Unlocking Data Insights - Introduction to Data-Centric AI

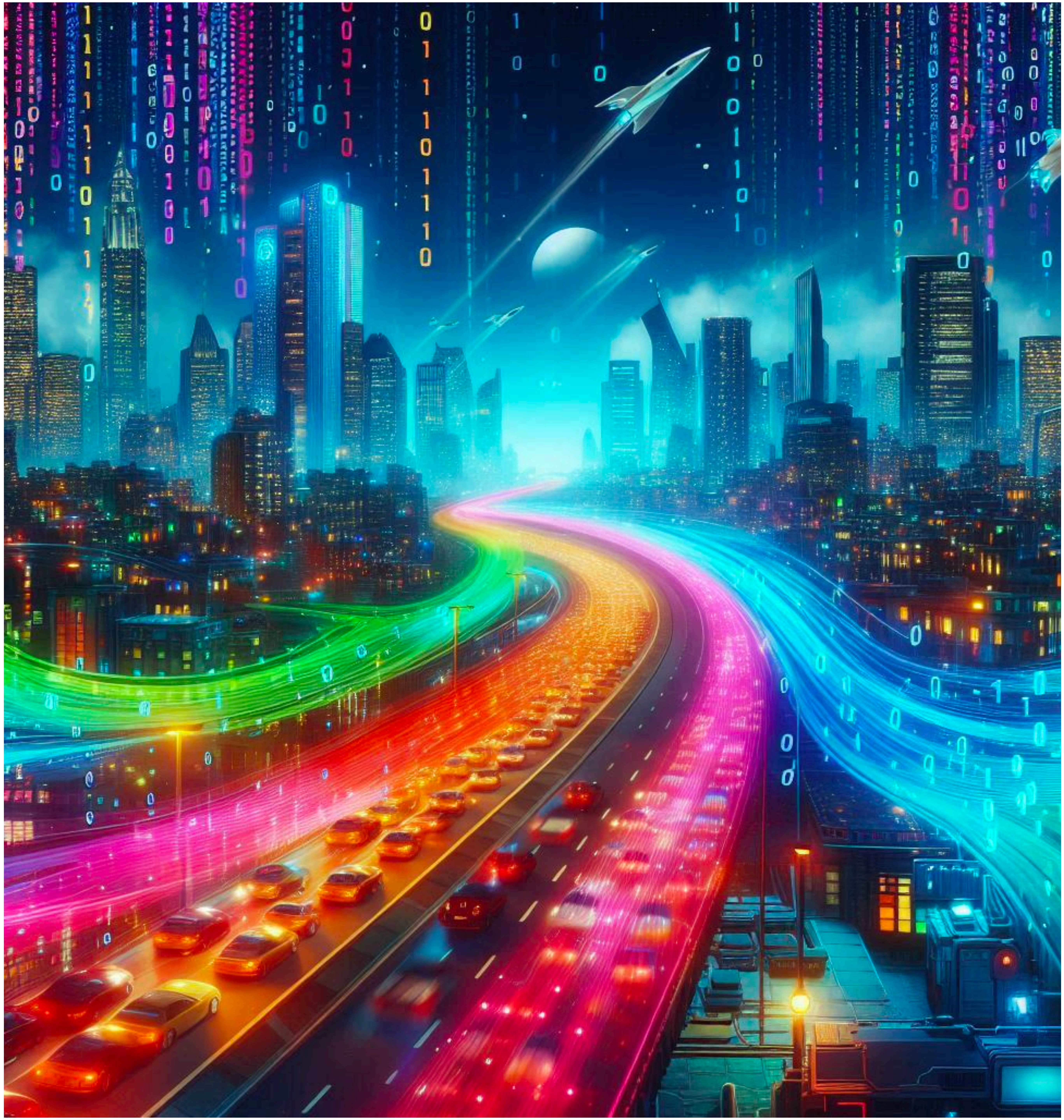
Learning from data streams: Use cases

refererce: Deep Learning with PyTorch and Lightning - Chapter 8

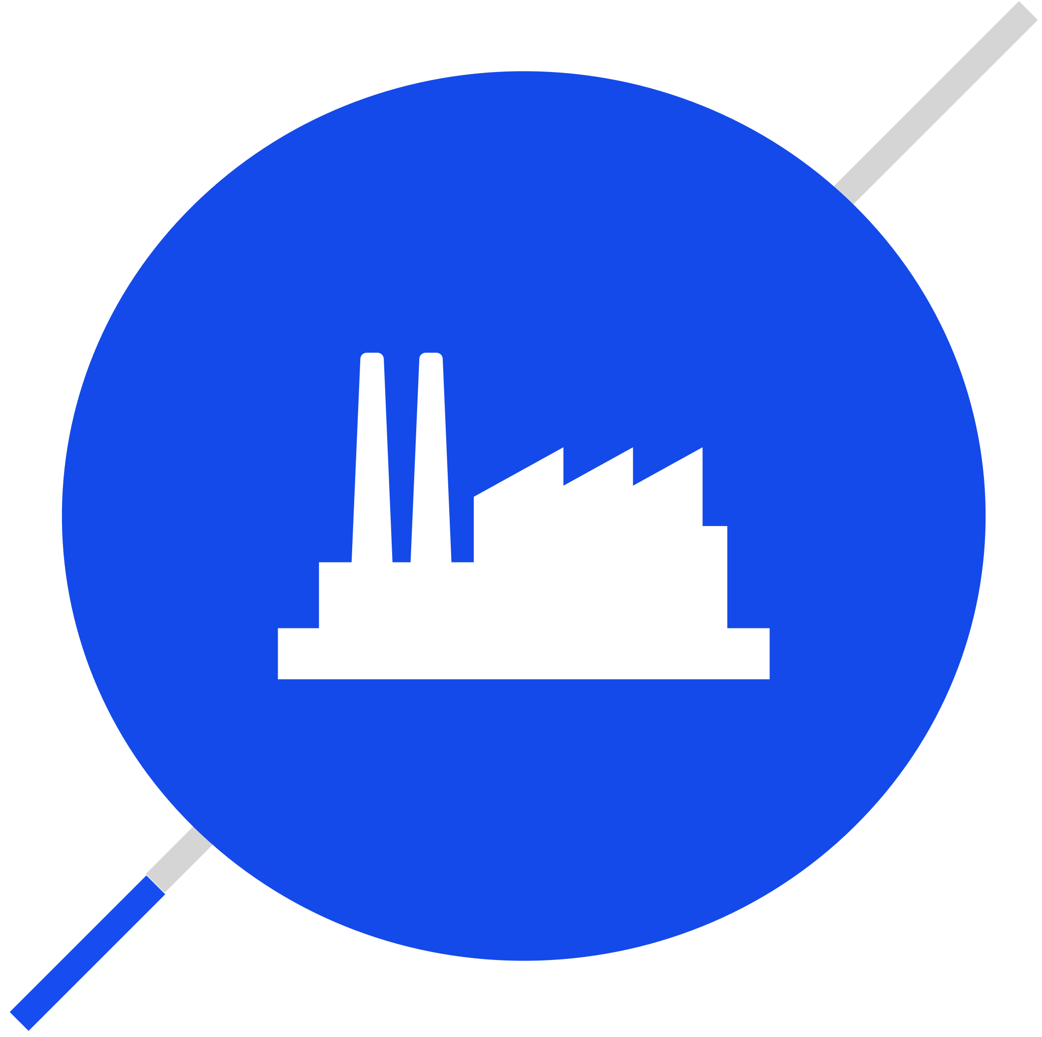
Learning from data streams: Use cases

Executive Summary

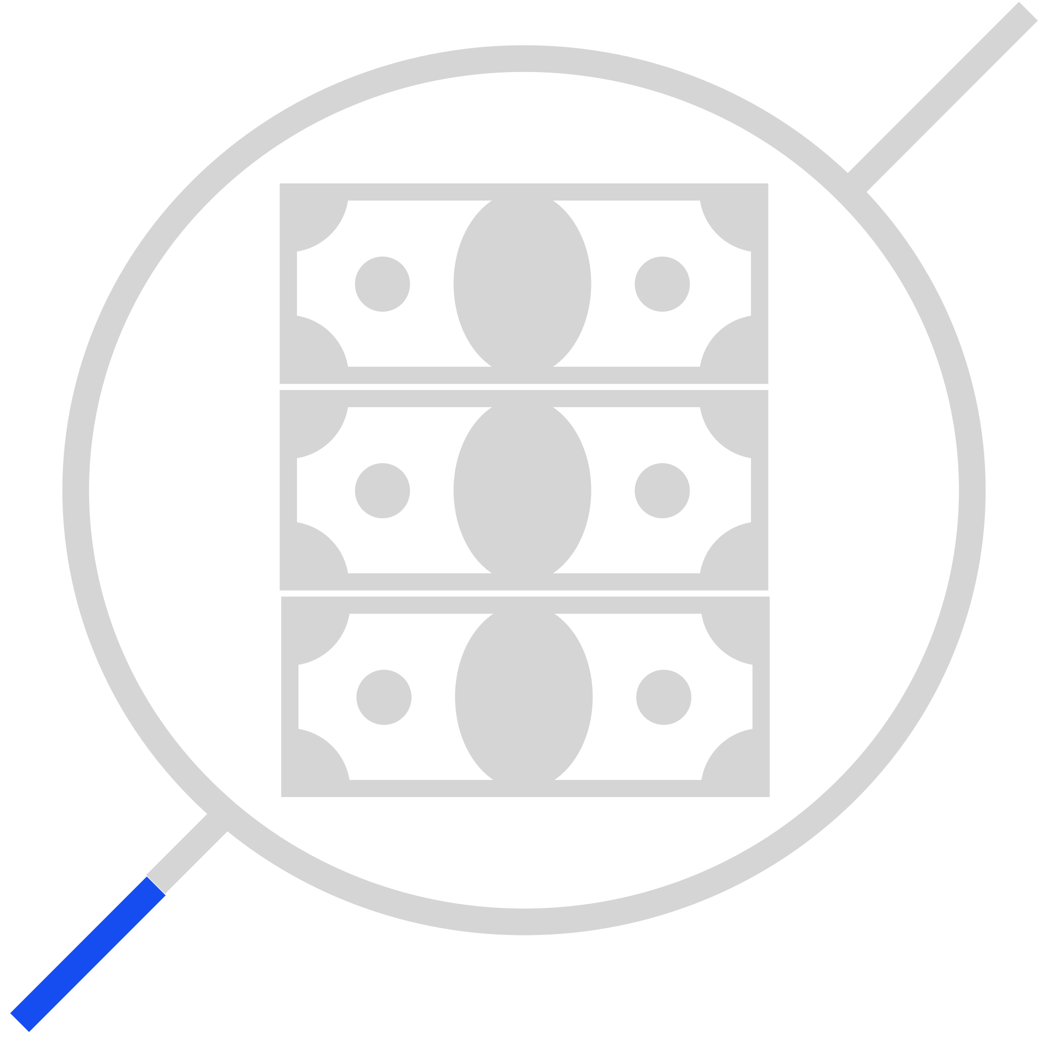
- Chunk online learning - **Retail sector**
- Stream online learning - **Business Process Management**



What is the problem?



E-commerce, retail
etc.



New customer
acquisition

Preventing
customer churns

Churn prediction: SOTA

Machine Learning
(Random Forest, XGBoost,etc.)

RFM-based approach

Neural Network
(LSTM, CNN, etc.)

Several studies in different domains

Telecommunication

Online Gaming

Banking&Insurance

Multimedia

Churn prediction: SOTA

Machine Learning
(Random Forest, XGBoost,etc.)

RFM-based approach

Neural Network
(LSTM, CNN, etc.)

But, a few studies have explored the problem in

Retail

Data-Centric AI Challenges

1. Smart data from raw data
2. Distribution shift
3. Model adaptation
4. Customer profile explanation

1. Smart data from raw data

1.1 Purchase Basket Item

,3, €5.00

1.2 Purchase Basket

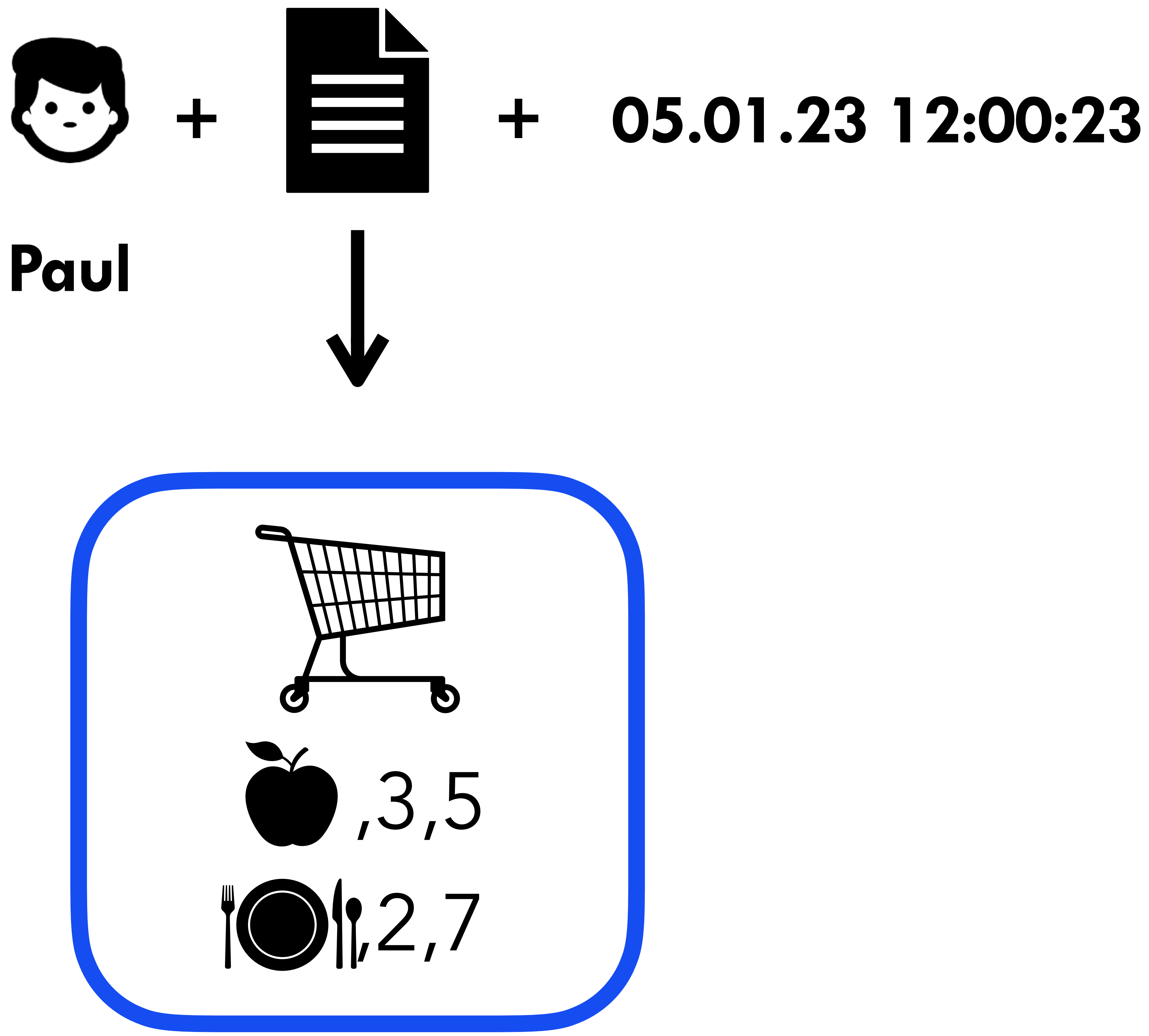
,1, €2.00

,1, €7.00

,4, €7.00

1. Smart data from raw data

1.3 Sale Receipt



1. Smart data from raw data

1.4 Sale Receipt Stream

Customer	Timestamp	Purchase basket	
Paul	05.01.23 12:00:23	,3,5	,2,7
Mary	05.01.23 12:01:23	,3,5	,1,4
Elliot	05.01.23 20:00:01	,3,7	
Paul	05.02.23 08:00:25	,5,6	
Mary	05.03.23 09:00:25	,3,2	,2,6
Paul	05.03.23 19:01:05	,4,7	,1,8
Paul	05.03.23 20:01:05	,2,7	
Paul	05.03.23 20:01:05	,5,1	
Mary	05.04.23 21:01:05	,2,1	
Mary	05.06.23 11:21:00	,1,1	,3,4
Elliot	05.07.23 19:00:05	,5,1	
Elliot	05.07.23 20:01:05	,5,1	,2,4

1. Smart data from raw data

1.5 Customer Trace

Paul

Timestamp	Purchase basket	
05.01.23 12:00:23	,3,5	,2,7
05.02.23 08:00:25	,5,6	
05.03.23 19:01:05	,4,7	,1,8
05.03.23 20:01:05	,2,7	
05.03.23 20:01:05	,5,1	

1. Smart data from raw data

1.6 RFM Model

Paul

Timestamp	**Purchase basket**	
05.01.23 12:00:23	,3,5	,2,7
05.02.23 08:00:25	,5,6	
05.03.23 19:01:05	,4,7	,1,8
05.03.23 20:01:05	,2,7	
05.03.23 20:01:05	,5,1	

CURRENT TIME = 05.04.23 24:00:00

0 **Recency**

5 **Frequency**

114 **Monetary**

1. Smart data from raw data

First scenario - last receipt

1.7 Churn Status

Churn alert
2 days

Paul
05.04.23 20:01:05
Last receipt

Paul
05.06.24 24:00:00
Churn

The churn status of a customer is evaluated at midnight of each day

1. Smart data from raw data

Second scenario - temporary churn status

1.7 Churn Status

Churn alert
2 days

Eliot
05.01.23 20:00:01
Last receipt

Eliot
05.03.23 24:00:00
Churn

Churn/No Churn?
Eliot
05.07.23 19:00:05

The churn status of a customer is evaluated at midnight of each day

1. Smart data from raw data

Customer Trace			Timestamp	Label
Paul	05.01.23 12:00:23	,3,5 ,2,7	05.01.23 24:00:00	Non-churn
Mary	05.01.23 12:01:23	,3,5 ,1,4	05.01.23 24:00:00	Non-churn
Mary	05.01.23 12:01:23	,3,5 ,1,4	05.02.23 24:00:00	Non-churn
Elliot	05.01.23 20:00:01	,3,7	05.01.23 24:00:00	Churn
Elliot	05.01.23 20:00:01	,3,7	05.02.23 24:00:00	Churn
Paul	05.01.23 12:00:23 05.01.23 12:00:23	,3,5 ,2,7 ,3,5	05.02.23 24:00:00	Non-churn
...			...	...

2. Distribution shift

Handling drifting data

ADWIN to monitor the performance of a churn predictive model along a Sale Receipt Stream

Dataset description

	UK retail	Brazilian
#customers	5853	2913
#products	4619	33041
#sale receipts	36597	6159
#basket items	776637	7483
#daily sale receipts (avg ± stdev)	60.59±23.64	10.21±6.01
#basket items per sale receipt (avg ± stdev)	21.22±22.97	1.21±0.69
time between sale receipts of a customer (avg ± stdev)	115.96±108.28	88.55±110.31

2. Distribution shift

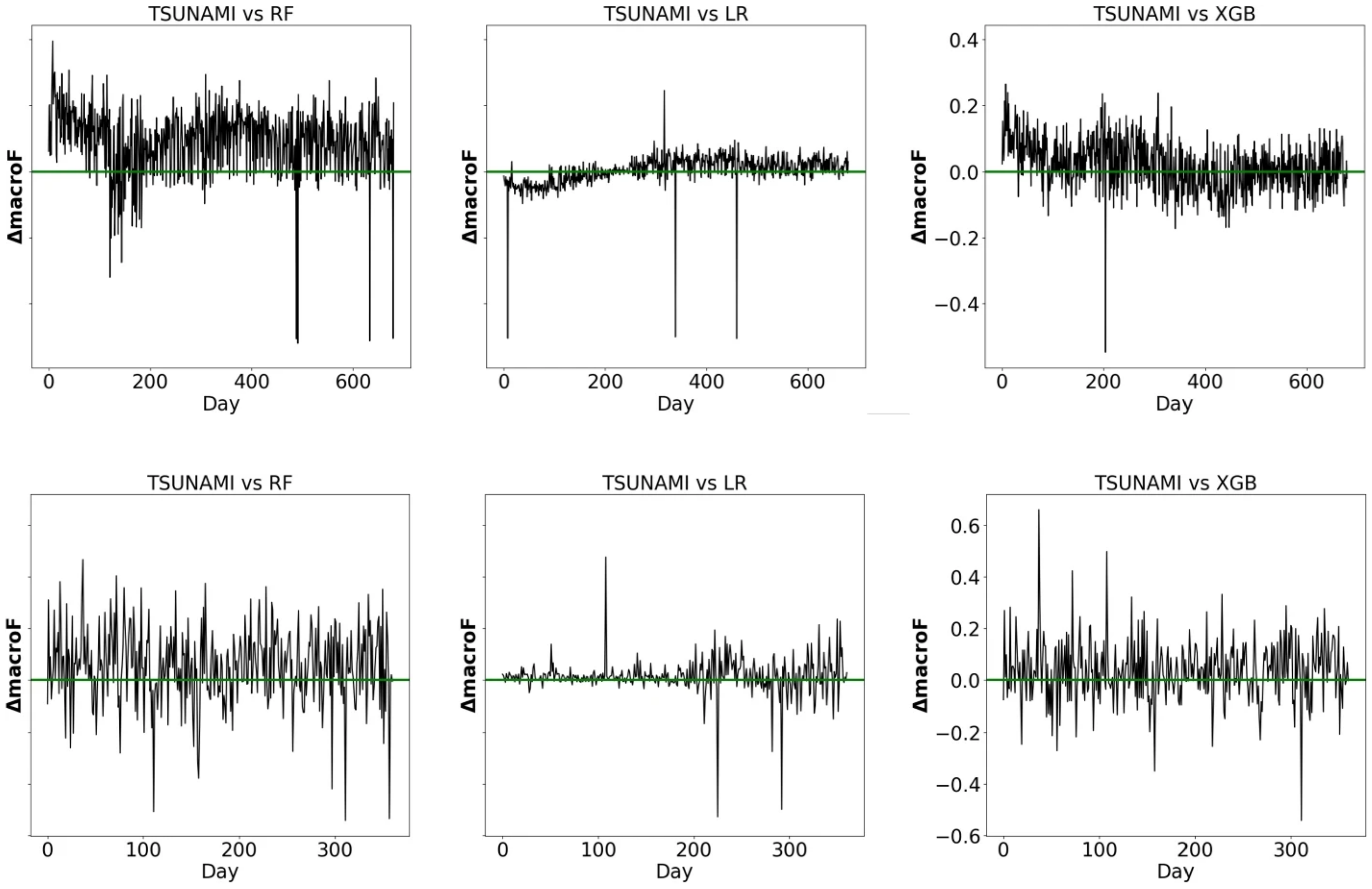
TSUNAMI vs Baseline

Dataset	Conf.	drift	F non-churn	F churn	macroF
UK retail	TSUNAMI	4	**0.69**	0.67	**0.68**
	RF	1	0.65	0.49	0.57
	LR	1	**0.69**	0.64	0.66
	XGB	0	0.62	**0.72**	0.67
Brazilian retail	TSUNAMI	2	**0.54**	**0.89**	**0.71**
	RF	3	0.41	0.89	0.65
	LR	1	0.51	0.87	0.69
	XGB	1	0.45	0.88	0.66

3. Model Adaptation

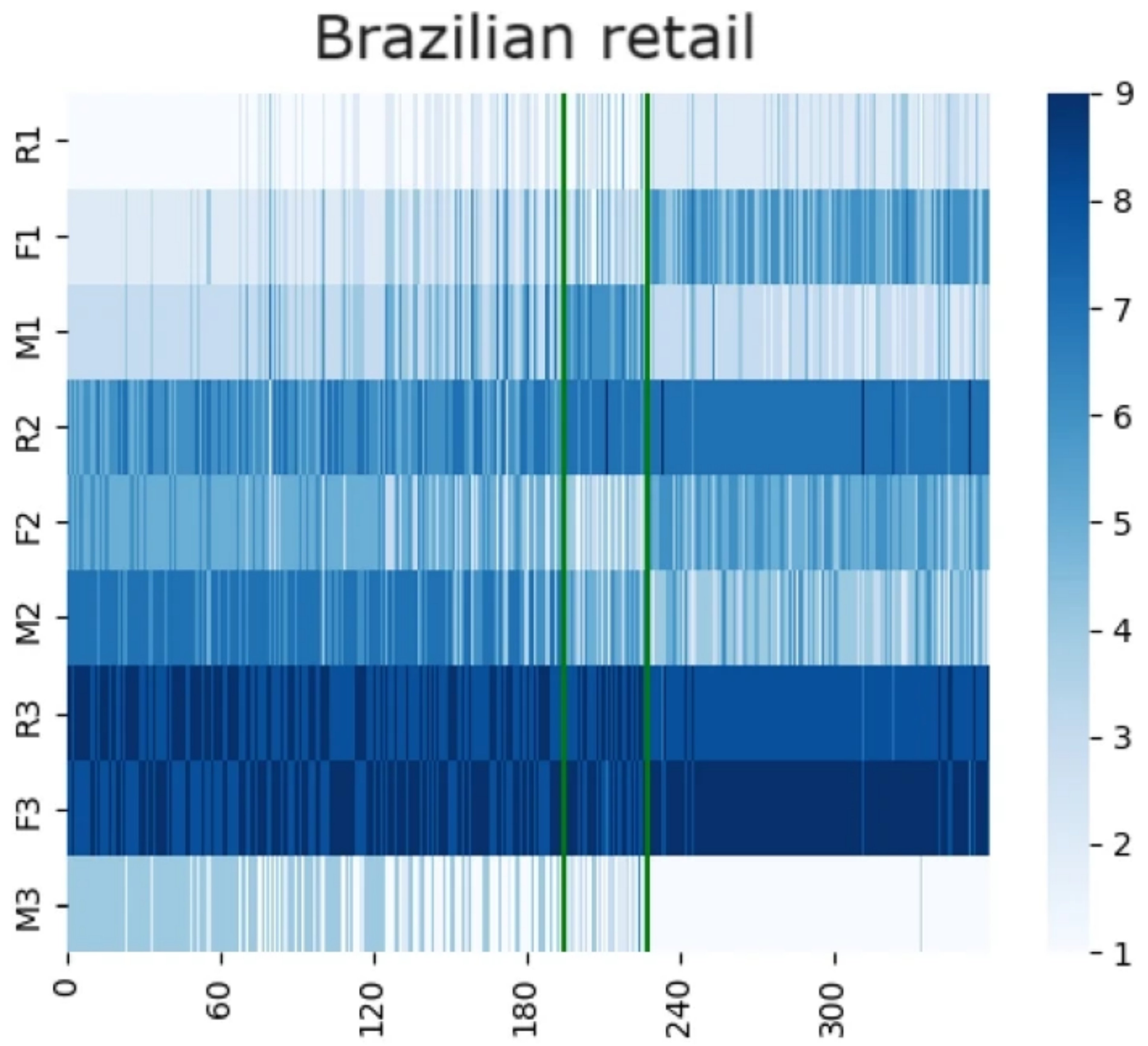
Keeping the model accurate

Fine-tuning to update the deep neural predictive model



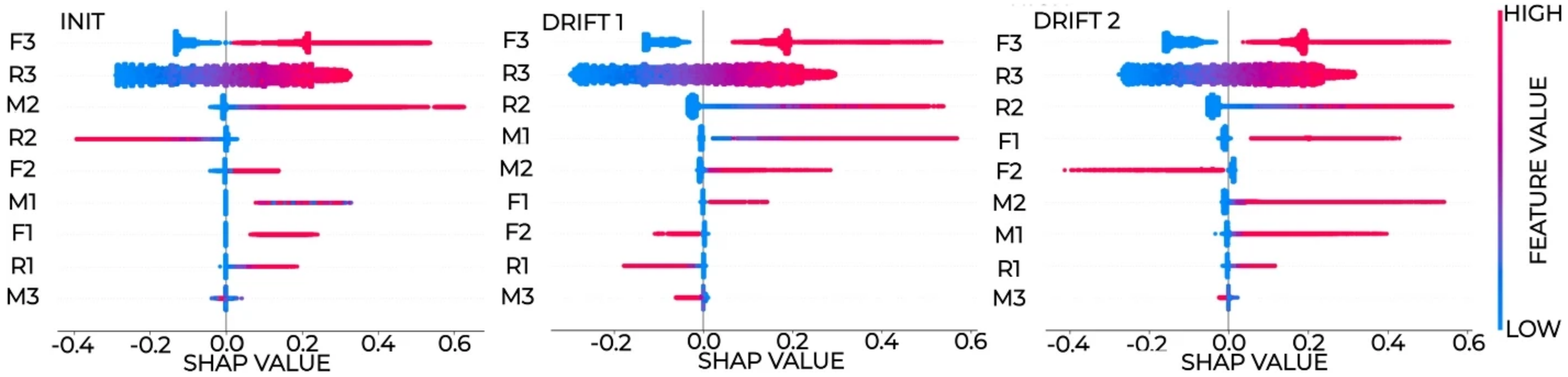
ΔmacroF (axis Y) of TSUNAMI vs RF, LR and XGB measured on customer traces labeled daily during the online stage (axis X)

4. Customer profile explanation



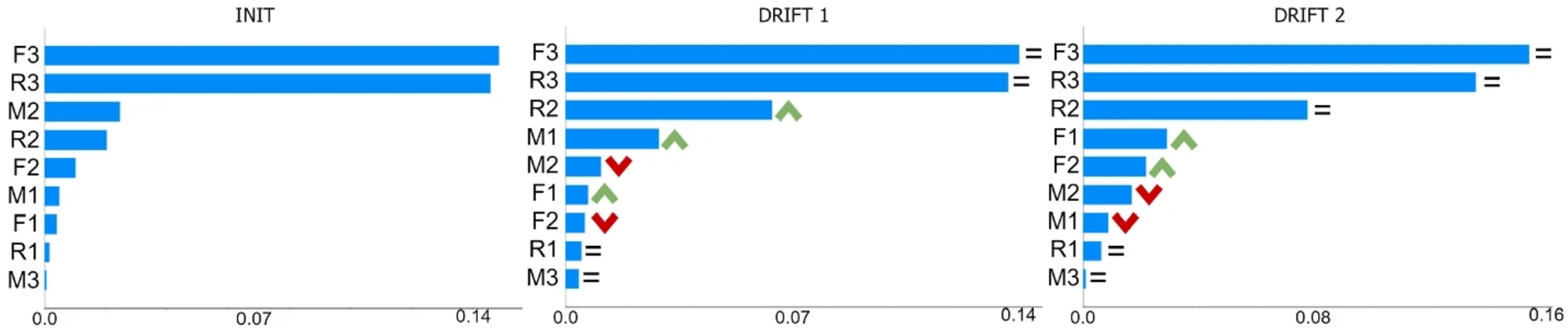
The heat-map of the daily ranking of the importance of RFM features in Brazilian retail

4. Customer profile explanation

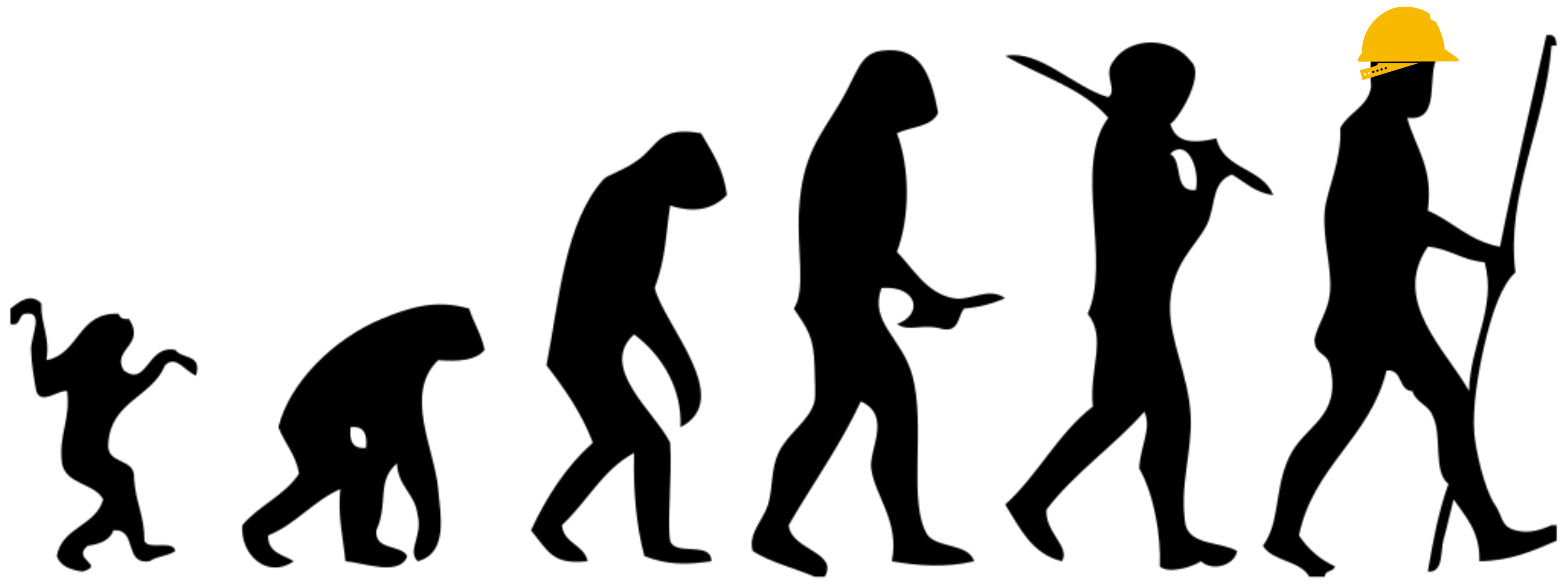


Brazilian retail: Shapley values (axis X) of input features (axis Y) computed for decisions produced with the online churn classification model learned by TSUNAMI. Decisions explanations are plotted with respect to the feature value and grouped with respect to the model used to produce decisions

4. Customer profile explanation



Brazilian retail: global feature importance plot grouped with respect to the churn classification model used to produce decisions



DARWIN

DARWIN: An online deep learning approach to handle concept drifts in predictive process monitoring Vincenzo Pasquadibisceglie, Annalisa Appice, Giovanna Castellano, Donato Malerba, Engineering Applications of Artificial Intelligence,Volume 123, Part C, 2023

What’s problem

Predictive Process Monitoring (PPM)* concerns a set of techniques developed in the area of process mining, in order to predict the outcome of a business process based on historical raw data

**C. Di Francescomarino, C. Ghidini, Predictive process monitoring, Process Mining Handbook. LNBIP 448 (2022) 320–346*

Type of prediction

e.g. What is the outcome of this trace?

Type of prediction

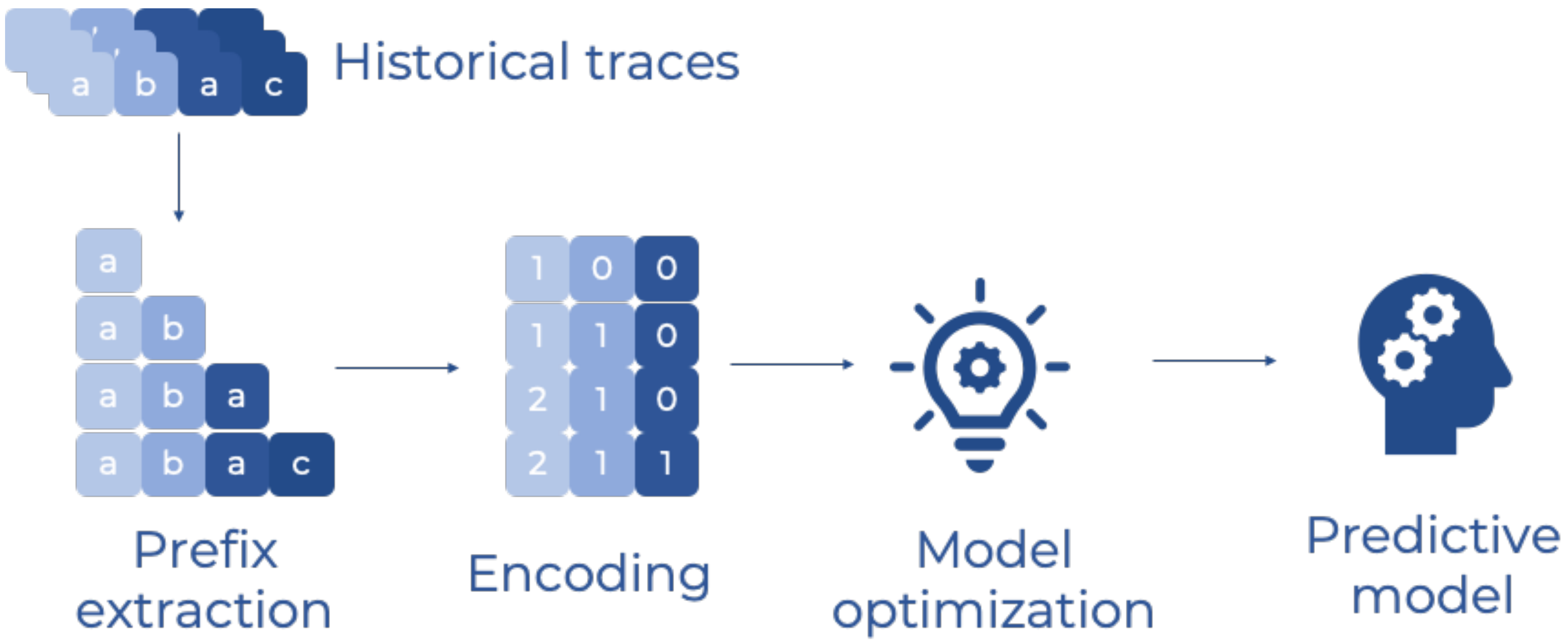
e.g. What is the completion time of the trace?

Type of prediction

e.g. What is the next activity?

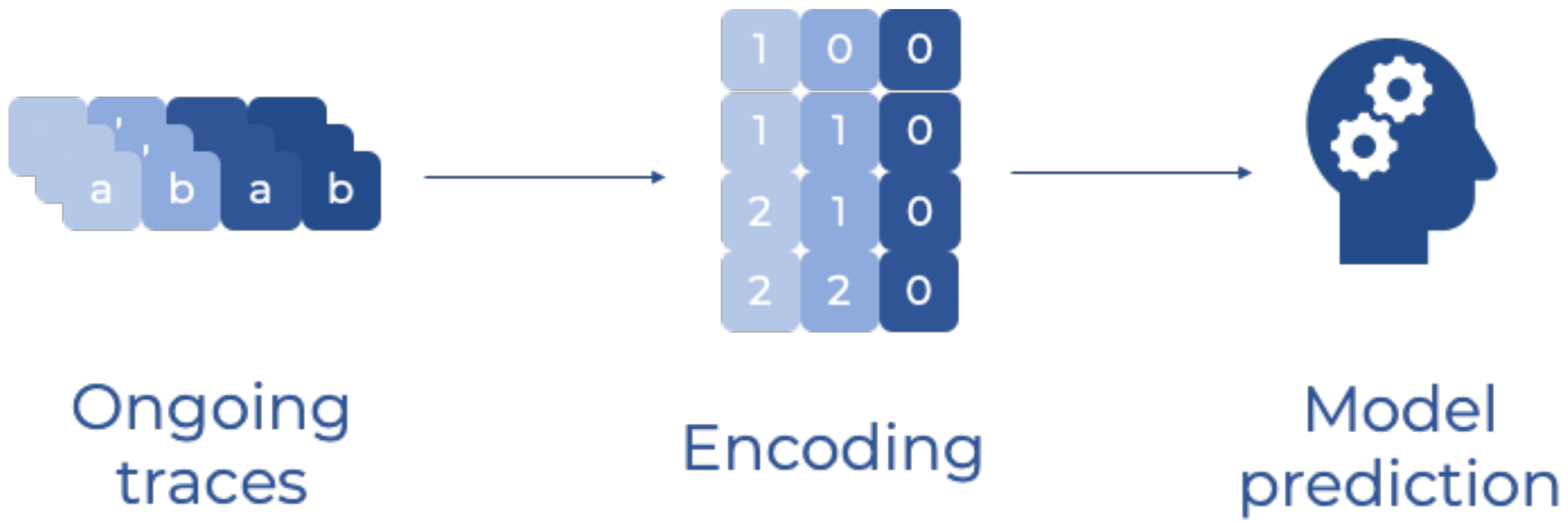
The pipeline of PPM project

Training Stage



The pipeline of PPM project

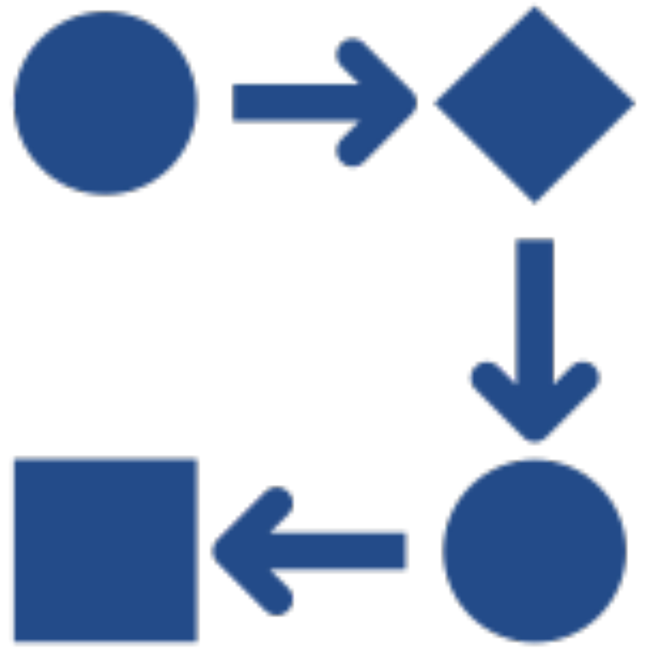
Runtime Stage



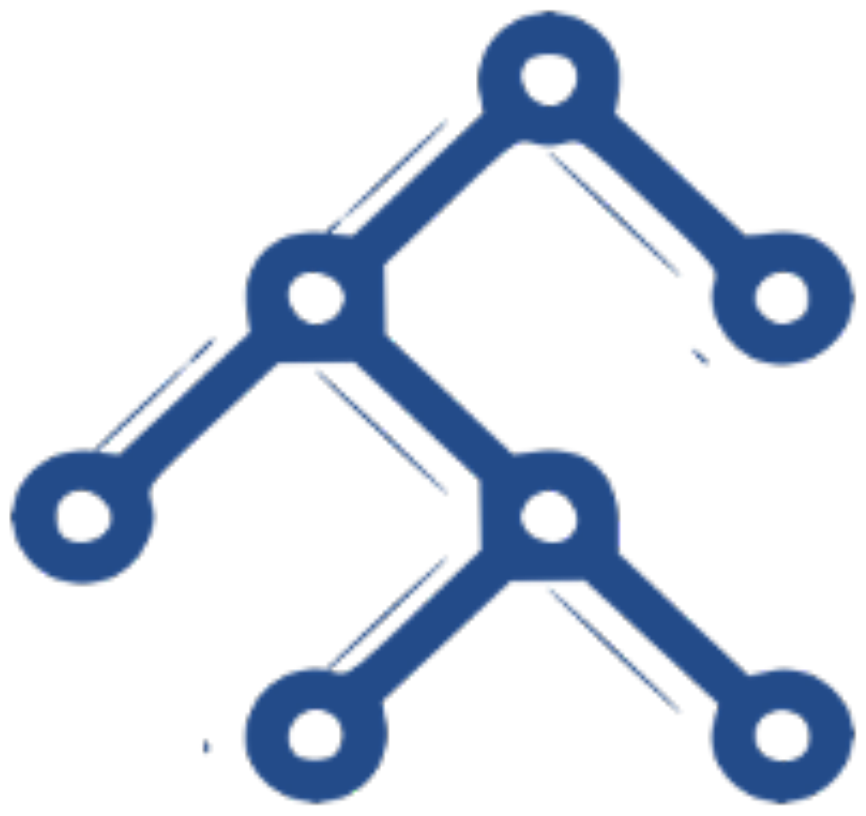
What's problem

The majority of PPM approaches operates in a static context where the analyst has the entire event log to analyze

Type of approach

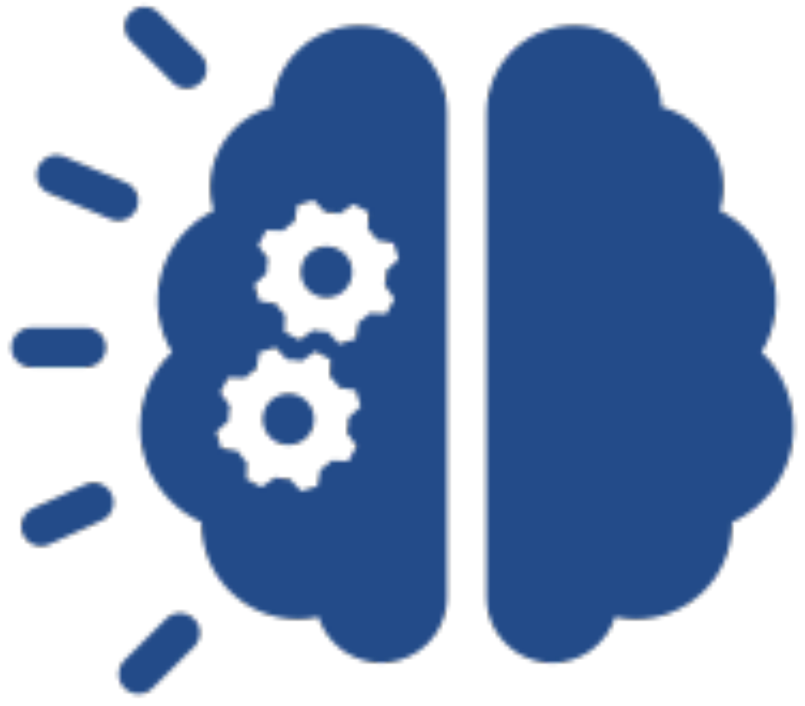


Petri Nets & Transition System



Machine learning alghoritms

Random Forest, XGB, etc.



Deep neural networks

LSTM, CNN, GNN etc.

The existing work in online setting

Next-activity prediction

Pauwels and Calders (2021)

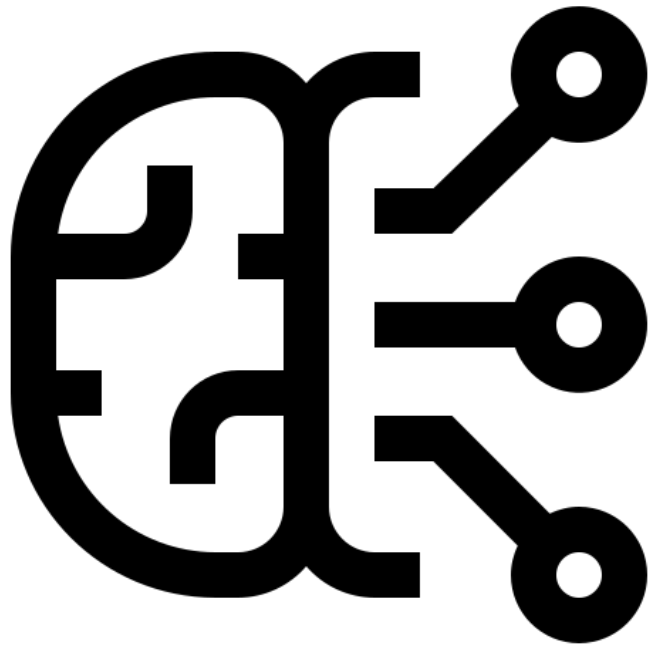
Outcome prediction

Maisenbacher and Weidlich (2017)

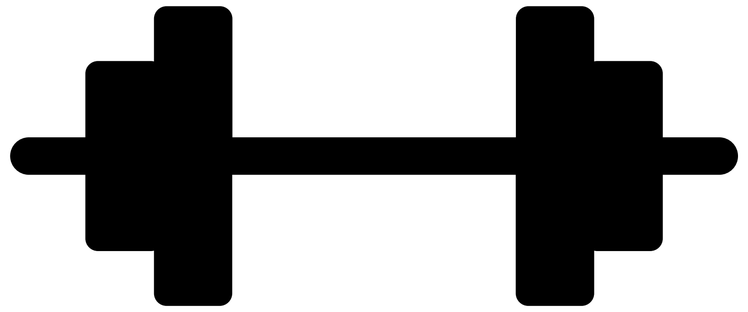
Rizzi et al. (2022)

The existing work in online setting

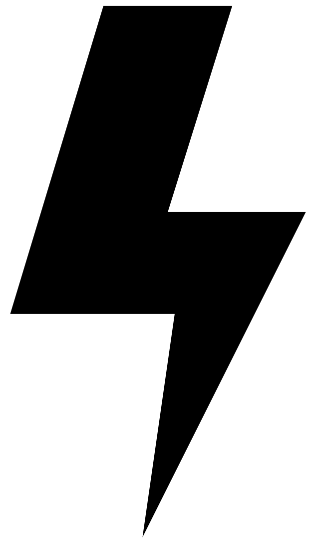
Pauwels and Calders (2021)



Simple Neural Network and dynamic Bayesian networks



Re-train and Fine-tuning



Concept drift based on work of Bose et al.*

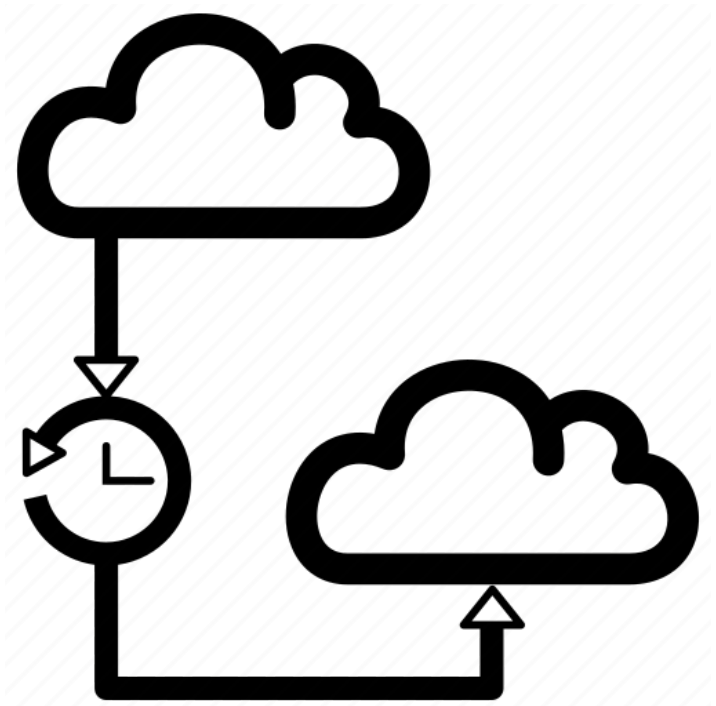
Bose R.P.J.C., van der Aalst W.M.P., Zliobaite I., Pechenizkiy M. Dealing with concept drifts in process mining IEEE Trans. Neural Netw. Learn. Syst., 25 (2014), pp. 154-171

The existing work in online setting

Pauwels and Calders (2021)

010
001
100

One-Hot encoding

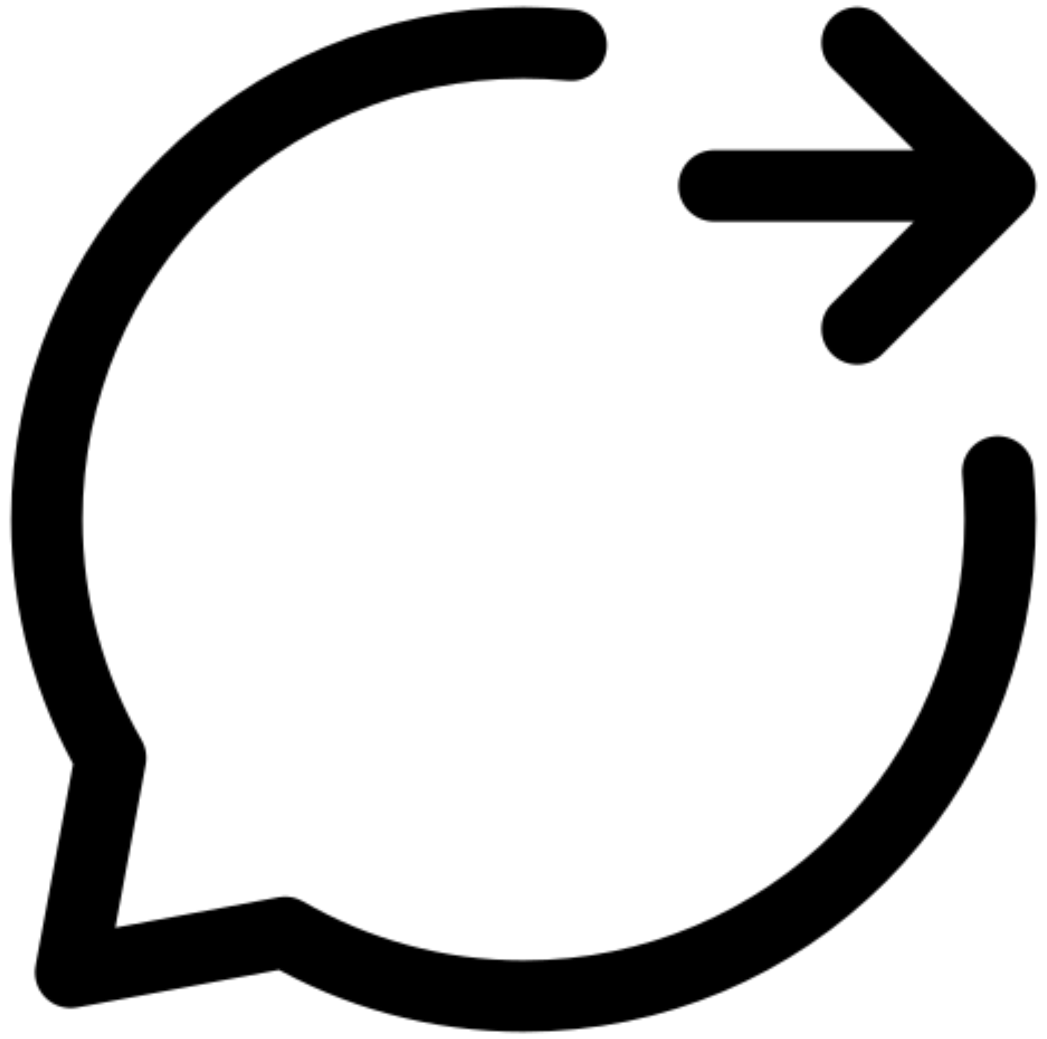


Latency missing

*Bose R.P.J.C., van der Aalst W.M.P., Zliobaite I., Pechenizkiy M. Dealing with concept drifts in process mining IEEE Trans. Neural Netw. Learn. Syst., 25 (2014), pp. 154-171

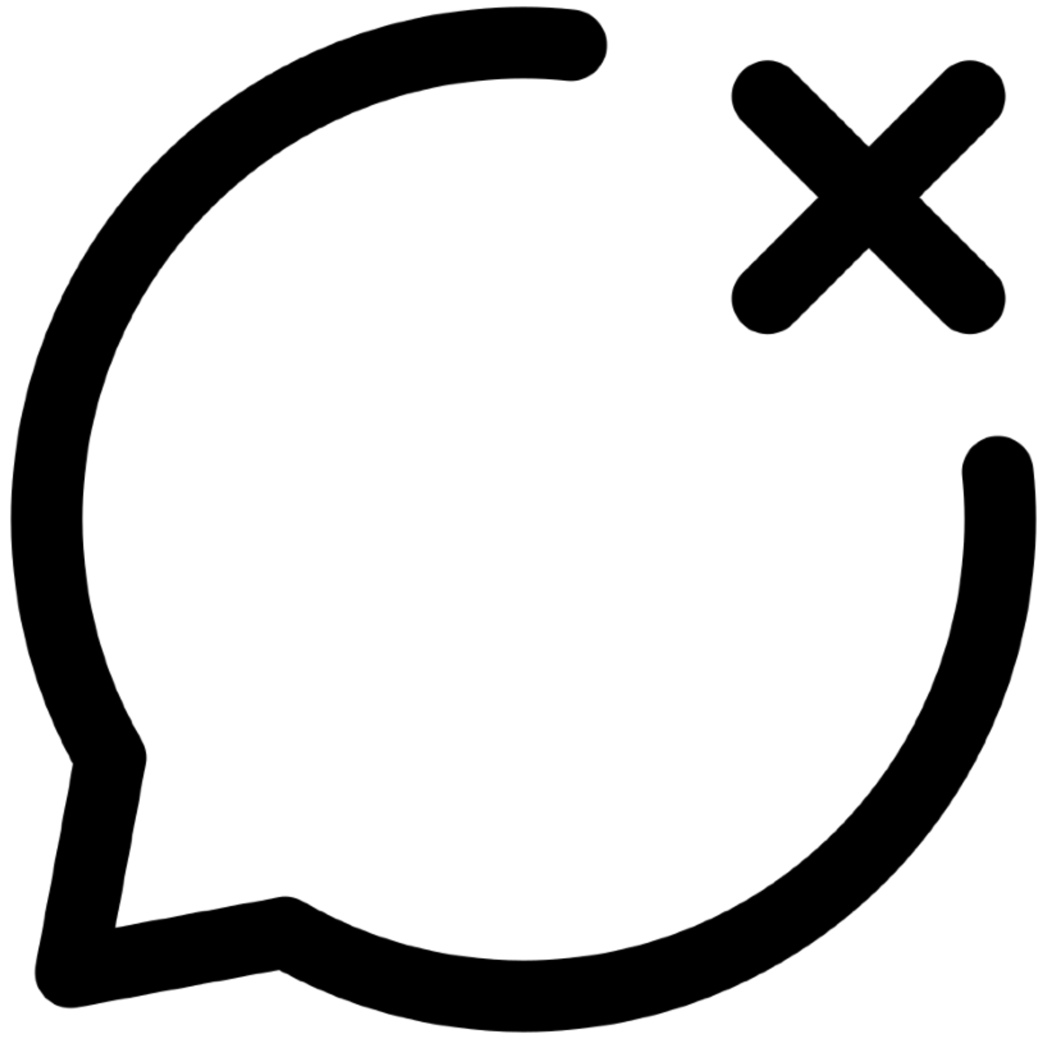
The existing work in online setting

Outcome



Tree based solution

Adaptive Hoeffding tree
Adaptive Random Forest



*Bose R.P.J.C., van der Aalst W.M.P., Zliobaite I., Pechenizkiy M. Dealing with concept drifts in process mining IEEE Trans. Neural Netw. Learn. Syst., 25 (2014), pp. 154-171

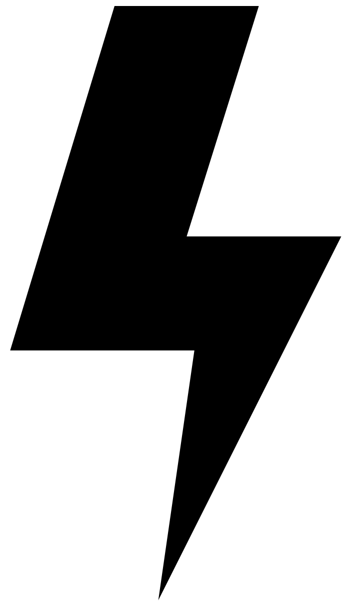
Solution

Develop a Predictive Process Monitoring approach that analyses an event stream, in order to update the predictive model over time

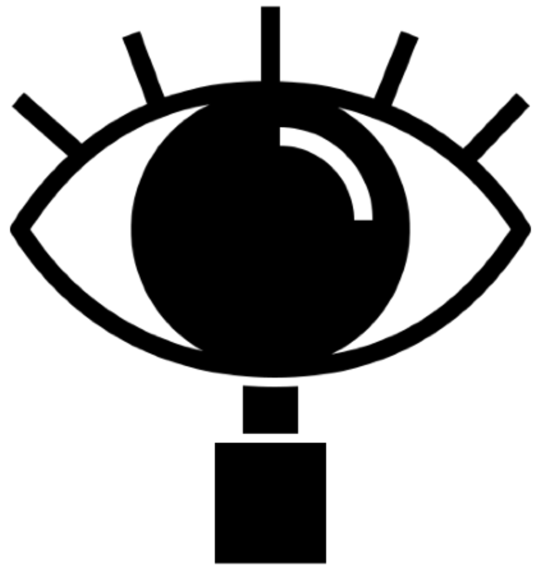
The challenges

Extract smart data from raw event data

Keeping the model accurate on new data

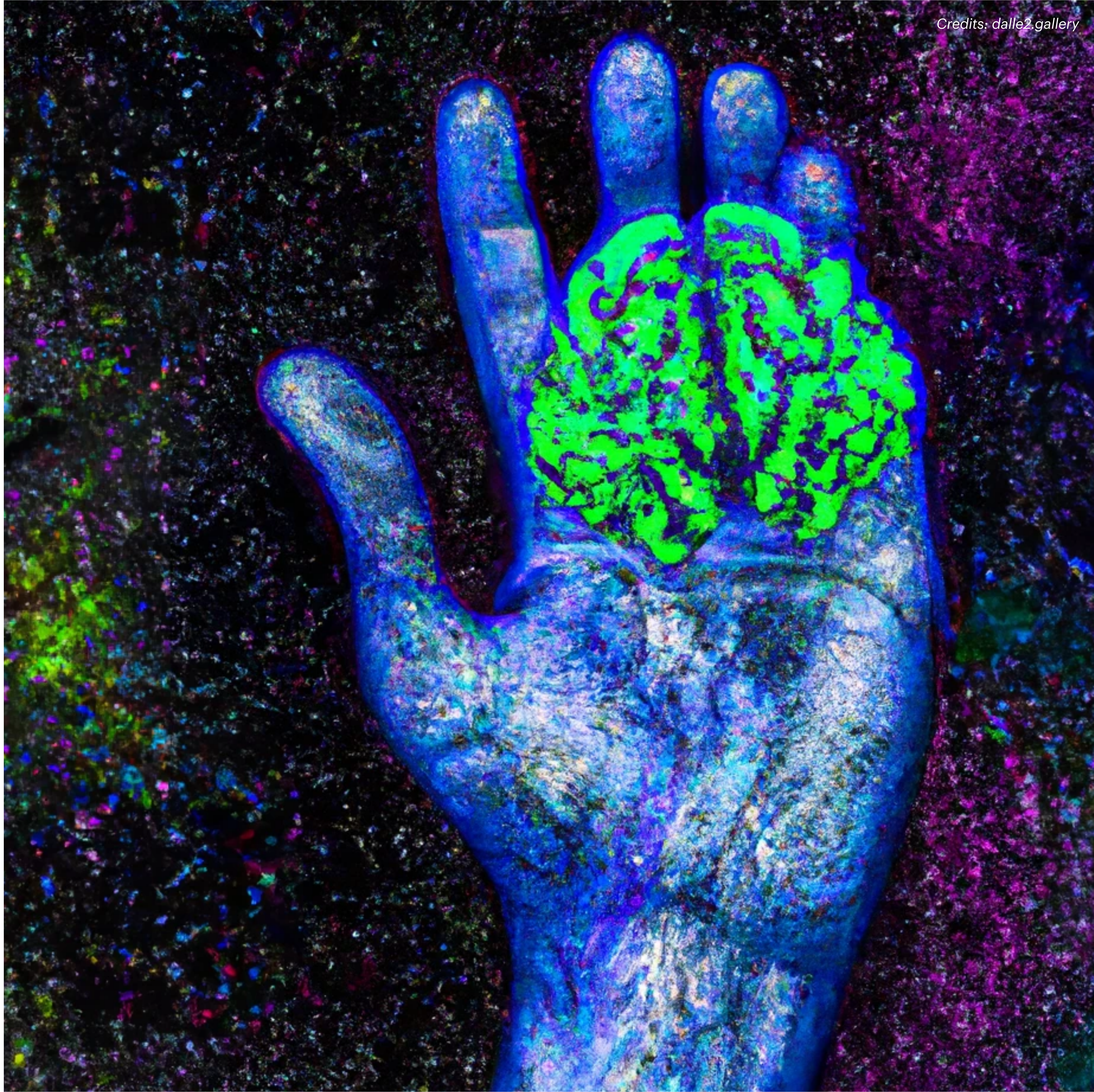


Handling drifting data



Concept drift understanding

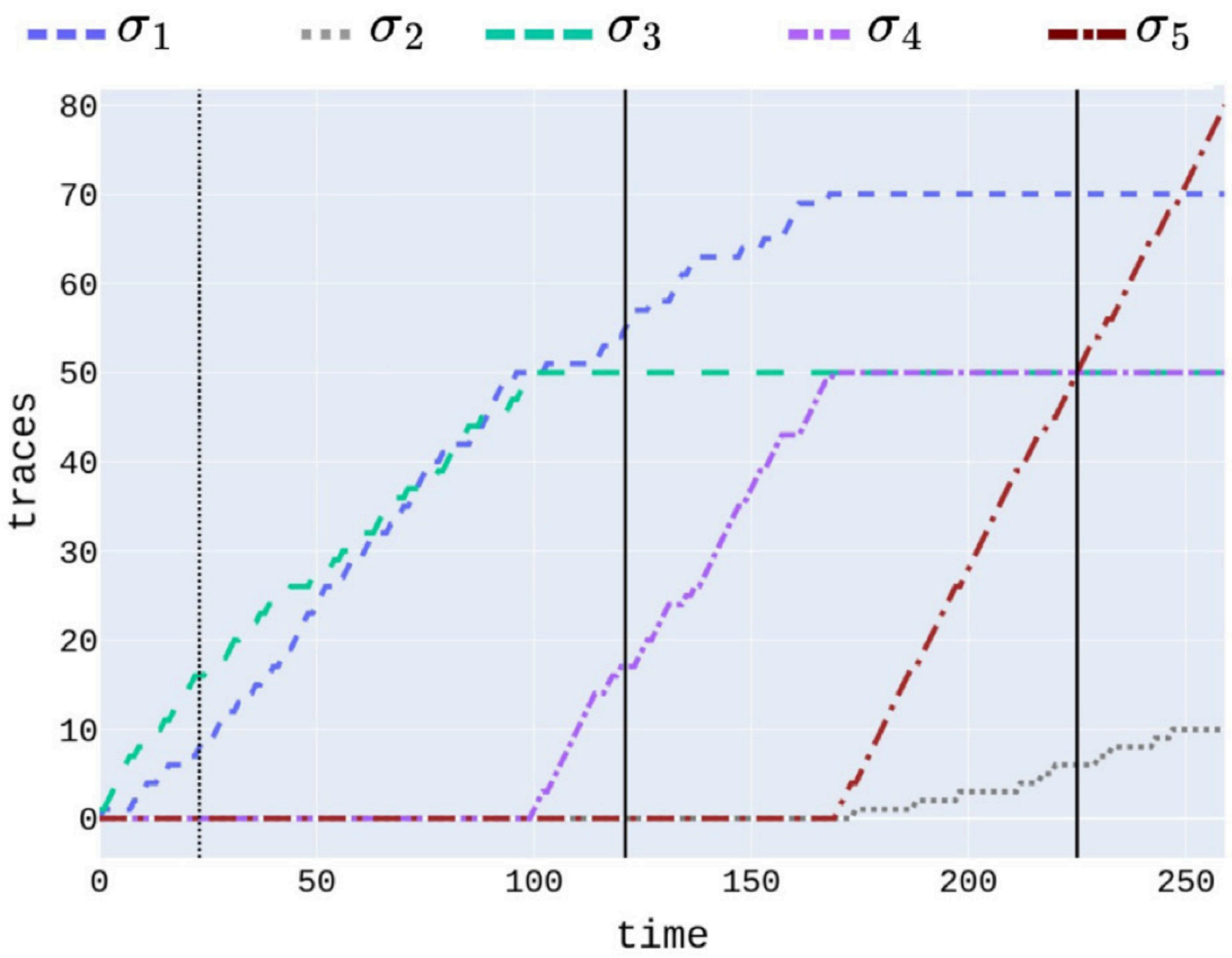
A PPM method that detects concept drifts and adapts a deep neural model to concept drifts



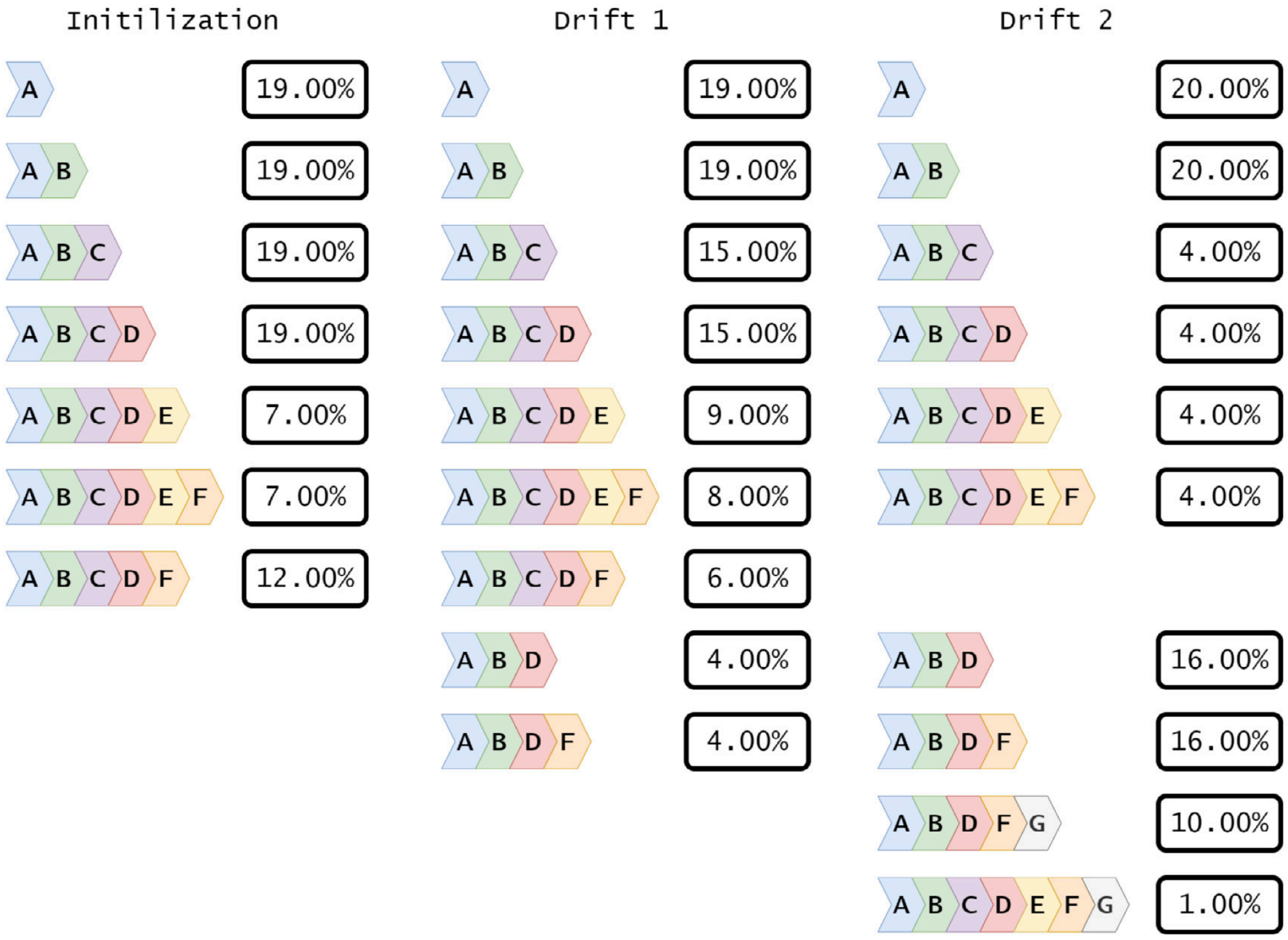
Motivating Example

$$\sigma_1 : \langle A, B, C, D, E, F, \perp \rangle^{70}$$
$$\sigma_2 : \langle A, B, C, D, E, F, G, \perp \rangle^{10}$$
$$\sigma_3 : \langle A, B, C, D, F, \perp \rangle^{50}$$
$$\sigma_4 : \langle A, B, D, F, \perp \rangle^{50}$$
$$\sigma_5 : \langle A, B, D, F, G, \perp \rangle^{80}$$

Motivating Example



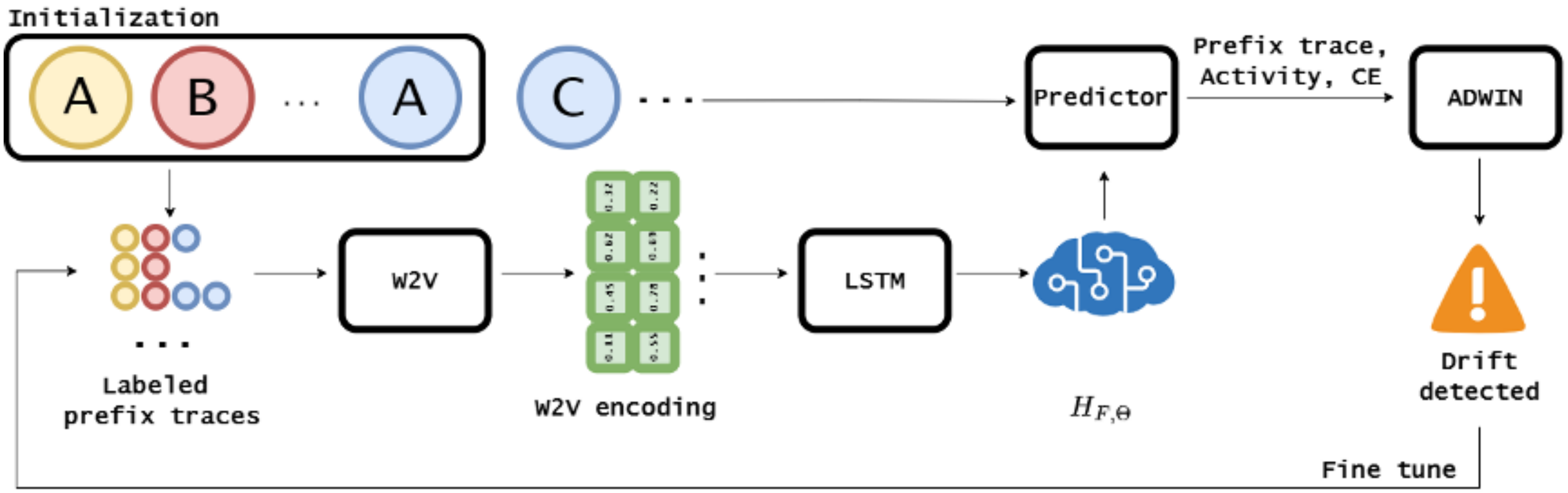
Motivating Example



Motivating Example

Activity	**Static**	DARWIN	Support
B	**1.000**	**1.000**	234
C	0.618	**0.660**	105
D	0.447	**0.691**	235
E	0.000	**0.521**	71
F	0.145	**0.732**	235
G	0.000	**0.512**	90
$\perp$	0.739	**0.781**	235

DARWIN pipeline



DARWIN - addressed challenges

Handling drifting data

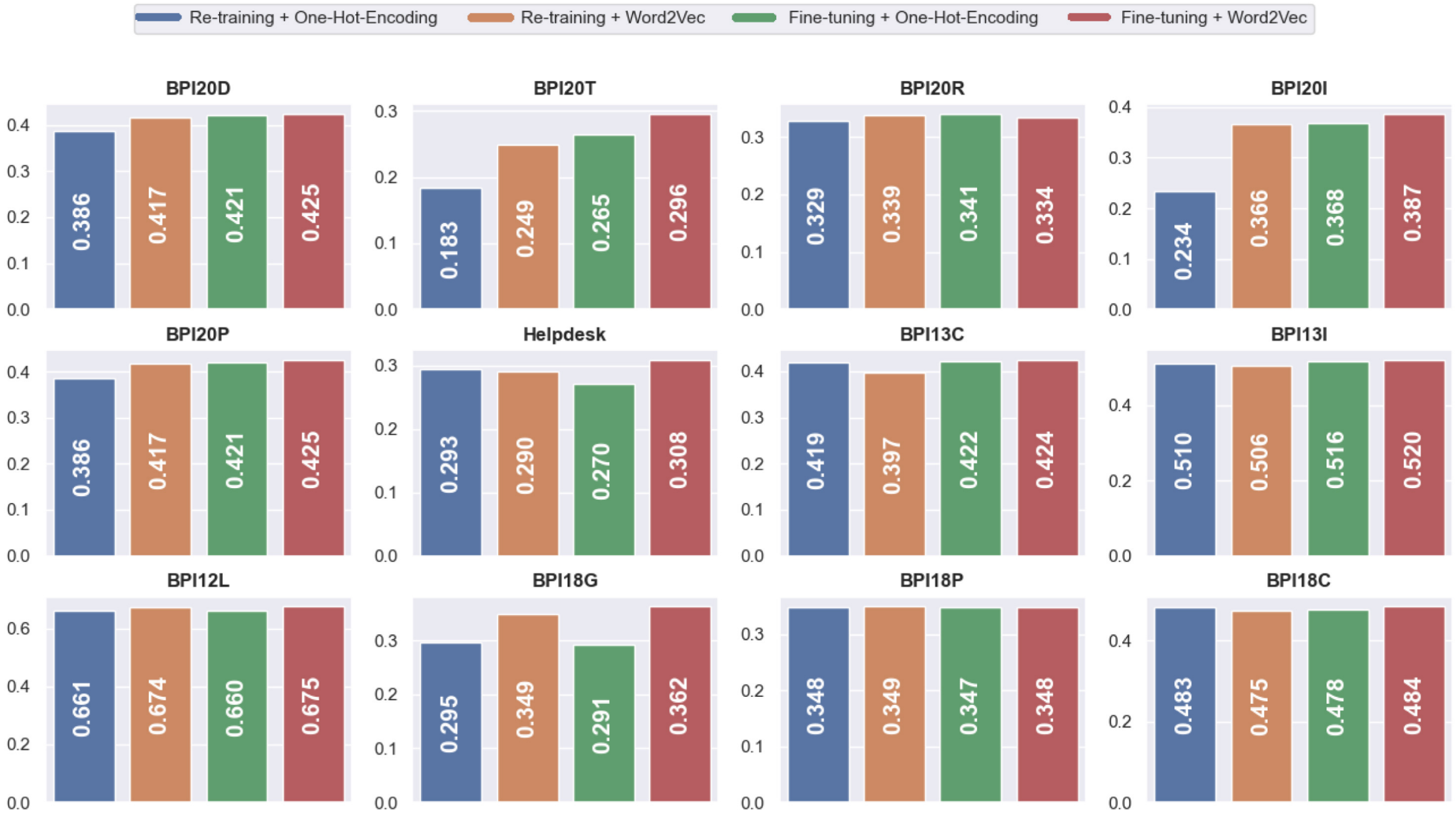
ADWIN to monitor the performance of a next-activity predictive model along an event stream

DARWIN - addressed challenges

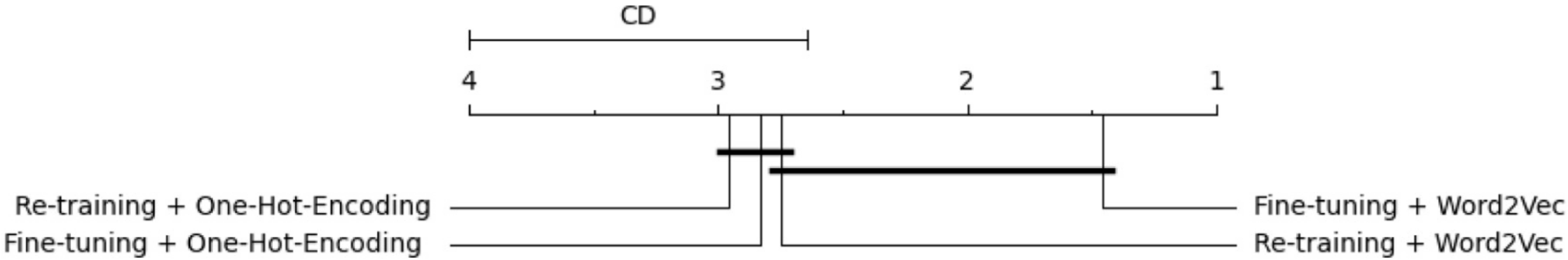
Extracting smart data from raw event data

Word2Vec embedding, to handle categorical data (i.e. sequence of activities)

W2V vs One-Hot



W2V vs One-Hot



DARWIN - addressed challenges

Keeping the model accurate

Fine-tuning to update the deep neural predictive model and the W2V model on the new data by mitigating the catastrophic forgetting

DARWIN vs Baseline

Keeping the model accurate

Event stream	A	R	E	T	V	T/V	T_{avgL}	T_{avgD} (days)	Σ_{totD} (days)
BPI20D (van Dongen, 2020)	17	2	56437	10500	99	106.06	5	11.5	889
BPI20T (van Dongen, 2020)	51	2	86581	7065	1478	4.78	12	87.4	1792
BPI20R (van Dongen, 2020)	19	2	36796	6886	89	77.37	5	8.2	941
BPI20I (van Dongen, 2020)	34	2	72151	6449	753	8.56	11	86.5	1313
BPI20P (van Dongen, 2020)	29	2	18246	2099	202	10.39	12	36.8	772
Helpdesk (Polato, 2017)	14	22	21348	4580	226	20.26	5	40.9	1451
BPI13C (Steeman, 2013)	7	585	6660	1487	327	4.54	4	179.2	2332
BPI13I (Steeman, 2013)	13	1440	65533	7554	2278	3.31	9	12.1	784
BPI12L (van Dongen, 2012)	36	69	262200	13087	4366	2.99	20	8.6	165
BPIC18G (van Dongen and Borchert, 2018)	23	117	569209	29059	9372	19.58	20	143.5	576
BPIC18P (van Dongen and Borchert, 2018)	10	111	132963	14750	3615	9.01	9	196	976
BPIC18C (van Dongen and Borchert, 2018)	7	113	161296	43808	59	742.51	4	57.3	1051

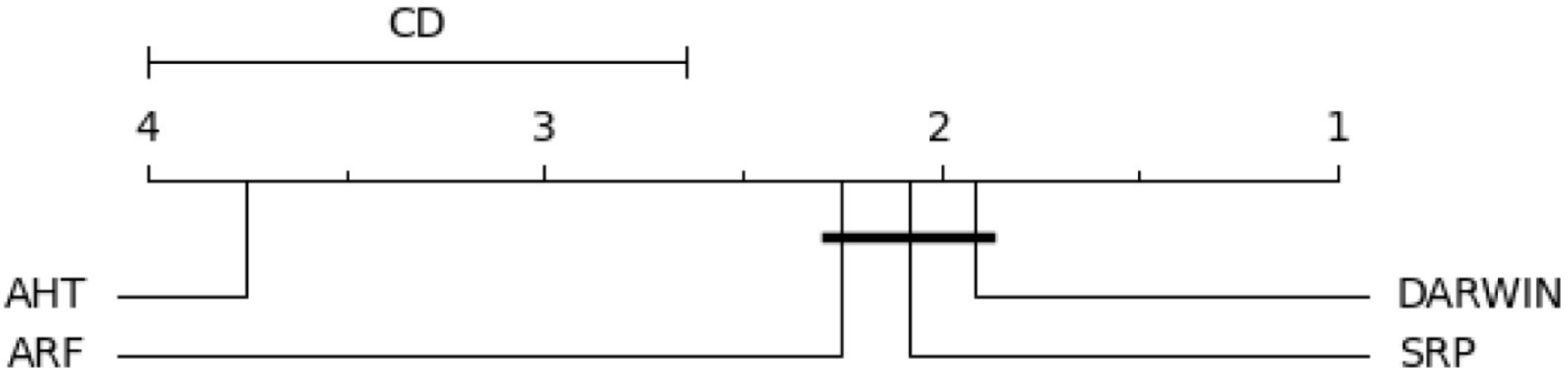
DARWIN vs Baseline

Keeping the model accurate

Stream	Fscore				TIME			
	DARWIN	AHT	ARF	SRP	DARWIN	AHT	ARF	SRP
BPI20D	**0.425**	0.397	0.402	0.415	5572.26	**77.02**	179.76	245.51
BPI20T	0.296	0.291	0.338	**0.341**	8036.72	**229.78**	582.37	667.84
BPI20R	0.334	0.353	0.351	**0.360**	2709.17	**33.86**	81.65	134.01
BPI20I	0.387	0.373	0.401	**0.414**	5977.03	**150.33**	359.09	482.28
BPI20P	0.411	0.403	0.451	**0.457**	1439.05	**11.64**	31.56	63.73
Helpdesk	**0.308**	0.285	0.306	0.299	1819.20	**13.77**	32.93	69.70
BPI13C	**0.424**	0.234	0.371	0.397	762.55	**2.19**	7.72	17.79
BPI13I	**0.520**	0.419	0.453	0.391	4374.57	**153.34**	243.25	392.78
BPI12L	**0.675**	0.452	0.637	0.644	22952.66	**1921.17**	3235.36	2942.48
BPIC18G	0.362	0.361	**0.402**	0.390	53934.60	**9746.56**	14453.67	14567.99
BPIC18P	**0.348**	0.304	0.341	0.336	6230.80	**612.04**	855.52	1043.50
BPIC18C	**0.484**	0.446	0.482	0.469	10117.42	**929.35**	934.17	1254.31

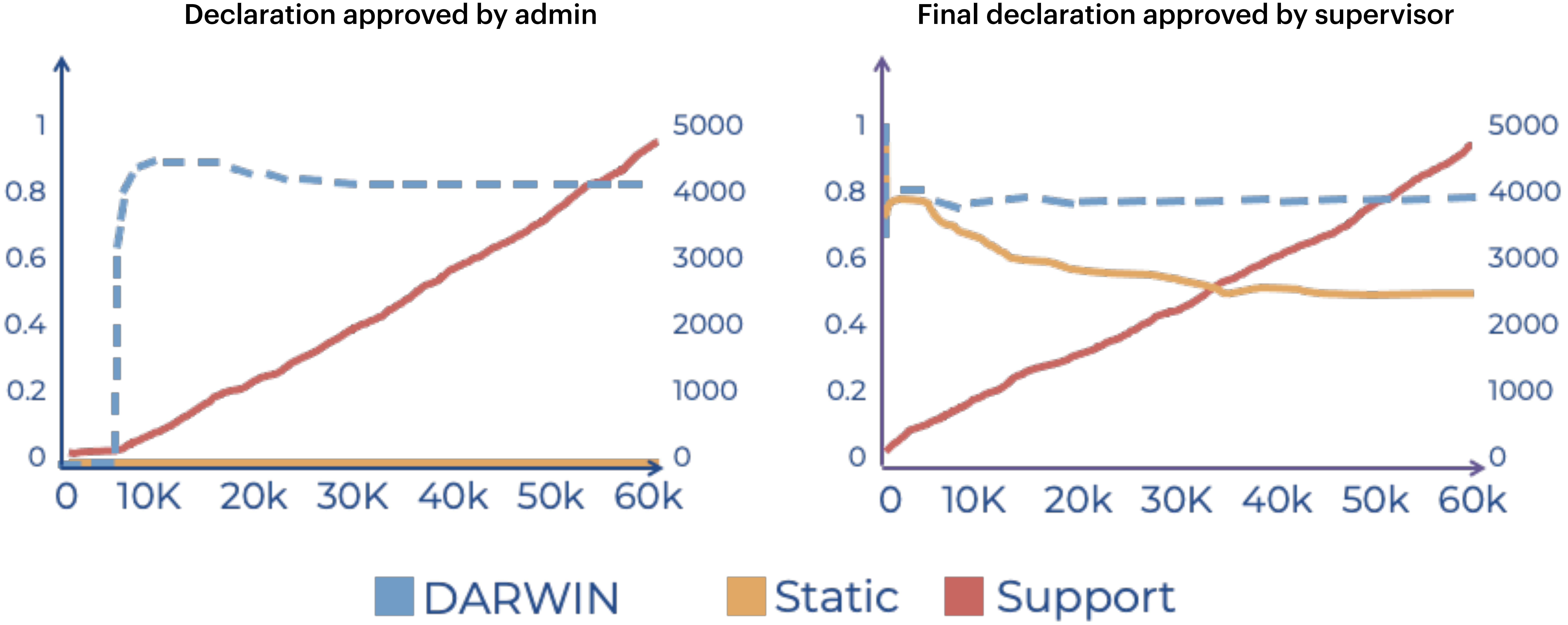
DARWIN vs Baseline

Keeping the model accurate



DARWIN - addressed challenges

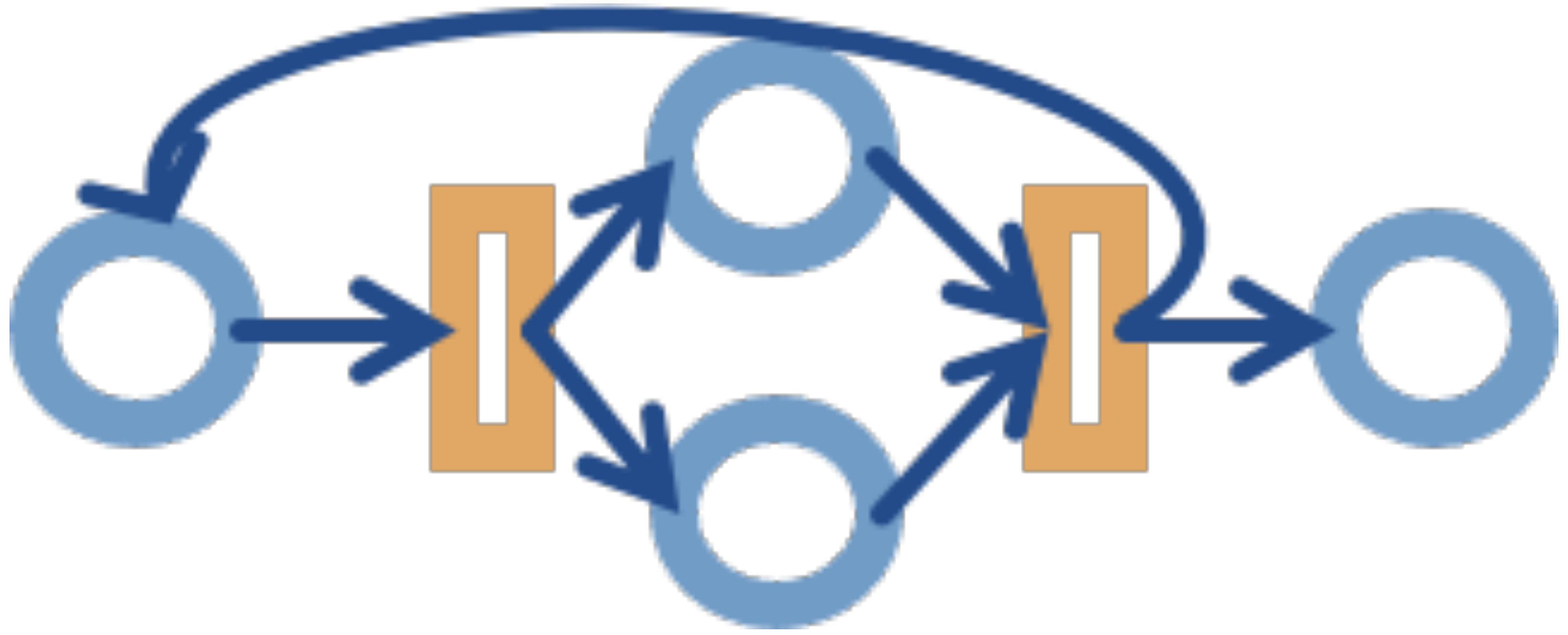
Keeping the model accurate



DARWIN - addressed challenges

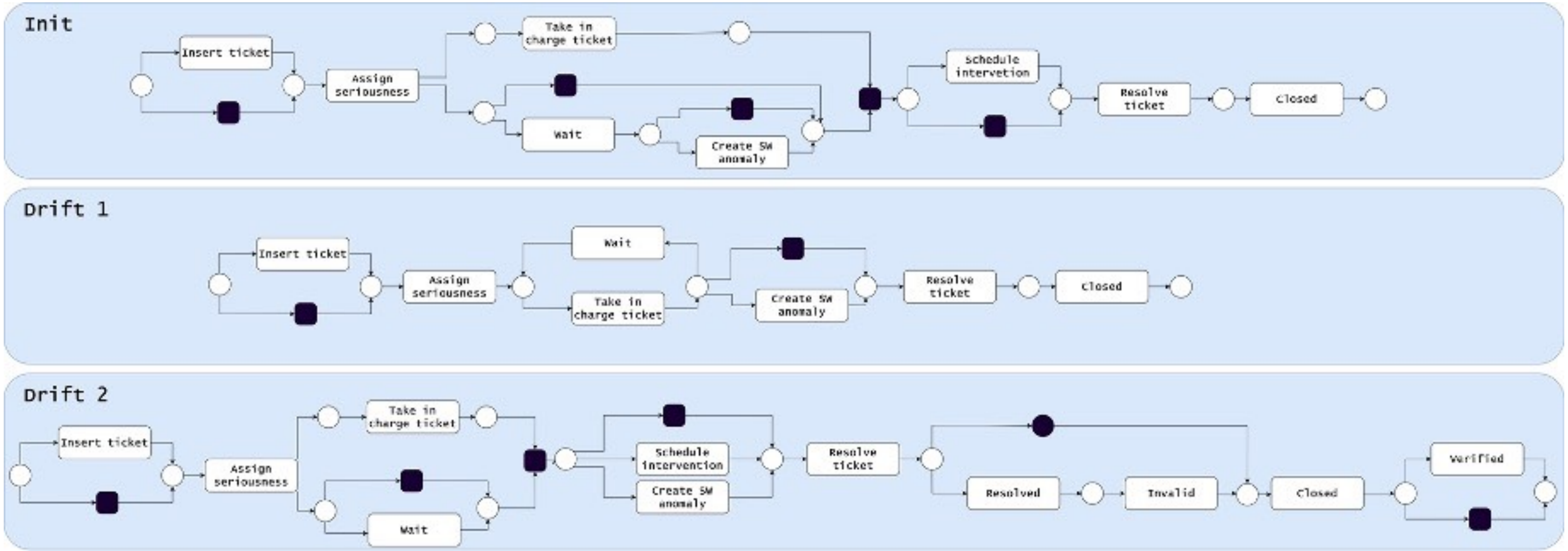
Concept drift understanding

To **understand** how the process model changes by causing the concept drift



DARWIN - addressed challenges

Concept drift understanding



Unlocking Data Insights - Introduction to Data-Centric AI

Self-Supervised Learning

Funded by the European Union NextGenerationEU

Ministero dell'Università e della Ricerca

Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA

FAIR Future Artificial Intelligence Research

UniBa UNIVERSITÀ DEGLI STUDI DI BARI ALDO MORO

dib

KDDE

refererce: Deep Learning with PyTorch and Lightning - Chapter 8

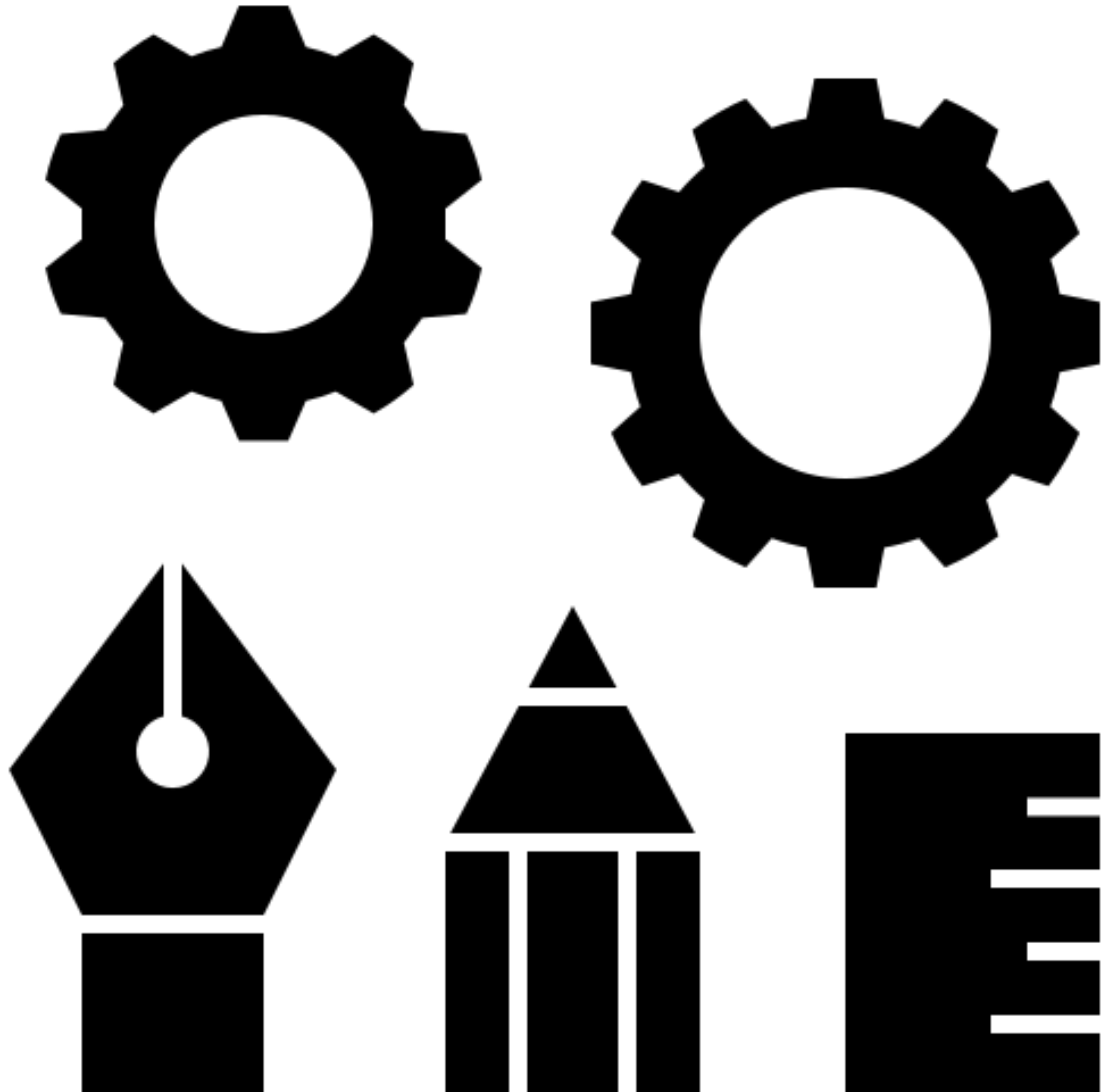
Self-Supervised Learning
Executive Summary

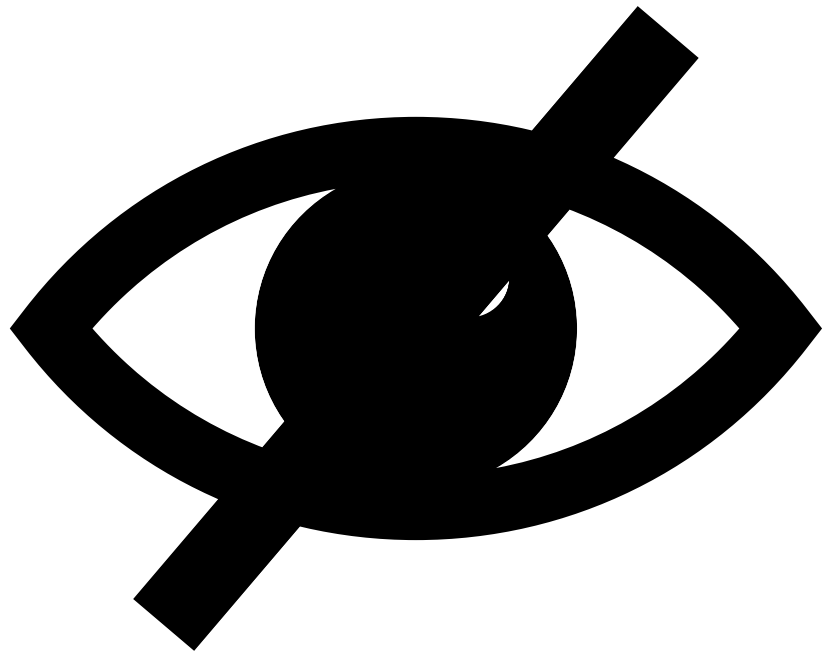
- Getting started with Self-Supervised Learning
- What is Contrastive Learning?
- SimCLR architecture

What’s problem

Supervised

Machine
Learning





Unsupervised

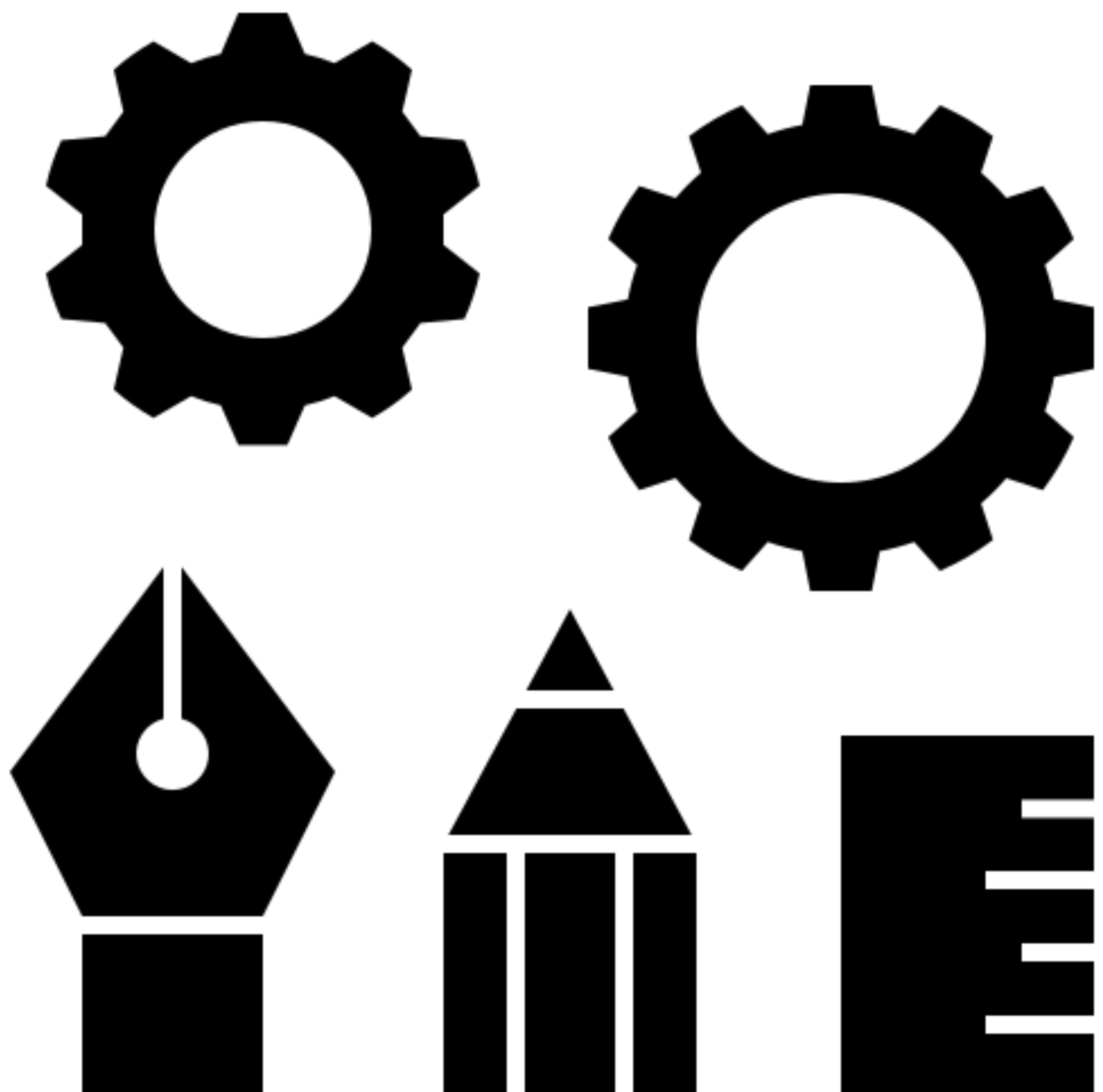
What's problem

Supervised

Labels cost time and money

Unsupervised

A lot of unlabelled data



What's problem

ImageNet dataset

Over 14 million images

But...

+ workers

+ times

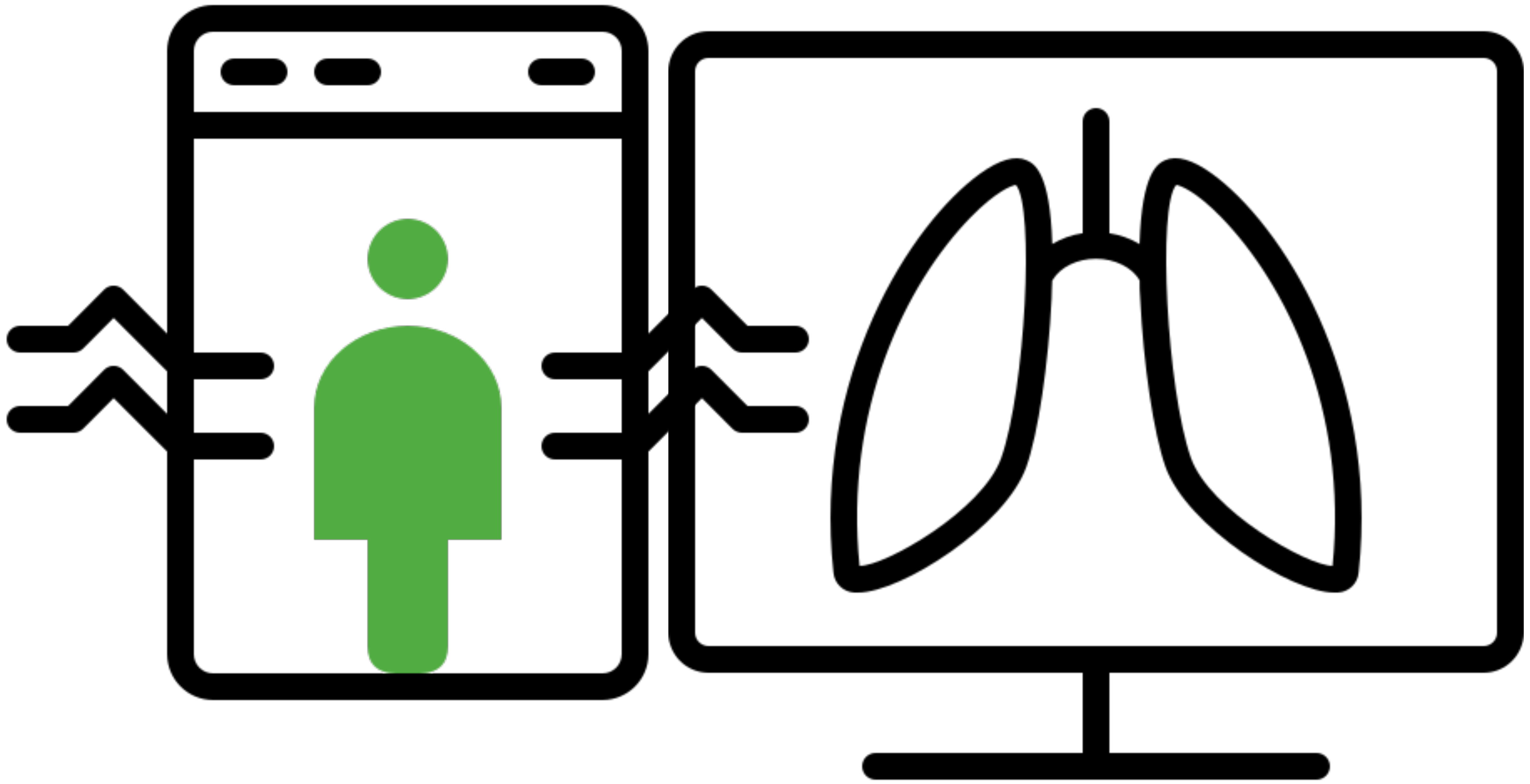
What's problem

Medical dataset

A lot of images and scans

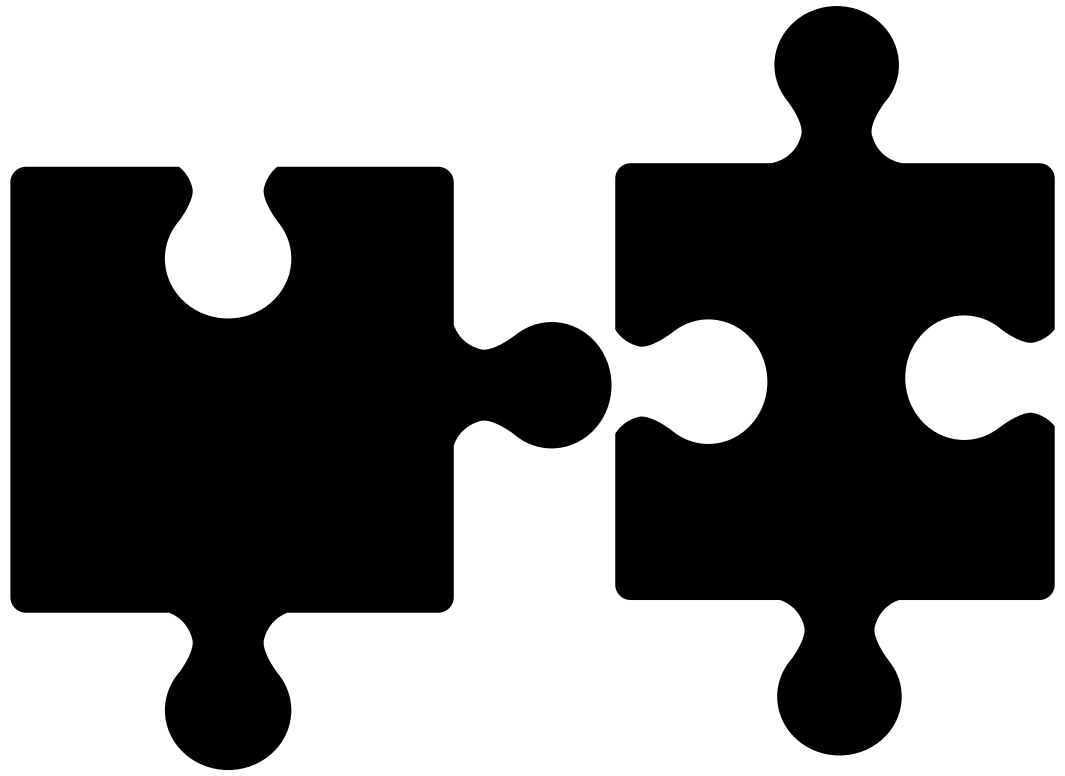
But...

labeling very hard



The question is

What if we can come up with a new method that can work without needing so many labels, such as unsupervised learning, but gives output that is as high-impact as supervised learning?



Self-supervised learning

Self-supervised learning

Self-Supervised Learning is the latest paradigm in Machine Learning and is the most advanced frontier. While it has been theorized for a few years, it's only in the last year that it has been able to show results comparable to supervised learning and has become touted as the future of Machine Learning.

Self-supervised learning

- The foundation of Self-Supervised Learning for images is that we can make machines learn a true representation even without labels
- With a minuscule number of labels (as low as 1% of the dataset), we can achieve as good results as supervised models can
- This unlocks the untapped potential in millions of datasets that are sitting unused due to the lack of high-quality labels

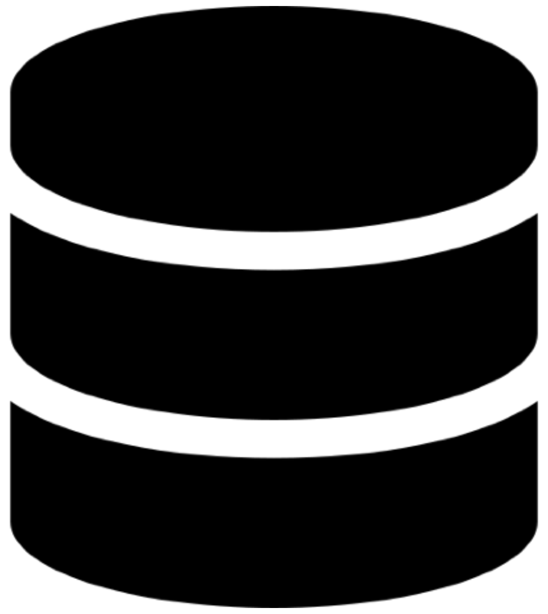
Credits: Brett Jordan - Unsplash

Getting started with Self-Supervised Learning

The **future** of Machine Learning has been hotly contested given the spectacular success of **Deep Learning** methods such as CNN and RNN in recent years

Getting started with Self-Supervised Learning

They don't compare very well to humans on tasks such as **reasoning**, **deduction**, and **comprehension**



They require an **enormous amount of well-labeled** training data to learn even something as simple as image recognition

Getting started with Self-Supervised Learning

Unsurprisingly, that is not the way humans learn. A child does not need millions of labeled images as input before it can recognize objects. The incredible ability of the human brain to generate its own new labels based on a minuscule amount of initial information is unparalleled when compared to Machine Learning

Getting started with Self-Supervised Learning

Solutions for AI

Reinforcement Learning

The focus is on constructing a game-like environment and making a machine learn how to navigate through the environment without explicit directions

Self-Supervised Learning

The focus is on making the machine learn a bit like a human by trying to create its own labels and continue to learn adaptively

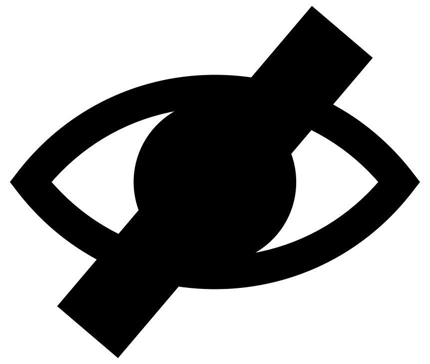
Self-Supervised Learning is the brainchild of Yann LeCun

Getting started with Self-Supervised Learning

The core concept is that we can learn in multiple dimensions of data

Getting started with Self-Supervised Learning

In supervised learning, we have data (x) and labels (y), and we can do a lot of things, such as prediction, classification, and object detection

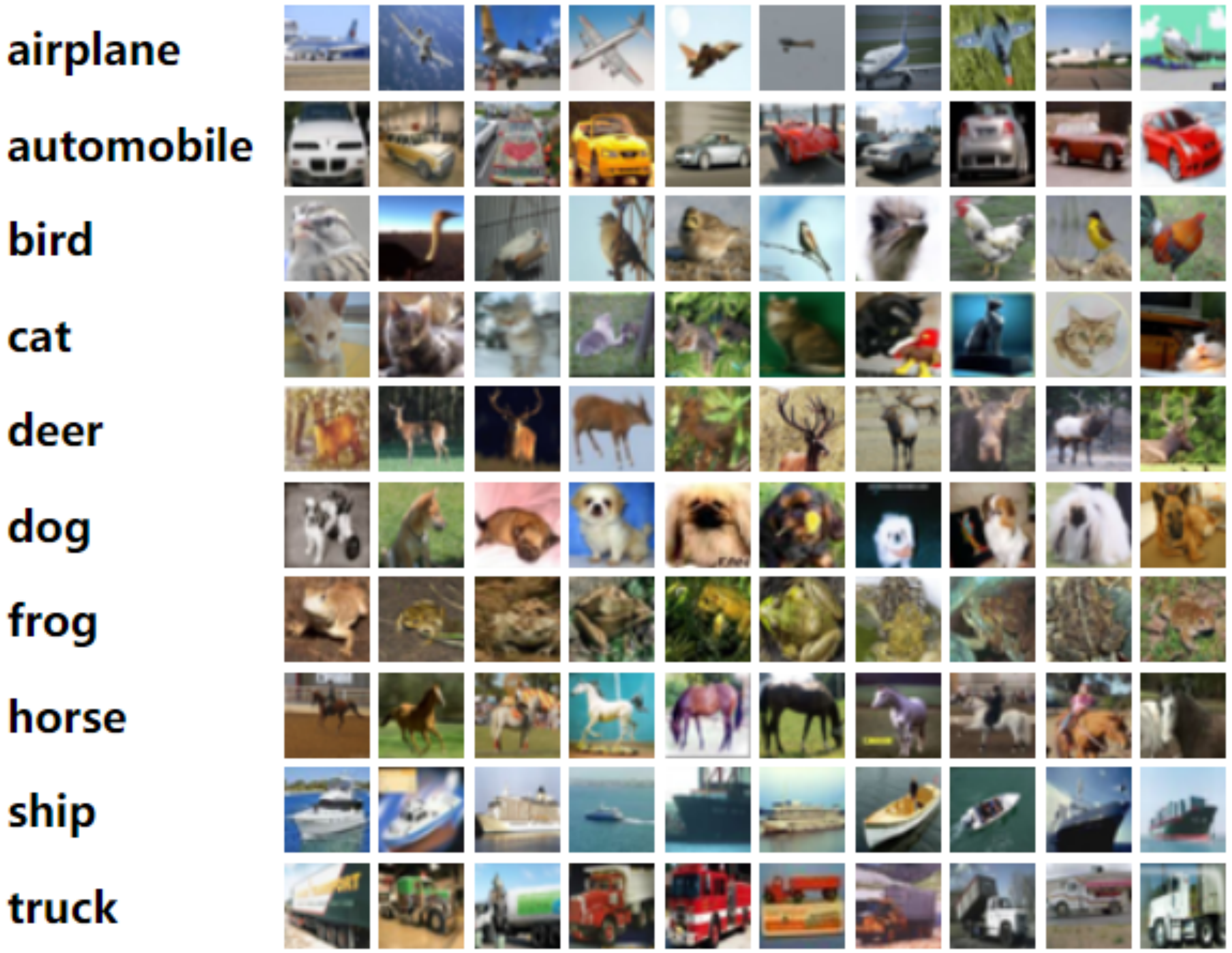


In unsupervised learning, we only have data (x), and we can only do clustering types of models. In unsupervised learning, we have the advantage that we don't need costly labels, but the kinds of models we can build are limited

What if we start in an unsupervised manner and then move on to supervised learning?

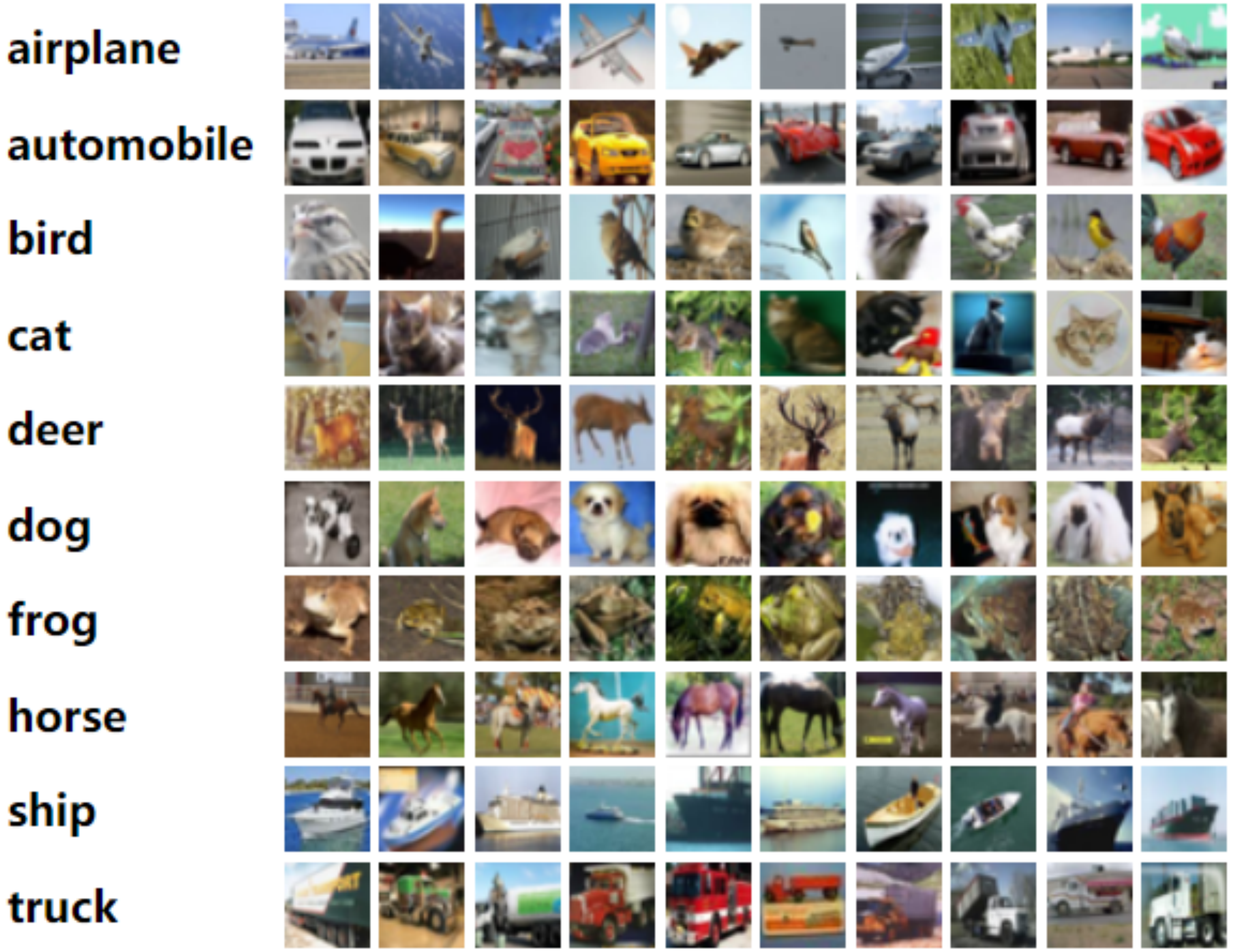
Getting started with Self-Supervised Learning

Suppose we have an image dataset such as CIFAR-10 that consists of 10 image classes (classes such as bird, plane, dog and cat) distributed over 65.000 labels



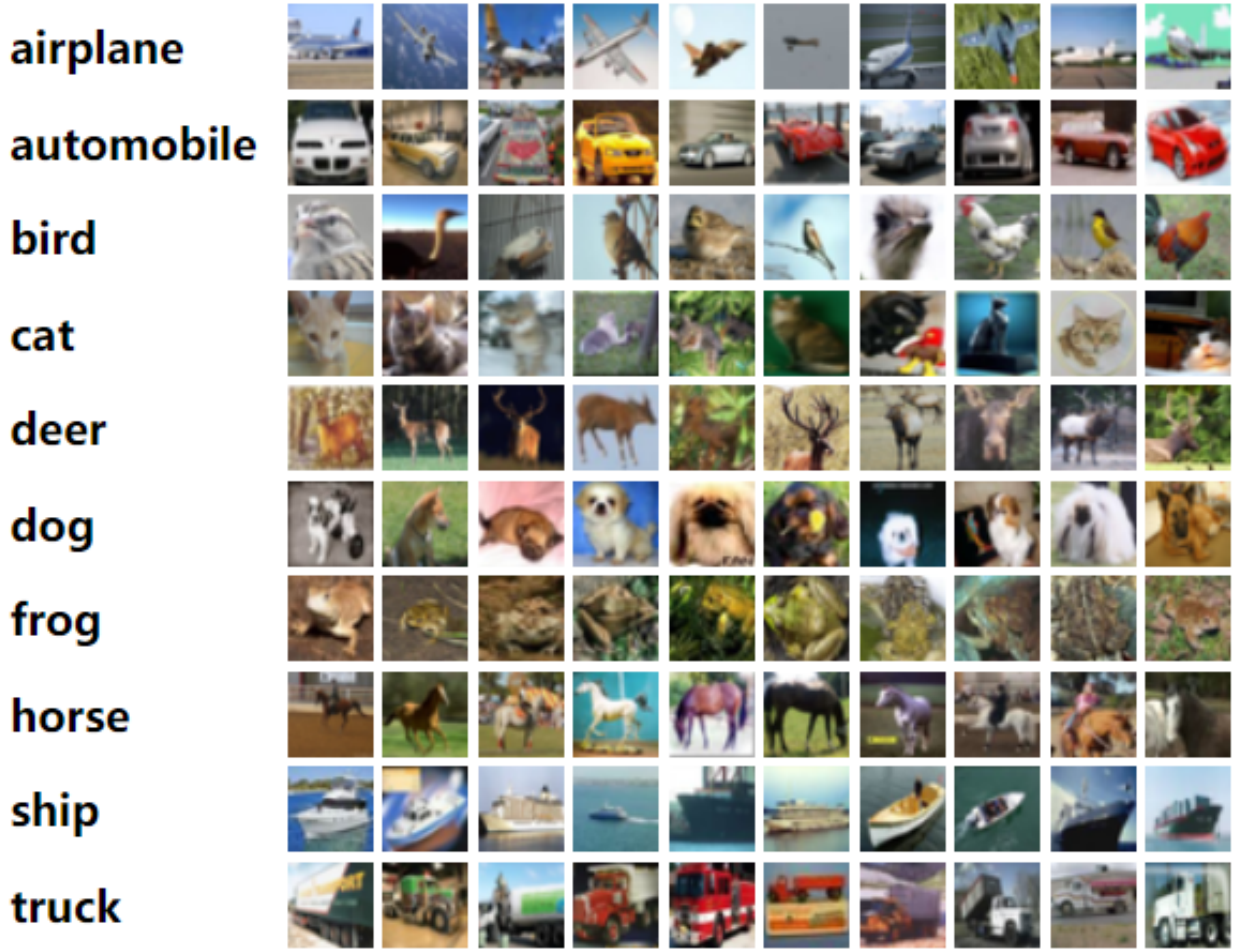
Getting started with Self-Supervised Learning

The machine must learn to recognize these 10 classes. What if instead of providing 65.000 labels, as we did for this dataset, We only provide **10 labels (one for each class)** and the machine finds images similar to those classes and adds labels to them?

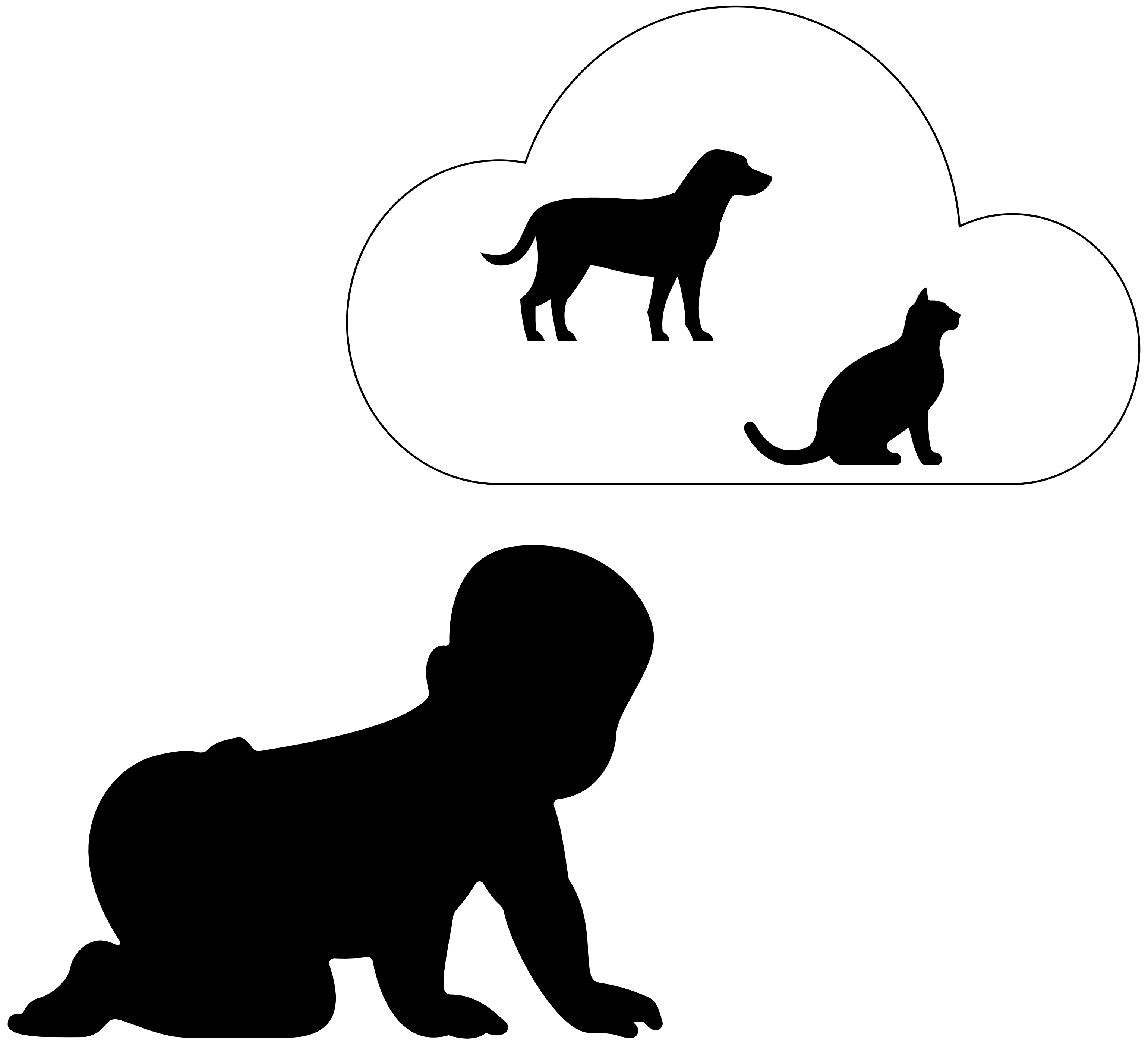


Getting started with Self-Supervised Learning

If we can do that, then the machine is self-supervising its learning process and will be able to solve previously unsolved problems



Contrastive Learning



- The idea of understanding an image is to get an image of a particular kind (say a dog) and then we can recognize all other dogs by reasoning that they share the same representation or structure
- For example, if you show a child who is not yet able to talk or understand language (say, less than 2 years old) a picture of a dog (or a real dog for that matter) and then give them a pack of cards with a collection of animals, which includes dogs, cats, elephants, and birds, and ask the child which picture is similar to the first one, it is most likely that the child could easily pick the card with a dog on it
- And the child would be able to do so even without you explaining that this picture equals "dog" (in other words, without supplying any new labels)

Contrastive Learning

There have been various proposed architectures for contrastive learning that have had spectacular results. Some popular ones are **SimCLR, CPC, YADIM, and NOLO**. In next slides , we will see the architecture that is fast becoming a de facto standard for contrastive learning – **SimCLR**

SimCLR architecture

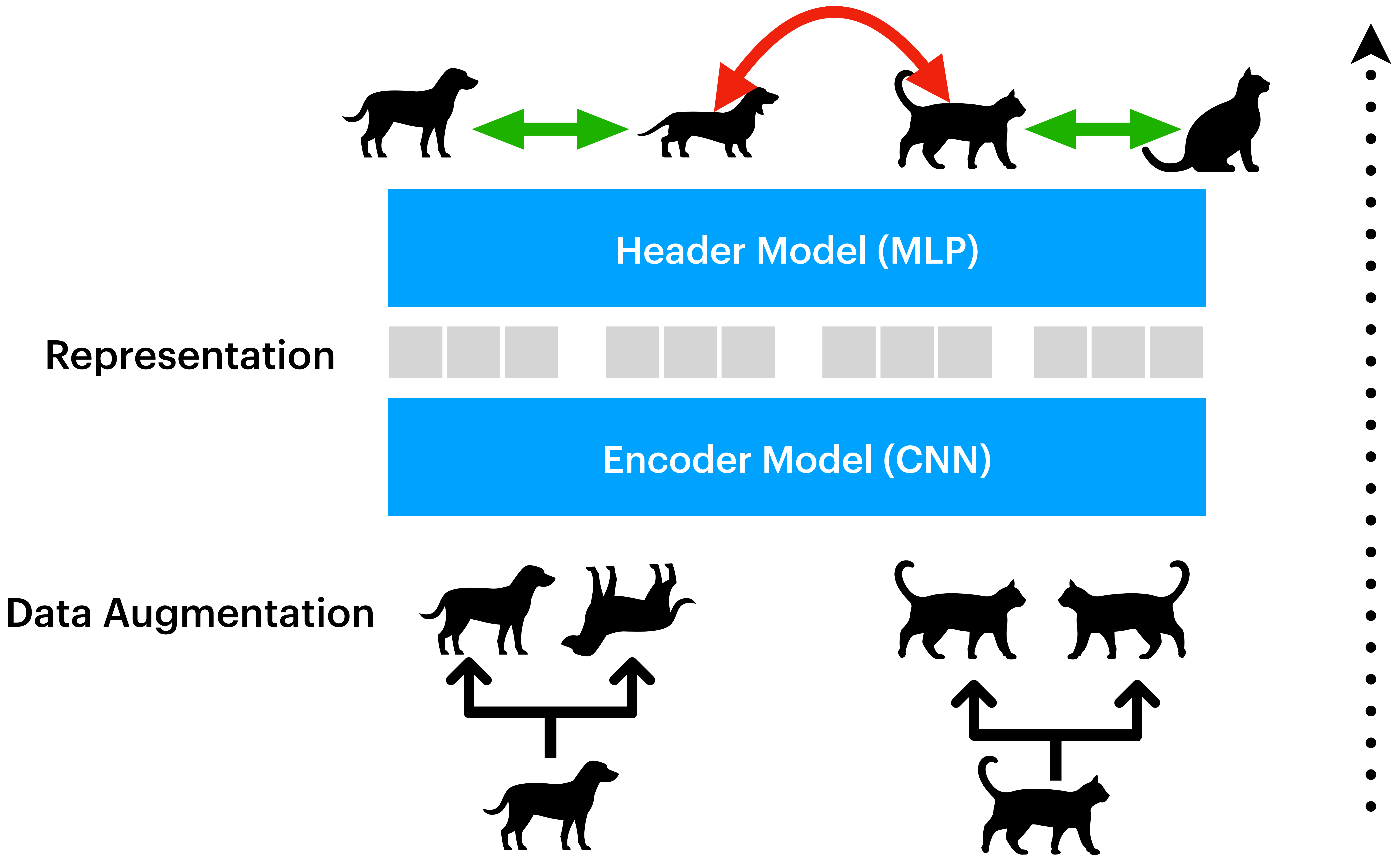
SimCLR stands for Simple Contrastive Learning Architecture*

The architecture has shown in relation to the ImageNet dataset that we can achieve 93% accuracy with just 1% of labels

**Chen, Ting, et al. "A simple framework for contrastive learning of visual representations." International conference on machine learning. PMLR, 2020.*

SimCLR architecture

How does SimCLR work?



SimCLR architecture
How does SimCLR work?

The architecture consists of the following building blocks:

1. As a first step, **data augmentation** is performed on the group of random images. Various data augmentation tasks are performed. Some of them are standard ones, such as rotating the images, cropping them, and changing the color by making them grayscale. Other more complex data augmentation tasks, such as Gaussian Blur, are also performed

SimCLR architecture

How does SimCLR work?

The architecture consists of the following building blocks:

This data augmentation step is very important since we want to make the model learn the true representation reliably and consistently. Another, and rather important, reason is that we don't have labels in the dataset. So, we have no way of knowing which images are actually similar to each other and which are dissimilar.
And thus, having various augmented images from a single image creates a "true" set of similar images for the model that we can be apriori sure of

SimCLR architecture

How does SimCLR work?

The architecture consists of the following building blocks:

2. The next step is then to create a batch of images that contains similar and dissimilar images. As an analogy, this can be thought of as a batch that has some positive ions and some negative ions, and we want to isolate them by moving a magical magnet over them (SimCLR).

SimCLR architecture

How does SimCLR work?

The architecture consists of the following building blocks:

3. This process is followed by an encoder that is nothing but a CNN architecture. ResNet architectures such as ResNet-18 or ResNet-50 are most commonly used for this operation. However, we strip away the last layer and use the output after the last average pool layer. This encoder helps us learn the image representations

SimCLR architecture

How does SimCLR work?

The architecture consists of the following building blocks:

4. This is followed by the header module (also known as projection head), which is a Multi-Layer Perceptron (MLP) model. This is used to map contrastive loss to the space where the representations from the previous step are applied. Our MLP can be a single hidden layer neural network (as in SimCLR) or a 3-layer network (as it is in SimCLR2). You can even experiment with larger neural networks. This step is used to balance alignment (keeping similar images together) and uniformity (preserve the maximum amount of information)

SimCLR architecture

How does SimCLR work?

The architecture consists of the following building blocks:

5. The key in this step is the contrastive loss function that is used for contrastive prediction. Its job is to identify other positive images in a dataset. The specialized loss function used for this is **NT-Xent (the normalized temperature-scaled, cross-entropy loss)**. This loss function helps us measure how the system is learning in subsequent epochs

SimCLR architecture

How does SimCLR work?

The architecture consists of the following building blocks:

These steps describe the SimCLR architecture and, as you may have noted, it works purely on unlabeled images. The magic of SimCLR is realized when you fine-tune it for a downstream task such as image classification. This architecture can learn features for you, and then you can use those features for any task

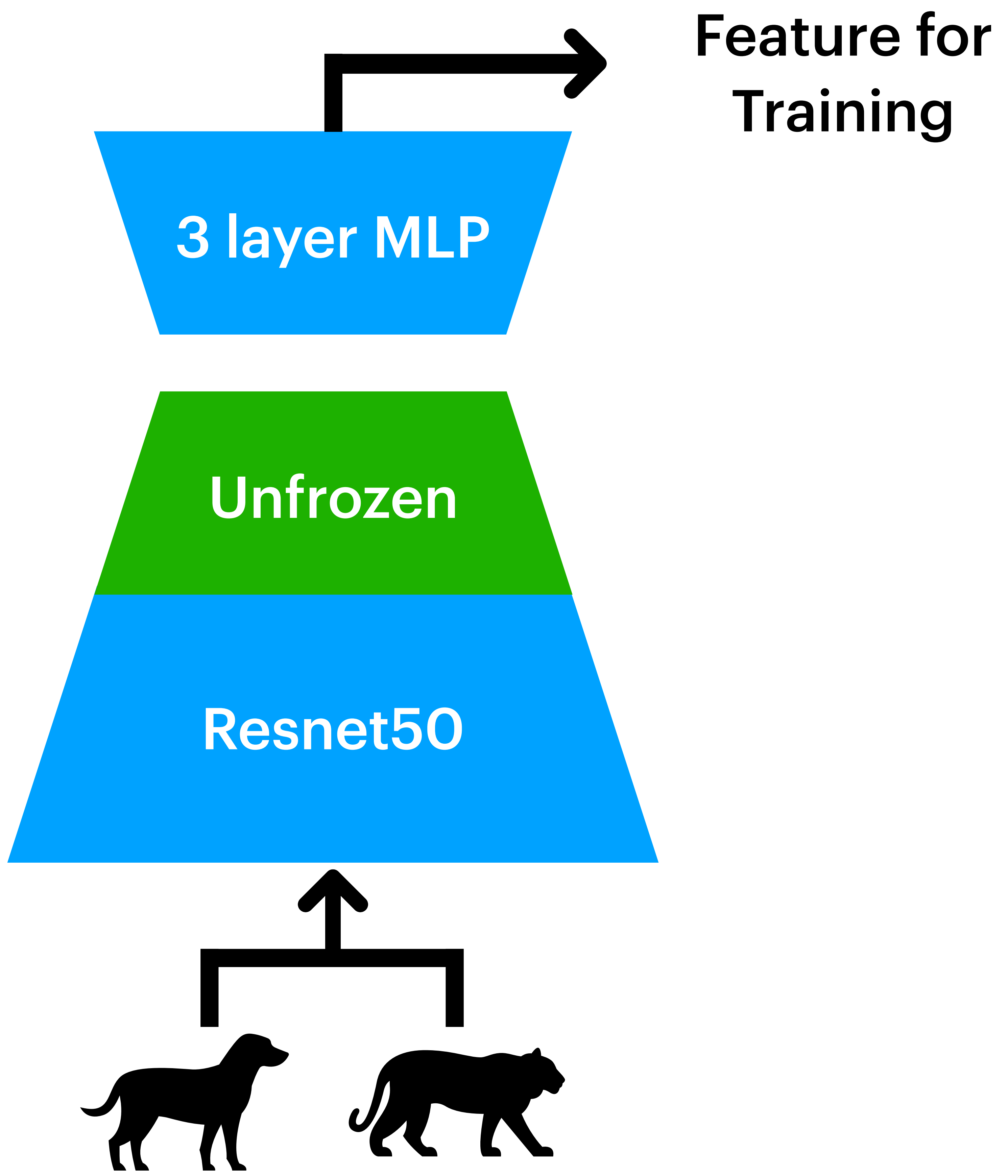
SimCLR architecture

SimCLR model for image recognition

We have seen that SimCLR can do the following:

The SimCLR architecture involves the following steps, which we implement in code:

1. Collecting the dataset
2. Setting up data augmentation
3. Loading the dataset
4. Configuring training
5. Training the SimCLR model
6. Evaluating the performance



Thank for the attention

vincenzo.pasquadibisceglie@uniba.it

https://www.linkedin.com/in/vincenzo-pasquadibisceglie-252923ba/