previous timestep, LSTMs require a continuous chain of inputs to meaningfully update their internal memory.

- **The Transformer:** Introduced by Vaswani et al. [VSP+17], the Transformer architecture bypasses recurrent step-by-step processing entirely. Instead, it relies on a "Self-Attention" mechanism. Self-attention calculates attention weights between every timestep in the sequence simultaneously, allowing the model to dynamically focus on the most discriminative phenological periods regardless of their distance from one another in time.

  Because the Transformer does not process data recurrently, it utilizes positional encodings to keep track of chronological order. Furthermore, the architecture can use "Attention Masking." By applying a boolean mask to the attention matrix, the network can be instructed to ignore specific timesteps (such as unobserved future weeks or missing data), preventing them from influencing the classification output.

- **CNN-LSTM:** While 1D CNNs excel at extracting localized temporal features and LSTMs are optimized for modeling long-term sequential memory, the CNN-LSTM hybrid integrates

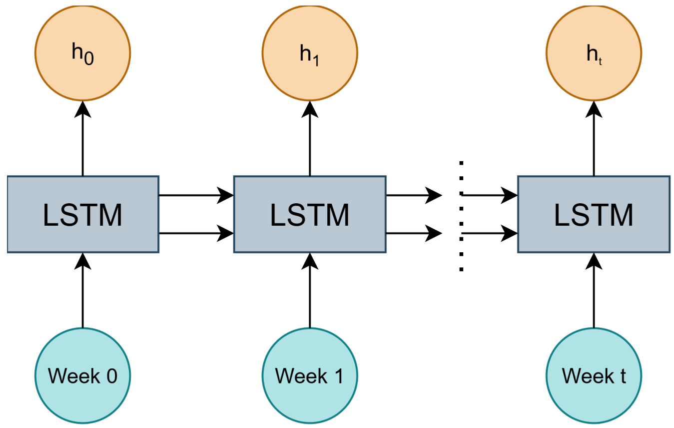


Figure 1: An unrolled Long Short-Term Memory (LSTM) architecture across t weeks. The multi-spectral observations (Weeks 0 to t) are processed sequentially. The network maintains a memory by passing the hidden state (h_t) and cell state horizontally between timesteps. (Adapted from Olah, 2015)

both mechanisms into a sequential pipeline. In this architecture, 1D convolutional layers act as a feature extractor, evaluating the multi-spectral bands to automatically derive high-level spectral representations. The sequence of these extracted feature vectors is then fed into an LSTM network, which maps the development trajectory over the growing season.

2.2.3 Spatio-Temporal Models

While sequential architectures successfully capture the chronological progression of crop phenology, they fundamentally process the landscape on a per-pixel basis. The input SITS is treated strictly as a two-dimensional matrix ($T \times F$) representing a single point on the Earth's surface over time. However, agricultural environments are inherently spatial: a single pixel rarely exists in isolation and is instead often part of a larger field. Neighboring pixels are more likely to contain the same crop type and exhibit similar phenological responses to local micro-climate and soil conditions.

To leverage this neighborhood context, Spatio-Temporal Models expand the network's receptive field by processing multidimensional data blocks. The input data structure is transformed from a single pixel's time series into a sequence of image patches, represented as a tensor of shape $T \times W \times H \times F$, where W and H denote the spatial width and height of the patch. By integrating both the spatial and temporal dimensions, these architectures can reduce classification noise and improve overall robustness.

This study uses two primary architectures within this paradigm:

- **3D Convolutional Neural Network (3D CNN):** While 1D CNNs slide across a single dimension, such as time, and 2D CNNs slide across two spatial dimensions, the 3D CNN [TBF+15] extracts features across both domains simultaneously. It employs a three-dimensional convolutional kernel that slides across spatial width, spatial height, and chronological time concurrently. By computing convolutions in three dimensions, the 3D CNN learns joint spatio-temporal representations natively, capturing complex multidimensional dependencies that purely sequential or purely spatial models cannot observe easily.

- **3D Fully Convolutional Network (3D FCN):** Building upon the standard 3D CNN, the 3D FCN addresses a critical computational bottleneck in mapping large regions. A standard 3D CNN traditionally flattens its final spatio-temporal feature maps into dense, fully connected layers, outputting a single classification label for the entire input patch. The 3D FCN [LSD15] modifies this architecture by removing the dense layers entirely, utilizing exclusively convolutional layers throughout the network. By maintaining a spatially structured output, the 3D FCN can ingest the spatio-temporal tensor and directly output a spatial prediction map ($W \times H \times C$) where every pixel is assigned a crop type simultaneously. This structural shift reduces the number of trainable parameters associated with dense layers and increases computational efficiency for large-scale inference.

2.3 Strategies for Handling Missing Temporal Data

A practical constraint of many artificial neural networks is their requirement for fixed-size input tensors. In a standard SITS crop mapping scenario, a model might be designed to ingest a full growing season, represented by a tensor with a fixed temporal dimension. However, when deploying these models for in-season, early-warning classification, the full sequence is unavailable because later observations have not yet occurred. If an inference is required at week 4, the subsequent weeks represent unobserved, missing data.

To process an incomplete time series without altering the underlying architecture of the network, the missing temporal data must be algorithmically handled. These methodologies broadly fall into two computer science paradigms: Imputation-based methods and Algorithmic Masking.

2.3.1 Statistical Imputation

Imputation strategies estimate replacement values for missing timesteps to satisfy the network's requirement for a complete tensor [LR19]. By filling the unobserved future positions, the network can process the sequence in the same fixed-length format as a full-season input.

- **Zero Padding:** The most basic computational approach is to fill all unobserved future timesteps with zeros. While computationally inexpensive, it introduces artificial values that do not correspond to realistic spectral observations, potentially leading the network to misinterpret the missing timesteps as meaningful signal rather than absent data.

- **Mean Imputation:** This strategy calculates the historical average of the spectral bands for a given location and inserts those statistical means into the missing timesteps. While it smooths the sequence, it risks pulling the unique temporal signature of the real data toward a generic baseline, potentially erasing anomalous early-season indicators.

- **Last Value Imputation:** This technique takes the final observed spectral value (e.g., the data from week 4) and copies it sequentially across all remaining unobserved timesteps. It effectively "freezes" the crop's phenological state. While it prevents the sudden data drops associated with zero padding, it forces the network to interpret an unrealistically flat biological curve.

2.3.2 Algorithmic Masking

Unlike imputation, which estimates replacement values to fill the tensor, masking strategies alter how the network processes the sequence, allowing it to ignore the missing values

- **Variable Sequence Truncation (Raw Slicing):** For recurrent architectures like the LSTM, the strict requirement for a fixed sequence length can sometimes be bypassed by dynamically truncating the input tensor. If only 4 weeks are available, the network is instructed to unroll its hidden states (h_t) for exactly 4 steps and immediately output a classification. However, because the network's weights were optimized to expect a full biological progression, premature truncation can severely disrupt the learned distributions of the final hidden state.
- **Attention Masking:** Supported by Transformer architectures [ZJP+21], this approach preserves the fixed tensor size but modifies the self-attention matrix. A boolean mask is applied to prevent the network from attending to unobserved future timesteps. When the network calculates the Softmax probabilities to determine which weeks are most relevant, the masked weeks are omitted, stopping them from influencing the output. This allows the network to process the incomplete season using only the observed data, removing the need for synthetic imputation methods.

Understanding how these data-handling strategies interact with different architectural structures is crucial. The method chosen to handle the unobserved future influences how accurately and efficiently a network can predict from a truncated dataset.

3 Related Work

The previous section established the mathematical and architectural foundations required to process a Satellite Image Time Series (SITS). However, the application of these architectures to agricultural monitoring has not been static; it has evolved alongside advancements in sensor resolution and computational power. This section reviews the literature surrounding crop type mapping, tracing the paradigm shift from traditional machine learning models to modern spatio-temporal deep learning. Furthermore, it examines the emerging focus on early-season classification, ultimately identifying the gap in current literature regarding how missing data handling strategies influence a model's ability to achieve high performance early in the season.

3.1 The Evolution of Crop Mapping

Historically, the standard approach to large-scale crop type mapping relied heavily on static machine learning algorithms, with the Random Forest (RF) and Support Vector Machine (SVM) emerging as the predominant baselines [PVI+16]. Early SITS classification methodologies rarely utilized raw multi-spectral optical bands directly. Instead, researchers relied on hand-crafted, domain-specific features known as Vegetation Indices (VIs), such as the Normalized Difference Vegetation Index (NDVI). By calculating these indices at various points throughout the year, researchers constructed temporal profiles that static models could use to draw decision boundaries.

While computationally lightweight and highly interpretable, this traditional approach possessed significant limitations. Extracting hand-crafted features requires substantial domain expertise and inherently results in the loss of subtle, multi-spectral information present in the raw sensor data. Furthermore, as established in Section 2.2.1, static architectures are forced to flatten the temporal sequence. Consequently, these early models were unable to explicitly model the chronological progression of crop phenology, relying entirely on the aggregate statistics of the flattened vegetation indices.

The introduction of Deep Learning (DL) to Earth Observation disrupted the use of this methodology. Instead of relying on manual feature engineering, deep learning architectures possess the capacity to automatically learn hierarchical representations directly from multi-temporal satellite data. Seminal works in the field, such as those by Zhong et al. [ZHZ19], demonstrated that a Conv1D-based deep-learning model could outperform traditional machine-learning baselines by automatically learning hierarchical temporal features from Landsat EVI time series.

This transition marked a shift in remote sensing: rather than explicitly telling the algorithm what biological markers to look for, researchers began designing architectures capable of discovering those markers by itself. By feeding multi-temporal optical observations directly into the networks, the field laid the groundwork for the adoption of advanced sequence models capable of modeling chronological information.

3.2 Advancements in Spatio-Temporal Sequence Modeling

As deep learning became increasingly adopted for Satellite Image Time Series (SITS) analysis, researchers recognized that treating multi-spectral observations as static, unordered features discarded the most critical biological information: the chronological growth trajectory of the crop.

To address this, the field rapidly adopted Sequential Deep Learning models explicitly designed to process ordered data.

Initial advancements focused heavily on one-dimensional architectures. 1D Convolutional Neural Networks (1D CNNs) were utilized to extract localized temporal patterns, effectively identifying sudden phenological shifts. However, it was the adoption of Recurrent Neural Networks (RNNs), specifically the Long Short-Term Memory (LSTM) network, that marked a significant development. By maintaining recurrent states across the time series, LSTMs could successfully map the long-term dependencies of a full agricultural season. Rußwurm and Körner [RK18] demonstrated that sequential recurrent encoders could outperform static baselines on optical time series by natively modeling these temporal dynamics.

As the complexity of crop mapping tasks increased, researchers began addressing the inherent memory bottlenecks of standard recurrent architectures. Xu et al. [XYX+21] conducted a comprehensive evaluation of multi-temporal deep learning models, explicitly highlighting the power of Attention mechanisms. By augmenting LSTMs with attention, or by utilizing purely self-attention-based Transformer architectures, networks were no longer forced to compress the entire sequence into a single final hidden state. Instead, they could assign higher mathematical weights to the most discriminative phenological periods.

While sequence models successfully captured chronological phenology, many of these architectures processed the landscape on a pixel-by-pixel basis. An advancement within this 1D sequential paradigm was introduced by Xu et al. [XZZ+20], who developed DeepCropMapping. This framework utilized a modified Attention-LSTM architecture to achieve robust spatial generalizability, allowing models trained in one geographic region to be successfully transferred and applied to different regions. By dynamically weighting the importance of different phenological stages, their approach set a high standard for interpreting complex, multi-temporal crop signatures without relying on spatial neighborhood data.

Despite the success of advanced sequential models, pixel-level spectral time series cannot explicitly model the spatial structure of agricultural fields. To overcome this limitation, some of the literature shifted toward Spatio-Temporal Models. Researchers began stacking spatial image patches into multidimensional tensors to leverage local neighborhood context. By utilizing 3D Convolutional Neural Networks (3D CNNs) to process spatial dimensions and temporal sequences concurrently, algorithms could learn spatio-temporal representations from multi-temporal remote sensing imagery [JZX+18].

3.3 The Shift to In-Season Classification

Despite the advances of spatio-temporal deep learning, full-season crop type classification possesses a fundamental limitation: it is retrospective. Models requiring a complete temporal sequence cannot generate predictions until after harvest has concluded. While historically useful for post-season yield accounting and land-use analysis, this delayed inference offers little actionable intelligence for real-world agricultural stakeholders. Precision agriculture, market forecasting, and food security early-warning systems require accurate crop identification months before the season ends.

Consequently, the focus of agricultural remote sensing literature has increasingly shifted toward also including within-season early classification. The primary challenge of in-season mapping lies in the truncation of the temporal data; an algorithm must confidently classify a pixel using only the partial phenological signature of the spring emergence or early summer growth, entirely lacking the highly discriminative features of peak greenness and senescence.

Recent literature has proven that waiting for these late-season features is not always necessary. Konduri et al. [KKH+20] conducted extensive evaluations of in-season mapping across the United States, demonstrating that machine learning models could accurately identify crop types weeks or even months prior to harvest. Their work established a precedent for evaluating models not just on their absolute full-season accuracy but on their temporal convergence, specifically identifying the point in the season where a model reaches 90% of its maximum reliable performance.

Building upon this, Rußwurm et al. [RCE+23] formalized the Early Classification of Time Series (ECTS) for agricultural mapping. They recognized that standard evaluation metrics failed to capture the operational value of time. To address this, they introduced an "earliness reward" framework, discounting the value of late-season predictions while incentivizing networks to extract discriminative features as early in the sequence as possible.

This shift in the literature provides a new objective for modern crop mapping. The goal is no longer simply achieving the highest possible accuracy with a complete Satellite Image Time Series, but to make accurate predictions as early in the sequence as possible. In this context, model performance is evaluated through the trade-off between predictive accuracy and earliness. This reflects the operational value of producing actionable crop-type predictions from truncated early-season data.

3.4 Gap Analysis and Problem Statement

A review of the current literature reveals a clear trajectory in agricultural remote sensing: traditional static models have been largely replaced with deep learning architectures capable of explicitly modeling time (LSTMs, Transformers) and joint spatio-temporal context (3D CNNs). At the same time, the operational focus of the field has shifted from retrospective full-season mapping to proactive in-season classification.

However, a critical intersection between these two advancements remains underexplored. While the superiority of Spatio-Temporal Models for full-season mapping is well-documented, their behavior under the data constraints of in-season inference is not fully understood. Specifically, there is a distinct lack of comprehensive benchmarking regarding how different architectural paradigms handle missing future timesteps during early-season classification.

As established in Section 2.3, architectures must utilize disparate mechanics ranging from statistical imputation to algorithmic masking to process an incomplete Satellite Image Time Series. The existing literature rarely isolates how these data-handling mechanics directly influence a model's early-season convergence rate. For instance, while a 3D FCN may outperform a Transformer when the sequence is complete, it is unclear if that relationship holds when both architectures are used to evaluate a truncated sequence using imputation strategies.

This study aims to directly address this gap in the literature. By benchmarking Static Models, Sequential Deep Learning Models and Spatio-Temporal Models against one another under progressive temporal truncation, this research isolates the interaction between architectural design and missing data mechanics. The ultimate objective is to come closer to defining the optimal architectural and data-handling framework for reliable, early-warning crop type classification.

4 Methodology

This section presents the dataset we used, the models we tested and how they were implemented, how training and testing was done and what metrics the models were evaluated on.

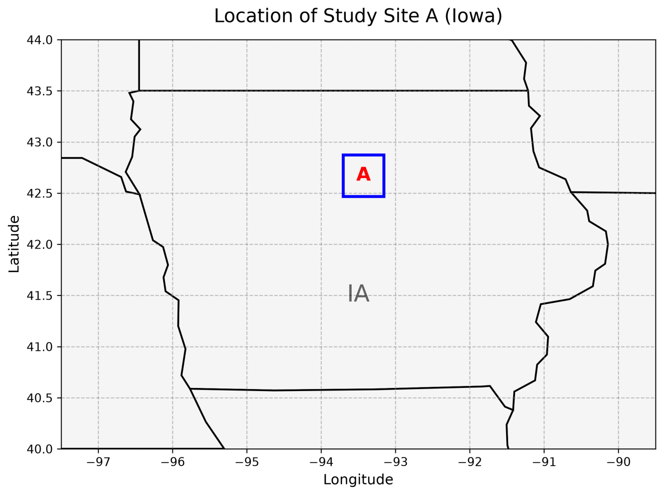


Figure 2: Approximate location of the area in Iowa from which the data was collected.

4.1 Dataset and Preprocessing

The data utilized for this study consisted of Satellite Image Time Series (SITS) derived from Landsat Analysis Ready Data (ARD) surface reflectance composites [MBS24]. Specifically, the dataset combined imagery from the Landsat 8 Operational Land Imager (OLI) and the Landsat 7 Enhanced Thematic Mapper Plus (ETM+) at a 30-meter spatial resolution. The study focused on a primary agricultural region in Iowa, designated as Site A [Fig. 2], to classify three distinct target categories: Corn, Soybean, and a merged "Other" class representing alternative land covers and minor crop types.

Year	Valid Pixels	Validity	Corn	Soybean	Other
2020	2,249,845	99.99%	55.73%	27.91%	16.36%
2021	2,180,602	96.92%	50.60%	32.23%	17.17%

Table 1: Statistics of valid pixels and target class proportions for the Site A (Iowa) dataset during the 2020 and 2021 growing seasons [MBS24].

For each observation, six optical bands were extracted to capture the spectral reflectance characteristics of the vegetation: blue, green, red, near-infrared (NIR), shortwave infrared 1 (SWIR1), and shortwave infrared 2 (SWIR2). In the original dataset, the temporal sequence spanned 28 weeks at seven-day intervals. The data was preprocessed following a methodology established by Xu et al. [XZZ+20]: pixel observations labeled as cloudy, shadow, filled or unclear were discarded, and any pixels which did not have at least two valid observations in every third of the growing season were marked as invalid and filtered out of the dataset, resulting in the statistics shown in Table 1. Any remaining pixels with missing observations were linearly interpolated to fill in the gaps

The original dataset covers the period between the 20th of March and the 25th of September, because it included areas where planting started in March and areas where harvesting started in September. Because this study focuses only on the Iowa dataset, where planting does not occur until mid-April, the decision was made to trim the first 4 weeks from the data [Fig. 3]. This reduces the amount of noise in the dataset as the first 4 weeks do not contain any crops to classify.

To prepare the raw Landsat optical bands, the pixel values were calibrated to true Surface Reflectance. The standard USGS scaling factor was applied to all temporal observations ($x_{calibrated} = x \times 0.0000275 - 0.2$) [U.S21]. This conversion placed the multi-spectral inputs on a physically meaningful reflectance scale, reducing differences in numerical magnitude across bands and supporting more stable neural-network training.

Finally, the data was structurally adapted to accommodate the different spatial requirements of the evaluated models. For the baseline machine learning methods, the data was processed as isolated pixel time series, mathematically flattened into 1D arrays of 144 features (24 timesteps $\times$ 6 bands). In contrast, the Sequential Deep Learning Models processed the isolated pixel time series in their original 2D matrix format (24 timesteps $\times$ 6 bands) to preserve the chronological progression of the observations. However, to evaluate the 3D FCN and 3D CNN, local spatial context was required. Spatial patches were generated using a spatial validity mask, which identified the locations of valid pixels and allowed the physical 2D grid of the study area to be reconstructed. This allowed for the extraction of 3x3 spatial windows around each target pixel across time, reshaping the input into a 5D tensor suitable for the 3D CNN. To facilitate segmentation for the 3D FCN, the data pipeline was designed to partition the reconstructed grid into larger 64×64 regional tiles. This structuring allowed the 3D FCN to ingest broad spatial landscapes and output prediction maps simultaneously, with any invalid pixels explicitly flagged to be ignored during the calculation of the training loss.

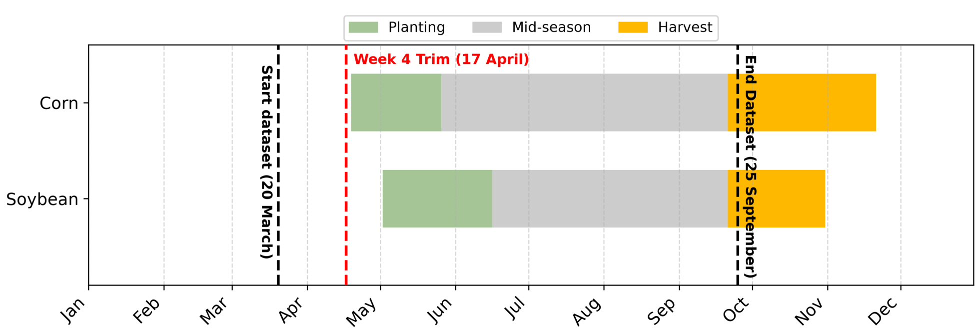


Figure 3: Corn and soybean crop calendar in Iowa based on progress reports [Uni10] with dataset timings [MBS24] in black dotted lines. The red dotted line is the trim date chosen to reduce noise.

4.2 Model Architectures

To evaluate early-season crop type mapping performance, this study used nine distinct classification models. These models were selected to represent the progression of methodologies in the field of Remote Sensing, ranging from established traditional machine learning techniques to modern deep learning architectures. All neural network models were implemented using the PyTorch framework.

Traditional Non-Sequential Machine Learning Classifiers

To establish a robust performance baseline, three foundational classification algorithms were evaluated: Random Forest (RF), Support Vector Machine (SVM), and a Multi-Layer Perceptron (MLP).

The Random Forest classifier was selected because of its proven robustness in handling SITS. As an ensemble learning method, it constructs a great number of decision trees during training and outputs the mode of the individual tree predictions. This architecture makes it inherently more robust against overfitting and capable of managing the complex, high-dimensional feature spaces typical of SITS.

Alongside RF, a Support Vector Machine utilizing a Radial Basis Function (RBF) kernel was implemented. The RBF kernel allows the SVM to project the non-linear temporal spectral signatures into a higher-dimensional space to find an optimal separating hyperplane between crop classes. However, it is important to note that the SVM proved to be computationally heavy; the algorithmic complexity of calculating the kernel matrix across the large volume of pixel samples resulted in significant training and testing delays, highlighting some limitations of traditional kernel methods for large-scale crop mapping.

Finally, a Multi-Layer Perceptron was included to serve as the structural bridge between traditional machine learning and modern deep learning. The MLP is a standard feed-forward artificial neural network consisting of fully connected layers.

Because these models do not possess inherent mechanisms to process sequential or spatial dimensions, the dataset required structural modification prior to training. For each isolated pixel, the temporal matrix of 24 timesteps and 6 spectral bands was mathematically flattened into a single, static 1D feature vector of 144 elements. This flattening process allows the network to evaluate the entire season simultaneously, though it sacrifices the sequential ordering of the phenological stages.

Sequential Deep Learning Models

To explicitly capture the sequential nature of crop phenological development, two foundational Sequential Deep Learning Models were implemented: a 1D Convolutional Neural Network (1D CNN) and an Attention-augmented Long Short-Term Memory (LSTM) network. Unlike the MLP, which treats the sequence as a static block, the 1D CNN preserves the chronological structure by employing convolutional filters that slide exclusively across the time dimension. This mechanism excels at extracting localized temporal features, effectively identifying rapid, short-term phenological shifts in the multi-spectral bands.

Conversely, the Attention-LSTM processes the time series sequentially to model the broader growing season. As a specialized Recurrent Neural Network (RNN), the LSTM mitigates the vanishing gradient problem and allows the network to remember long-term dependencies, mathematically linking early-season planting signatures with mid-season peak growth patterns. The specific implementation of this model, adapted from [XYX+21], utilizes a bidirectional architecture coupled with a custom attention mechanism. By passing the LSTM outputs through an attention layer, the network can dynamically weight the importance of different temporal states before final classification.

To capitalize on the strengths of both convolutional and recurrent architectures, a hybrid CNN-LSTM model was evaluated. This approach utilized 1D convolutional layers as a front-end feature extractor to automatically derive high-level spectral representations from the raw multi-spectral bands at each timestep. The extracted feature sequence was subsequently passed into an LSTM network, which modeled the temporal evolution of these features across the growing season. By partitioning the learning process, the CNN-LSTM aimed to capture both localized phenological patterns and long-term crop development trajectories within a single pipeline.

Building upon the capabilities of these sequence models, a temporal Transformer architecture, also adapted from [XYX+21], was evaluated. While LSTMs process time steps sequentially, which can occasionally lead to information bottlenecking over long sequences, the Transformer relies entirely on a Self-Attention mechanism. Utilizing a custom exponential positional encoding strategy to maintain temporal order, the Self-Attention layers calculate the mathematical correlation between every single timestep in the 24-week sequence simultaneously. This architecture offers a highly parallelized approach, dynamically assigning higher importance weights to the most discriminative phenological periods without relying on recurrent loops.

Spatio-Temporal Models

To address the spatial limitations of the pixel-based models, a 3D Convolutional Neural Network (3D CNN) was introduced to establish a baseline for spatio-temporal feature extraction. The model processes local 3×3 spatial patches over time, utilizing 3D convolutional filters with a 3x3x3 receptive field to simultaneously extract features across both the spatial dimensions and the temporal sequence. To manage the high dimensionality of the resulting feature maps, a 3D Max Pooling operation (using a kernel size of 2x1x1) was applied exclusively along the temporal axis, reducing temporal resolution while preserving the spatial dimensions of the patch throughout the convolutional layers. The extracted multi-dimensional feature maps were then mathematically flattened and processed through a series of dense, fully connected layers, utilizing dropout regularization to mitigate overfitting before generating the final class probabilities.

Finally, to optimize the spatio-temporal approach, a 3D Fully Convolutional Network (3D FCN) was implemented. Unlike the standard 3D CNN, which relies on a bottleneck of dense layers to classify features, the 3D FCN adopts an encoder-decoder architecture. The encoder component utilizes a series of 3D convolutional and batch normalization layers to extract hierarchical spatio-temporal features. The decoder component then transforms these abstract feature maps into class predictions. By avoiding fully connected layers and using convolutional operations to produce spatially structured outputs, the model preserves the spatial organization of the input tile and can perform segmentation over regional tiles. This reduces the number of dense-layer parameters and may lower the risk of overfitting, while enabling simultaneous prediction for multiple pixels.

4.3 Experimental Setup

The experiment required a custom training and testing setup which fit all models.

Training Setup

To ensure the reproducibility of the benchmarked architectures, a standardized training protocol was established across all models. The neural networks were implemented using the PyTorch deep learning framework and trained on the LIACS Research and Education Lab (REL) server. Hardware acceleration was provided by an NVIDIA GeForce RTX 3090 GPU. Given the shared nature of the environment, batch sizes were dynamically scaled depending on real-time node availability and the specific memory constraints of the model being trained.

The models were optimized to minimize the standard multi-class Cross-Entropy Loss, which is well-suited for the mutually exclusive target categories of Corn, Soybean, and Other. Network weights were updated using the Adam optimizer (Adaptive Moment Estimation), configured with a fixed learning rate of 0.0001 to ensure stable convergence across both the sequential and Spatio-Temporal Models.

A maximum training duration of 150 epochs was defined for all deep learning models. However, to actively prevent model overfitting and minimize unnecessary computational expenditure, an early stopping mechanism was implemented. The training process was automatically halted if the validation loss failed to improve by a minimum threshold of 0.0002 for five consecutive epochs (patience = 5). The model weights from the epoch with the lowest validation loss were then restored and saved for final evaluation.

In-Season Testing Simulation Setup

While standard evaluation metrics provide insight into post-season model accuracy, the primary objective of this study was to assess early-season classification capabilities. To simulate real-time, mid-season deployment, an iterative in-season evaluation protocol was developed. To establish a performance baseline, all architectures were evaluated on the complete 24-week temporal sequence. Following that the dataset was progressively truncated to simulate real-time deployment. Models were evaluated at every single timestep from Week 1 to Week 24, recording the sequential changes in the classification accuracy and Macro-F1 Score.

Because most of the sequential and Spatio-Temporal Models require a fixed-length input tensor (24 timesteps), missing future observations during the early-season inference steps had to be artificially generated. To ensure a complete evaluation of how different architectures handle missing future context, three distinct temporal imputation strategies were implemented, while Zero-Padding was discarded as a strategy due to it destroying the phenological trajectory of the crops, as stated in [Sec. 2.3.1]:

- **Mean Imputation:** In this strategy, future timesteps were populated using historical, timestep-specific averages derived from the 2020 training dataset. Rather than applying a single global average, this strategy filled each missing week with its corresponding historical mean, providing the models with a statistically stable phenological baseline for the remainder of the season, while running the risk of the model being influenced by the imputed average phenological signature heavily reflecting the 'Corn' majority class.
- **Last Value Imputation:** This strategy propagated the last valid spectral observation forward across all remaining future timesteps. This approach retains more of the pixels' unique spectral identity, while sacrificing the future phenological patterns represented by the mean imputation strategy.

- **No Imputation:** Unlike the architectures that require fixed-dimensional inputs, sequence-based models like the Attention-LSTM and Transformer possess the capability to process truncated data. Because these models can dynamically handle sequence steps or attention masking, predictions could be extracted directly at the current timestep without appending any imputed future data. This strategy was applied to these two models to evaluate their specific architectural strength when processing incomplete time series.

By evaluating the models across these three distinct imputation scenarios, the evaluation protocol effectively measured not only the predictive power of the algorithms but also their robustness to incomplete temporal sequences. The pseudocode for the evaluation protocol can be found in the appendix [Algorithm 1]

4.4 Evaluation Metrics

To evaluate the predictive performance of the benchmarked architectures, two primary classification metrics were recorded at each inference timestep: Overall Accuracy and the Macro-F1 Score.

Overall Accuracy was utilized as a baseline metric to measure the raw proportion of correctly identified pixels across the entire test set. However, given the inherent class imbalance of the dataset, where the "Corn" class represents a significant majority of the study area, relying exclusively on overall accuracy can yield overly optimistic results. A model could theoretically achieve high accuracy simply by predicting the majority class while ignoring the minority classes. To account for this, the Macro-F1 Score was used as the primary performance metric. The Macro-F1 computes the F1-score (the harmonic mean of precision and recall) for each individual crop class independently, and then calculates their unweighted average:

$$\text{Macro-F1} = \frac{1}{N} \sum_{i=1}^{N} F1_i$$

where $N = 3$, representing the Corn, Soybean, and Other classes. This unweighted averaging ensures that the minority classes contribute equally to the final evaluation metric, penalizing models that fail to learn the less dominant spectral signatures.

To determine the practical deployment viability of the models for real-world agricultural monitoring, a relative operational performance threshold was established. The threshold was defined as 90% of a model's full-season performance. For a given model, the target threshold (T_{90}) is calculated as:

$$T_{90} = 0.90 \times MF1_T$$

where $MF1_T$ represents the model's final Macro-F1 Score achieved at the complete 24-week sequence. Reaching this threshold indicates that the network has acquired sufficient phenological evidence to make predictions that are 90% as reliable as those made with the complete dataset. This threshold has been established as sufficient for real-world use in earlier research [KKH+20]. This relative scaling allows for a fair, normalized comparison of convergence speed across architectures with varying full-season performance.

Alongside the 90% performance threshold, the models were also evaluated using a cumulative metric that scores performance across the entire growing season [MBS24]. To reflect the operational priority of rapid inference, this metric heavily weights early-season accuracy. The resulting *Absolute Earliness-Weighted Score* (S_E) is calculated as follows:

$$S_E = \frac{2}{T+1} \sum_{t=0}^{T} MF1_t \times \left(1 - \frac{t}{T}\right)$$

where T represents the total sequence length minus 1 (23), t denotes the specific inference timestep (from 0 to 23), and $MF1_t$ is the Macro-F1 Score achieved at that timestep. The linear time-decay multiplier, $\left(1 - \frac{t}{T}\right)$, is adapted from the "earliness reward" established by [RCE+23]. As t approaches T, this multiplier approaches zero, mathematically enforcing the principle that late-season predictions hold diminishing practical value. The resulting value from $\sum_{t=0}^{T} MF1_t \times \left(1 - \frac{t}{T}\right)$ lies between 0 and $\frac{T+1}{2}$, so we normalize it to a 0–1 score by multiplying the value by its reciprocal. By calculating this score, the benchmarked architectures can be ranked on a scale that balances both classification accuracy and prediction speed.

Because this thesis is primarily about how quickly a model can get close to its own Full-Season Performance (FSP), a relative version of this score was also calculated. We calculate this *Relative Earliness-Weighted Score* (RS_E) as follows:

$$RS_E = \frac{2}{T+1} \sum_{t=0}^{T} (MF1_t / MF1_T) \times \left(1 - \frac{t}{T}\right)$$

Where everything is the same as the S_E calculation, except that the Macro-F1 Scores are calculated to be in proportion to their FSP. It is important to note that this score can theoretically go above 1, as a model can overperform during the season only to end on a lower Macro-F1 Score.

5 Results

This section presents the results obtained using the methodology [Sec. 4], going over the full-season performance of the models, their in-season testing results, the impact of using different data imputation methods and the models' rankings based on the earliness scores.

5.1 Full-Season Baseline Performance

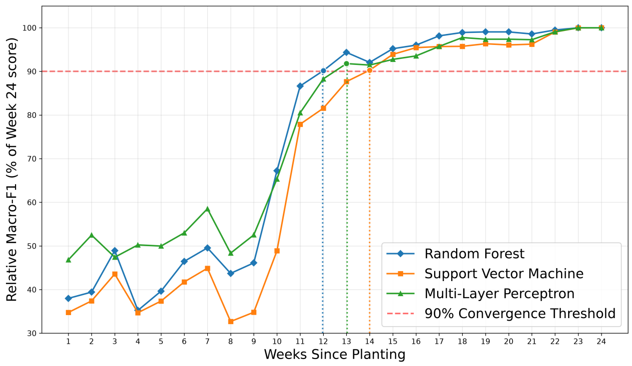
Model Architecture	Overall Accuracy	Macro-F1 Score	T_{90}
Traditional Non-Sequential Machine Learning Classifiers			
Random Forest (RF)	0.939	0.933	0.840
Support Vector Machine (SVM)	**0.958**	**0.950**	**0.855**
Multi-Layer Perceptron (MLP)	0.955	0.947	0.852
Sequential Deep Learning Models			
1D CNN	0.928	0.921	0.829
Attention-LSTM	0.952	0.943	0.849
Transformer	0.940	0.930	0.837
CNN-LSTM	0.942	0.931	0.838
Spatio-Temporal Models			
3D CNN	0.936	0.928	0.835
3D FCN	0.952	0.945	0.851

Table 2: Full-Season Baseline Performance (Week 24). Results represent the classification metrics when utilizing the complete sequence of observations. Numbers in **bold** are the maximum value in their column

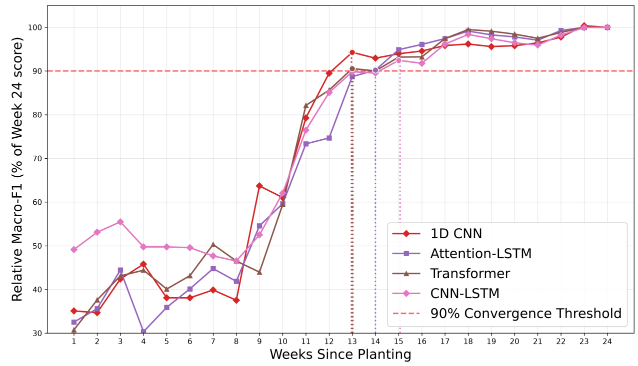
Before testing the in-season performance of the models, the models were tested on the full 24 week sequence to establish whether the models were capable of mapping the crop classes effectively when given access to the full season of data. The results of these baseline tests are shown in Table 2.

The results show a remarkable trend in the impact of model complexity. Contrary to expectations, the less complex traditional machine learning baseline methods actually perform better on the full season: The SVM achieved an overall accuracy of 95.8% and a Macro-F1 Score of 0.950, with the MLP showing similar results. This indicates that, in the full-season setting evaluated in this study, simpler models were able to achieve strong performance without requiring complex feature extraction.

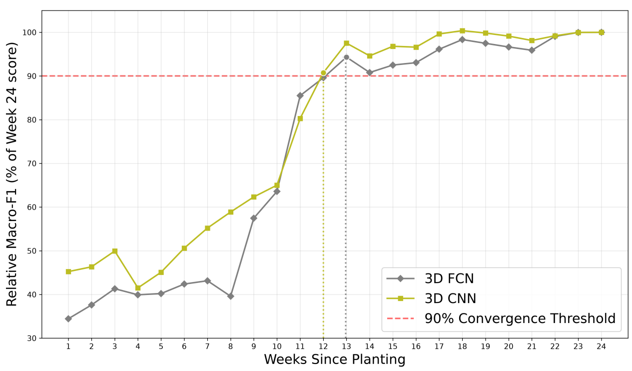
Among the Sequential Deep Learning Models, the Attention-LSTM proved to be the best model when measuring full-season performance (0.943 Macro-F1), beating the 1D CNN and Transformer models. The 1D CNN recorded the lowest Macro-F1 Score (0.921) among the tested models. The Spatio-Temporal Models performed similarly to the other classes, with the 3D FCN being on the higher end of the results with a Macro-F1 Score of 0.945 and the 3D CNN recording being on the lower end with a Macro-F1 Score of 0.928.



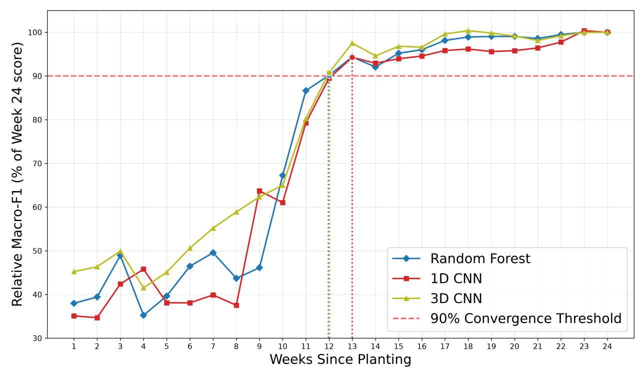
(a) Traditional Non-Sequential Machine Learning Classifiers



(b) Sequential Deep Learning Models



(c) Spatio-Temporal Models



(d) Comparison of best from (a), (b) and (c)

Figure 4: Relative In-Season Convergence organized by architectural class using Last Value Imputation. The y-axis represents the proportion of maximum Week 24 Macro-F1 Score achieved at each timestep. The dashed red line indicates the 90% operational convergence threshold. Dotted drop-lines indicate the week of operational threshold entry.

5.2 In-Season Test Results

To effectively compare the models on the in-season performance without the graphs becoming unreadable, this section is separated into 4 sections: 3 for the established groups of models [Sec. 4.2] and 1 comparing the best performing models from the 3 groups. All models were evaluated on their best performing imputation strategy, which happened to be the Last Value Imputation strategy for all models (More on this in [Sec. 5.3]).

(a) **Traditional Non-Sequential Machine Learning Classifiers**
When looking at convergence speed alone the comparison between the machine learning models shows a clear winner, that being the Random Forest (RF) model [Fig. 4a]. This model reaches 90% of its FSP in Week 12, at least a week before its competitors in this class. The only mitigating circumstance for the other models is their FSP: both the Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) had a full-season Macro-F1 Score of around 0.95, while the Random Forest trailed behind at 0.933 [Table 2]. Even when taking

this into account, the RF performs better in Week 12 than both other models with an MF1 of 0.841, with the SVM and MLP having 0.775 and 0.836 respectively in Week 12.

(b) **Sequential Deep Learning Models**
Declaring a winner in the Sequential Deep Learning category required a bit more work, as there are 2 models which reach 90% FSP in the same week, being the 1D CNN and the Transformer model in Week 13 [Fig. 4b]. To break this tie, two factors were considered: the percentage of FSP the model is at in Week 13 and the real Macro-F1 Score attained in Week 13. The transformer is at 90.6% of their FSP and has an Macro-F1 Score of 0.842 in Week 13, while the 1D CNN is at 94.3% and has an MF1 of 0.868. This shows that while they both converge to 90% in Week 13, the 1D CNN does so more convincingly.

(c) **Spatio-Temporal Models**
The Spatio-Temporal Models can easily be ranked by looking at the graph [Fig. 4c]. The 3D CNN outperforms the 3D FCN, reaching 90% in Week 12 compared to Week 13.

(d) **Comparison of best from (a), (b), (c)**
To select a best model out of all 9 models with regards to in-season performance, the best models from the three categories were compared [Fig. 4d]. This comparison contained another tie, this time between the RF and the 3D CNN. We broke this tie in the same way as before, using the percentage FSP and MF1 in Week 12. The RF was at 90.2% with an MF1 of 0.841, while the 3D CNN was at 90.7% with an MF1 of 0.842. This leads us to declare the 3D CNN as the best model regarding in-season performance, when looking at 90% convergence of FSP as the defining metric.

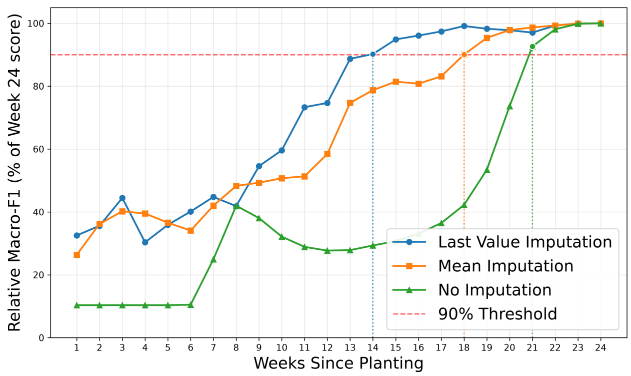
5.3 Evaluating the Impact of Imputation Strategies

Having established the 3D CNN utilizing Last Value Imputation as the optimal baseline for early convergence (achieving the operational threshold at Week 12), the analysis now shifts to evaluating how alternative handling of unobserved future timesteps impacts model performance. Specifically, the Last Value strategy was benchmarked against Mean Imputation and No Imputation across representative architectures. No imputation was only applied on the 2 architectures that could perform it, and was done using Masking on the Transformer model and using raw slicing on the Attention-LSTM.

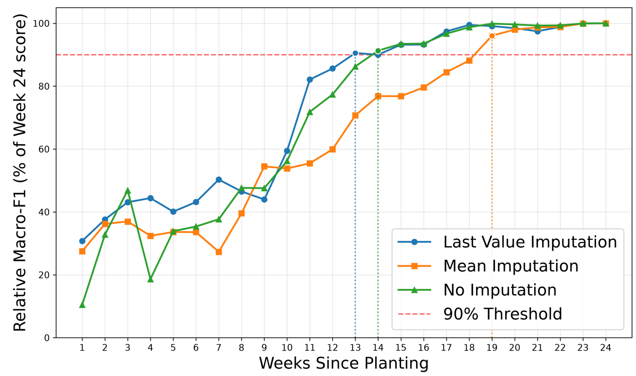
5.3.1 The Impact of Mean Imputation

The results demonstrate that Mean Imputation severely degrades early-season convergence speed across all architectural classes. As observed in the performance curves for the Attention-LSTM, Transformer, and 3D CNN, replacing unobserved future timesteps with the historical dataset mean delays the 90% convergence point to Weeks 18 and 19 [Fig. 5c].

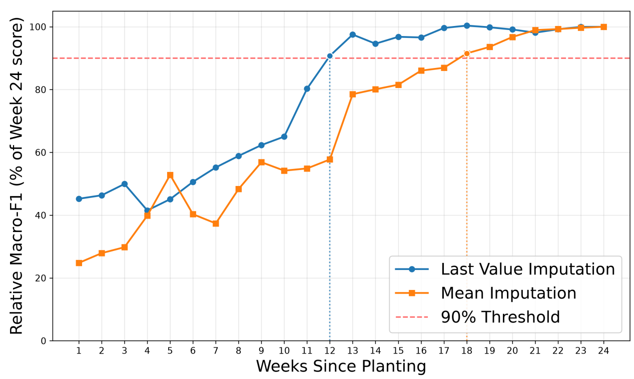
This four to six-week delay compared to the Last Value strategy can be attributed to a phenomenon of signal dilution. In the early and mid-season (Weeks 1–15), the actual observed satellite data contains subtle but still discriminative phenological signatures. However, when the remainder of the 24-week sequence is filled with mean values, a large proportion of the input sequence represents a generic averaged crop signature, which may partly reflect the dominant "Corn"



(a) Attention-LSTM



(b) Transformer



(c) 3D CNN

Figure 5: Analysis of the impact of different data imputation strategies across representative architectures. The y-axis represents the proportion of maximum Week 24 Macro-F1 Score achieved at each timestep. The dashed red line indicates the 90% operational convergence threshold. Dotted drop-lines indicate the week of operational threshold entry.

class in the training dataset. This artificial averaging drowns out the valid early-season evidence, resulting in the models relying on 'fake' evidence until Week 18/19 when the real observations finally outweigh the imputed averages enough to make confident predictions.

5.3.2 The Impact of No Imputation

Testing the models without any imputation (feeding exclusively the truncated observed sequence) exposed differing architectural sensitivities to sequence length.

For the Transformer network, utilizing a No Imputation approach through attention-masking to handle missing future data proved relatively effective [Fig. 5b]. The masked Transformer successfully reached its 90% convergence threshold at Week 14. Because the self-attention mechanism computes global dependencies, the application of a boolean mask allows the model to ignore non-existent future timesteps without penalty, focusing on the available early-season data. The results are still slightly worse overall when compared to its Last Value Imputation counterpart, but it is still an encouraging outcome.

Conversely, the Attention-LSTM performed poorly under the no-imputation setting, where only the observed portion of the sequence was provided to the model, while all unavailable future timesteps were omitted [Fig. 5a]. The Attention-LSTM remains at a very low Macro-F1 Score of approximately 0.15 during the first six weeks, shows a brief improvement in Weeks 7 and 8, declines again in Week 9, and does not cross the 90% full-season performance threshold until Week 21. This severe delay highlights a limitation of recurrent architectures: Attention-LSTMs require a sufficient number of sequential timesteps to meaningfully update their hidden state and cell memory. When the sequence is sliced in the early season, the recurrent network is starved of the temporal depth it was trained to fully utilize.

5.3.3 Summary of Findings

The comparative analysis proves that the choice of imputation is as critical as the choice of model architecture for early-season crop mapping. Last Value Imputation is strongly suggested to be the superior strategy. By extending the most recent valid observation into the future, it provides a stable assumption (maintaining the current state of the pixel) that allows both sequential and Spatio-Temporal Models to converge up to nine weeks earlier than alternative methods.

5.4 Earliness Rankings

Model Architecture	S_E	RS_E
Traditional Non-Sequential Machine Learning Classifiers		
Random Forest (RF)	0.568	0.609
Support Vector Machine (SVM)	0.528	0.556
Multi-Layer Perceptron (MLP)	**0.616**	0.650
Sequential Deep Learning Models		
1D CNN	0.542	0.589
Attention-LSTM	0.535	0.567
Transformer	0.549	0.590
CNN-LSTM	0.597	0.641
Spatio-Temporal Models		
3D CNN	0.606	**0.653**
3D FCN	0.560	0.593

Table 3: Final Operational Efficiency Rankings calculated using Last Value Imputation. The S_E score measures the time-discounted performance, while the RS_E score measures time-discounted convergence toward the model's own Week 24 Macro-F1 Score. Higher scores indicate superior early-season efficiency and the highest values are marked in **bold**.

To rank the architectures for practical agricultural deployment, the final step of the in-season evaluation is to summarize the temporal performance into time-discounted cumulative scores. Table 3 presents the Absolute Earliness-Weighted Score (S_E) and Relative Earliness-Weighted Score (RS_E) for all models, utilizing the established optimal Last Value Imputation strategy.

5.4.1 Absolute Earliness (S_E): Maximum Operational Utility

When evaluating purely based on the time-discounted absolute Macro-F1 Scores (S_E), the Multi-Layer Perceptron (MLP) emerges as the highest-performing architecture with a score of 0.616, closely followed by the 3D CNN (0.606) and the CNN-LSTM (0.597).

The MLP's dominance in this metric is driven by its high performance in the very early timesteps and its above-average overall performance. Because the S_E score does not normalize for a model's maximum potential, architectures that maintain a higher Macro-F1 Score throughout the season get greater scores. As shown in Table 2 and [Fig. 4a], the MLP has a high full-season performance while it also trends higher than other models in the very early timesteps, which count the most towards S_E.

5.4.2 Relative Earliness (RS_E): Architectural Learning Efficiency

Examining the Relative Earliness Score (RS_E), which focuses on relative learning speed by normalizing each timestep against the model's own Full-Season Performance (FSP), reveals a shift in the architectural hierarchy.

Under the RS_E metric, the 3D CNN (0.653) slightly outperforms the MLP (0.650) as the most efficient model in the study. This demonstrates that the 3D CNN is more "front-loaded" in its learning trajectory. It extracts the necessary spatio-temporal features to come close to its own FSP earlier than any other architecture evaluated [Fig. 4c]. The 3D CNN results demonstrate the advantage of using spatial neighborhood features for exploiting early-season spatial and temporal signals compared to strictly 1D sequence models or some Traditional Non-Sequential Machine Learning Classifiers. Additionally, the 3D FCN secures fifth place (0.593).

5.4.3 Summary of Findings

The divergence between the S_E and RS_E rankings highlights the distinction between operational performance and relative convergence. Under the S_E metric, the MLP ranks marginally above the 3D CNN, indicating slightly stronger absolute early-season Macro-F1 performance. Under the RS_E metric, however, the 3D CNN ranks marginally above the MLP, suggesting a slightly more front-loaded convergence pattern relative to its own full-season performance. Because both differences are very small, these rankings should be interpreted cautiously rather than as evidence of architectural superiority one way or the other. These results suggest that the MLP is the strongest model in the study from an operational perspective, while the 3D CNN shows promising relative convergence behavior that warrants further investigation under additional tuning and repeated runs.

6 Conclusions and Further Research

Model Architecture	FSP (Macro-F1)	90% Convergence	S_E	RS_E
Support Vector Machine (SVM)	**0.950**	Week 14	0.528	0.556
Multi-Layer Perceptron (MLP)	0.947	Week 13	**0.616**	0.650
Random Forest (RF)	0.933	**Week 12**	0.568	0.609
3D CNN	0.928	**Week 12**	0.606	**0.653**

Table 4: Summary of results, featuring the top-performing models across metrics discussed in [Sec. 5]. Bold values indicate the highest achieved result in each respective category. Only models that top a metric are included.

This research sought to identify the optimal architectural and data-handling framework for early-season crop type mapping. As summarized in Table 4, the findings indicate that early-season deployment presents a trade-off between absolute accuracy and relative convergence speed. When prioritizing absolute overall performance (S_E), the Multi-Layer Perceptron (MLP) provides the maximum operational utility ($S_E = 0.616$), with the 3D Convolutional Neural Network (3D CNN) following closely behind (0.606). Conversely, when evaluating architectural learning efficiency (RS_E), the 3D CNN paired with Last Value Imputation marginally outperforms the MLP (0.653 versus 0.650), achieving the earliest operational convergence threshold at Week 12 compared to Week 13 for the MLP. Because the margins between these two models are small, neither should be framed as definitively superior: rather, the MLP serves as a baseline for operational deployment, while the 3D CNN's front-loaded convergence points to great potential following further architectural tuning.

A counter-intuitive insight emerged during baseline testing: traditional models (specifically the SVM and MLP) performed similarly or better than complex Spatio-Temporal Models when evaluating the full season. This suggests that, in the full-season setting of this study, the complete set of week-specific spectral observations already contained enough phenological information for accurate crop classification, even when the temporal sequence was treated as a flattened feature vector. At a 30-meter spatial resolution, applying 3D convolutions to a complete timeline provides some value to the 3D FCN (which performs comparably to the best architectures) and likely overcomplicated the feature space for the 3D CNN (which ranked in the bottom 2 with regards to FSP). Complex spatial feature extraction seems to be a tool better utilized to compensate for missing temporal data than to supplement complete data.

These results answer the research question posed in the literature gap analysis: architectural paradigms and missing data mechanics are intertwined and cannot be evaluated in isolation. The mathematical handling of the unobserved future largely dictates early-season success. Applying Mean Imputation delayed operational convergence by four to six weeks compared to Last Value Imputation due to signal dilution, while feeding truncated sequences to the Attention-LSTM caused it to break. For architectures that require fixed-length inputs, their effectiveness in within-season mapping is therefore strongly conditioned by the chosen imputation strategy.

Translating these findings into practice, for an organization looking to predict crops at scale during a growing season (for example, to create a food security early-warning system), the choice of architecture should be tailored to operational priorities. While a 3D CNN paired with Last Value Imputation demonstrates excellent relative convergence by Week 12, the MLP provides highly competitive absolute early-season accuracy. Given the tiny performance margins between these approaches, this study advises that large-scale operational deployment would require further validation across additional regions, seasons, and crop distributions before establishing a definitive framework.

Limitations

While this study establishes a baseline for early-season classification, several limitations must be acknowledged:

- **Geographical Homogeneity:** The dataset was constrained to a single agricultural site in Iowa, dominated by corn and soybean. The generalizability of these models to regions with larger crop mixes or smaller field sizes remains unknown.
- **Computational Impracticality of SVM:** While the SVM demonstrated strong full-season performance, the $O(n^2)$ to $O(n^3)$ algorithmic complexity of calculating its kernel matrix makes it impractical for large-scale inference.
- **Fixed Spatial Context:** The 3D CNN was evaluated exclusively using a fixed 3×3 spatial neighborhood. It is currently unknown how varying this patch size would alter the balance between spatial noise and contextual feature extraction.
- **Unoptimized Models:** The tested models were either adapted from other work or the most simple version of a given model's architecture and were trained using standardized hyperparameters to ensure fair comparison. Performing hyperparameter optimization for the individual architectures may yield different results than the ones found here.

Future Research Directions

Given the established interaction between model architecture and missing data, future research should focus on the following:

- **Assessing Transferability:** The models must be evaluated on geographically diverse sites to test spatial transferability.
- **Learned Imputation Models:** Future studies should move beyond naive statistical baselines and explore "learned" imputation. Utilizing generative deep learning models to dynamically synthesize missing future timesteps based on early-season trajectories could prevent the signal dilution seen with Mean Imputation.
- **Impact of Hyperparameter Optimization on Learning Trajectories:** The architectures in this study were trained using standardized hyperparameters to ensure a fair baseline comparison. Future research should investigate whether tuning a model to maximize its full-season performance alters the shape of its in-season convergence curve. It remains to be determined whether a high Relative Earliness Score (RS_E) is an inherent property of Spatio-Temporal Models like the 3D CNN, or if sequential models can be mathematically tuned to shift their feature extraction efficiency earlier in the growing season.

References

[Bre01] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[BYMC11] Claire Boryan, Zhengwei Yang, Rick Mueller, 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.

[CV95] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[GZ21] Feng Gao and Xiaoyang Zhang. Mapping crop phenology in near real-time using satellite remote sensing: Challenges and opportunities. *Journal of Remote Sensing*, 2021, 2021.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[JZX+18] Shunping Ji, Chi Zhang, Ameng Xu, Yun Shi, and Yulin Duan. 3d convolutional neural networks for crop classification with multi-temporal remote sensing images. *Remote Sensing*, 10(1):75, 2018.

[KKH+20] Venkata Shashank Konduri, Jitendra Kumar, William W. Hargrove, Forrest M. Hoffman, and Auroop R. Ganguly. Mapping crops within the growing season across the united states. *Remote Sensing of Environment*, 251:112048, 2020.

[LCC+22] Licong Liu, Ruyin Cao, Jin Chen, Miaogen Shen, Shuai Wang, Ji Zhou, and Binbin He. Detecting crop phenology from vegetation index time-series data by improved shape model fitting in each phenological stage. *Remote Sensing of Environment*, 277:113060, 2022.

[LR19] Roderick JA Little and Donald B Rubin. *Statistical analysis with missing data*. John Wiley & Sons, 2019.

[LSD15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[MBS24] Sina Mohammadi, Mariana Belgiu, and Alfred Stein. A source-free unsupervised domain adaptation method for cross-regional and cross-time crop mapping from satellite image time series. *Remote Sensing of Environment*, 314:114385, 2024.

[Mei18] Uwe Meier. *Growth stages of mono- and dicotyledonous plants: BBCH Monograph*. Julius Kühn-Institut, 2018.

[PVI+16] Charlotte Pelletier, Silvia Valero, Jordi Inglada, Nicolas Champion, and Gérard Dedieu. Assessing the robustness of random forests to map land cover with high resolution satellite image time series over large areas. *Remote Sensing of Environment*, 187:156–168, 2016.

[RCE+23] 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.

[RHW86] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[RK18] Marc Rußwurm and Marco Körner. Multi-temporal land cover classification with sequential recurrent encoders. *ISPRS International Journal of Geo-Information*, 7(4):129, 2018.

[RWL+14] D.P. Roy, M.A. Wulder, T.R. Loveland, Woodcock C.E., R.G. Allen, M.C. Anderson, D. Helder, J.R. Irons, D.M. Johnson, R. Kennedy, T.A. Scambos, C.B. Schaaf, J.R. Schott, Y. Sheng, E.F. Vermote, A.S. Belward, R. Bindschadler, W.B. Cohen, F. Gao, J.D. Hipple, P. Hostert, J. Huntington, C.O. Justice, A. Kilic, V. Kovalskyy, Z.P. Lee, L. Lymburner, J.G. Masek, J. McCorkel, Y. Shuai, R. Trezza, J. Vogelmann, R.H. Wynne, and Z. Zhu. Landsat-8: Science and product vision for terrestrial global change research. *Remote Sensing of Environment*, 145:154–172, 2014.

[SSL+12] Francois Spoto, Omar Sy, Paolo Laberinti, Philippe Martimort, Valerie Fernandez, Olivier Colin, Bianca Hoersch, and Aime Meygret. Overview of sentinel-2. In *2012 IEEE International Geoscience and Remote Sensing Symposium*, pages 1707–1710, 2012.

[TBF+15] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[Uni10] United States Department of Agriculture, National Agricultural Statistics Service. Crop progress and planting report. Release file, USDA NASS, October 2010.

[U.S21] U.S. Geological Survey. Landsat collection 2 level-2 science products. Fact Sheet 2021-3055, U.S. Geological Survey, Reston, VA, 2021.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[XYX+21] Jinfan Xu, Jie Yang, Xingguo Xiong, Haifeng Li, Jingfeng Huang, K.C. Ting, Yibin Ying, and Tao Lin. Towards interpreting multi-temporal deep learning models in crop mapping. *Remote Sensing of Environment*, 264:112599, 2021.

[XZZ+20] Jinfan Xu, Yue Zhu, Renhai Zhong, Zhixian Lin, Jialu Xu, Hao Jiang, Jingfeng Huang, Haifeng Li, and Tao Lin. Deepcropmapping: A multi-temporal deep learning approach with improved spatial generalizability for dynamic corn and soybean mapping. *Remote Sensing of Environment*, 247:111946, 2020.

[ZHZ19] Liheng Zhong, Lina Hu, and Hang Zhou. Deep learning based multi-temporal crop classification. *Remote sensing of environment*, 221:430–443, 2019.

[ZJP+21] George Zerveas, Srideepika Jayaraman, Dhaval Patel, Anuradha Bhamidipaty, and Carsten Eickhoff. A transformer-based framework for multivariate time series representation learning. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 2114–2124, 2021.

[ZWX+20] Linglin Zeng, Brian D. Wardlow, Daxiang Xiang, Shun Hu, and Deren Li. A review of vegetation phenological metrics extraction using time-series, multispectral satellite data. *Remote Sensing of Environment*, 237:111511, 2020.

Algorithm 1 In-Season Evaluation Loop

Require: *model*: Trained classification model
Require: *data_loader*: Iterable dataset containing $(batch\_x, batch\_y)$
Require: *total_weeks*: Total number of timesteps in the full sequence (24)
Require: *imputation_strategy*: Method for handling unobserved future data (Last Value or Mean Imputation)
Require: *global_mean*: historical mean tensor for mean imputation

$accuracies \leftarrow [\,]$
$f1\_scores \leftarrow [\,]$
for $t = 1$ **to** $total\_weeks$ **do**
    $all\_preds \leftarrow [\,]$
    $all\_targets \leftarrow [\,]$
    for each $(batch\_x, batch\_y)$ **in** $data\_loader$ **do**
        $masked\_x \leftarrow \mathrm{clone}(batch\_x)$
        if $t < total\_weeks$ **then**
            if $imputation\_strategy ==$ "LastValue" **then**
                $last\_observed \leftarrow masked\_x[:, t-1:t, \ldots]$
                $masked\_x[:, t:, \ldots] \leftarrow last\_observed$
            else if $imputation\_strategy ==$ "Mean" **then**
                $masked\_x[:, t:, \ldots] \leftarrow global\_mean[:, t:, \ldots]$
            end if
        end if
        $logits \leftarrow model(masked\_x)$
        $predictions \leftarrow \mathrm{argmax}(logits)$
        append $predictions$ to all_preds
        append $batch\_y$ to $all\_targets$
    end for
    $accuracy \leftarrow \mathrm{compute\_accuracy}(all\_targets, all\_preds)$
    $macro\_f1 \leftarrow \mathrm{compute\_macro\_f1}(all\_targets, all\_preds)$
    append $accuracy$ to $accuracies$
    append $macro\_f1$ to $f1\_scores$
end for
return $accuracies, f1\_scores$