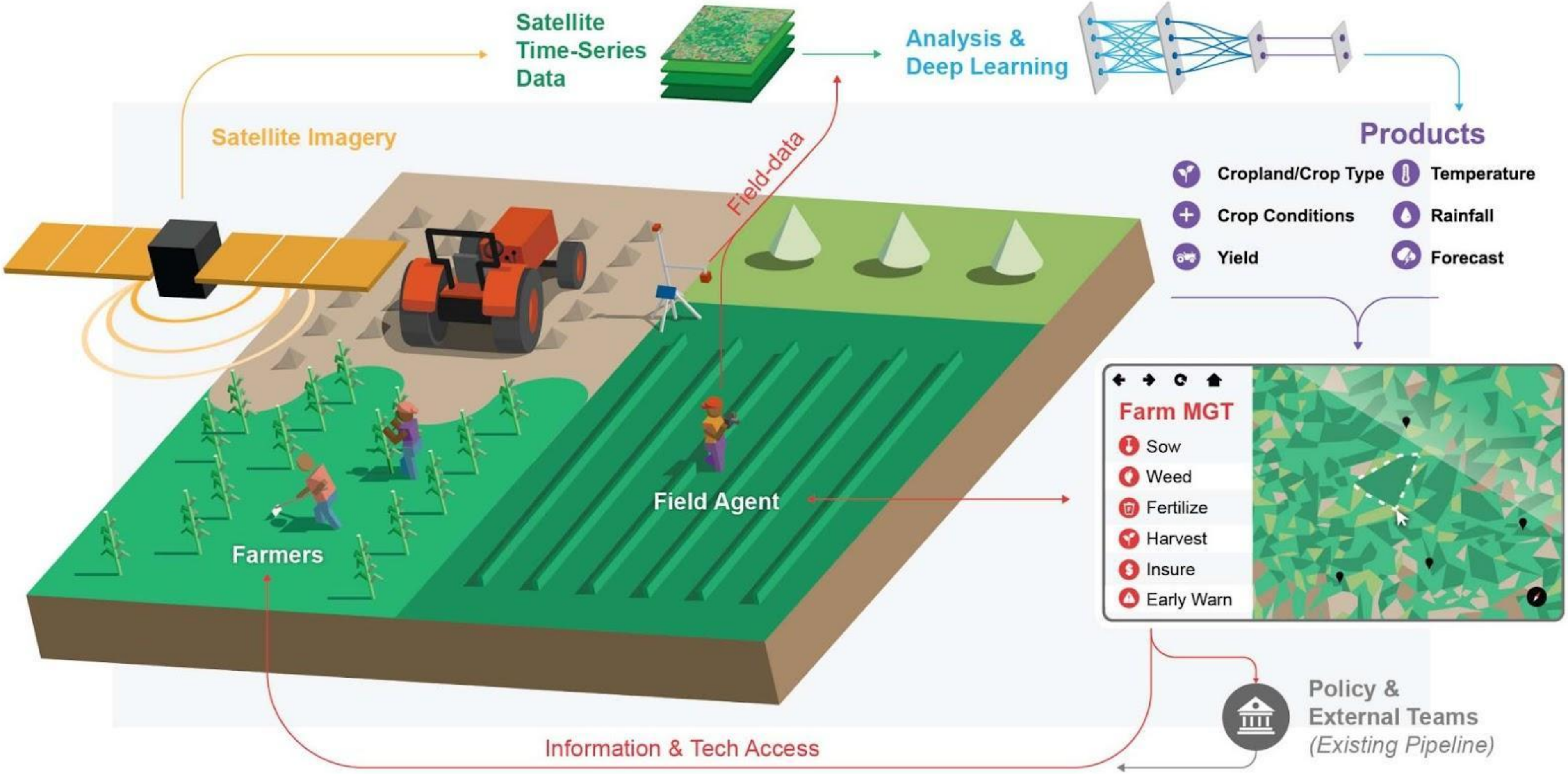
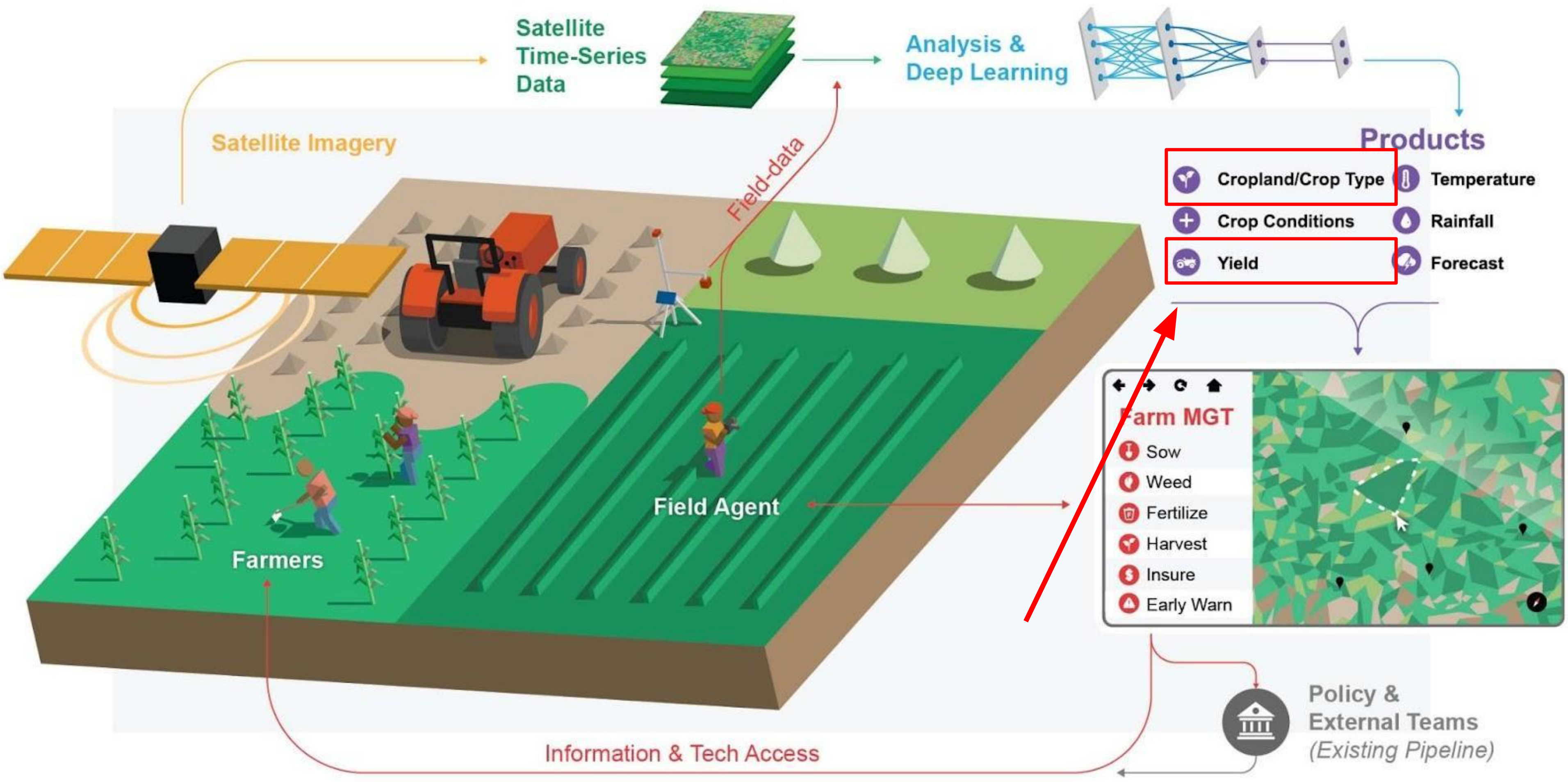
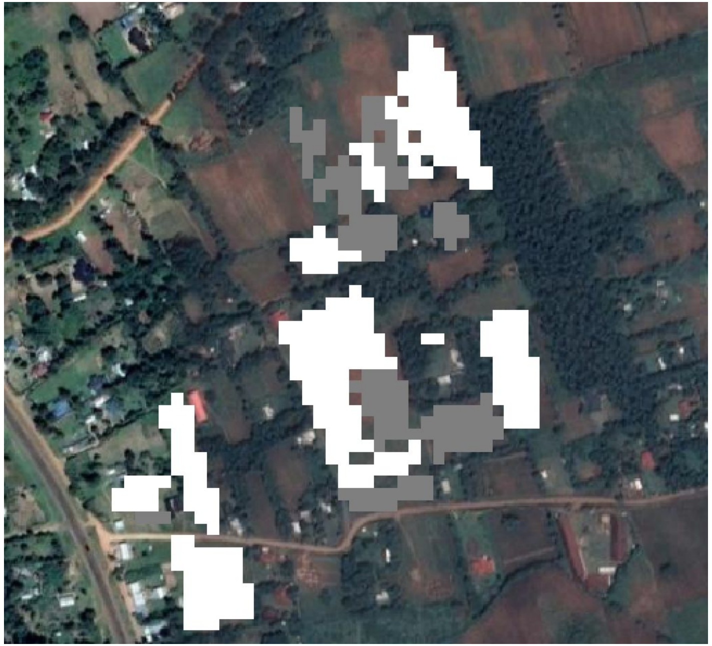
Task-Informed Meta-Learning for agriculture

Gabriel Tseng, Hannah Kerner, David Rolnick
NASA Harvest, University of Maryland, McGill University, Mila

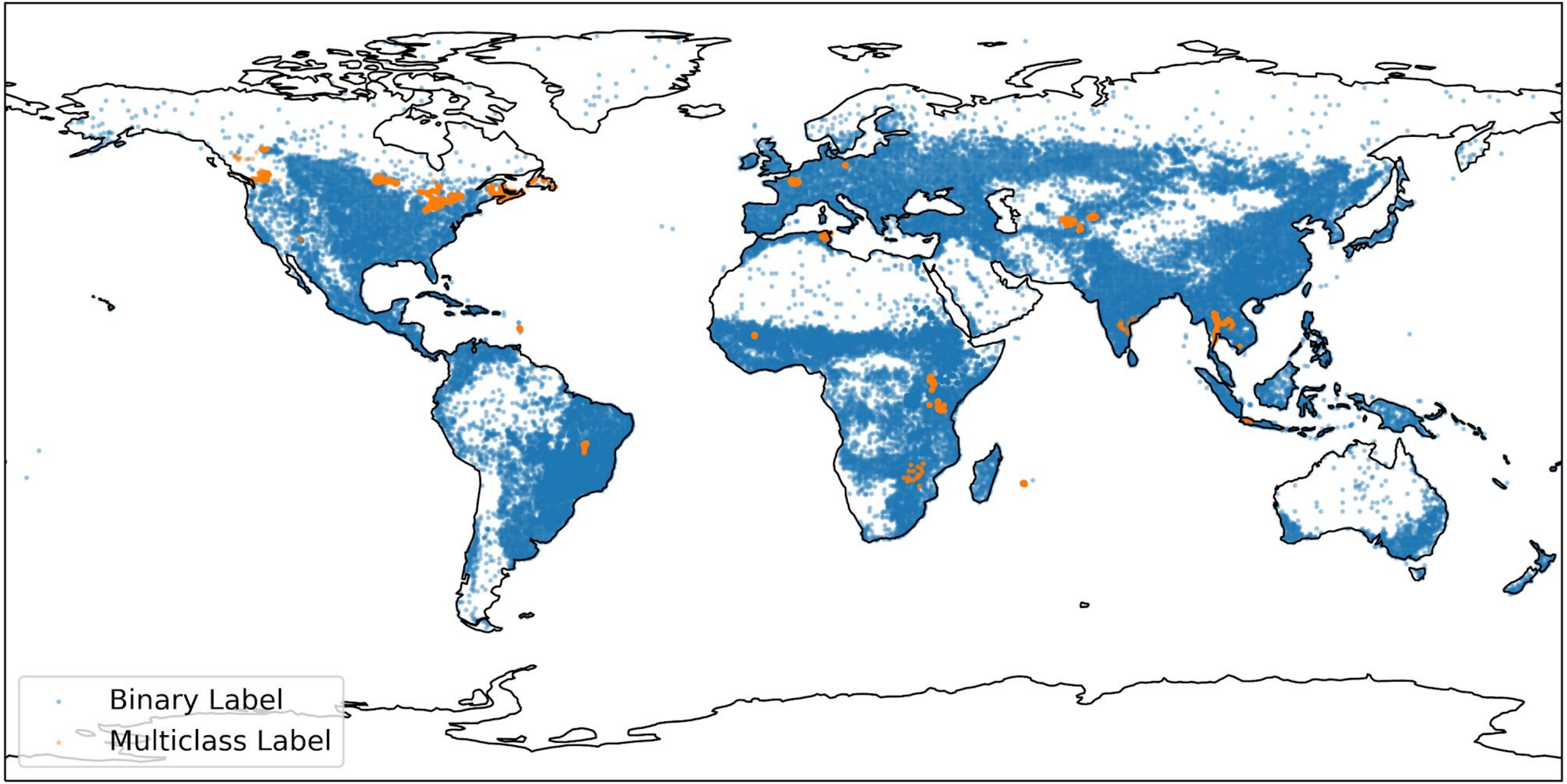

Satellite
Time-Series
Data
Analysis &
Deep Learning
Satellite Imagery
Field-data
Products
Cropland/Crop Type
Temperature
Crop Conditions
Rainfall
Yield
Forecast
Farm MGT
Sow
Weed
Fertilize
Harvest
Insure
Early Warn
Field Agent
Farmers
Policy &
External Teams
(Existing Pipeline)
Information & Tech Access


Satellite
Time-Series
Data
Analysis &
Deep Learning
Satellite Imagery
Field-data
Products
Cropland/Crop Type
Temperature
Crop Conditions
Rainfall
Yield
Forecast
Farm MGT
Sow
Weed
Fertilize
Harvest
Insure
Early Warn
Field Agent
Farmers
Policy &
External Teams
(Existing Pipeline)
Information & Tech Access

For crop type (as opposed to crop / non crop) mapping, photo-interpretation is hard / impossible so better leveraging (very) small datasets becomes more important



There is plentiful global data to learn from



The CropHarvest dataset

GeoWiki

Radiant Earth Foundation

EARTH IMAGERY FOR IMPACT

PlantVillage

https://github.com/nasaharvest/cropharvest

CropHarvest: a global satellite dataset for crop type classification

Gabriel Tseng
NASA Harvest
gabrieltseng95@gmail.com

Ivan Zvonkov
University of Maryland, College Park
izvonkov@umd.edu

Catherine Nakalembe
University of Maryland, College Park
cnakalem@umd.edu

Hannah Kerner
University of Maryland, College Park
hkerner@umd.edu

Abstract

Remote sensing datasets pose a number of interesting challenges to machine learning researchers and practitioners, from domain shift (spatially, semantically and temporally) to highly imbalanced labels. In addition, the outputs of models trained on remote sensing datasets can contribute to positive societal impacts, for example in food security and climate change. However, there are many barriers that limit the accessibility of satellite data to the machine learning community, including a lack of large labeled datasets as well as an understanding of the range of satellite products available, how these products should be processed, and how to manage multi-dimensional geospatial data. To lower these barriers

The labels are coupled to remote sensing products

Each label has the following inputs per month, for 12 months:

B2	B3	B4	B5	B6	B7	B8	B8A	B9	B11	B12	VV	VH	precip	temp	elevation	topography	NDVI

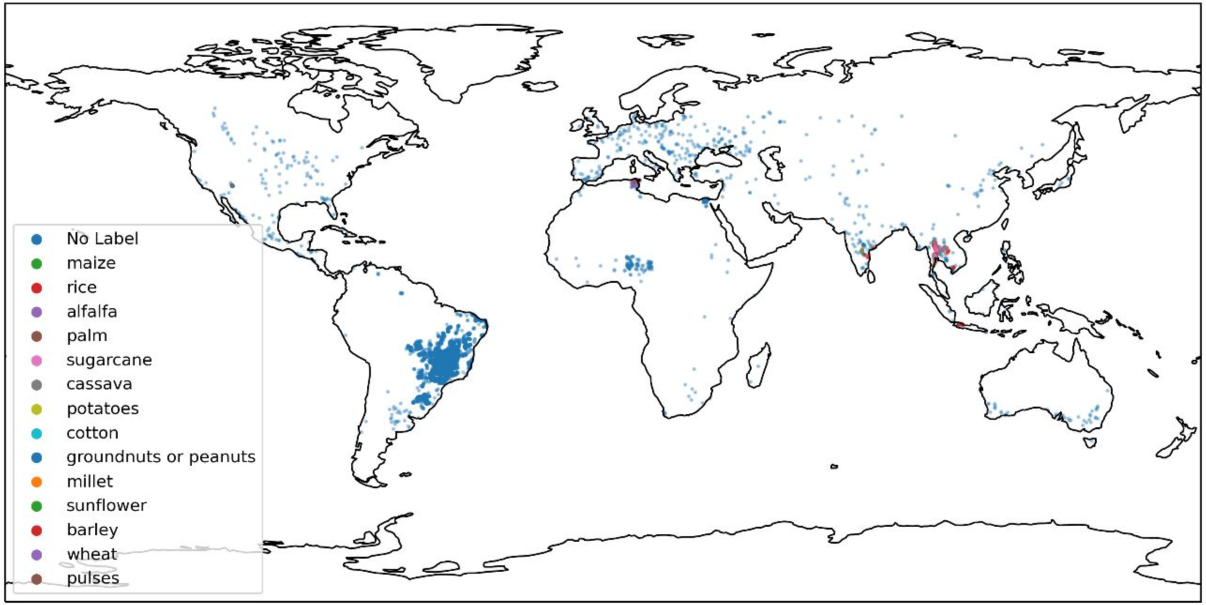
Sentinel 2 L1C: Take the least cloudy pixel over the 30 day window

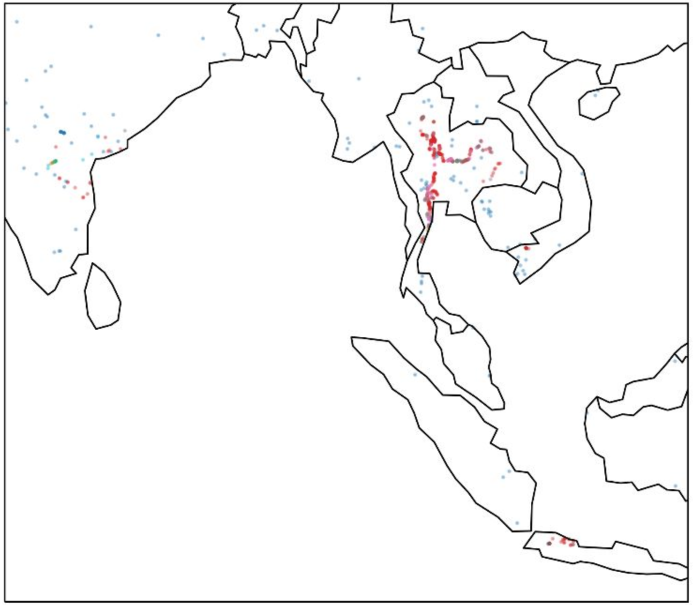
Sentinel 1: Take the median of available images (and the closest available one if none is)

SRTM DEM

ERA5 monthly means

This data is very heterogenous





GFSAD: https://www.croplands.org/app/data/search

Meta-learning optimizes for many different tasks, instead of a single task

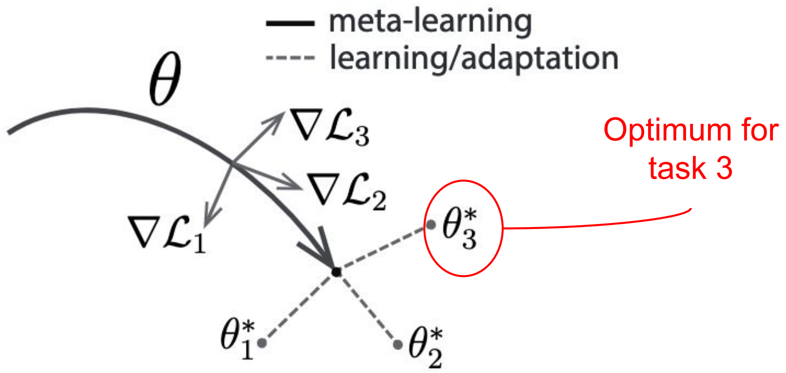
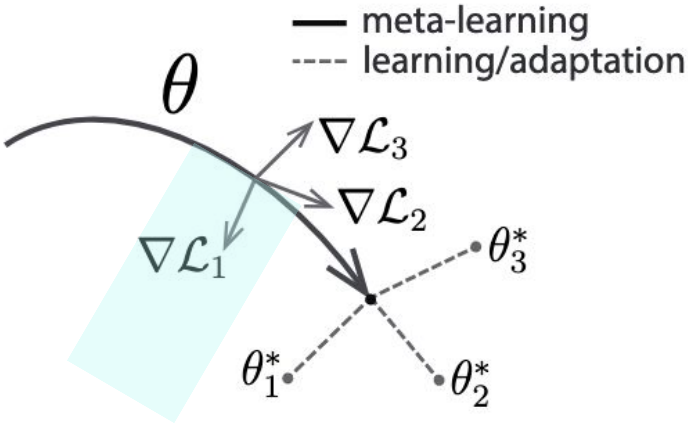


Figure 1. Diagram of our model-agnostic meta-learning algorithm (MAML), which optimizes for a representation θ that can quickly adapt to new tasks.

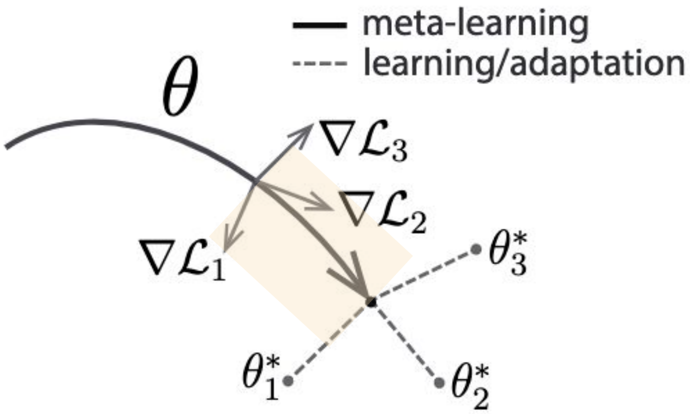
Finn, Chelsea, Pieter Abbeel, and Sergey Levine. "Model-agnostic meta-learning for fast adaptation of deep networks." *International conference on machine learning*. PMLR, 2017.



Algorithm 1 Model-Agnostic Meta-Learning

Require: $p(\mathcal{T})$: distribution over tasks
Require: α, β: step size hyperparameters

```
randomly initialize θ
while not done do
    Sample batch of tasks T_i ~ p(T)
    for all T_i do
        Evaluate ∇_θ L_{T_i}(f_θ) with respect to K examples
        Compute adapted parameters with gradient descent: θ'_i = θ − α∇_θ L_{T_i}(f_θ)
    end for
    Update θ ← θ − β∇_θ Σ_{T_i~p(T)} L_{T_i}(f_{θ'_i})
end while
```



Algorithm 1 Model-Agnostic Meta-Learning

Require: $p(\mathcal{T})$: distribution over tasks
Require: α, β: step size hyperparameters
1: randomly initialize θ
2: **while** not done **do**
3: Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T})$
4: **for all** $\mathcal{T}_i$ **do**
5: Evaluate $\nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$ with respect to K examples
6: Compute adapted parameters with gradient descent: $\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$
7: **end for**
8: Update $\theta \leftarrow \theta - \beta \nabla_\theta \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_i'})$
9: **end while**

This allows us to take advantage of all the information in the labels, even if they are specific to a region

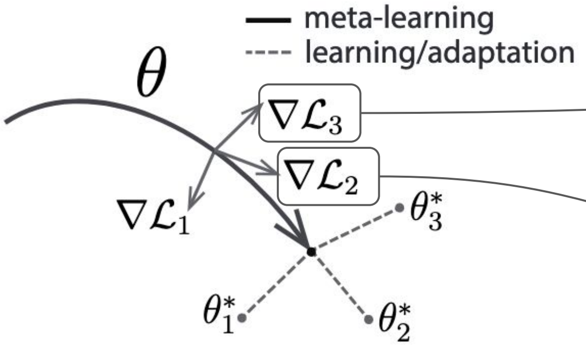


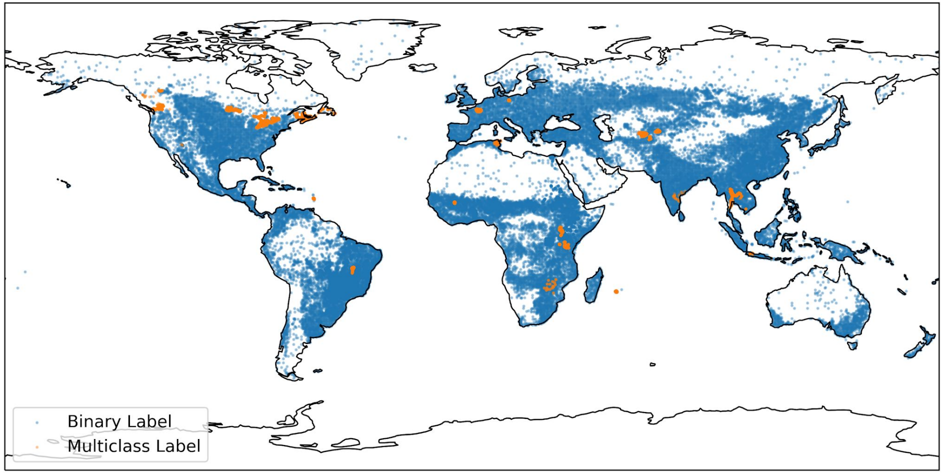
Figure 1. Diagram of our model-agnostic meta-learning algorithm (MAML), which optimizes for a representation θ that can quickly adapt to new tasks.

Kenya task, with groundnut class

India task, with “peanuts or groundnuts” class

We define two types of tasks to train the model

All tasks have spatial delinatations drawn using **bounding boxes** for countries

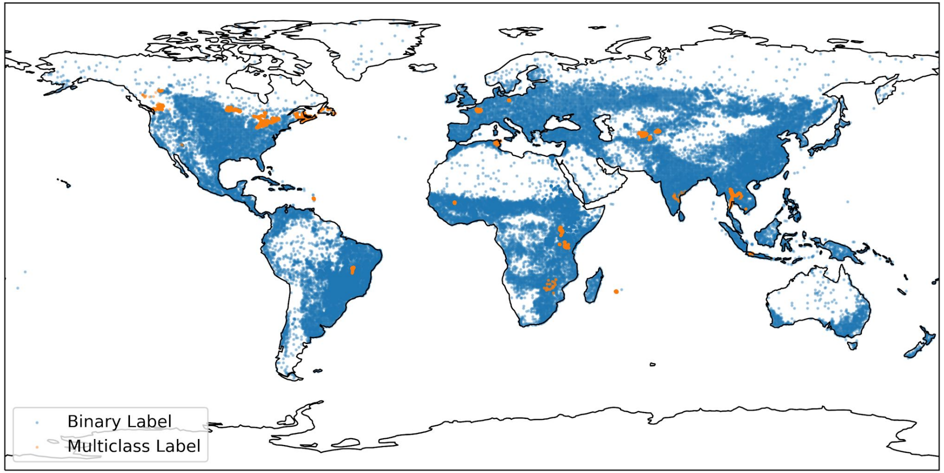


We define two types of tasks to train the model

All tasks have spatial delinatations drawn using **bounding boxes** for countries

- Crop vs. non crop tasks

All data points have a crop / non crop label. All data points within a country's bounding box are included.

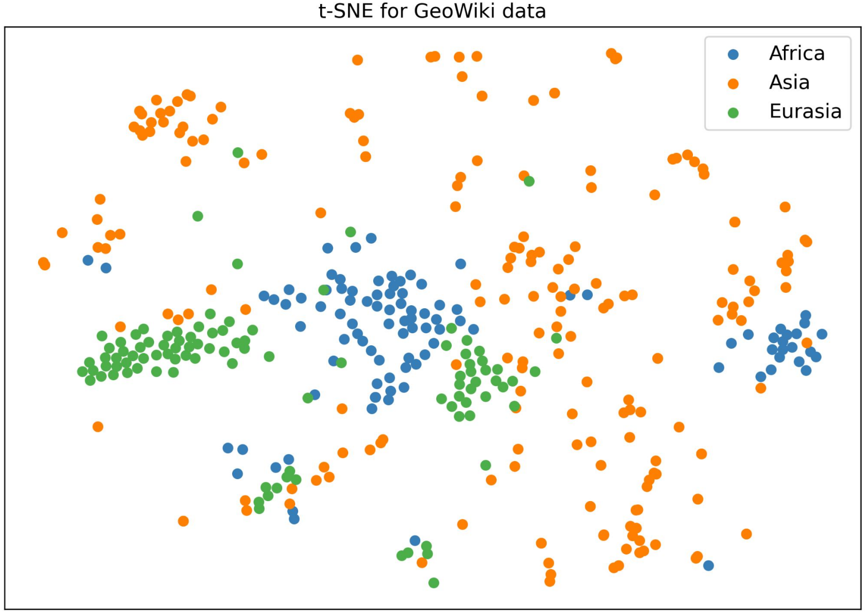


- Land use type vs. rest tasks

Negative examples are constructed from the other crop type labels, and the non-crop labels.

Also includes specific land uses, such as "Cerrado" - natural grassland in Brazil. Since this is non-crop, negative examples are constructed from crop labels.

There is additional information about the tasks we might want to pass to the model



Geographic clustering

How can we pass this information to the model?

Task information: Location and crop classification encoding

x1	x2	x3	Crop classification encoding

A 3D representation of latitude and longitude so that distance is respected (i.e. longitude=180 is close to longitude=-180 in this input space)

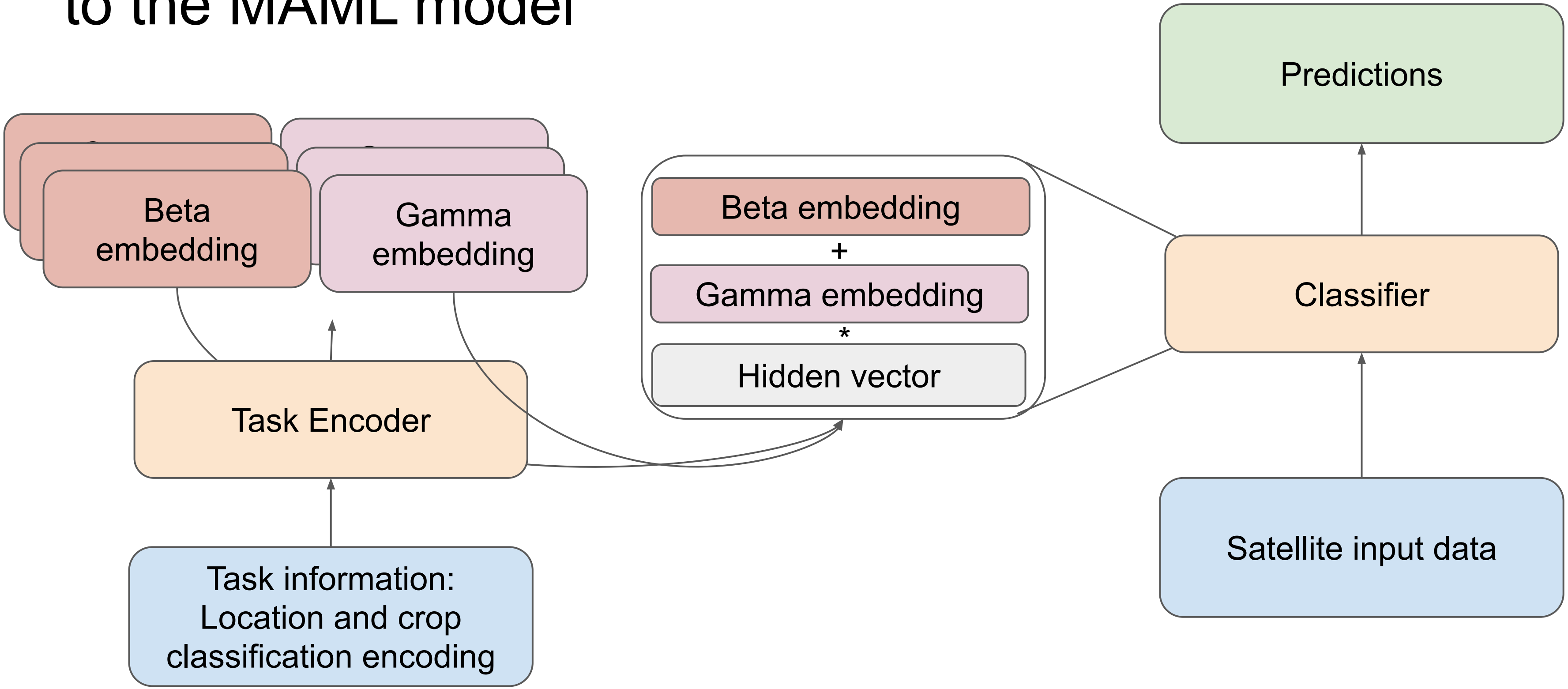
$$\begin{bmatrix} \text{lat} \\ \text{lon} \end{bmatrix} \rightarrow \begin{bmatrix} cos(\text{lat}) \times cos(\text{lon}) \\ cos(\text{lat}) \times sin(\text{lon}) \\ sin(\text{lat}) \end{bmatrix}$$

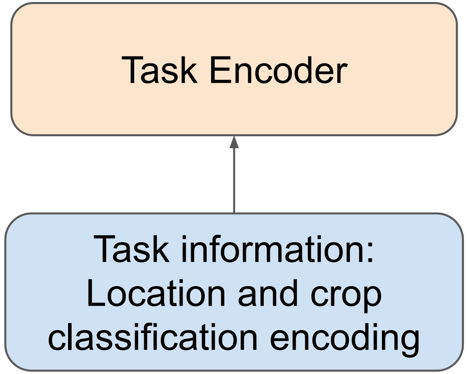
A one hot encoding (for crop type) describing the major crop type (or non crop) being classified. In the case of crop vs. non crop, all crop encoding values are equal to 1/n.

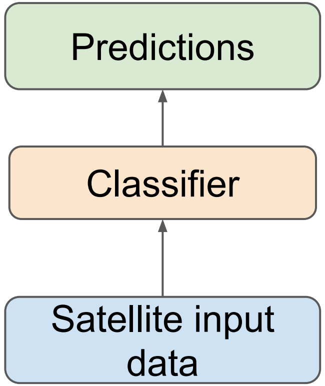
https://stats-class.fao.uniroma2.it/caliper/classification-page/43

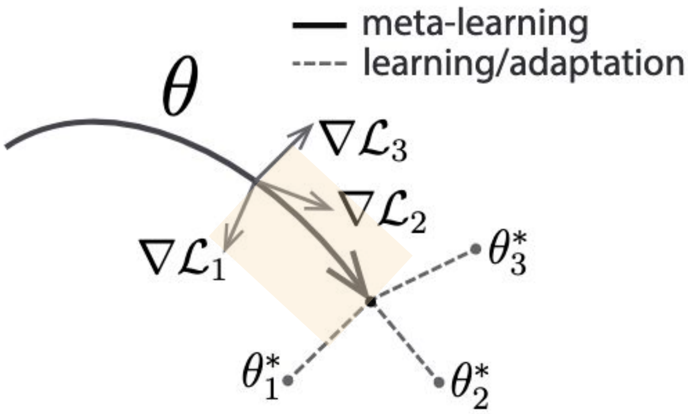
FAO FIAT PANIS

Task aware meta-learning communicates this task vector to the MAML model



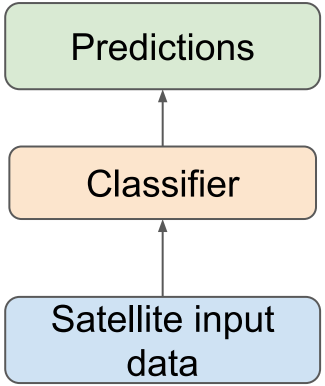


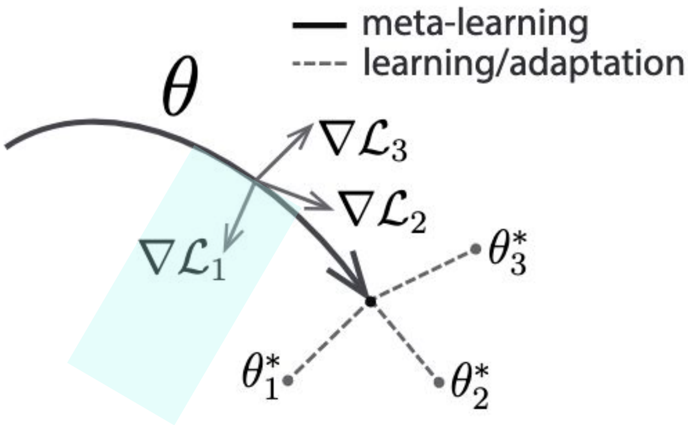




Algorithm 1: Task-Informed Meta-Learning

Require: $p(\mathcal{T})$: Distribution over tasks
Require: α, β: step size hyperparameters
randomly initialize meta model θ_m, task encoder θ_e
while not done **do**
  Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T})$ with task information t_i
  for all $\mathcal{T}_i, t_i$ **do**
    Generate task embeddings $\mu_i = f(t_i; \theta_e)$
    Evaluate $\nabla_{\theta_m} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_m}, \mu_i)$ with respect to K examples
    Compute adapted meta parameters with gradient descent: $\theta'_{m_i} \leftarrow \theta_m - \alpha \nabla_{\theta_m} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_m}, \mu_i)$
  Update $\theta_m \leftarrow \theta_m - \beta \nabla_{\theta_m} \Sigma_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta'_{m_i}}, \mu_i)$
  Update $\theta_e \leftarrow \theta_m - \beta \nabla_{\theta_e} \Sigma_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta'_{m_i}}, \mu_i)$





Algorithm 1: Task-Informed Meta-Learning

Require: $p(\mathcal{T})$: Distribution over tasks
Require: α, β: step size hyperparameters
randomly initialize meta model θ_m, task encoder θ_e
while not done **do**
  Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T})$ with task information t_i
  for all $\mathcal{T}_i, t_i$ **do**
    Generate task embeddings $\mu_i = f(t_i; \theta_e)$
    Evaluate $\nabla_{\theta_m}\mathcal{L}_{\mathcal{T}_i}(f_{\theta_m}, \mu_i)$ with respect to K examples
    Compute adapted meta parameters with gradient descent: $\theta'_{m_i} \leftarrow \theta_m - \alpha\nabla_{\theta_m}\mathcal{L}_{\mathcal{T}_i}(f_{\theta_m}, \mu_i)$
  Update $\theta_m \leftarrow \theta_m - \beta\nabla_{\theta_m}\Sigma_{\mathcal{T}_i\sim p(\mathcal{T})}\mathcal{L}_{\mathcal{T}_i}(f_{\theta'_{m_i}}, \mu_i)$
  Update $\theta_e \leftarrow \theta_m - \beta\nabla_{\theta_e}\Sigma_{\mathcal{T}_i\sim p(\mathcal{T})}\mathcal{L}_{\mathcal{T}_i}(f_{\theta'_{m_i}}, \mu_i)$

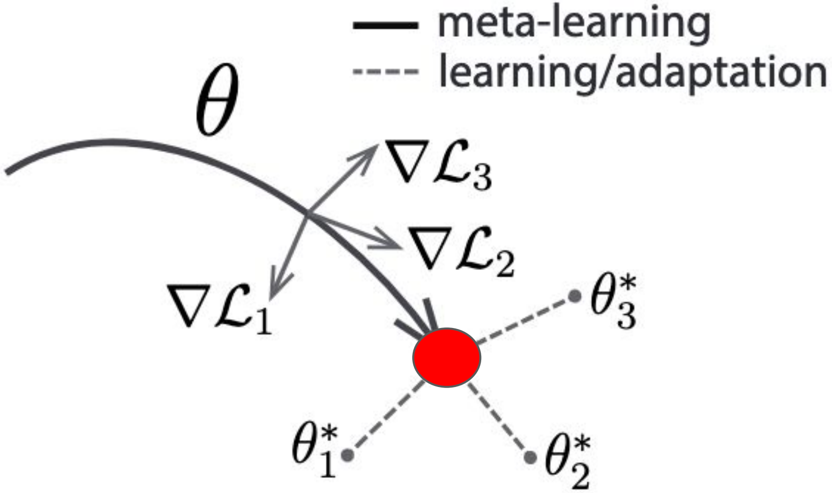


Figure 1. Diagram of our model-agnostic meta-learning algorithm (MAML), which optimizes for a representation θ that can quickly adapt to new tasks.

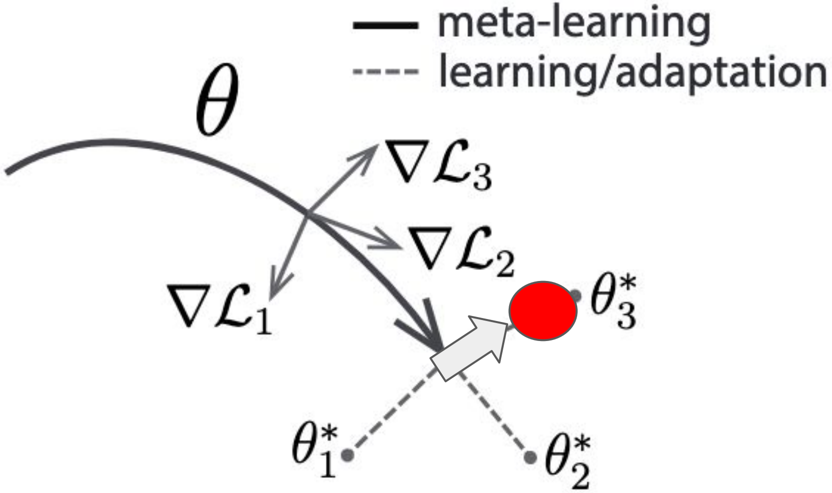
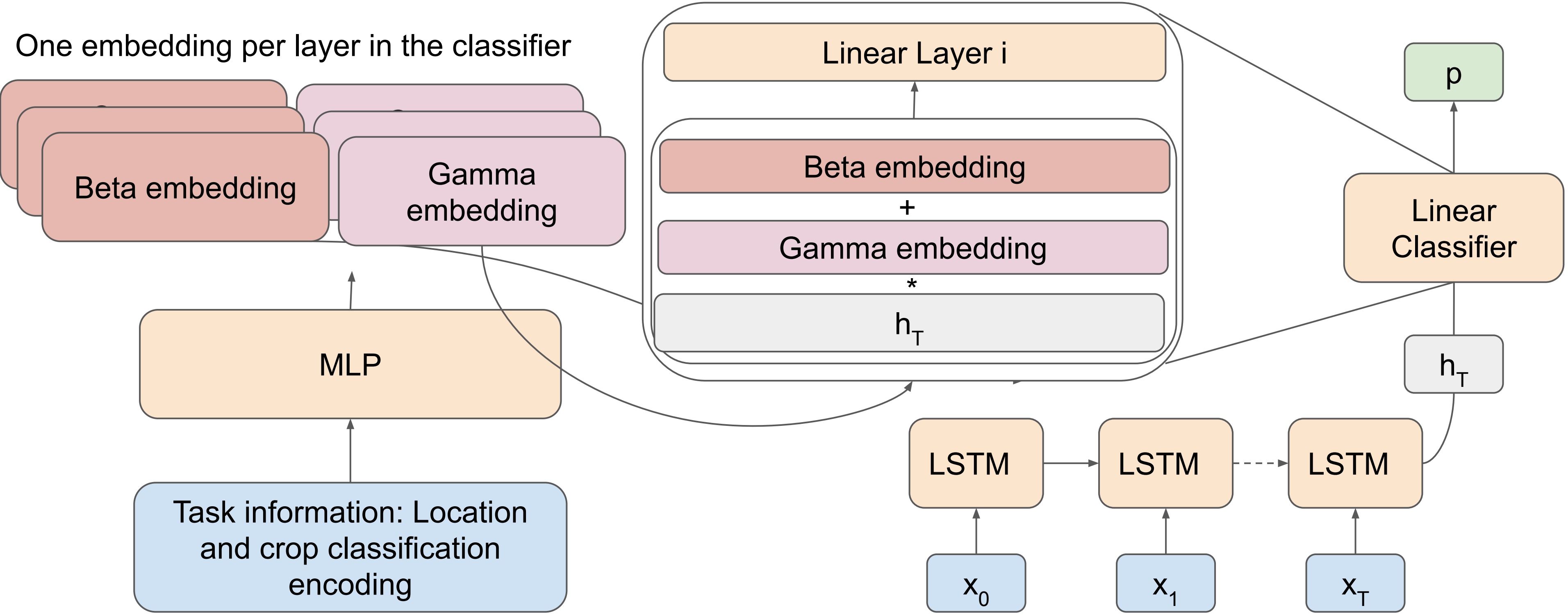
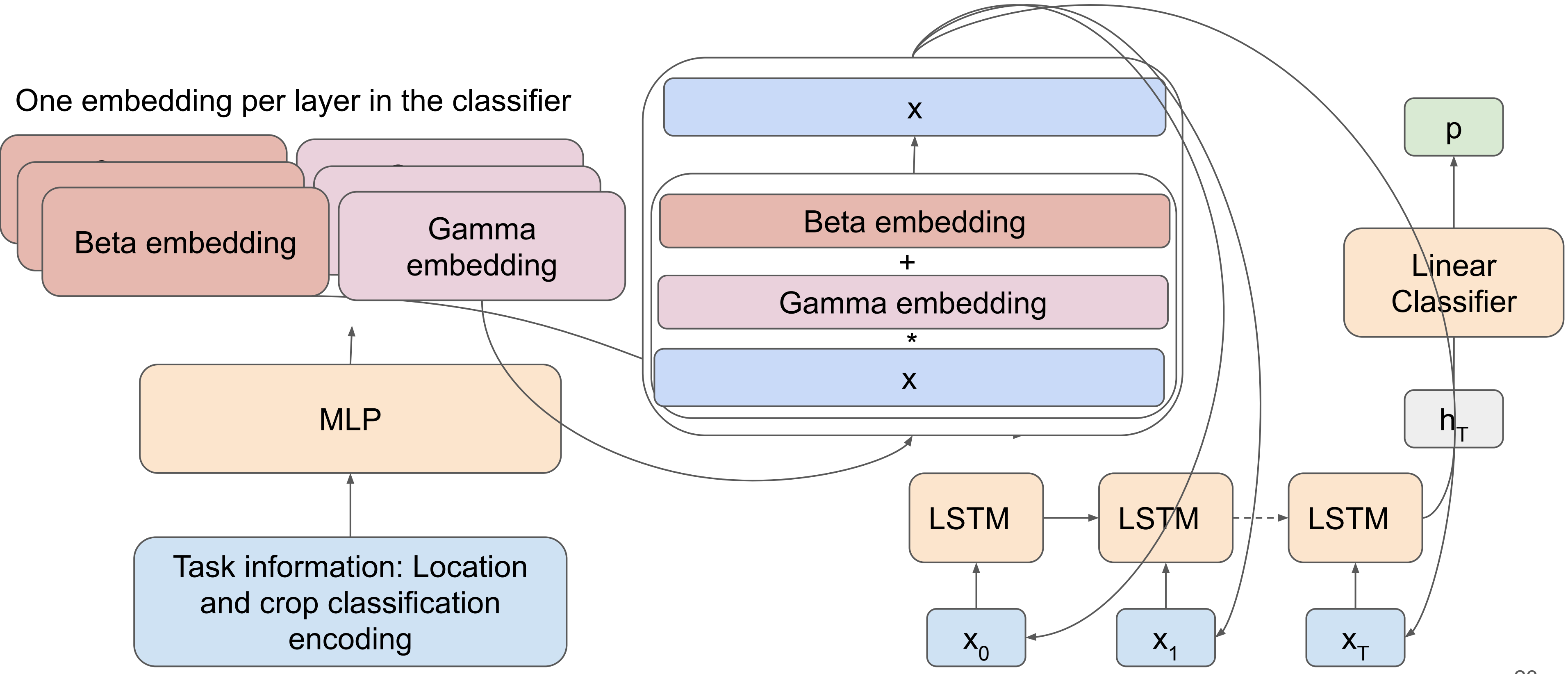


Figure 1. Diagram of our model-agnostic meta-learning algorithm (MAML), which optimizes for a representation θ that can quickly adapt to new tasks.

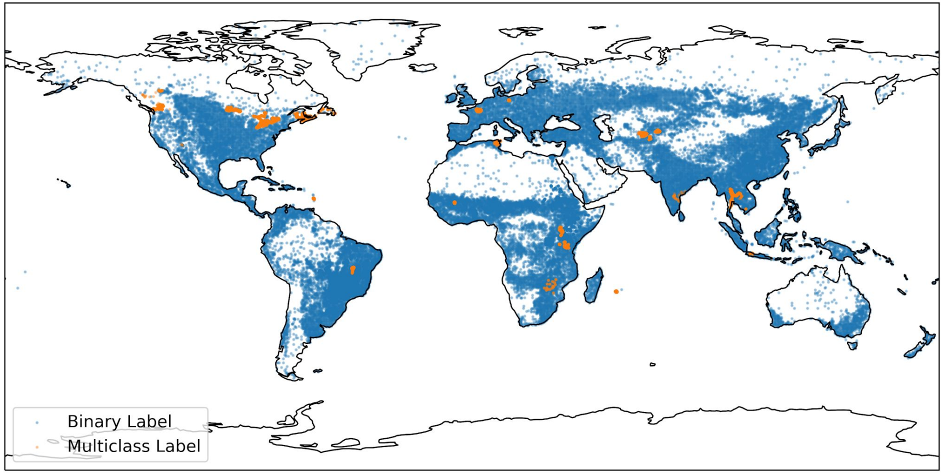
Task informed meta-learning



Task informed meta-learning



Forgetfulness consists of removing tasks the model has memorized



- Crop vs. non crop tasks

All data points have a crop / non crop label. All data points within a country's bounding box are included.

- Land use type vs. rest tasks

Negative examples are constructed from the other crop type labels, and the non-crop labels.

Also includes specific land uses, such as "Cerrado" - natural grassland in Brazil. Since this is non-crop, negative examples are constructed from crop labels.

We benchmark this algorithm against a number of others

- **MAML**: Normal MAML without the task information
- **Pretrained**: Pretrain the model on crop / non crop with all datapoints, and finetune on the test task
- **Random**: Start with random weights, and finetune on the test tasks
- And a **Random Forest**

We benchmark this algorithm against a number of others

- **MAML**: Normal MAML without the task information
- **Pretrained**: Pretrain the model on crop / non crop with all datapoints, and finetune on the test task
- **Random**: Start with random weights, and finetune on the test tasks
- And a **Random Forest**

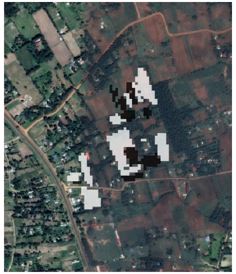
And ablations:

- **No forgetfulness**
- **No encoder**
- **No task information or encoder**

We use 3 test-tasks which evaluate the model in a variety of regimes

Togo crop vs. non crop.

1,319 training samples with a randomly sampled test set of 350 points labelled by 4 labellers



Kenya maize vs. rest

Evaluated using polygons, with 266 positive datapoints and 1075 negative datapoints.

Used 17 positive polygons for testing

Brazil coffee vs. rest

Evaluated using polygons, with 21 positive datapoints and 206 negative points.

Used 18 positive polygons used for testing.

Headline results: TIML performs best across all 3 tasks

	Model	Kenya	Brazil	Togo	Mean
AUC ROC	Random Forest	0.578 ± 0.006	0.941 ± 0.004	0.892 ± 0.001	0.803
	No pre-training	0.329 ± 0.011	0.898 ± 0.010	0.861 ± 0.002	0.700
	Crop pre-training	0.694 ± 0.001	0.820 ± 0.002	**0.894 ± 0.000**	0.801
	MAML	**0.729 ± 0.001**	0.831 ± 0.005	0.878 ± 0.001	0.843
	TIML	**0.794 ± 0.003**	**0.988 ± 0.001**	0.890 ± 0.000	**0.890**
	no forgetfulness	0.779 ± 0.003	0.877 ± 0.003	0.893 ± 0.001	0.850
	no encoder	0.712 ± 0.001	**0.977 ± 0.002**	**0.895 ± 0.000**	**0.862**
	no task info or encoder	0.690 ± 0.001	0.977 ± 0.002	0.876 ± 0.001	0.848
F1 score	Random Forest	0.559 ± 0.003	0.000 ± 0.000	**0.756 ± 0.002**	0.441
	No pre-training	0.782 ± 0.000	**0.764 ± 0.012**	0.720 ± 0.005	**0.734**
	Crop pre-training	0.819 ± 0.001	0.619 ± 0.005	0.713 ± 0.002	0.613
	MAML	0.828 ± 0.001	0.496 ± 0.001	0.662 ± 0.001	0.652
	TIML	**0.838 ± 0.000**	**0.835 ± 0.012**	0.732 ± 0.002	**0.802**
	no forgetfulness	**0.840 ± 0.000**	0.537 ± 0.002	**0.764 ± 0.002**	0.724
	no encoder	**0.840 ± 0.000**	0.473 ± 0.002	0.691 ± 0.001	0.691
	no task info or encoder	0.837 ± 0.001	0.473 ± 0.001	0.645 ± 0.002	0.652

Headline results: TIML performs best across all 3 tasks

	Model	Kenya	Brazil	Togo	Mean
AUC ROC	Random Forest	0.578 ± 0.006	0.941 ± 0.004	0.892 ± 0.001	0.803
	No pre-training	0.329 ± 0.011	0.898 ± 0.010	0.861 ± 0.002	0.700
	Crop pre-training	0.694 ± 0.001	0.820 ± 0.002	$\mathbf{0.894 \pm 0.000}$	0.801
	MAML	$\mathbf{0.729 \pm 0.001}$	0.831 ± 0.005	0.878 ± 0.001	0.843
	TIML	$\mathbf{0.794 \pm 0.003}$	$\mathbf{0.988 \pm 0.001}$	0.890 ± 0.000	**0.890**
	no forgetfulness	0.779 ± 0.003	0.877 ± 0.003	0.893 ± 0.001	0.850
	no encoder	0.712 ± 0.001	$\mathbf{0.977 \pm 0.002}$	$\mathbf{0.895 \pm 0.000}$	**0.862**
	no task info or encoder	0.690 ± 0.001	0.977 ± 0.002	0.876 ± 0.001	0.848
F1 score	Random Forest	0.559 ± 0.003	0.000 ± 0.000	$\mathbf{0.756 \pm 0.002}$	0.441
	No pre-training	0.782 ± 0.000	$\mathbf{0.764 \pm 0.012}$	0.720 ± 0.005	**0.734**
	Crop pre-training	0.819 ± 0.001	0.619 ± 0.005	0.713 ± 0.002	0.613
	MAML	0.828 ± 0.001	0.496 ± 0.001	0.662 ± 0.001	0.652
	TIML	$\mathbf{0.838 \pm 0.000}$	$\mathbf{0.835 \pm 0.012}$	0.732 ± 0.002	**0.802**
	no forgetfulness	$\mathbf{0.840 \pm 0.000}$	0.537 ± 0.002	$\mathbf{0.764 \pm 0.002}$	0.724
	no encoder	$\mathbf{0.840 \pm 0.000}$	0.473 ± 0.002	0.691 ± 0.001	0.691
	no task info or encoder	0.837 ± 0.001	0.473 ± 0.001	0.645 ± 0.002	0.652

Transfer learning isn't guaranteed to help!

	Model	Kenya	Brazil	Togo	Mean
AUC ROC	Random Forest	0.578 ± 0.006	0.941 ± 0.004	0.892 ± 0.001	0.803
	No pre-training	0.329 ± 0.011	0.898 ± 0.010	0.861 ± 0.002	0.700
	Crop pre-training	0.694 ± 0.001	0.820 ± 0.002	$\mathbf{0.894 \pm 0.000}$	0.801
	MAML	$\mathbf{0.729 \pm 0.001}$	0.831 ± 0.005	0.878 ± 0.001	0.843
	TIML	$\mathbf{0.794 \pm 0.003}$	$\mathbf{0.988 \pm 0.001}$	0.890 ± 0.000	**0.890**
	no forgetfulness	0.779 ± 0.003	0.877 ± 0.003	0.893 ± 0.001	0.850
	no encoder	0.712 ± 0.001	$\mathbf{0.977 \pm 0.002}$	$\mathbf{0.895 \pm 0.000}$	**0.862**
	no task info or encoder	0.690 ± 0.001	0.977 ± 0.002	0.876 ± 0.001	0.848
F1 score	Random Forest	0.559 ± 0.003	0.000 ± 0.000	$\mathbf{0.756 \pm 0.002}$	0.441
	No pre-training	0.782 ± 0.000	$\mathbf{0.764 \pm 0.012}$	0.720 ± 0.005	**0.734**
	Crop pre-training	0.819 ± 0.001	0.619 ± 0.005	0.713 ± 0.002	0.613
	MAML	0.828 ± 0.001	0.496 ± 0.001	0.662 ± 0.001	0.652
	TIML	$\mathbf{0.838 \pm 0.000}$	$\mathbf{0.835 \pm 0.012}$	0.732 ± 0.002	**0.802**
	no forgetfulness	$\mathbf{0.840 \pm 0.000}$	0.537 ± 0.002	$\mathbf{0.764 \pm 0.002}$	0.724
	no encoder	$\mathbf{0.840 \pm 0.000}$	0.473 ± 0.002	0.691 ± 0.001	0.691
	no task info or encoder	0.837 ± 0.001	0.473 ± 0.001	0.645 ± 0.002	0.652

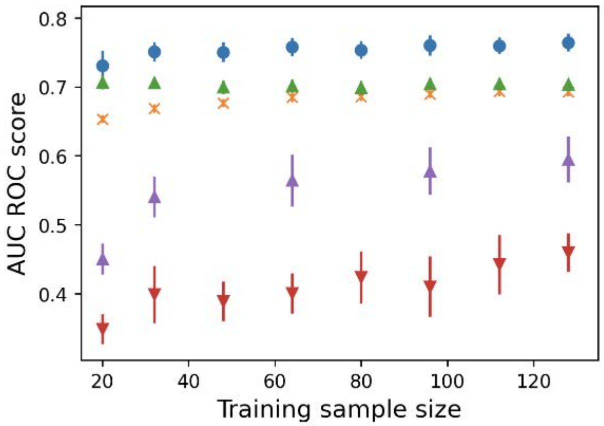
Transfer learning isn't guaranteed to help!

	Model	Kenya	Brazil	Togo	Mean
AUC ROC	Random Forest	0.578 ± 0.006	0.941 ± 0.004	0.892 ± 0.001	0.803
	No pre-training	0.329 ± 0.011	0.898 ± 0.010	0.861 ± 0.002	0.700
	Crop pre-training	0.694 ± 0.001	0.820 ± 0.002	**0.894 ± 0.000**	0.801
	MAML	**0.729 ± 0.001**	0.831 ± 0.005	0.878 ± 0.001	0.843
	TIML	**0.794 ± 0.003**	**0.988 ± 0.001**	0.890 ± 0.000	**0.890**
	no forgetfulness	0.779 ± 0.003	0.877 ± 0.003	0.893 ± 0.001	0.850
	no encoder	0.712 ± 0.001	**0.977 ± 0.002**	**0.895 ± 0.000**	**0.862**
	no task info or encoder	0.690 ± 0.001	0.977 ± 0.002	0.876 ± 0.001	0.848
F1 score	Random Forest	0.559 ± 0.003	0.000 ± 0.000	**0.756 ± 0.002**	0.441
	No pre-training	0.782 ± 0.000	**0.764 ± 0.012**	0.720 ± 0.005	**0.734**
	Crop pre-training	0.819 ± 0.001	0.619 ± 0.005	0.713 ± 0.002	0.613
	MAML	0.828 ± 0.001	0.496 ± 0.001	0.662 ± 0.001	0.652
	TIML	**0.838 ± 0.000**	**0.835 ± 0.012**	0.732 ± 0.002	**0.802**
	no forgetfulness	**0.840 ± 0.000**	0.537 ± 0.002	**0.764 ± 0.002**	0.724
	no encoder	**0.840 ± 0.000**	0.473 ± 0.002	0.691 ± 0.001	0.691
	no task info or encoder	0.837 ± 0.001	0.473 ± 0.001	0.645 ± 0.002	0.652

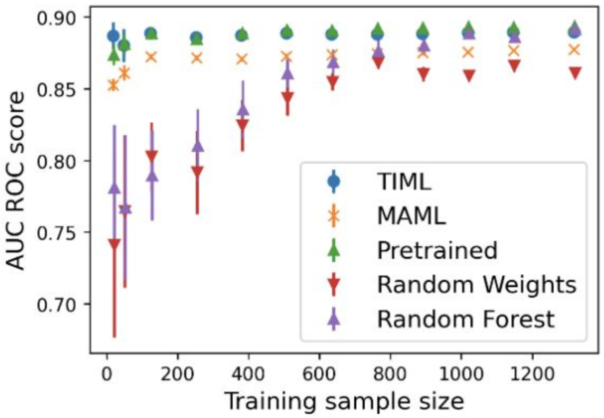
Forgetfulness especially helps in Brazil

	Model	Kenya	Brazil	Togo	Mean
AUC ROC	Random Forest	0.578 ± 0.006	0.941 ± 0.004	0.892 ± 0.001	0.803
	No pre-training	0.329 ± 0.011	0.898 ± 0.010	0.861 ± 0.002	0.700
	Crop pre-training	0.694 ± 0.001	0.820 ± 0.002	**0.894 ± 0.000**	0.801
	MAML	**0.729 ± 0.001**	0.831 ± 0.005	0.878 ± 0.001	0.843
	TIML	**0.794 ± 0.003**	**0.988 ± 0.001**	0.890 ± 0.000	**0.890**
	no forgetfulness	0.779 ± 0.003	0.877 ± 0.003	0.893 ± 0.001	0.850
	no encoder	0.712 ± 0.001	**0.977 ± 0.002**	**0.895 ± 0.000**	**0.862**
	no task info or encoder	0.690 ± 0.001	0.977 ± 0.002	0.876 ± 0.001	0.848
F1 score	Random Forest	0.559 ± 0.003	0.000 ± 0.000	**0.756 ± 0.002**	0.441
	No pre-training	0.782 ± 0.000	**0.764 ± 0.012**	0.720 ± 0.005	**0.734**
	Crop pre-training	0.819 ± 0.001	0.619 ± 0.005	0.713 ± 0.002	0.613
	MAML	0.828 ± 0.001	0.496 ± 0.001	0.662 ± 0.001	0.652
	TIML	**0.838 ± 0.000**	$\boxed{\mathbf{0.835 \pm 0.012}}$	0.732 ± 0.002	**0.802**
	no forgetfulness	**0.840 ± 0.000**	$\boxed{0.537 \pm 0.002}$	**0.764 ± 0.002**	0.724
	no encoder	**0.840 ± 0.000**	0.473 ± 0.002	0.691 ± 0.001	0.691
	no task info or encoder	0.837 ± 0.001	0.473 ± 0.001	0.645 ± 0.002	0.652

TIML improves model performance across a range of finetuning dataset sizes

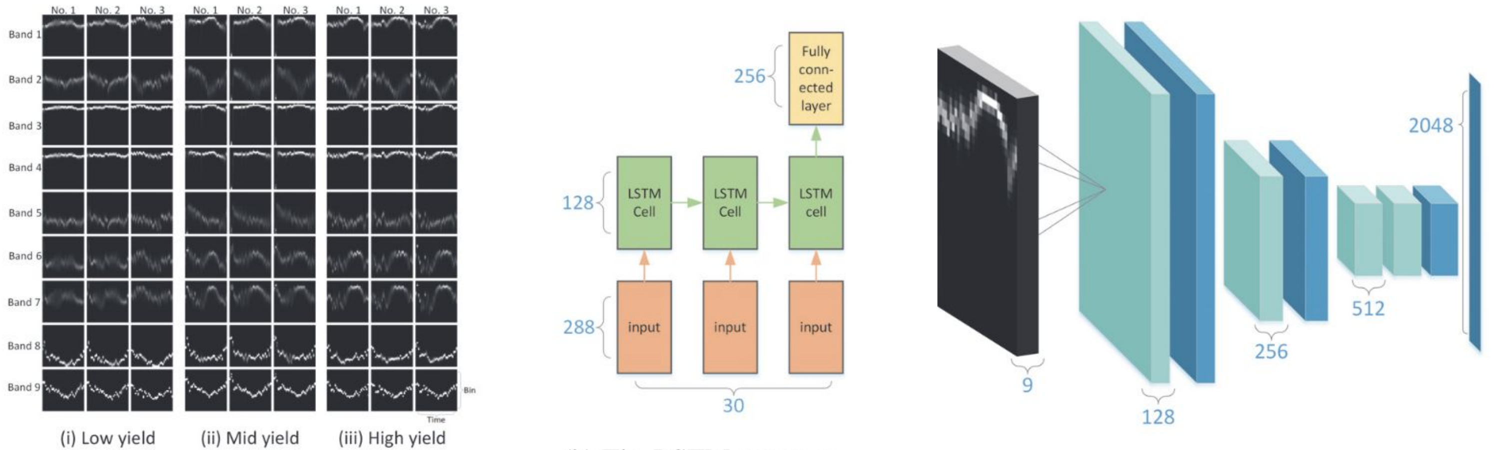


(a) Kenya: Maize vs. Rest



(b) Togo: Crop vs. Non Crop

TIML applies elsewhere too: yield estimation (regression)



(a) 3-D histogram visualization

(b) The LSTM structure

(c) The CNN structure

Figure 1: Visualization of the input data and used architectures. **Left:** Figures of typical 3-D histograms $\mathcal{H} \in \mathbb{R}^{b \times T \times d}$ flattened in the band dimension d under (i) low crop yield, (ii) mid crop yield, and (iii) high crop yield conditions are shown in the left panel. Each row corresponds to a different spectral band, while each column represents an individual data point. Each square is a slice of $\mathcal{H}$, where the x-axis corresponds to the "time" dimension T, and the y-axis to the "bin" dimension b. Brighter pixels indicate higher pixel counts in that bin. There exists distinctive visual differences between high yield and low yield conditions (for example in the second and the seventh bands). **Mid:** The adopted LSTM structure. **Right:** The adopted CNN structure, where stride-1 convolutional layers are in light blue, stride-2 convolutional layers are in dark blue and a fully connected layer is attached at the end.

You, J., Li, X., Low, M., Lobell, D., and Ermon, S. Deep Gaussian process for crop yield prediction based on remote sensing data. Proceedings of the AAAI Conference on Artificial Intelligence, 2017

Overall, TIML performs well in this regime too

Model	2011	2012	2013	2014	2015	Mean
LSTM	5.62 ± 0.10	6.60 ± 0.29	5.57 ± 0.21	6.63 ± 0.13	6.69 ± 0.31	6.22
+ GP	5.32 ± 0.10	$\mathbf{5.83 \pm 0.18}$	5.70 ± 0.19	5.61 ± 0.12	$\mathbf{5.24 \pm 0.14}$	**5.54**
+ MAML	26.90 ± 0.01	30.97 ± 0.01	29.57 ± 0.01	30.84 ± 0.01	32.02 ± 0.01	30.06
+ TIML	$\mathbf{5.16 \pm 0.03}$	$\mathbf{5.77 \pm 0.05}$	$\mathbf{5.39 \pm 0.02}$	5.24 ± 0.04	$\mathbf{4.89 \pm 0.04}$	**5.29**
CNN	6.08 ± 0.77	6.94 ± 1.83	6.42 ± 1.23	$\mathbf{4.80 \pm 0.83}$	5.57 ± 0.38	5.96
+ GP	5.55 ± 0.14	6.18 ± 0.49	6.44 ± 0.67	$\mathbf{4.87 \pm 0.31}$	6.02 ± 0.26	5.81
+ MAML	12.93 ± 0.05	8.28 ± 0.07	7.98 ± 0.04	12.05 ± 0.05	7.69 ± 0.06	9.79
+ TIML	$\mathbf{5.23 \pm 0.02}$	6.59 ± 0.02	$\mathbf{5.34 \pm 0.01}$	4.93 ± 0.02	6.35 ± 0.01	5.69
(You et al., 2017)						
LSTM + GP	5.77	6.23	5.96	5.70	5.49	5.83
CNN + GP	5.70	5.68	5.83	4.89	5.67	5.55

We rerun the GP models, since the MODIS data has incremented versions

Model	2011	2012	2013	2014	2015	Mean
LSTM	5.62 ± 0.10	6.60 ± 0.29	5.57 ± 0.21	6.63 ± 0.13	6.69 ± 0.31	6.22
+ GP	5.32 ± 0.10	$\mathbf{5.83 \pm 0.18}$	5.70 ± 0.19	5.61 ± 0.12	$\mathbf{5.24 \pm 0.14}$	**5.54**
+ MAML	26.90 ± 0.01	30.97 ± 0.01	29.57 ± 0.01	30.84 ± 0.01	32.02 ± 0.01	30.06
+ TIML	$\mathbf{5.16 \pm 0.03}$	$\mathbf{5.77 \pm 0.05}$	$\mathbf{5.39 \pm 0.02}$	5.24 ± 0.04	$\mathbf{4.89 \pm 0.04}$	**5.29**
CNN	6.08 ± 0.77	6.94 ± 1.83	6.42 ± 1.23	$\mathbf{4.80 \pm 0.83}$	5.57 ± 0.38	5.96
+ GP	5.55 ± 0.14	6.18 ± 0.49	6.44 ± 0.67	$\mathbf{4.87 \pm 0.31}$	6.02 ± 0.26	5.81
+ MAML	12.93 ± 0.05	8.28 ± 0.07	7.98 ± 0.04	12.05 ± 0.05	7.69 ± 0.06	9.79
+ TIML	$\mathbf{5.23 \pm 0.02}$	6.59 ± 0.02	$\mathbf{5.34 \pm 0.01}$	4.93 ± 0.02	6.35 ± 0.01	5.69
(You et al., 2017)						
LSTM + GP	5.77	6.23	5.96	5.70	5.49	5.83
CNN + GP	5.70	5.68	5.83	4.89	5.67	5.55

TIML is the best performing algorithm for each architecture

Model	2011	2012	2013	2014	2015	Mean
LSTM	5.62 ± 0.10	6.60 ± 0.29	5.57 ± 0.21	6.63 ± 0.13	6.69 ± 0.31	6.22
+ GP	5.32 ± 0.10	$\mathbf{5.83 \pm 0.18}$	5.70 ± 0.19	5.61 ± 0.12	$\mathbf{5.24 \pm 0.14}$	**5.54**
+ MAML	26.90 ± 0.01	30.97 ± 0.01	29.57 ± 0.01	30.84 ± 0.01	32.02 ± 0.01	30.06
+ TIML	$\mathbf{5.16 \pm 0.03}$	$\mathbf{5.77 \pm 0.05}$	$\mathbf{5.39 \pm 0.02}$	5.24 ± 0.04	$\mathbf{4.89 \pm 0.04}$	**5.29**
CNN	6.08 ± 0.77	6.94 ± 1.83	6.42 ± 1.23	$\mathbf{4.80 \pm 0.83}$	5.57 ± 0.38	5.96
+ GP	5.55 ± 0.14	6.18 ± 0.49	6.44 ± 0.67	$\mathbf{4.87 \pm 0.31}$	6.02 ± 0.26	5.81
+ MAML	12.93 ± 0.05	8.28 ± 0.07	7.98 ± 0.04	12.05 ± 0.05	7.69 ± 0.06	9.79
+ TIML	$\mathbf{5.23 \pm 0.02}$	6.59 ± 0.02	$\mathbf{5.34 \pm 0.01}$	4.93 ± 0.02	6.35 ± 0.01	5.69
(You et al., 2017)						
LSTM + GP	5.77	6.23	5.96	5.70	5.49	5.83
CNN + GP	5.70	5.68	5.83	4.89	5.67	5.55

TIML does well even when MAML performs poorly

Model	2011	2012	2013	2014	2015	Mean
LSTM	5.62 ± 0.10	6.60 ± 0.29	5.57 ± 0.21	6.63 ± 0.13	6.69 ± 0.31	6.22
+ GP	5.32 ± 0.10	$\mathbf{5.83 \pm 0.18}$	5.70 ± 0.19	5.61 ± 0.12	$\mathbf{5.24 \pm 0.14}$	**5.54**
+ MAML	26.90 ± 0.01	30.97 ± 0.01	29.57 ± 0.01	30.84 ± 0.01	32.02 ± 0.01	30.06
+ TIML	$\mathbf{5.16 \pm 0.03}$	$\mathbf{5.77 \pm 0.05}$	$\mathbf{5.39 \pm 0.02}$	5.24 ± 0.04	$\mathbf{4.89 \pm 0.04}$	**5.29**
CNN	6.08 ± 0.77	6.94 ± 1.83	6.42 ± 1.23	$\mathbf{4.80 \pm 0.83}$	5.57 ± 0.38	5.96
+ GP	5.55 ± 0.14	6.18 ± 0.49	6.44 ± 0.67	$\mathbf{4.87 \pm 0.31}$	6.02 ± 0.26	5.81
+ MAML	12.93 ± 0.05	8.28 ± 0.07	7.98 ± 0.04	12.05 ± 0.05	7.69 ± 0.06	9.79
+ TIML	$\mathbf{5.23 \pm 0.02}$	6.59 ± 0.02	$\mathbf{5.34 \pm 0.01}$	4.93 ± 0.02	6.35 ± 0.01	5.69
(You et al., 2017)						
LSTM + GP	5.77	6.23	5.96	5.70	5.49	5.83
CNN + GP	5.70	5.68	5.83	4.89	5.67	5.55

In conclusion:

- We introduce Task-Informed Meta-Learning, an algorithm which:
    - **Encodes task information** to inform meta-learning algorithms
    - Uses **forgetfulness** to boost performance in rare tasks

- We introduce CropHarvest, an aggregate dataset of agricultural land cover coupled with remote sensing data

- We demonstrate the applicability of TIML on **crop classification** and **yield estimation**, on a variety of model architectures

Code and models available on github

https://github.com/nasaharvest/timl

https://github.com/nasaharvest/cropharvest