Multi-modal Contrastive Learning for Crop Classification Using Sentinel2 and Planetscope

Ankit Patnala, Scarlet Stadtler, Martin G. Schultz, Juergen Gall

***Abstract*—Remote sensing has enabled large-scale crop classification to understand agricultural ecosystems and estimate production yields. Since a few years, machine learning is increasingly used for automated crop classification. However, in most approaches the novel algorithms are applied to custom datasets containing information of few crop fields covering a small region and this often leads to models that lack generalization capability. In this work, we propose a multi-modal contrastive self-supervised learning approach to obtain a pre-trained model for crop-classification without the use of labeled data. Such multi-modal self-supervised learning exploits the synergies of different data sources to obtain a richer representation of the data. We build our analysis by adapting the DENETHOR dataset developed for a part of Eastern Germany to our usecase. We use the publicly available Sentinel2 and commercial Planetscope data. While Sentinel2 has higher spectral resolution, Planetscope has finer spatial resolution. For an end-user application, only one source is required. In this work, we analyze and compare the performance of our multi-modal self-supervised model against the uni-modal contrastive self-supervised model using the SCARF algorithm. In addition, we also compare our multi-modal self-supervised model with a supervised model. We find that our multi-modal pre-trained model surpasses the uni-modal and supervised models in almost all test cases.**

***Index Terms*—Optical remote sensing, crop classification, contrastive learning, multi-modal contrastive learning, time-series, self-supervised learning**

I. Introduction

Crop classification is a method of identifying the type of agricultural plants at a particular location. Typically, in remote sensing, researchers use information from various public landcover satellite missions such as Sentinel2 [1] and Landsat[1] which cover the entire globe at a regular time interval over many years. Crops exhibit clear temporal signatures due to phenological traits i.e. the pattern of their growth stages from seed to sprout through budding, growing and then ripening [2]. Therefore, crop classification methods rely extensively on temporal patterns from the large archives of these public remote sensing satellite missions. The analysis of crops from remote sensing data can be used to optimize farming practices, assess damage, and increase yields.

The availability of public satellite missions facilitates large scale crop mapping suitable for machine learning. However, labeling of crops as it is necessary for classical machine learning approaches is time consuming and requires skilled human efforts. Most works on vegetation remote sensing therefore only use a small region consisting of few crop fields for crop analysis. The conventional machine learning methods such as random forest generate good results on the same field but can not be generalized to other fields with different geographical properties [4] or even when tested on the same fields at a different time [5].

The need for models that generalize well without using additional manual annotations and the development of advanced algorithms in the field of deep learning has motivated the development of techniques such as self-supervised learning. Self-supervised learning facilitates pre-training using a large amount of unlabeled data. The weights of this pre-trained model are then used as a starting point to solve tasks containing a few annotated data samples. This technique is called transfer learning. It is found that the performance of pre-trained models surpasses the performance of equivalent models where weights are randomly initialized [6]. Self-supervised learning relies on pretext tasks, and with recent advancement, the contrastive learning [7] has shown promising results. For contrastive self-supervised learning, the pretext task is to align the output from two different viewpoints of the data simultaneously ensuring the output from all the other data are pushed away Since the loss functions such as mean square error (MSE) would lead to trivial solution as there are no labels, alternative loss functions such as InfoNCE [8] are used in order to obtain non-trivial weights. The contrastive learning method relies on augmentation of a data sample and aims to maximize similarity of the data sample and its augmented version. However, such augmentation for raw satellite data is non-trivial [9]. In this work, we thus propose multi-modal contrastive learning where the augmented version of the data is obtained from another source [10]. Recently, datasets like Breizhcrops [11], TimeSen2Crop [12], Eurocrops [13], DENETHOR [14] etc. have been proposed to facilitate large scale crop classification. In this work, we specifically use DENETHOR, which provides surface reflectance from multiple optical remote sensing sensors such as Sentinel2 [1] and Planetscope [15]. Thus, DENETHOR is feasible for multi-modal self-supervised learning. Though two data sources are used for pre-training, the user does not require both sources while applying the pre-trained model. We term the model used for pre-training as backbone model and the model used for

A.Patnala, S.Stadtler, and M.G.Schultz are with Institute of Advanced Simulation at Forschungszentrum Jülich, Germany(correspondence mail: a.patnala@fz-juelich.de) J.Gall is with Department of Information Systems and Artificial Intelligence at University of Bonn and Lamarr Institute for Machine Learning and Artificial Intelligence, Germany.

We acknowledge the funding from the German Federal Ministry for the Environment, Nature Conservation and Nuclear Safety under grant no 67KI2043 (KISTE). Computing time for this study was kindly provided by the Julich SupercomputerCentre under project DeepACF. JG is supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - SFB 1502/1-2022 - Projektnummer : 450058266

Manuscript received April 19, 2021; revised August 16, 2021.

[1]https://landsat.gsfc.nasa.gov/appendix/references/

downstream tasks as base models.

Sentinel2 is an ESA satellite mission. Its multi spectral instrument (MSI) consists of 12 bands including visible, thermal and infrared bands ranging from $400\ \mathrm{nm}$ to $2190\ \mathrm{nm}$. Sentinel2 has a spatial resolution of $10\ \mathrm{m}$, which means that each pixel covers an area of $100\ \mathrm{m}^2$. Data are publicly available using either the Copernicus API or Google Earth Engine [16]. Although the cloud masks are available, cloud-free images are hard to obtain. On the other hand, Planetscope is a commercial satellite mission. Planetscope comes with higher pixel resolution when compared to Sentinel2 i.e. $3\ \mathrm{m/px}$. The instrument takes multiple snapshots of a particular region and uses the "best scene on top" algorithm[2]. This algorithm ensures that we obtain an image least affected from cloud, haze, and other image variability. A downside of Planetscope is the lower spectral resolution i.e., only 4 channels (R,G,B and NIR) are available. In this work, we design a strategy to develop a multi-modal self-supervised pre-trained model by complementing the higher spectral resolution of Sentinel2 with the finer spatial resolution of Planetscope. Although two data sources, i.e., Sentinel2 and Planetscope, are used for pre-training, the pre-trained model can be applied to only one source. For evaluation, we use crop classification on Sentinel2 data as downstream task. Our experiments show that the proposed multi-modal contrastive self-supervised pre-training improves the crop classification accuracy.

The main contributions of this paper are:

- We have designed a setup for multi-modal self-supervised learning with two different optical remote sensing image sources for crop classification.
- We show that our multi-modal contrastive self-supervised pre-trained model provides higher accuracy for crop classification compared to the conventional uni-modal contrastive learning using random feature corruption as an augmentation for tabular data.
- We analyze and test our pre-trained models by evaluating the learned representation using three different types of networks, namely a convolutional, a recurrent and a transformer networks.

II. Related Works

Recently, there has been much work on contrastive self-supervised learning on remote sensing images. SeCo [17] is one of them where the authors found that their pre-trained model outperformed a model pre-trained on Imagenet [**?**] on several benchmark datasets. This work focused on contrastive learning but it did not explore channels beyond RGB and they considered only single time spatial land cover classification. Their work used only one source, i.e. Sentinel2, and used standard transformations to obtain augmented data for contrastive learning. They used an image size of 224×224. The authors of [18] used a multi-modal contrastive learning approach on remote sensing images. They aligned an optical remote sensing image (Sentinel2) with a radar image (Sentinel1). To our knowledge, there is no work that applies contrastive learning to time series tasks in the field of remote sensing data. In the machine learning community, there are few works on tabular data. For example, SCARF [19] uses random feature corruption techniques to obtain an augmented version of a tabular data. The authors have tested this augmentation on 69 datasets from the public OPENML-CC18 [20] benchmark data. They compared their proposed methods to SCARF+autoencoder, autoencoder, denoising autoencoder [21] and found the performance to be better than all the alternatives. SAINT [22] is another contrastive self-supervised learning method developed for tabular data. They propose a modification in Tabtransformer [23] to incorporate both categorical and continuous data. With the help of Cutmix [24] in the raw data space and Mixup [25] at the embedding space, they produced a representation of an augmented data. They further use two projections where one is meant for the contrastive loss and the other is used to obtain a reconstruction. By optimizing the SimCLR [26] contrastive loss and MSE as reconstruction loss, they obtain a pre-trained model. They use intersample attention, i.e. data attends other data in the batch for both pre-training and fine-tuning. Due to intersample attention in fine-tuning, they always need few labelled data samples even for prediction. VIME [27] is also a self-supervised method developed for tabular data. Unlike SCARF and SAINT, they do not rely on contrastive learning. They use feature corruption and maksing to corrupt the data. They train two pretext tasks in parallel. The feature estimator aims to reconstruct the original data and the mask estimator aims to predict whether a feature has been masked.

III. Datasets

We use the training and validation sets of the DENETHOR dataset to prepare our own custom data set to perform multi-modal self-supervised learning experiments.

DENETHOR's training dataset covers a region of 24×24 km^2 in eastern Germany. This region is located in the state of Brandenburg. This dataset contains multiple sources of the same data that includes both Sentinel2 and Planetscope. The training data covers the entire year 2018. Given the pixel resolution of $10\ \mathrm{m/px}$ for Sentinel2 and $3\ \mathrm{m/px}$ for Planetscope, the dimensions of the measurements are represented as 2400×2400 and 8000×8000, respectively. We have 144 Sentinel2 timestamps of the region for a given year and we filtered the Planetscope data for the same 144 timestamps of the year. DENETHOR's validation dataset also covers a $24 \times 24\ \mathrm{km}^2$ region in Brandenburg but from a different region. The validation dataset covers the entire year 2019 with 144 timestamps for Sentinel2.

The total training data consists of 2534 crop fields and the validation data consists of 2064 crop fields for 9 crop types. In both the training and validation regions, there are locations where no crops are grown. The pixels belonging to these locations are masked.

We make a 70-21-9 random split of DENETHOR's training dataset to obtain the pre-training data and the data for downstream task1. A 70-30 random split is done on DENETHOR's validation dataset to obtain data for our downstream task2. The use of two downstream tasks is to evaluate the performance of a pre-trained model on test data from a different time and region.

[2]https://developers.Planet.com/docs/data/visual-basemaps/

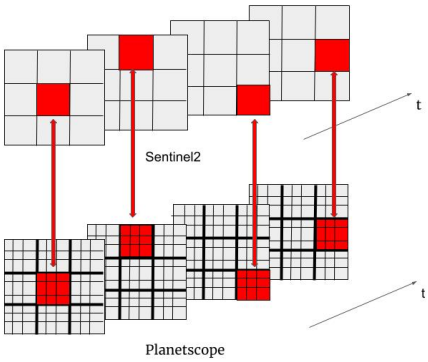


Fig. 1: **Description of data alignment for multi-modal self-supervised learning.** The top row and the bottom row show a spatial region of the same geographical region from Sentinel2 and Planetscope. For each timestamp, pixels are randomly selected from Sentinel2. The pixel data is aligned to 3×3 pixels from the same corresponding region for the same timestamp from Planetscope.

Figure 2 shows a visual description of our splitting strategy. Subsection III-A describes the creation of the pre-training data and Subsection III-B gives an overview of the data created for the two downstream tasks.

A. Data for Pre-training

For pre-training, we need unlabeled data. To obtain this, we use 70% of the random crop fields obtained from the 70-21-9 split. We do not use the crop labels.

For our pre-training, we iterate through each of the 144 Sentinel2 timestamps and randomly selected 100,000 pixels from the 70% split. This process yielded 14,400,000 data samples for our multi-modal self-supervised experiment. Since one pixel of Sentinel2 covers $100\ \mathrm{m}^2$, whereas a pixel of Planetscope covers $9\ \mathrm{m}^2$, we assigned a pixel of Sentinel2 to 3×3 pixels of Planetscope for alignment as shown in Figure 1.

B. Data for Downstream Tasks

The pre-trained model is tested on two different sets of Sentinel2 data to evaluate its generalizability. We create two crop classification downstream tasks using DENETHOR's training and validation dataset. The 21% and 9% of the data of the given 70-21-9 split of the DENETHOR's training dataset is used for the crop classification downstream task1. The 21-9 split is done to separate training and validation crop fields for our downstream task1. For each crop, 5000 pixels are randomly selected from their training cropfields in order to create a balanced dataset. For the validation set of downstream task1, we also created a balanced dataset by randomly selecting 1000 pixels of each crop from the validation cropfields. For our crop classification downstream task2, we used a 70-30 split on DENETHOR's validation set. We implemented a similar process as before to get a balanced dataset for our 2nd crop classification downstream task. Figure 2 shows the creation of the two downstream tasks.

IV. Methods

A. Multi-modal Self-Supervised Learning

Figure 3 shows the setup of our multi-modal contrastive learning method. In this approach, the networks are not shared between the two modalties as the two sources have different input dimensions. Here, the backbone network is denoted by $E_s : \mathbb{R}^{12} \rightarrow \mathbb{R}^{256}$ and $E_p : \mathbb{R}^{36} \rightarrow \mathbb{R}^{256}$ for data from Sentinel2 and Planetscope, respectively. Similarly, the projector network is denoted by $P_s : \mathbb{R}^{256} \rightarrow \mathbb{R}^{256}$ and $P_p : \mathbb{R}^{256} \rightarrow \mathbb{R}^{256}$ for Sentinel2 and Planetscope, respectively. Equation (1) shows a mathematical formulation of the SimCLR loss function [26] used in our work. For the pre-training, we adapt the SimCLR loss to our multi-modal setup as shown in the Equation (1a).

$$l_{x_{is},x_{ip}} = -log \frac{r_{iisp}}{\sum_{k=1,k\neq i}^{N} r_{ikss} + \sum_{m=1,m\neq i}^{N} r_{imsp}} \tag{1a}$$

$$sim(z_{is}, z_{jp}) = (z_{is}^T z_{jp})/(\|z_{is}\| \, \|z_{jp}\|) \tag{1b}$$

$$r_{ijsp} = exp(sim(z_{is}, z_{jp})/\tau) \tag{1c}$$

x_{is} represents the i^{th} Sentinel2 data sample and z_{is} represents the output obtained after passing through the encoder and projector part of the Sentinel2 network. Similarly, x_{ip} represents the i^{th} Planetscope data sample and z_{ip} represents the output obtained after passing through the encoder and projector part of the Planetscope network. τ denotes the temperature which controls the sensitivity of the loss function. In the original SimCLR equation [26], there is only one network and two augmented views share the same model. In contrast, in our multi-modal case there are separate networks for different views. The term r_{ikss} in the denominator of the Equation (1a) denotes the cosine distance between Sentinel2 data sample to other Sentinel2 data samples in the batch and similarly r_{imsp} denotes the cosine distance between the Sentinel2 data sample and the other Planetscope data samples in the batch. Figure 3 shows how to use the pre-trained Sentinel2 backbone for the downstream task of crop classification. 144 timestamps of each pixel are passed through the pre-trained model to obtain an abstract pixel representation. The time series formed with the representations of each timestamp is used as an input to the base models. As the multi-modal pre-training implicitly learns a mapping from Planetscope to Sentinel2 data, it suffices to feed only Sentinel2 data into the model for the classification downstream task. Thus, users can implicitly take advantage of Planetscope's finer spatial resolution.

We use the random feature corruption technique from SCARF [19] as transformation on both sources in our multi-modal self-supervised learning setup. In random feature corruption, with a given corruption rate "c", randomly c % of the features in the data are replaced by the empirical marginal

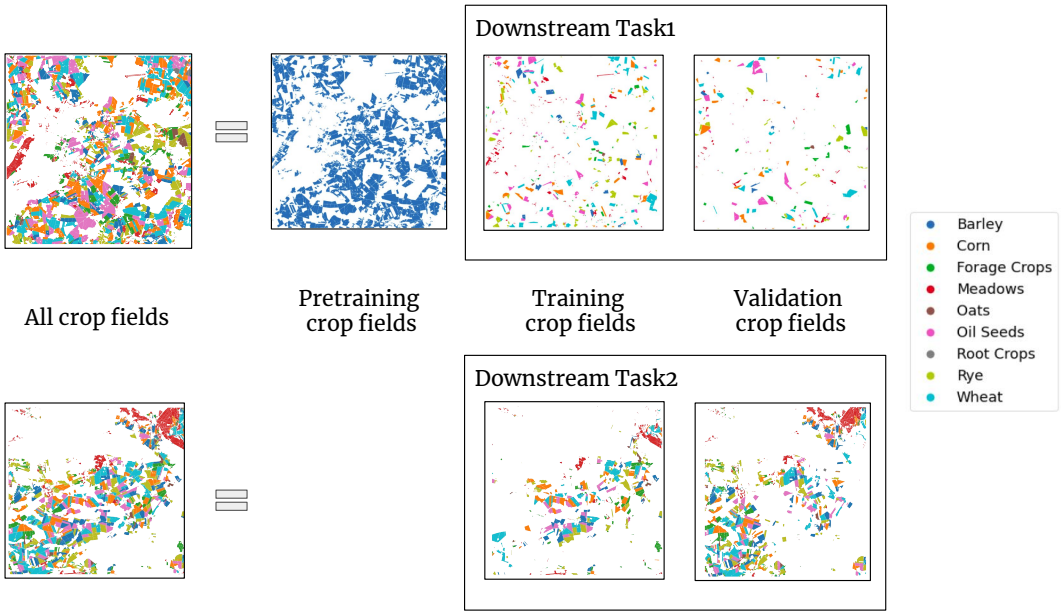


Fig. 2: **Dataset for the multi-modal self-supervised learning experiment.** The top row of the figure shows the splitting we used on the DENETHOR's training dataset. The top image shows all crop fields in the training dataset, which we divided into three parts. The first part is used for pre-training and is represented by blue crop fields. The blue color represents unlabeled data. The other two parts are training and validation data for our downstream task1. The bottom row shows the validation dataset for DENETHOR. For our downstream task2, we split the validation data into training and validation for our downstream task2. The downstream task2 is trained using the pre-trained model obtained from the same pre-training dataset to check its performance when region and time period are changed.

distribution of the corresponding features. Figure 4 shows a schematic diagram of the random feature corruption technique.

In our experiments, we use two different backbones for our multi-modal self-supervised learning, MLP and ResMLP. The most commonly used network for tabular data is MLP. Inspired by the skip connection mechanism of ResNets [28], we use skipped connection MLP and named it as ResMLP.

B. Base Models

In the following part, we will describe the networks used as base models for the downstream tasks.

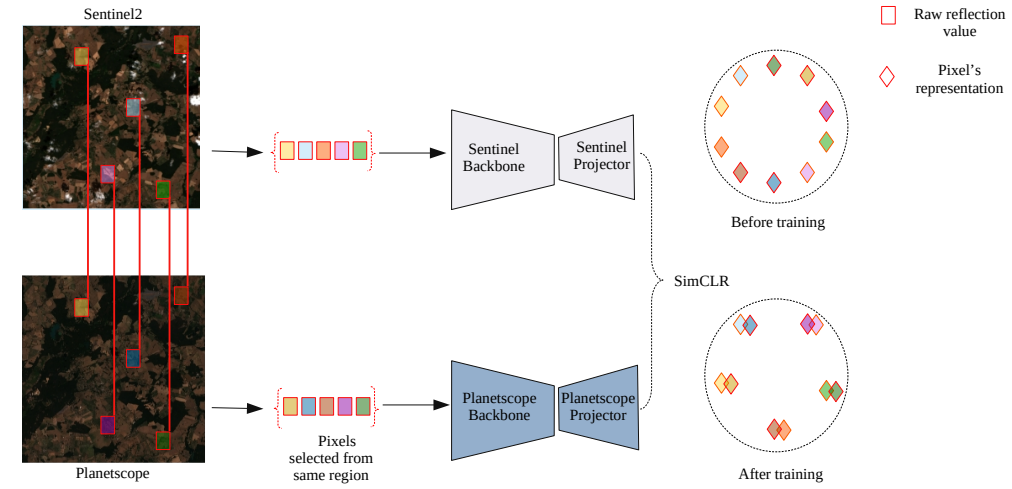
1) ***Bi-directional LSTM****:* LSTM [29] is an advanced recurrent neural network (RNN) designed to solve tasks related to language processing or time-series. It is an autoregressive model. An LSTM layer consists of multiple gated memory cells each consisting of multiple gates i.e. forget gate, input gate and output gate. Compared to vanilla RNN, LSTM increases the ability to capture long-term dependency and mitigates to some extent the problem of vanishing and exploding gradients. In a bi-directional LSTM, each LSTM layer receives input from both directions i.e. from initial timestamp to final timestamp and vice-versa from final timestamp to initial timestamp. LSTMs are trained by backpropagation through time.

2) ***Inception Time****:* Inception networks have originally been developed for images. It consists of repeating components called inception modules. Inception networks are a variant of CNNs where filters with different kernel sizes are processed in parallel. In the inceptiontime model [30], such inception modules are applied to our time series data. It is intended to capture different temporal scales of patterns.

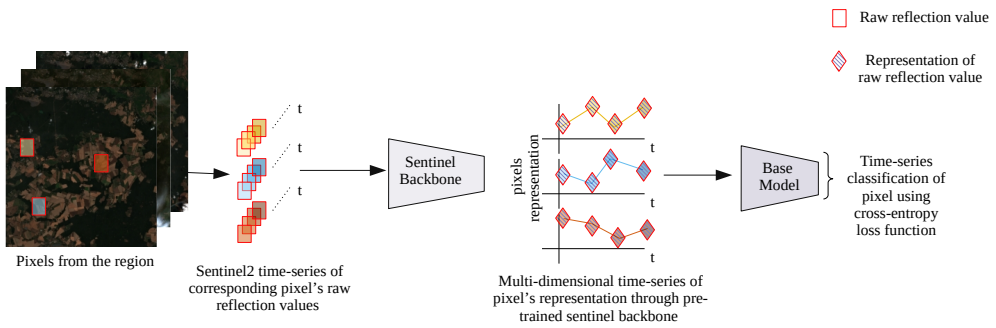
3) ***PE Transformer****:* The position encoded transformer is the encoder part of the original transformer network [31] developed for language translation problems. The attention mechanism used in this network is position invariant, so sinus and cosinus positional encoding are used. The attention mechanism helps to solve sequential problems such as time-series. We apply the network to each timestamp and perform max pooling over timestamps with an additional MLP layer.

C. Evaluation of Downstream Task Performance

We used two sets of experiments, supervised and uni-modal self-supervised experiments to compare our proposed multi-modal approach. In supervised experiments, the Sentinel2 data is directly passed to the base networks as there is no pre-training involved. For uni-modal self-supervised experiments, we use the random feature corruption technique from SCARF [19] to obtain an augmented view of the Sentinel2 data as shown in the Figure 4. This augmented data along with the original is then used for uni-modal self-supervised contrastive learning. The same MLP and ResMLP model which is used



(a)



(b)

Fig. 3: **Schematic setup of our multi-modal self-supervised experiment.** a) The top part shows the multi-modal setup where corresponding pixels shown with red boxes are randomly selected. Then, the Sentinel2 and Planetscope data are passed through their corresponding backbone and projector network. The outputs are finally aligned by optimizing the contrastive loss with the objective to attract similar and repel dissimilar pairs. This is shown with two spherical diagrams on the right side where before training the projection of the data are randomly distributed and with training, the similar pairs got aligned simultaneously maintaining uniformity in the latent hypersphere space. b) The bottom part shows the training of a downstream task where the raw data from Sentinel2 is fed through the pre-trained Sentinel2 backbone. The output is then fed to different base models for conventional supervised learning, which optimizes the standard cross entropy loss for multi-class classification.

as Sentinel2 backbone in the multi-modal setup, is used here. The cross-entropy loss function is used for all our baseline experiments.

V. Experiments

We divide this section into three parts: supervised experimentsV-A, uni-modal self-supervised experiments and multi-modal self-supervised experiments. For all our experiments, we use a 32GB NVIDIA Tesla V100 GPU. For our supervised and downstream experiments, we adopt all the base models from the implementation of [11].

A. Supervised Experiments

This is our first experiment setup. In this setup, we fed the raw reflectance values of Sentinel2 from the training and validation datasets directly to the networks. For each category of networks i.e. bi-directional LSTM, inceptiontime, and transformers, 10 different models were selected with varying hyperparamters. To obtain 10 different models for each of the three categories, we used the optuna [32] hyperparameter tuner on a specific hyperparameter search grid. For bi-directional LSTM, the hyperparameter space is as follows: dimensions of hidden layer in the category of [32,64,128,256], number of layers in the category of [2,3,4,5,6], and learning rate in continuous value between 10^{-5} and 10^{-3}. For inception

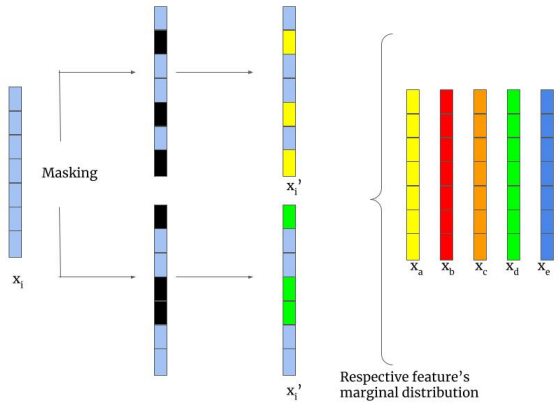


Fig. 4: **Random feature corruption mechanism in SCARF [19]**. At each time, few features are masked randomly as shown with black cells in the figure. These masked features are replaced by features from other samples in the batch as represented by $(x_i^{'})$.The two corrupted features are considered as a positive pair.

time, the hyperparameter space is as follows: number of layers in the category of [2,4,8], dimension of hidden layer in the category of [128,256,512,1024], kernel size in the category of [40,80,120,136], and learning rate in the continuous value between 10^{-5} and 10^{-3}. The hyperparameter space for transformers is as follows: dimension of model in the category of [32,64,128], number of attention heads in the category of [2,4,8] and number of layers in the category of [2,3,4,5,6], and the learning rate is a continuous value range between 10^{-5} and 10^{-3}. For all the supervised experiments, we trained the network for 20 epochs. We used the initial learning rate of 10^{-3} with the linear scheduler.

B. Uni-modal Self-Supervised Experiments

This is our second set of experiments. With these experiment sets, we intended to compare our proposed multi-modal self-supervised models with the one which already have seen pre-training data and gained additional information. We used uni-modal contrastive learning i.e., only using Sentinel2 during pre-training. Lack of transformation processes such as cropping, color jittering makes it difficult to obtain augmented data for raw reflectance value of a Sentinel2 at a location. Therefore, we used the random feature corruption technique of [19] to be able to perform contrastive learning for tabular data with one mode. In our uni-modal self-supervised experiment setup, we pre-trained the model on pre-training data. We ran it for 100 epochs. We used a SimCLR loss function with a temperature of 0.07. The learning rate is set to 10^{-3} . Such a contrastive loss needs a higher batch size to generalize well, so we took a batch size of 2048. We tested on different random feature corruption rates i.e. 20% and 60%.

C. Multi-Modal Self-Supervised Experiments

This is our proposed experimental setup. In contrast to the uni-modal self-supervised setup, we used different backbone networks for Sentinel2 and Planetscope. We ran this pre-training for 100 epochs. Similar to uni-modal self-supervised experiment setup, the initial learning was set to 10^{-3} with a temperature parameter of the SimCLR loss set to 0.07. In addition, we randomly applied SCARF algorithm on each source. In this case, we tried our experiments with corruption rate 0 (no corruption)), 20 and, 60.

VI. RESULTS

We used the evaluation protocol from [19] i.e. the use of win-matrix plot and the box plot to compare different models for both datasets. In the win-matrix plot, the value in the cell shows the ratio of experiments mentioned in the row outperform the one in the column as formulated in Equation (2); with i and j are competing methods, and N is the total number of experiments.

$$W_{ij} = \frac{\sum_{i=1}^{N} \mathbb{I}(val\_acc_i > val\_acc_j)}{N} \tag{2}$$

We presented results for all the three base models separately. The results for ResMLP as a backbone are shown in this section and for MLP, please refer to appendix.

Figure 5 shows the win-matrix and relative gain box plot for ResMLP pre-trained models on downstream task1. We found our multi-modal self-supervised model's performance to be better than uni-modal self-supervised and supervised models. We found that the random feature corruption technique, which shows improved performance on openml tabular data [20] for uni-modal contrastive learning pre-trained model, does not show promising results in the case of time-series crop classification data. The experiment comparisons are done with similar hyperparameters, i.e. same random feature corruption coefficient (20 & 60), so for each baseline type, we have 20 experiments (two variants of scarf for 10 variants of each base model type). As there is no pre-trained model involved in the supervised baseline model, so we used the same score for both instances. On comparing the self-supervised ResMLP model with the supervised setup for training data, the ratio of the number of wins was $17/20$, $19/20$, and $20/20$ for LSTM, inception, and transformer respectively. With more wins, we have shown that multi-modal self-supervised learning gains knowledge about the crop lands. When compared to uni-modal self-supervised ResMLP model, the multi-modal self-supervised's win-ratio is $\sim 20/20$ for all the three base models.

In the case of supervised models, the classification accuracies for LSTM, inception and transformer are $66.7\% \pm 2.53\%$, $25.84\% \pm 4.65\%$, and $71.39\% \pm 4.54\%$ respectively. The corresponding box plot shows the range of relative gain over the supervised setup. For LSTM, the range of relative gain over the supervised experiment is between -4.58% and 3.87% for uni-modal self-supervised versus -2.34% and 9.14% for multi-modal self-supervised. In the case of inception the range is between -14.8% and -5.2% for uni-modal self-supervised versus 3.18% and 13.90% for multi-modal self-supervised. For transformers, the range is between -5.57% and 11.76% for uni-modal self-supervised versus 0.66% and 17.04% for multi-modal self-supervised.

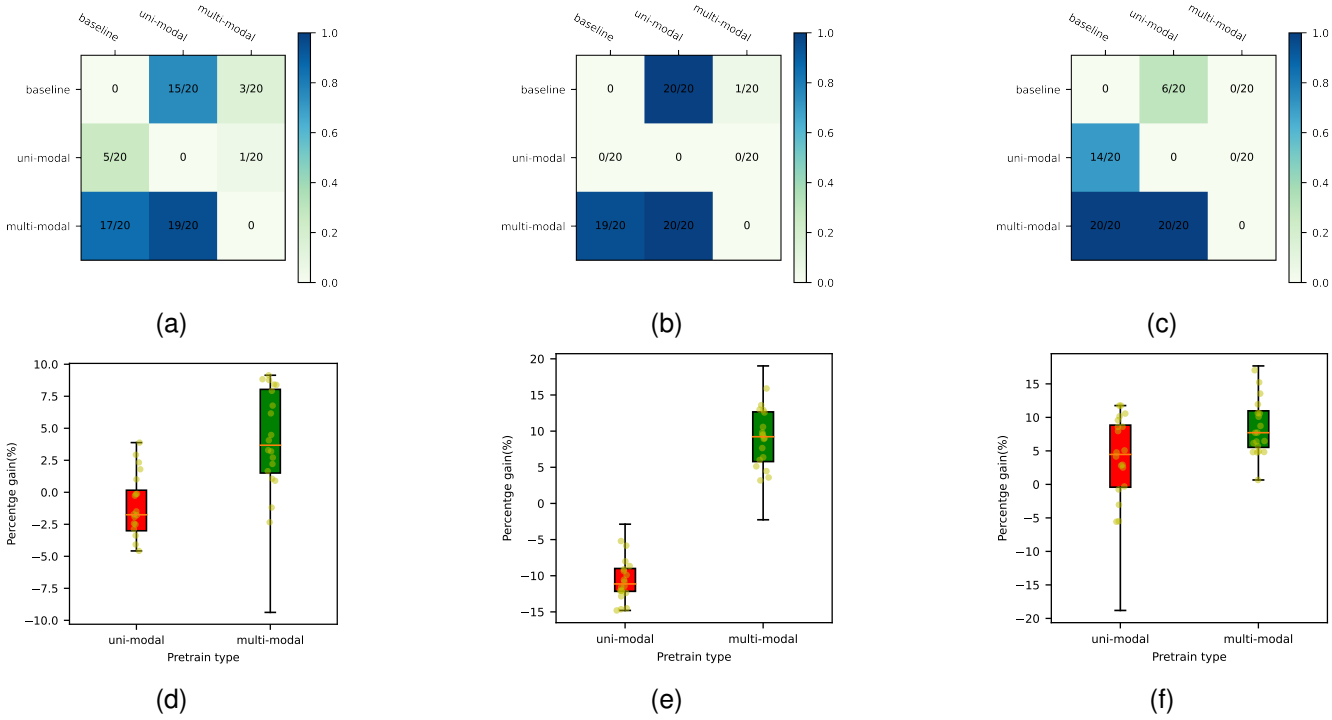


Fig. 5: **Win-matrix and box plot for ResMLP backbone model on downstream task1.** a, b & c shows the win-matrix for LSTM, inception and transformer respectively. d, e & f corresponds to box plot showing relative gain for both uni-modal and multi-modal self-supervised over the supervised experiments for LSTM, inception and transformer respectively.

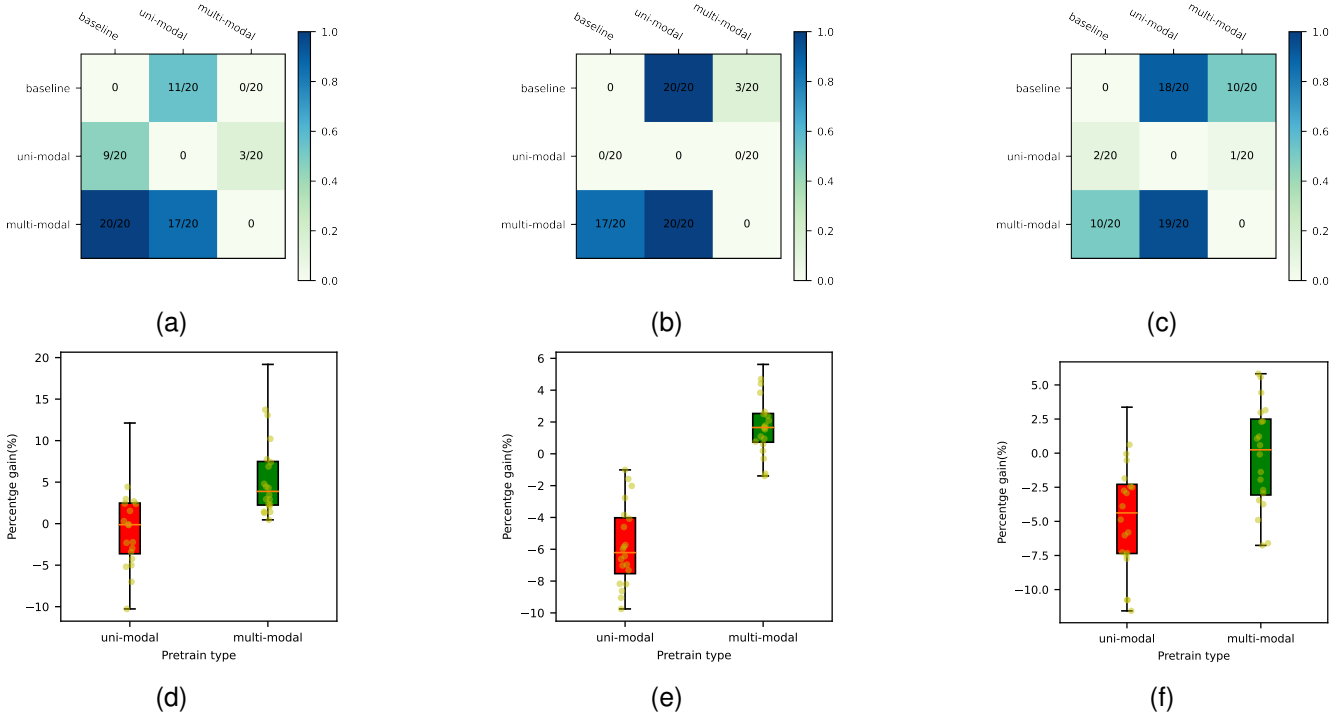


Fig. 6: **Win-matrix and box plot for ResMLP backbone model on downstream task2.** a, b & c shows the win-matrix for lstm, inception and transformer respectively. d, e & f corresponds to the box plot showing the relative gain for both uni-modal and multi-modal self-supervised over the supervised experiments for LSTM, inception and transformer respectively.

TABLE I: **Table showing the accuracy of the supervised setup and relative gain of uni-modal and multi-modal self-supervised experiment for different downstream tasks.**

		Supervised Accuracy (mean $\pm std$)	Relative gain for uni-modal (max/min)	Relative gain for multi-modal (max/min)
Downstream Task1	LSTM	66.70% $\pm 2.53\%$	-4.58% / 3.87%	-2.34% / 9.14%
	InceptionTime	25.84% $\pm 4.65\%$	-14.8% / -5.2%	3.18% / 13.90%
	Transformers	71.39% $\pm 4.54\%$	-5.57% / 11.76%	0.66% / 17.04%
Downstream Task2	LSTM	59.31% $\pm 5.75\%$	-10.27% / 4.43%	0.46% / 13.72%
	InceptionTime	20.43% $\pm 3.98\%$	-9.75% / -1.01%	-1.39% / 4.7%
	Transformers	80.83% $\pm 2.69\%$	-11.56% / 0.62%	-6.76% / 5.82%

Figure 6 shows the results and interpretation for both pre-trained models when tested on downstream task2. The purpose of this dataset was to check how the model behaves when tested on a data of different year and at different geographical region. We found that the uni-modal self-supervised model's performance were poor for ResMLP pre-trained models for all the three baseline models. Similar to the case of downstream task1, the multi-modal self-supervised model outperformed the uni-model self-supervised model for all the competing experiment setups. When comparing the multi-modal self-supervised MLP model with the base model the win ratio were $20/20$, $17/20$ & $10/20$ for LSTM, inception, and transformer model and with self-supervised model, the win-ratio were $17/20$, $20/20$ and $19/20$. In the case of supervised models, the classification accuracies for LSTM, inception and transformer are $59.31\% \pm 5.75\%$, $20.43\% \pm 3.98\%$ and $80.83\% \pm 2.69\%$ respectively. From the box plots in Figure 6, we found the range of relative gain over the supervised experiment in the case of LSTM to be lying between -10.27% and 4.43% for uni-modal self-supervised and for multi-modal self-supervised it is between 0.46% and 13.72%. In the case of inception the range is between -9.75% and -1.01% for uni-modal self-supervised versus -1.39% and 4.7% for multi-modal self-supervised. For transformers, the range lies between -11.56% and 0.62% for uni-modal self-supervised and between -6.76% and 5.82% for multi-modal self-supervised. Table I shows the supervised accuracy and relative gain for both uni-modal and multi-modal self-supervised learning on both downstream tasks.

To see similar plots for MLP, please refer to IX-A. In addition, we also found the performance of multi-modal self-supervised model without any random feature corruption technique also produced promising results, but had slightly lower performance when compared to the multi-modal trained with random feature corruption.

VII. Conclusion

In this work, we compare uni-modal self-supervised learning using only Sentinel2 data against multi-modal contrastive learning using Sentinel2 and Planetscope as multi-modal source. We tested our approach on three different base models for crop classification, i.e., bi-directional LSTM, inceptiontime and position encoded transformers. We used MLP and ResMLP as the backbone model for pre-training. Based on our results, we conclude that when contrastive learning was applied only on Sentinel2 using a random feature corruption technique, it was unable to learn an expressive representation for crop classification. On the other hand, when we used multi-modal contrastive self-supervised learning with Sentinel2 and Planetscope, we found a relative gain in performance for the bi-directional LSTM and inceptiontime models, while the gains were smaller for the transformer model on data from a different region and time period. Given the improvement in most test cases, we can conclude that multi-modal contrastive learning helps in learning an expressive representation for crop classification. The added advantage of multi-modal contrastive learning is that the end-user does not have to rely on the commercial Planetscope data for an application and can still benefit from the fine spatial resolution of Planetscope data.

References

[1] M. Drusch, U. Del Bello, S. Carlier, O. Colin, V. Fernandez, F. Gascon, B. Hoersch, C. Isola, P. Laberinti, P. Martimort, A. Meygret, F. Spoto, O. Sy, F. Marchese, and P. Bargellini, "Sentinel-2: Esa's optical high-resolution mission for gmes operational services," *Remote Sensing of Environment*, vol. 120, pp. 25–36, 2012, the Sentinel Missions - New Opportunities for Science. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425712000636

[2] U. Meier, H. Bleiholder, L. Buhr, C. Feller, H. Hack, M. Heß, P. Lancashire, U. Schnock, R. Stauß, T. Boom, E. Weber, and P. Zwerger, "The bbch system to coding the phenological growth stages of plants-history and publications," *Journal für Kulturpflanzen*, vol. 61, pp. 41–52, 01 2009.

[3] M. Račič, K. Oštir, D. Peressutti, A. Zupanc, and L. Čehovin Zajc, "Application of temporal convolutional neural network for the classification of crops on sentinel-2 time series," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLIII-B2-2020, pp. 1337–1342, 08 2020.

[4] C. Hütt, G. Waldhoff, and G. Bareth, "Fusion of sentinel-1 with official topographic and cadastral geodata for crop-type enriched lulc mapping using foss and open data," *ISPRS International Journal of Geo-Information*, vol. 9, no. 2, 2020. [Online]. Available: https://www.mdpi.com/2220-9964/9/2/120

[5] X. Yang, X. He, Y. Liang, Y. Yang, S. Zhang, and P. Xie, "Transfer learning or self-supervised learning? A tale of two pretraining paradigms," *CoRR*, vol. abs/2007.04234, 2020. [Online]. Available: https://arxiv.org/abs/2007.04234

[6] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2021. [Online]. Available: https://doi.org/10.1109\%2Ftkde.2021.3090866

[7] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *CoRR*, vol. abs/1807.03748, 2018. [Online]. Available: http://arxiv.org/abs/1807.03748

[8] S. Purushwalkam and A. Gupta, "Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/22f791da07b0d8a2504c2537c560001c-Abstract.html

[9] X. Yuan, Z. Lin, J. Kuen, J. Zhang, Y. Wang, M. Maire, A. Kale, and B. Faieta, "Multimodal contrastive training for visual representation learning," *CoRR*, vol. abs/2104.12836, 2021. [Online]. Available: https://arxiv.org/abs/2104.12836

[10] M. Rußwurm, S. Lefèvre, and M. Körner, "Breizhcrops: A satellite time series dataset for crop type identification," *CoRR*, vol. abs/1905.11893, 2019. [Online]. Available: http://arxiv.org/abs/1905.11893

[11] G. Weikmann, C. Paris, and L. Bruzzone, "Timesen2crop: A million labeled samples dataset of sentinel 2 image time series for crop-type classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 4699–4708, 2021.

[12] E. C. J. R. Centre., SatCen., and E. S. Agency., *Proceedings of the 2021 conference on Big Data from Space: 18 20 May 2021.* Publications Office, 2021. [Online]. Available: https://data.europa.eu/doi/10.2760/125905

[13] L. Kondmann, A. Toker, M. Rußwurm, A. Camero, D. Peressuti, G. Milcinski, P.-P. Mathieu, N. Longépé, T. Davis, G. Marchisio, L. Leal-Taixé, and X. X. Zhu, "DENETHOR: The dynamicearthNET dataset for harmonized, inter-operable, analysis-ready, daily crop monitoring from space," in *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. [Online]. Available: https://openreview.net/forum?id=uUa4jNMLjrL

[14] "Planet application program interface: In space for life on earth," Planet, 2017–. [Online]. Available: https://api.planet.com

[15] N. Gorelick, M. Hancher, M. Dixon, S. Ilyushchenko, D. Thau, and R. Moore, "Google earth engine: Planetary-scale geospatial analysis for everyone," *Remote Sensing of Environment*, 2017. [Online]. Available: https://doi.org/10.1016/j.rse.2017.06.031

[16] O. Mañas, A. Lacoste, X. Giró-i-Nieto, D. Vázquez, and P. Rodríguez, "Seasonal contrast: Unsupervised pre-training from uncurated remote sensing data," *CoRR*, vol. abs/2103.16607, 2021. [Online]. Available: https://arxiv.org/abs/2103.16607

[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *CoRR*, vol. abs/1409.0575, 2014. [Online]. Available: http://arxiv.org/abs/1409.0575

[18] L. Scheibenreif, J. Hanna, M. Mommert, and D. Borth, "Self-supervised vision transformers for land-cover segmentation and classification," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2022, New Orleans, LA, USA, June 19-20, 2022*. IEEE, 2022, pp. 1421–1430. [Online]. Available: https://doi.org/10.1109/CVPRW56347.2022.00148

[19] D. Bahri, H. Jiang, Y. Tay, and D. Metzler, "SCARF: self-supervised contrastive learning using random feature corruption," *CoRR*, vol. abs/2106.15147, 2021. [Online]. Available: https://arxiv.org/abs/2106.15147

[20] B. Bischl, G. Casalicchio, M. Feurer, P. Gijsbers, F. Hutter, M. Lang, R. G. Mantovani, J. N. van Rijn, and J. Vanschoren, "Openml: A benchmarking layer on top of openml to quickly create, download, and share systematic benchmarks," *NeurIPS*, 2021. [Online]. Available: https://openreview.net/forum?id=OCrD8ycKjG

[21] K. J. Geras and C. Sutton, "Scheduled denoising autoencoders," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1406.3269

[22] G. Somepalli, M. Goldblum, A. Schwarzschild, C. B. Bruss, and T. Goldstein, "SAINT: improved neural networks for tabular data via row attention and contrastive pre-training," *CoRR*, vol. abs/2106.01342, 2021. [Online]. Available: https://arxiv.org/abs/2106.01342

[23] X. Huang, A. Khetan, M. Cvitkovic, and Z. S. Karnin, "Tabtransformer: Tabular data modeling using contextual embeddings," *CoRR*, vol. abs/2012.06678, 2020. [Online]. Available: https://arxiv.org/abs/2012.06678

[24] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," *CoRR*, vol. abs/1905.04899, 2019. [Online]. Available: http://arxiv.org/abs/1905.04899

[25] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *CoRR*, vol. abs/1710.09412, 2017. [Online]. Available: http://arxiv.org/abs/1710.09412

[26] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," *CoRR*, vol. abs/2002.05709, 2020. [Online]. Available: https://arxiv.org/abs/2002.05709

[27] J. Yoon, Y. Zhang, J. Jordon, and M. van der Schaar, "Vime: Extending the success of self- and semi-supervised learning to tabular domain," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 11 033–11 043. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/7d97667a3e056acab9aaf653807b4a03-Paper.pdf

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[29] S. Cornegruta, R. Bakewell, S. Withey, and G. Montana, "Modelling radiological language with bidirectional long short-term memory networks," in *Proceedings of the Seventh International Workshop on Health Text Mining and Information Analysis*. Auxtin, TX: Association for Computational Linguistics, Nov. 2016, pp. 17–27. [Online]. Available: https://aclanthology.org/W16-6103

[30] H. I. Fawaz, B. Lucas, G. Forestier, C. Pelletier, D. F. Schmidt, J. Weber, G. I. Webb, L. Idoumghar, P. Muller, and F. Petitjean, "Inceptiontime: Finding alexnet for time series classification," *CoRR*, vol. abs/1909.04939, 2019. [Online]. Available: http://arxiv.org/abs/1909.04939

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[32] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," ser. KDD '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 2623–2631. [Online]. Available: https://doi.org/10.1145/3292500.3330701

VIII. Biography Section

Ankit Patnala received his B.Tech degree from National Institute of Technology, Rourkela, India in 2015 from the Department of Mechanical Engineering and M.S degree from RWTH Aachen, Aachen, Germany from the faculty of Mechanical Engineering. He is currently a Doctoral Researcher at Forschungszentrum Juelich, Juelich, Germany. His research interest lies in the field of self-supervised learning to remotes sensing images for various earth science applications.

Scarlet Stadtler Scarlet Stadtler received her M.Sc. degree in Physics of the Earth and Atmosphere and her Ph.D. degree in Meteorology from Bonn University, Bonn, Germany, in 2015 and 2018, respectively. Currently, she works as a postdoctoral researcher at the Jülich Supercomputing Centre (JSC), Forschungszentrum Jülich, Jülich, Germany, focusing on Machine and Deep Learning for Earth Science. Her primary research area involves Explainable Machine Learning for Meteorology and Atmospheric Chemistry. Her research interests include Representation Learning and Uncertainty Quantification.

Martin G. Schultz Martin G. Schultz has a PhD in physical chemistry and a habilitation in meteorology. He has been working on atmospheric chemistry and dynamics problems since more than 30 years and became highly cited researcher twice. Since 2017, he has been working at the Jülich Supercomputing Centre (JSC), where he established a research group to develop innovative deep learning solutions for problems in the Earth sciences. Schultz is co-founder of the AtmoRep consortium, which develops a large-scale foundation model for atmospheric dynamics.

Juergen Gall Juergen Gall (Member, IEEE) received the B.Sc. degree in mathematics from the University of Wales, Swansea, U.K., in 2004, the M.Sc. degree in mathematics from the University of Mannheim, Mannheim, Germany, in 2005, and the Ph.D. degree in computer science from Saarland University, Saarbrücken, Germany, and the Max-Planck-Institut für Informatik, Saarbrücken, in 2009. He was a Post-Doctoral Researcher with the Computer Vision Laboratory, ETH Zürich, Zürich, Switzerland, from 2009 to 2012, and a Senior Research Scientist with the Max Planck Institute for Intelligent Systems, Tübingen, Germany, from 2012 to 2013. Since 2013, he has been a Professor with the University of Bonn, Bonn, Germany, where he is currently the Head of the Computer Vision Group. He is also a member of the Lamarr Institute for Machine Learning and Artificial Intelligence, Dortmund, Germany.

IX. Appendices

A. Results on MLP

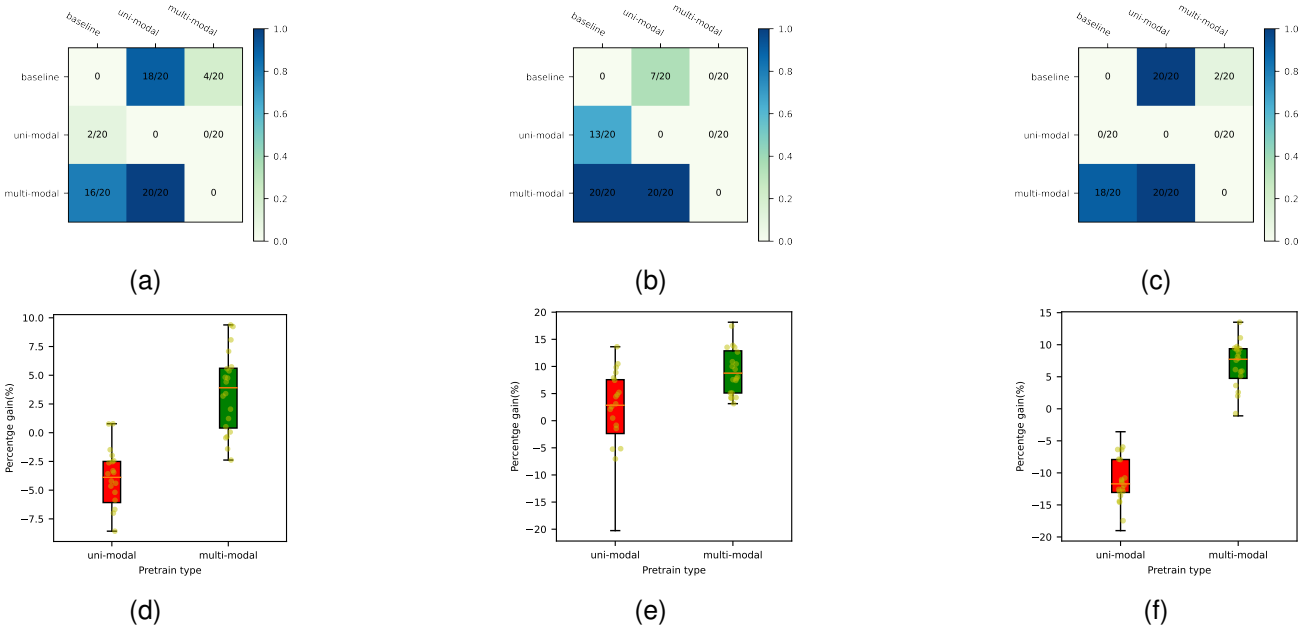


Fig. 7: **Win-matrix and box plot for MLP backbone model on downstream task 1.** a, b & c shows the win-matrix for LSTM, transformer and inception respectively. d, e & f corresponds to the box plot showing the relative gain for both uni-modal and multi-modal self-supervised over the supervised experiments for LSTM, transformer and inception respectively.

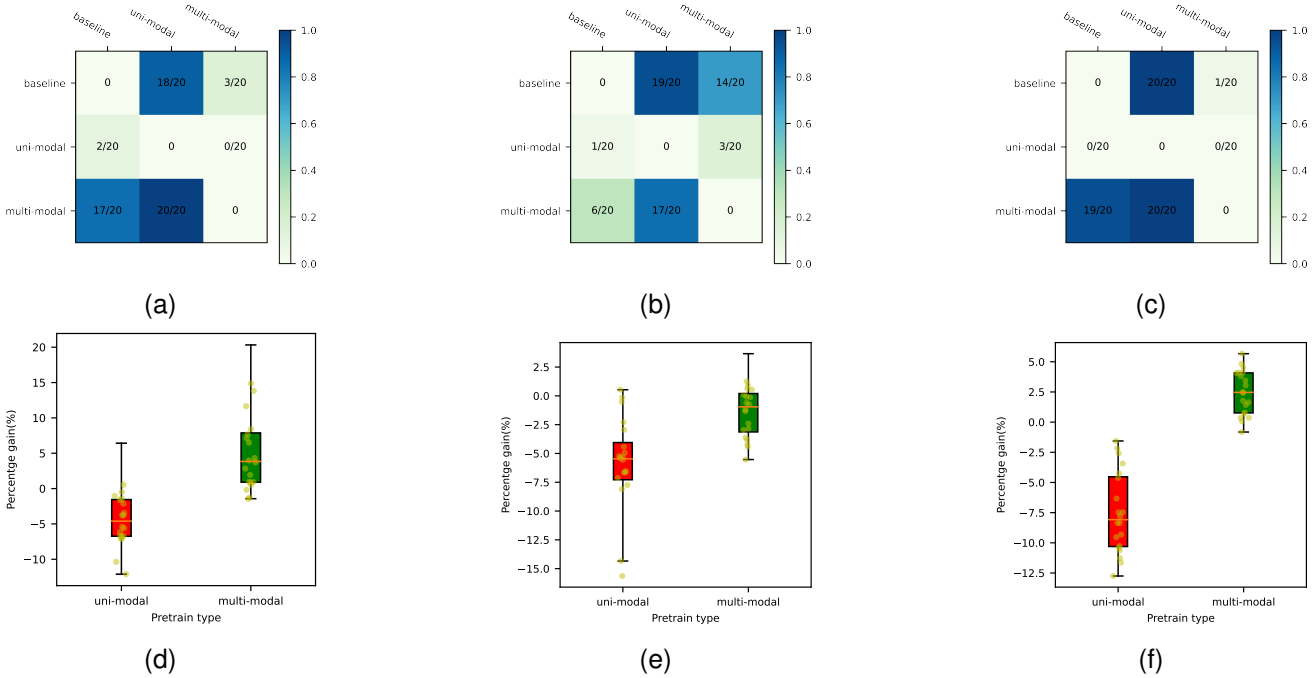


Fig. 8: **Win-matrix and box plot for MLP backbone model on downstream task 2.** a, b & c shows the win-matrix for LSTM, transformer and inception respectively. d, e & f corresponds to the box plot showing the relative gain for both uni-modal and multi-modal self-supervised over the supervised experiments for LSTM, transformer and inception respectively.