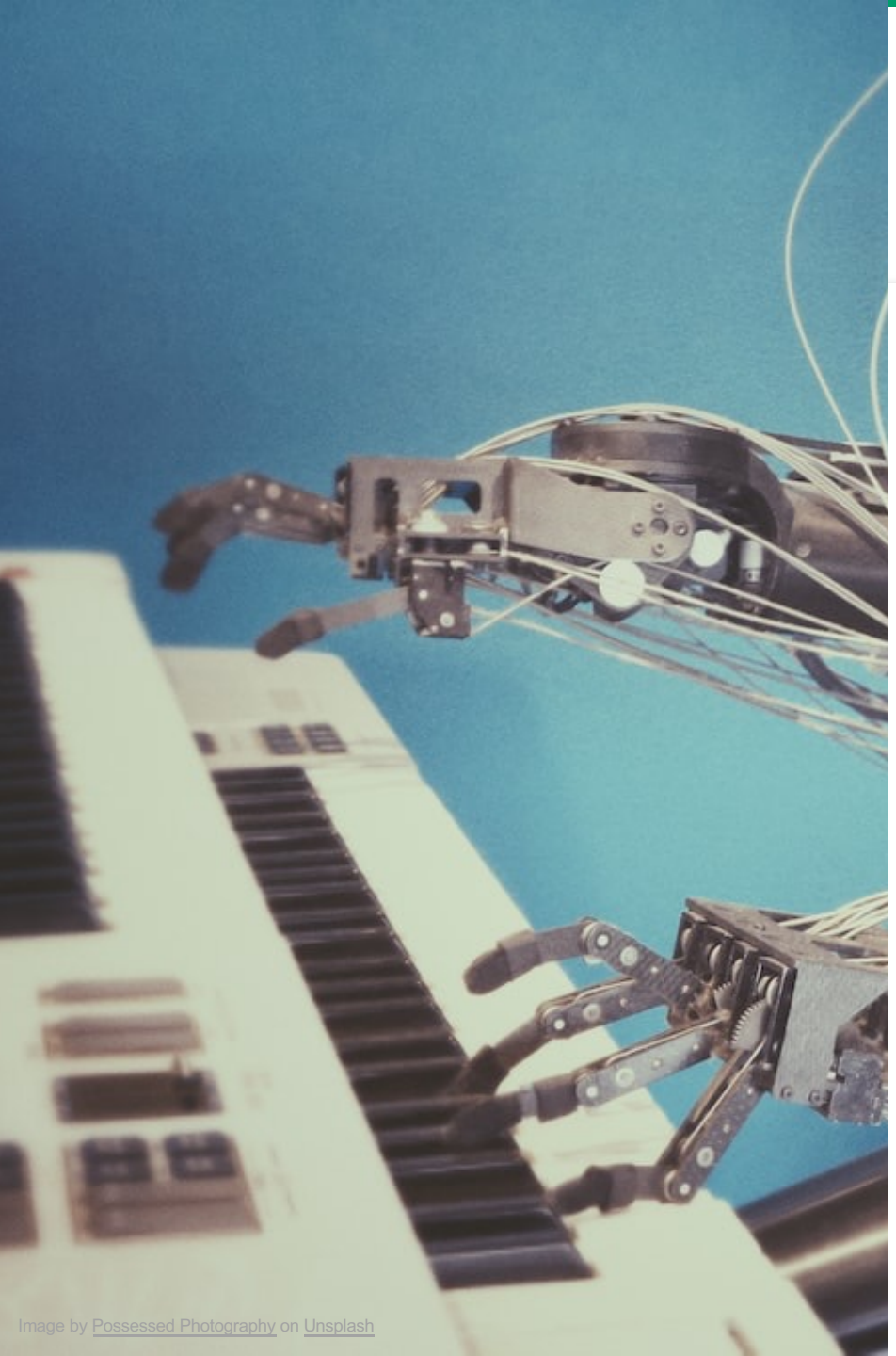


Applied Artificial Intelligence
07 - System-wide Learning

Univ.-Prof. Dr-Ing. habil. Niklas Kühl
www.niklas.xyz

University of Bayreuth

Karlsruhe Institute of Technology

TUM School of Management

Objectives
What are the learning goals of this lecture?

UNDERSTAND

Understand challenges of system-wide learning

LEARN

Get to know meta machine learning, transfer learning, federated learning and different sub-concepts

INTENSIFY

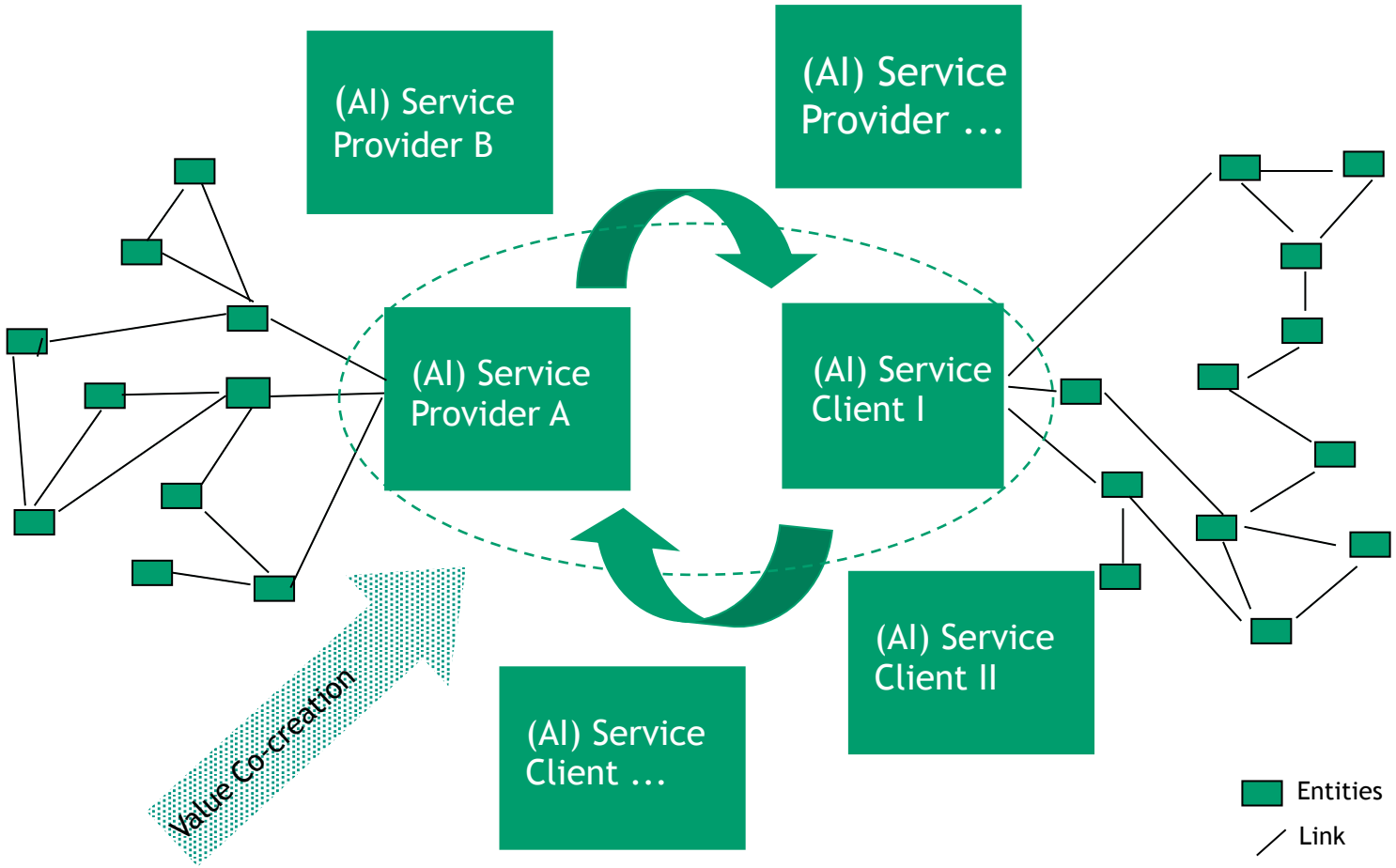
Get to know practical examples where meta learning was applied

APPLY

Be able to know when to apply which learning concept

1. AI Systems
2. System-wide Learning
3. Meta Machine Learning
4. Transfer Machine Learning
5. Federated Machine Learning

AI (Service) Systems
What are Service Systems?



A Service System comprises *"service providers and service clients working together to coproduce value in complex value chains or networks"*. [1]

[1] Spohrer et al.

1. (AI) Service Systems
2. System-wide Learning
3. Meta Machine Learning
4. Transfer Machine Learning
5. Federated Machine Learning

System-wide Learning
Data sources are often distributed across company borders.


[1]

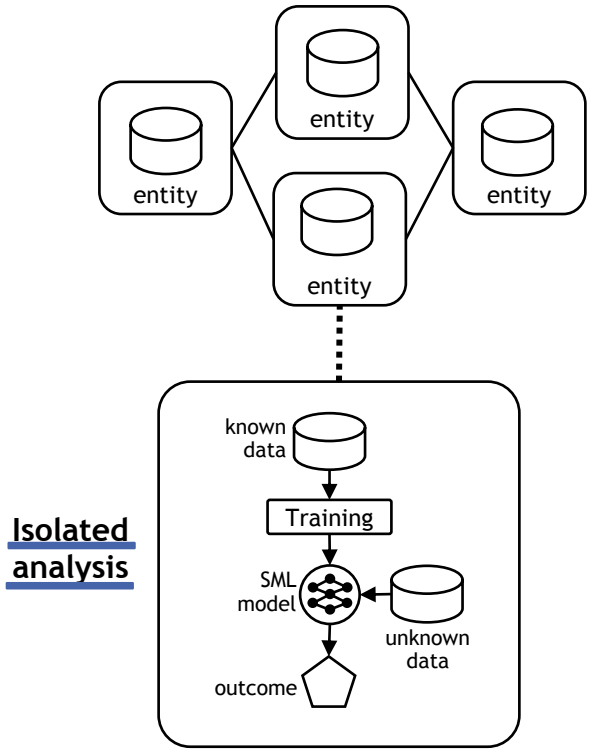
[2]

Question

What challenges can appear, when you analyze data coming from different entities?

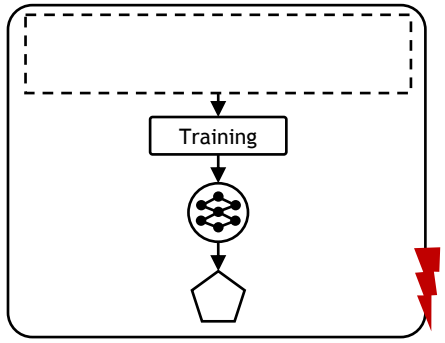
[1] Image source https://unsplash.com, free licence
[2] Source: https://www.kaggle.com/c/bosch-production-line-performance/

System-wide Learning
The isolation and distribution leads to challenges.



Isolated analysis

vs.



Centralized analysis

Data Exposure [4]

- Service systems are increasingly digitized and produce a growing amount of data which is isolated [1]
- Utilizing this data represents a major challenge for entities in service science and information systems literature [2]
- Machine learning represents a main driver for analyzing that data [3]

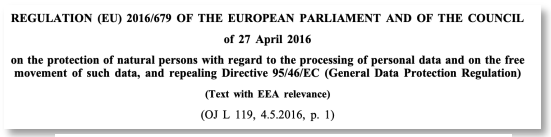
Entity of a service system — Relationship between entities — Data — SML model — Insight derived by SML model

[1] Barile & Polese, 2010, H. Chen & Storey, 2012
[2] T. H. Davenport, 2013; Böhmann et al., 2014; Porter & Heppelmann, 2015; Schüritz & Seebacher, 2017
[3] Jordan & Mitchell, 2015
[4] e.g., Miorandi et al., 2012

System-wide Learning
Challenges such as privacy, heterogeneity and volume.

1. Data privacy

Ex.: A manufacturing company does not want to make process parameters publicly available due to the danger of exposing IP [1]


REGULATION (EU) 2016/679 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL
of 27 April 2016
on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)
(Text with EEA relevance)
(OJ L 119, 4.5.2016, p. 1)

[5]

European Commission [6]

2. Data heterogeneity

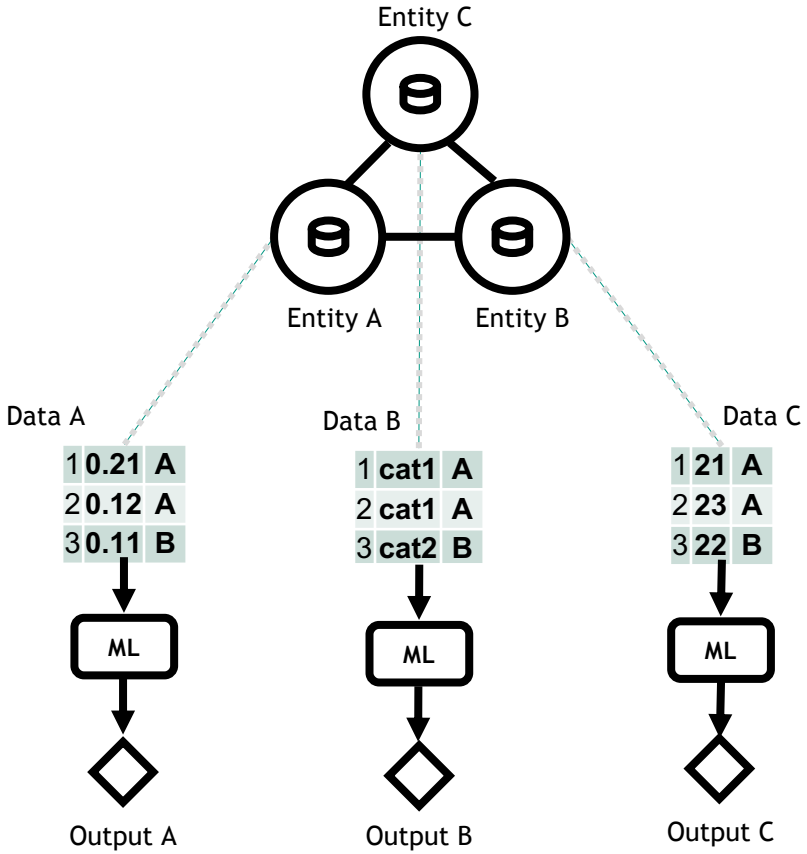
Ex.: In order to simultaneously analyze images and text, a mapping, weighting and conversion of both data sources is required. This can lead to inefficiencies and complicates the analysis [2]

3. Data velocity/volume

Ex.1: Sensors at a production line machine produce large data streams. As the number of sensors and their resolution increases, a transmission might not be possible. [3]
Ex. 2: Large data sets, such as image data bases, can not be exchanged without limitations, especially if the capacity of the processing unit doesn't allow it [4]

[1] ibmbigdatahub.com
[2] bbva.com
[3] tdwi.org
[4] kdnuggets.com
[5] eur-lex.europa.eu/legal-content/
[6] commission.europa.eu

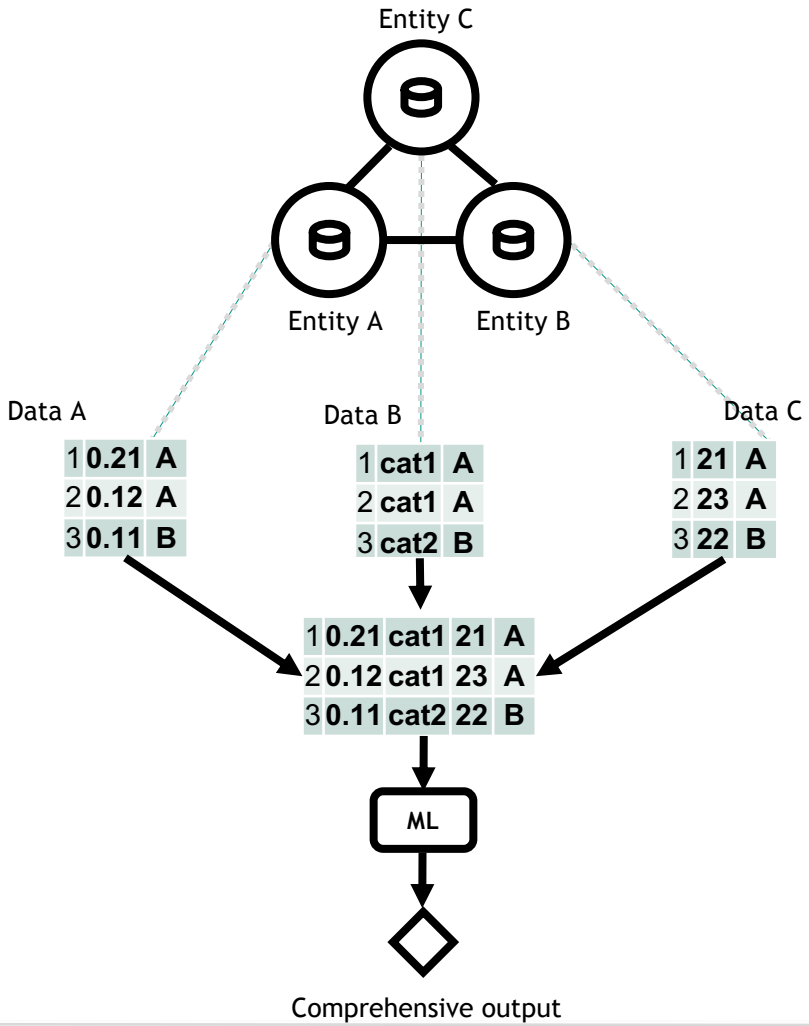
System-wide Learning
Exchanging no data - Isolated Analysis



Every entity analyzes its data separately:

+ No exposure of data/information
+ Decrease of processing complexity due to less heterogeneous data and smaller data set
– Outputs are based on incomplete information

System-wide Learning
Full exchange of all data – a centralized analysis

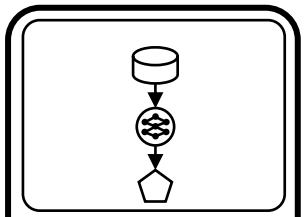


Entities fully exchange their data for a comprehensive analysis:

+ Outputs are based on complete information
– exposure of data/information
– Increase of processing complexity due to heterogeneous data and bigger data set
– Larger data sets are required to be transferred; problematic especially with data streams

System-wide Learning
Isolated vs. centralized: Two extremes in terms of data exchange

No data exchange ⟷ Full data exchange



Isolated analysis

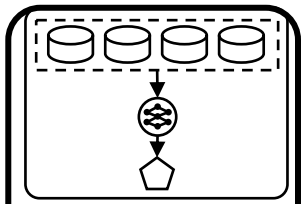
Use only a subset of available data of a system

1. Data privacy

2. Data heterogeneity

3. Data velocity/volume

How can we overcome those challenges and still be able to perform a comprehensive analysis?



Centralized analysis

Centralize (and expose) all data

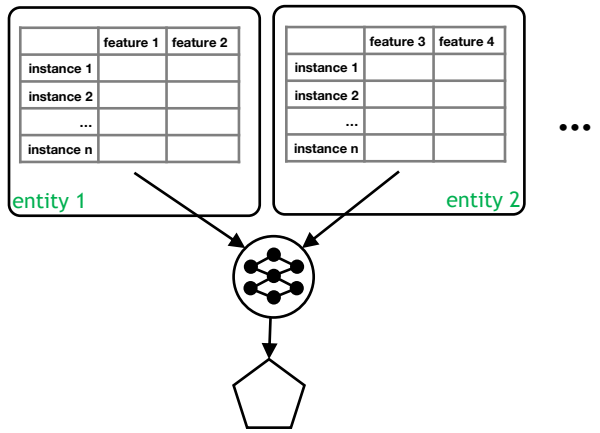
1. (AI) Service Systems
2. System-wide Learning
3. **Meta Machine Learning**
4. Transfer Machine Learning
5. Federated Machine Learning

Meta Machine Learning
Systemic challenges of ML in AI service systems

Aggregated data of a system (simplified)

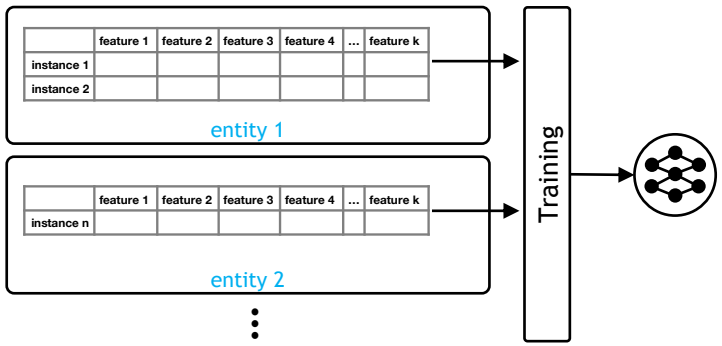
	feature 1	feature 2	feature 3	feature 4	...	feature k	
instance 1	x_{11}	x_{21}	x_{31}	x_{41}	...	x_{k1}	entity 1
instance 2	x_{12}	x_{22}	x_{32}	x_{42}	...	x_{k2}	
...	...	...	...	...	...	...	entity ...
instance n	x_{1n}	x_{2n}	x_{3n}	x_{4n}	...	x_{kn}	
	entity 1		entity 2			entity ...	

Challenge A:
Predicting outcomes based on distributed features



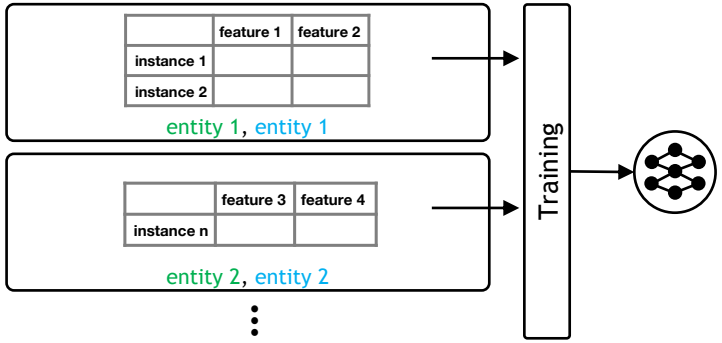
e.g., Demirkan & Delen (2013), G. Liu et al. (2015)

Challenge B:
Training models based on distributed instances



e.g. Brisimi et al. (2018), Moore et al. (2016)

Challenge C:
Training models based on distributed features and distributed instances



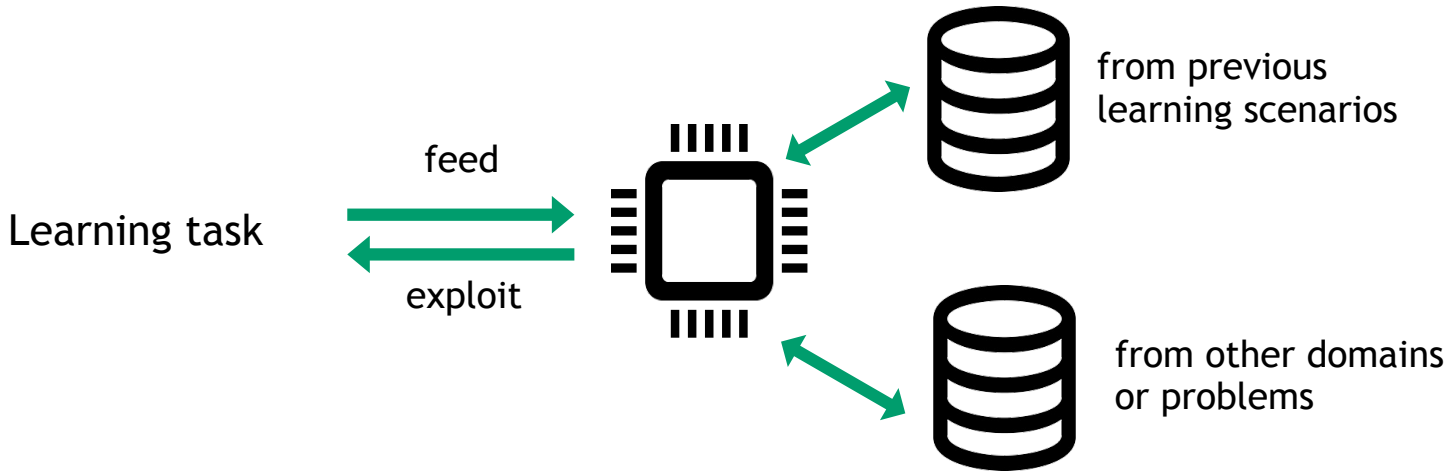
e.g. Brisimi et al. (2018), Moore et al. (2016)

Meta Machine Learning
Definition

Meta learning describes a concept in machine learning, where a learning algorithm can utilize metaknowledge gained through experience:

Meta Learning

A meta learning system must include a learning subsystem, which adapts with **experience**. Experience is gained by **exploiting metaknowledge** extracted in a previous learning episode on a single dataset and/or from different domains or problems.



[1] Lemke, Budka and Gabrys (2013)

Meta Machine Learning
Our focus lays on base learner combinations (ensembles).

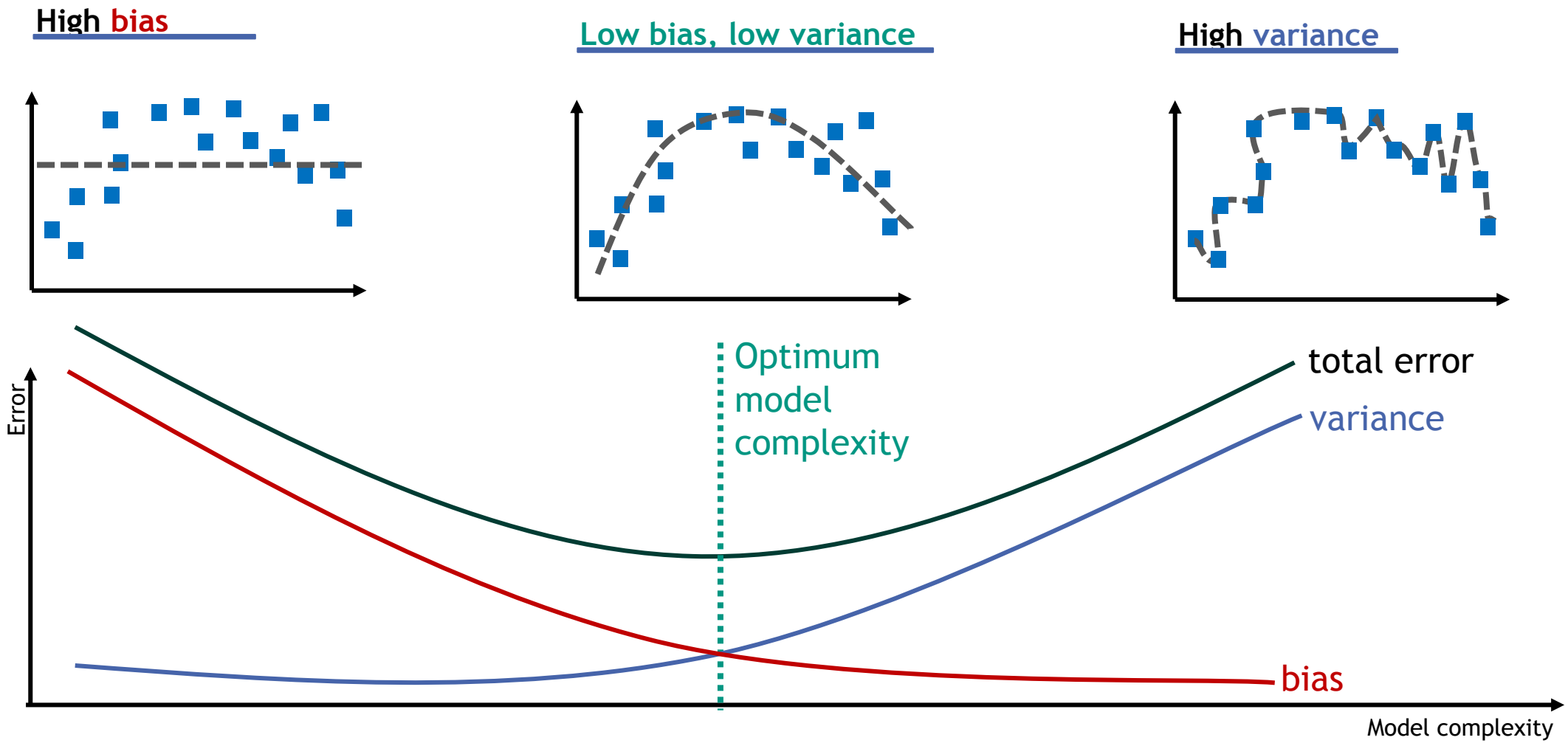
- Model combination is often used when several applicable algorithms for a problem are available.
- Instead of selecting a single algorithm for a problem, the risk of choosing the wrong one can be reduced by combining all or a subset of the available outcomes.

1 | **Bagging** (Breiman,1996)

2 | **Boosting** (Freund and Schapire,1997)

3 | **Stacked generalization, or "stacking"** (Wolpert,1992)

Meta Machine Learning
Recap: Bias Variance trade-off

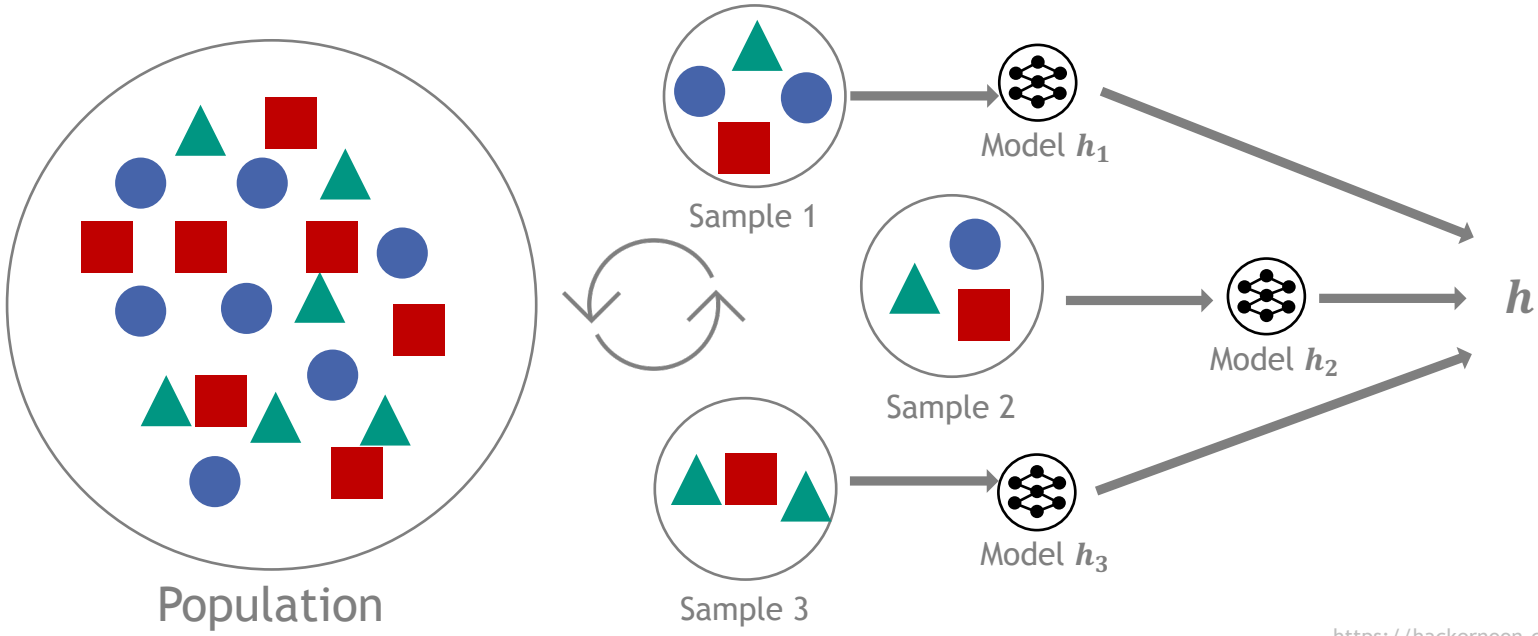


Meta Machine Learning: 1 | Bagging

Combine multiple classifiers to create a composite classifier.

Bagging (Bootstrap Aggregating)

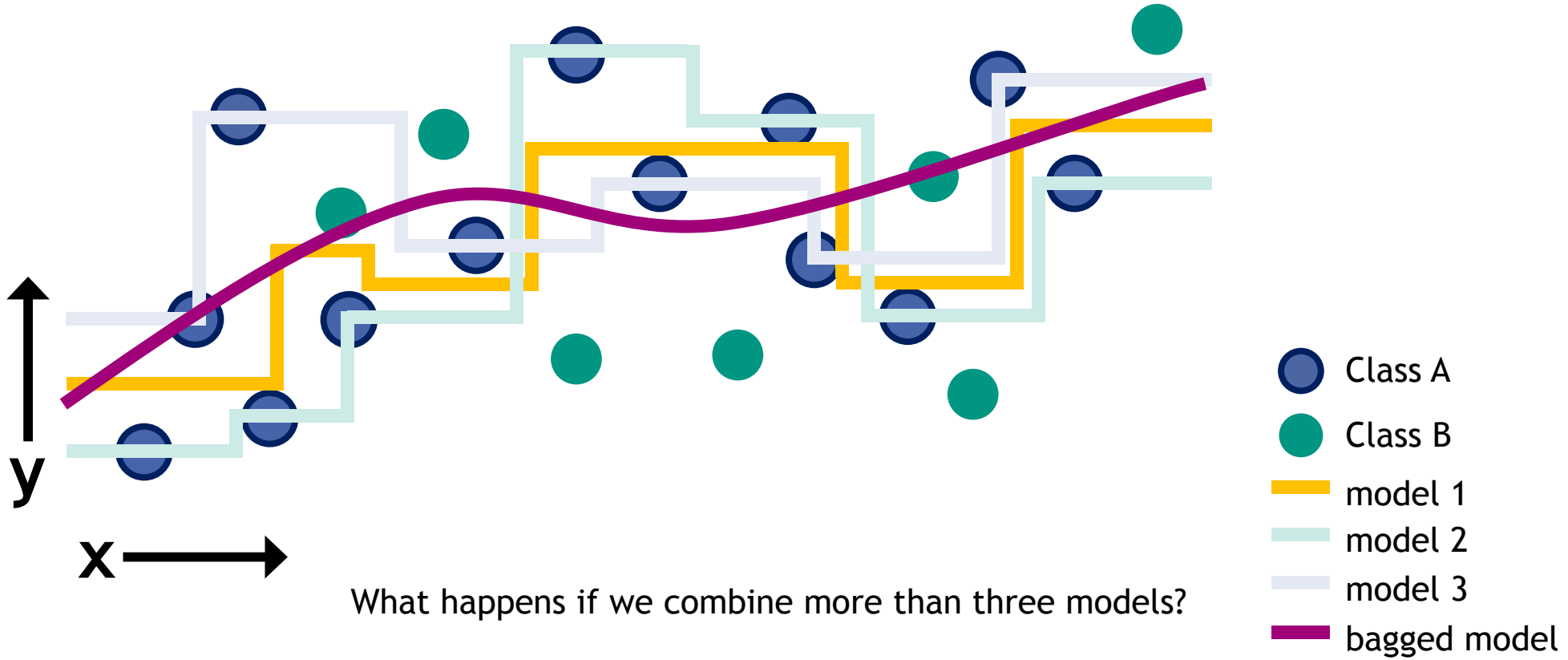
decreasing the variance of a prediction
Generating additional data from original dataset by combining with repetition
By increasing the size of training set variance is decreased (no increase of predictive force) → narrowly tune prediction



https://hackernoon.com/how-to-develop-a-robust-algorithm-c38e08f32201

Meta Machine Learning: 1 | Bagging

The combined bagged model is in between the individual models



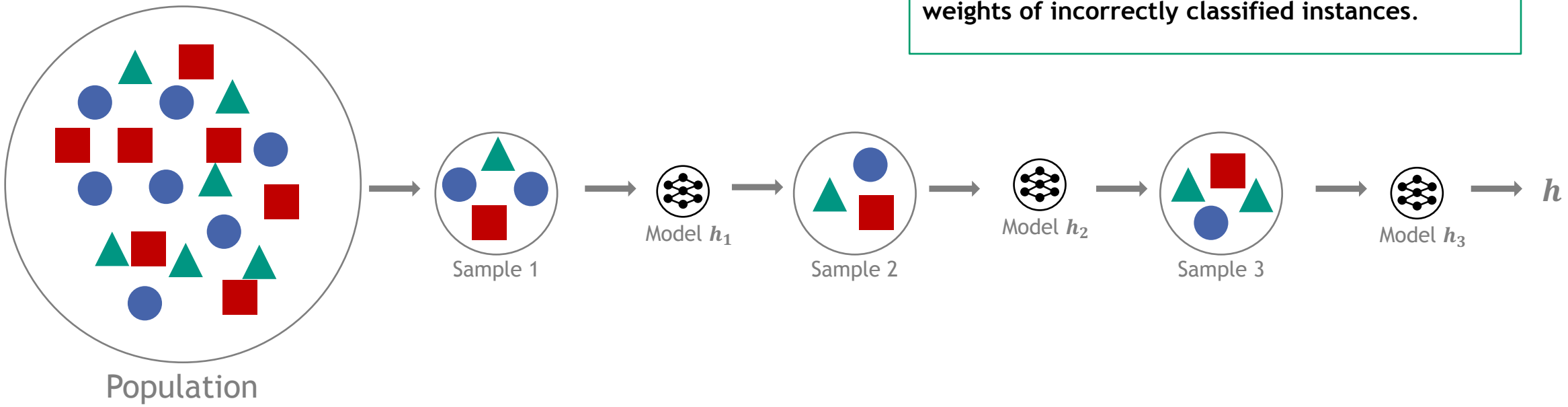
What happens if we combine more than three models?

Rokach (2009) ; Breiman (1996)

Meta Machine Learning: 2|Boosting
Combine classifications into a stringer model by reweighting

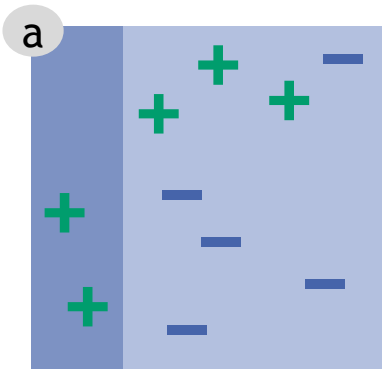
AdaBoost

AdaBoost is a popular boosting algorithm that works by decreasing the weight of correctly classified instances while **increasing the weights of incorrectly classified instances.**

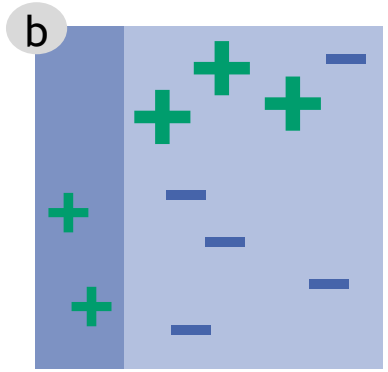


Rokach (2009) ; Viola and Jones (2002) ; Freund and Schapire (1996)

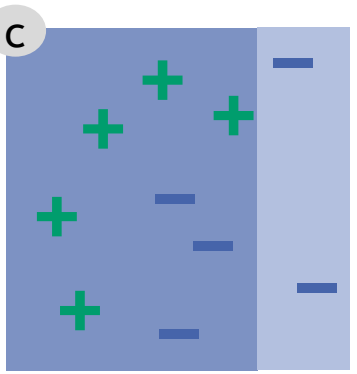
Meta Machine Learning: 2|Boosting
Combine classifications into an optimized model by reweighting



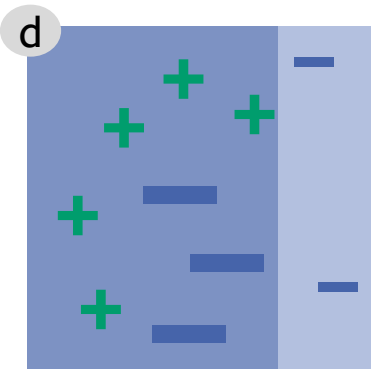
Initial classification



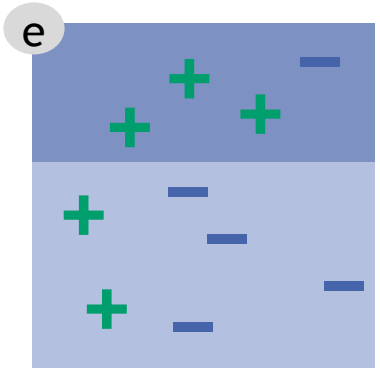
Increase weight of false classification



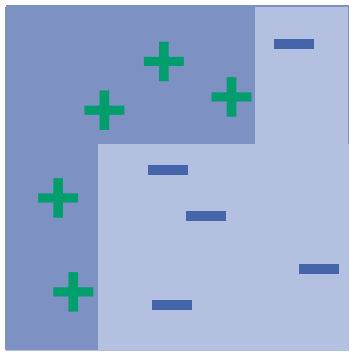
New classification regarding the weights



Increase weight of false classification



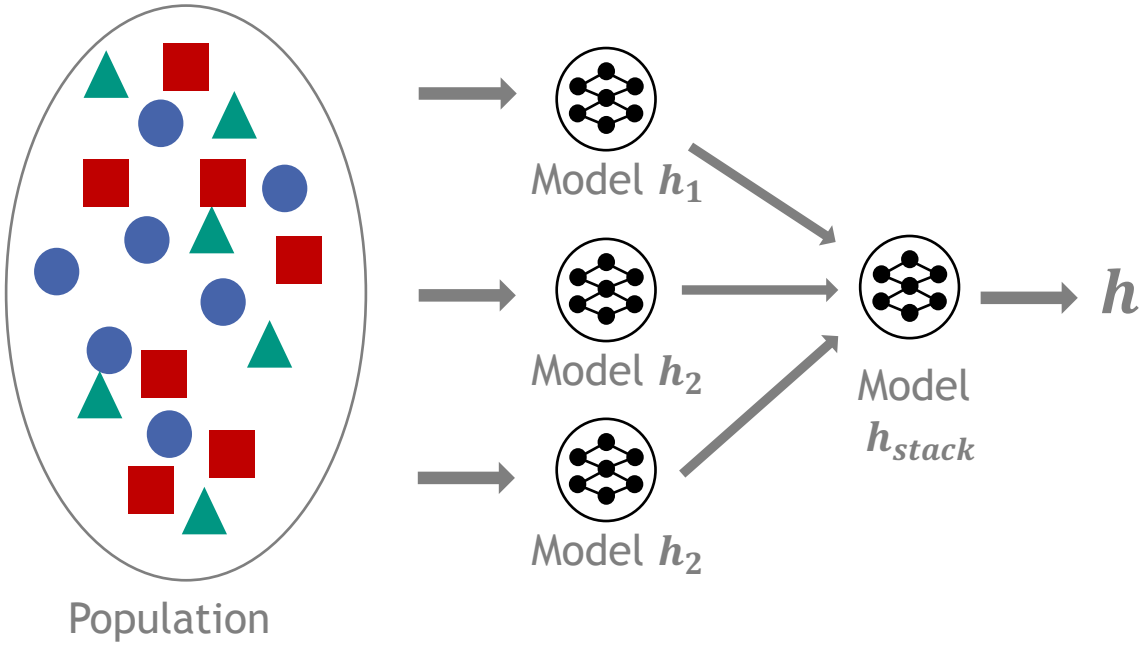
New classification regarding the weights



Combined classifier

Rokach (2009) ; Viola and Jones (2002) ; Freund and Schapire (1996)

Meta Machine Learning: 3 | Stacking
Combine multiple classifiers sequentially



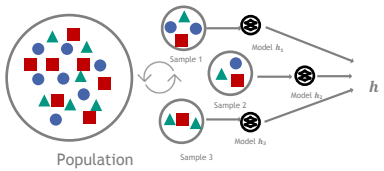
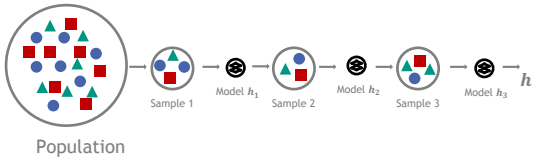
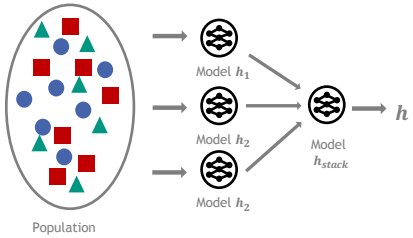
Stacking

- Stacked generalization (or stacking) employs a **second layer of machine learning to combine (heterogeneous) classifiers**
- The approach is that input data is modified from the original input data set as the stacking approach uses the **prediction of the other classifiers as an input** for the new model
- Aims for increasing the performance compared to selecting the single best classifier

→ Model predictions are used to create a new more capable model

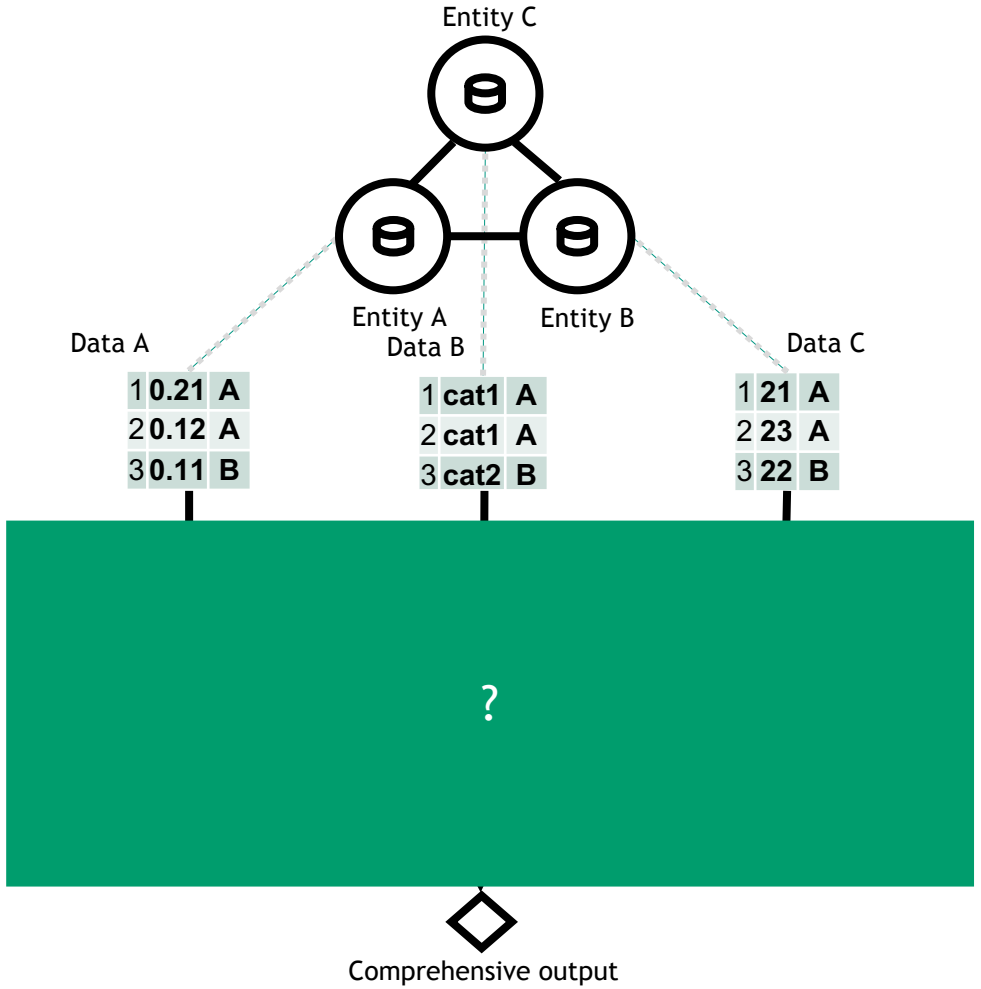
Rokach (2009) ; Džeroski and Ženko (2004)

Meta Machine Learning
Comparison of bagging, boosting and stacking

	Bagging	Boosting	Stacking
			
Partitioning of data into subsets	Randomly selected	Mis-classified samples have a higher probability to be re-selected	Various
Function that are used to combine single models	(weighted) average / majority vote	(Weighted) average / majority vote	Second layer of machine learning (e.g. logistic regression)
Ensemble type	Parallel ensemble	Sequential ensemble	Both
Aim	Decrease variance	Decrease bias	both

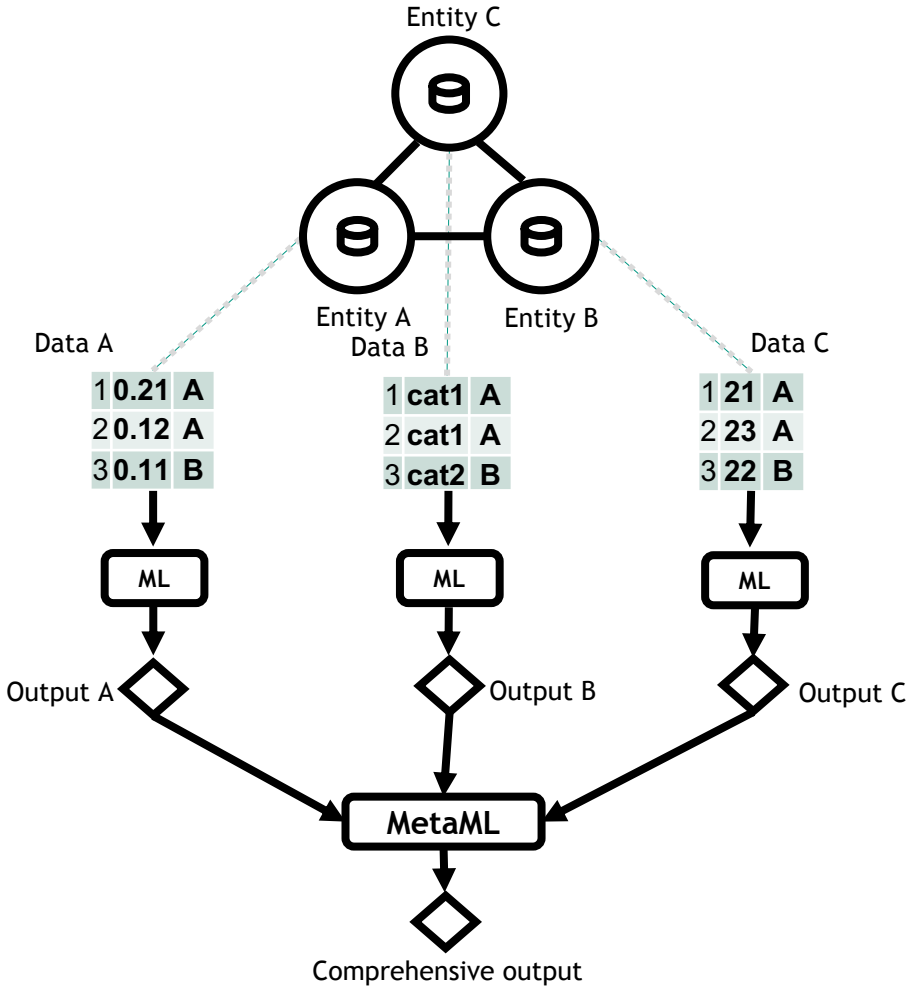
Meta Machine Learning: Stacking

Now: How can we use Meta ML (stacking) to solve our problem?



Is there an option “in-between”?

Meta Machine Learning: Stacking
Step-wise meta analysis of distributed data sources



Every entity analyzes its data separately:

+ No exposure of data/information
+ Decrease of processing complexity due to less heterogeneous data and smaller data set
– Outputs are based on incomplete information

Meta Machine Learning | Example 1
Quality prediction in industrial manufacturing case

https://www.kaggle.com/c/bosch-production-line-performance/

Meta Machine Learning | Example 1
Data is unbalanced and very high dimensional

Setting

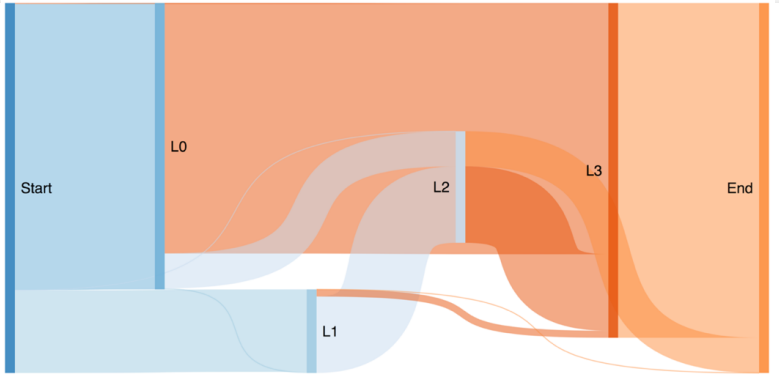
Each part goes through a varying sequence of 4 lines and their respective stations (52 stations in total)

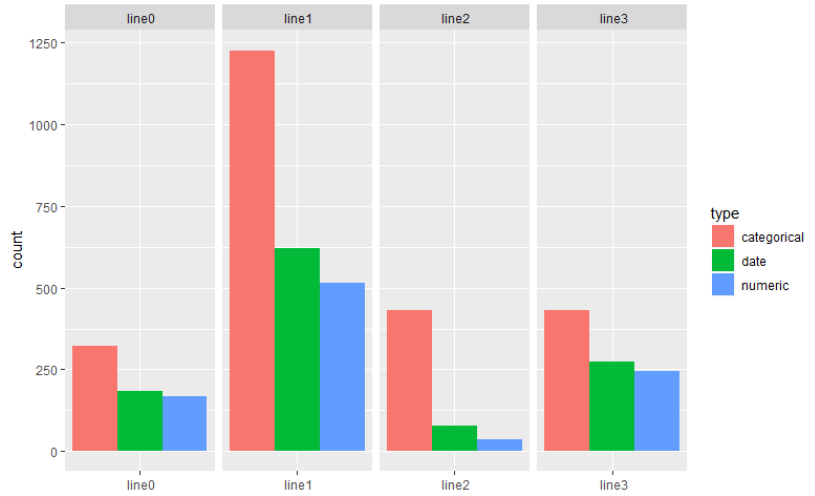
Goal

Predict future failures in time and intervene to reduce the number of faulty parts

Data

- **968** numeric features, **1.156** date features, **2.140** categorical features
- Contains 1,183,747 parts ("no scrap") or not ("scrap")
- Highly imbalanced data set: only **6,879 parts** labelled as faulty (**failure rate of 0.58 %**)

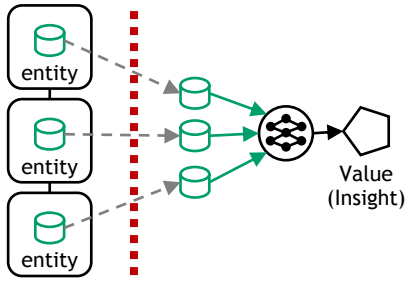




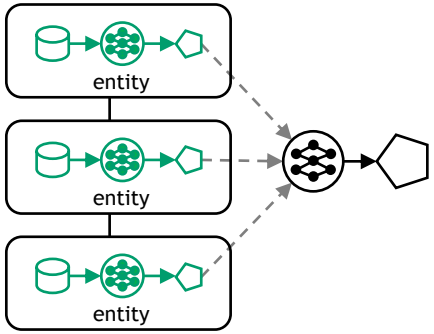
Hirt, R., Kühl, N., Martin, D., & Satzger, G. "Enabling inter-organizational analytics in business networks through meta machine learning" Information Technology and Management, Springer, 2023

Meta Machine Learning | Example 1
Evaluate the meta-ML framework on three scenarios

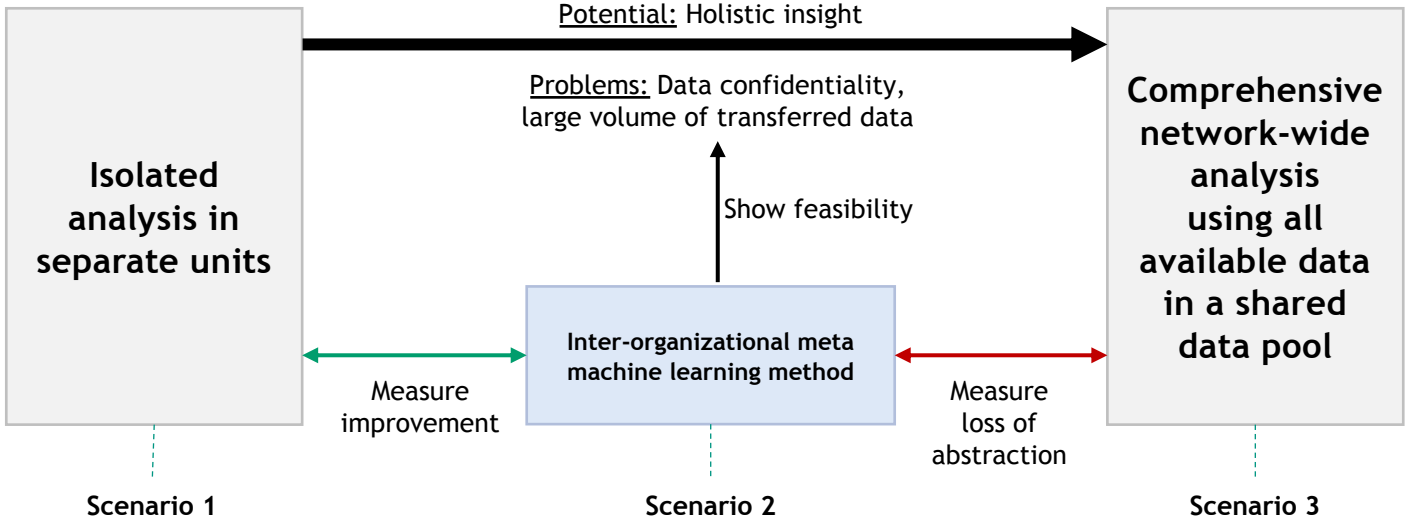
Problem: Analyzing distributed components



Tentative Design: Meta machine learning method



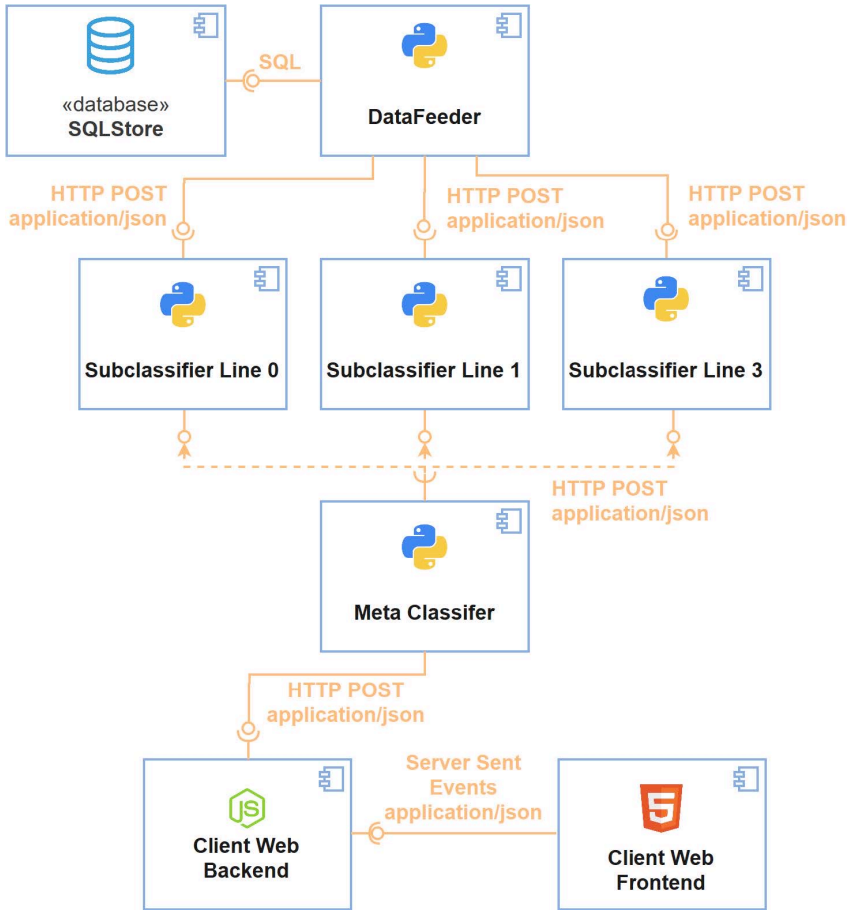
Evaluation



Hirt, R., Kühl, N., Martin, D., & Satzger, G. "Enabling inter-organizational analytics in business networks through meta machine learning" Information Technology and Management, Springer, 2023

Meta Machine Learning | Example 1

Framework enables a continuous analysis of distributed data.



Hirt, R., Kühl, N., Martin, D., & Satzger, G. “Enabling inter-organizational analytics in business networks through meta machine learning” Information Technology and Management, Springer, 2023

Meta Machine Learning | Example 1
Front-end hands-on experience

#	Predictor Line 0	Predictor Line 1	Predictor Line 3	Cognitive Predictor
18251	n.a.	false	false	false
10672	n.a.	false	false	false
12272	false	n.a.	false	false
11039	false	n.a.	true	false
11368	false	n.a.	true	false
4646	false	n.a.	true	false
3620	false	n.a.	true	false
17691	false	n.a.	true	false
18149	false	n.a.	true	false
11511	false	n.a.	true	false
7840	false	n.a.	true	false
12475	n.a.	false	false	false
1345	false	n.a.	true	false
15672	false	n.a.	true	false
3069	false	n.a.	true	false

Hirt, R., Kühl, N., Martin, D., & Satzger, G. “Enabling inter-organizational analytics in business networks through meta machine learning” Information Technology and Management, Springer, 2023

Meta Machine Learning | Example 1
Challenges of analyzing distributed data sources

1. Data privacy

Different legal entities might not be allowed or willing to exchange data across entity borders

2. Data heterogeneity

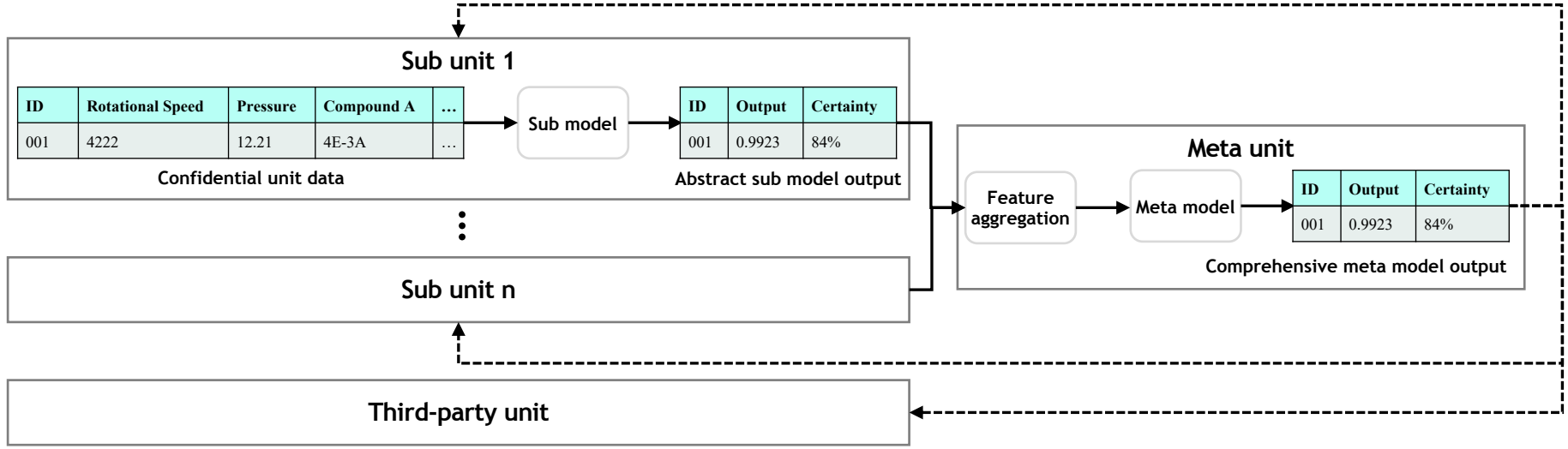
Data originating from different entities might be of different shape, thus a comprehensive analysis might be problematic

3. Data velocity/volume

A data stream is produced faster than it can be transmitted, or the transmission of a large data source is not feasible due to technical limitations

Meta Machine Learning | Example 1
Step-wise analysis preserves data confidentiality.

1.
Data privacy



- Sub entities/units only transfer abstracted information to a meta entity/unit
- The meta unit processes all incoming abstracted sub outputs to make a holistic meta prediction
- At no point, entities exchange or expose raw data outside entity borders
  → data confidentiality/privacy is preserved

Hirt, R., Kühl, N., Martin, D., & Satzger, G. "Enabling inter-organizational analytics in business networks through meta machine learning" Information Technology and Management, Springer, 2023

Meta Machine Learning | Example 1
Results

1.
Data privacy

No raw data exchange?

Yes, as only abstract prediction outputs are transferred.

How well does our method perform in comparison to meaningful scenarios?

Matthews correlation coefficient: $$\mathrm{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

	Scenario 1		Scenario 2		Scenario 3
	Isolated analysis in separate units	improvement +17.57 to +31.43%	meta machine learning method	loss of abstraction -4,83%	Analysis using all available data
MCC	0.1935 - 0.2326		0.2822		0.2965

Meta Machine Learning | Example 1
Challenges of analyzing distributed data sources

	1. Data privacy 	2. Data heterogeneity 	3. Data velocity/volume 
How meta machine learning copes with challenge	**Solution:** ▪ As only abstracted prediction outputs are being exposed outside an entities' border, data privacy and confidentiality is preserved. ▪ The abstract prediction outputs can be processed.		

Meta Machine Learning | Example 1

By only exchanging abstracted output, we can drastically reduce the required amount of transferred data

3.
Data velocity/volume

Notation

s	Space a feature requires
k	Sub entities
n	Number of input features
m	Number of output features

Assumptions

1. All features require the same amount of space **s**
2. There are **k** subunits with varying number of features

	Scenario "no exchange"	Scenario "meta machine learning"	Scenario "full exchange"
Formula	0	$k * m * s$	$\sum_{i=1}^{k} n_i * s$
Ex.: 4 entities each has 100 features sub outputs = 3 10 Bytes / feature	0	4*3*10 = 120 Bytes	4*100*10 = 4000 Bytes

Meta Machine Learning | Example 1
Challenges of analyzing distributed data sources

	1. Data privacy 	2. Data heterogeneity 	3. Data velocity/volume 
How meta machine learning copes with challenge	**Solution:** ▪ As only abstracted prediction outputs are being exposed outside an entities' border, data privacy and confidentiality is preserved. ▪ The abstract prediction outputs can be processed.		**Solution:** ▪ By only transferring prediction outputs, the amount of transferred data is significantly reduced. ▪ Note: this principle is basis to edge or fog computing.

Meta Machine Learning | Example 2
User profiling on Social Media

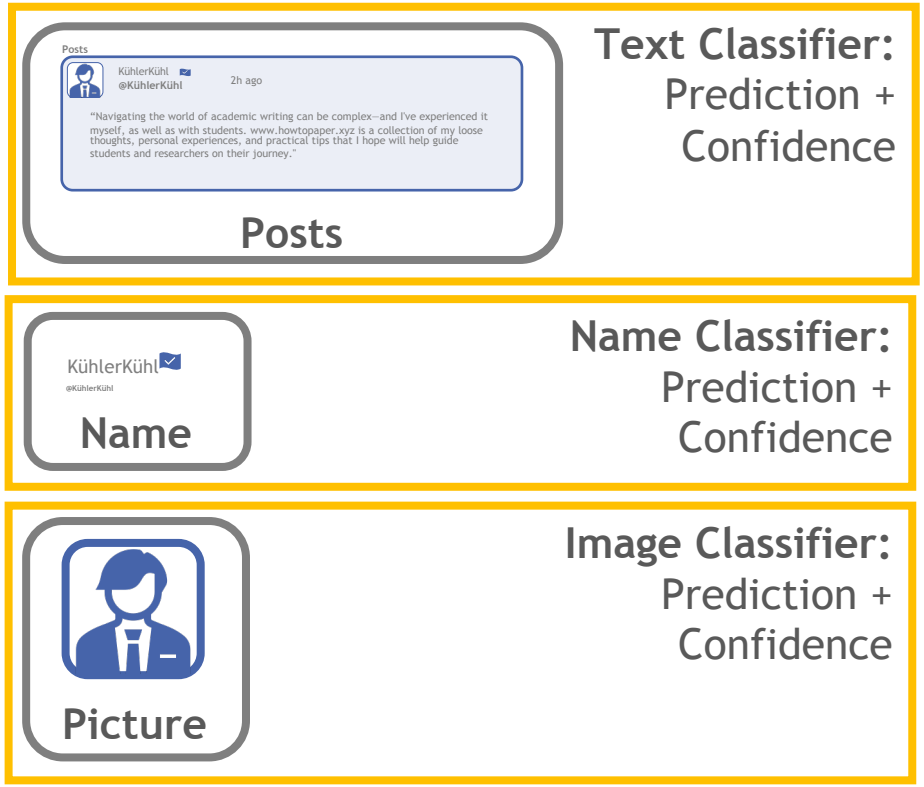
2.
Data
heterogeneity

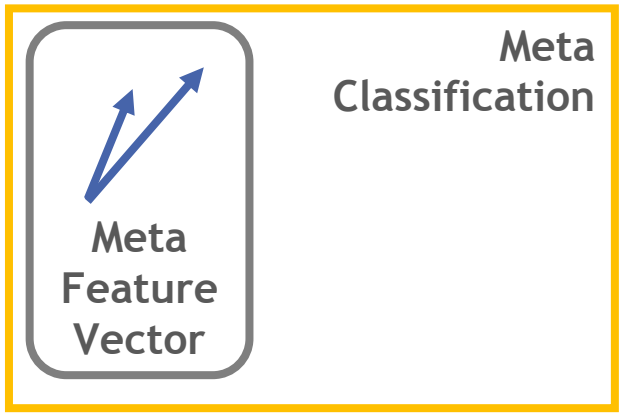
D. Hirt, R., Kühl, N., & Satzger, G. "Cognitive computing for customer profiling: meta classification for gender prediction" Electronic Markets, Springer, 2019

Meta Machine Learning | Example 2

Heterogeneous data sources can be processed flexibly:
The example of classifying Social Media users' demographics.

2.
Data
heterogeneity





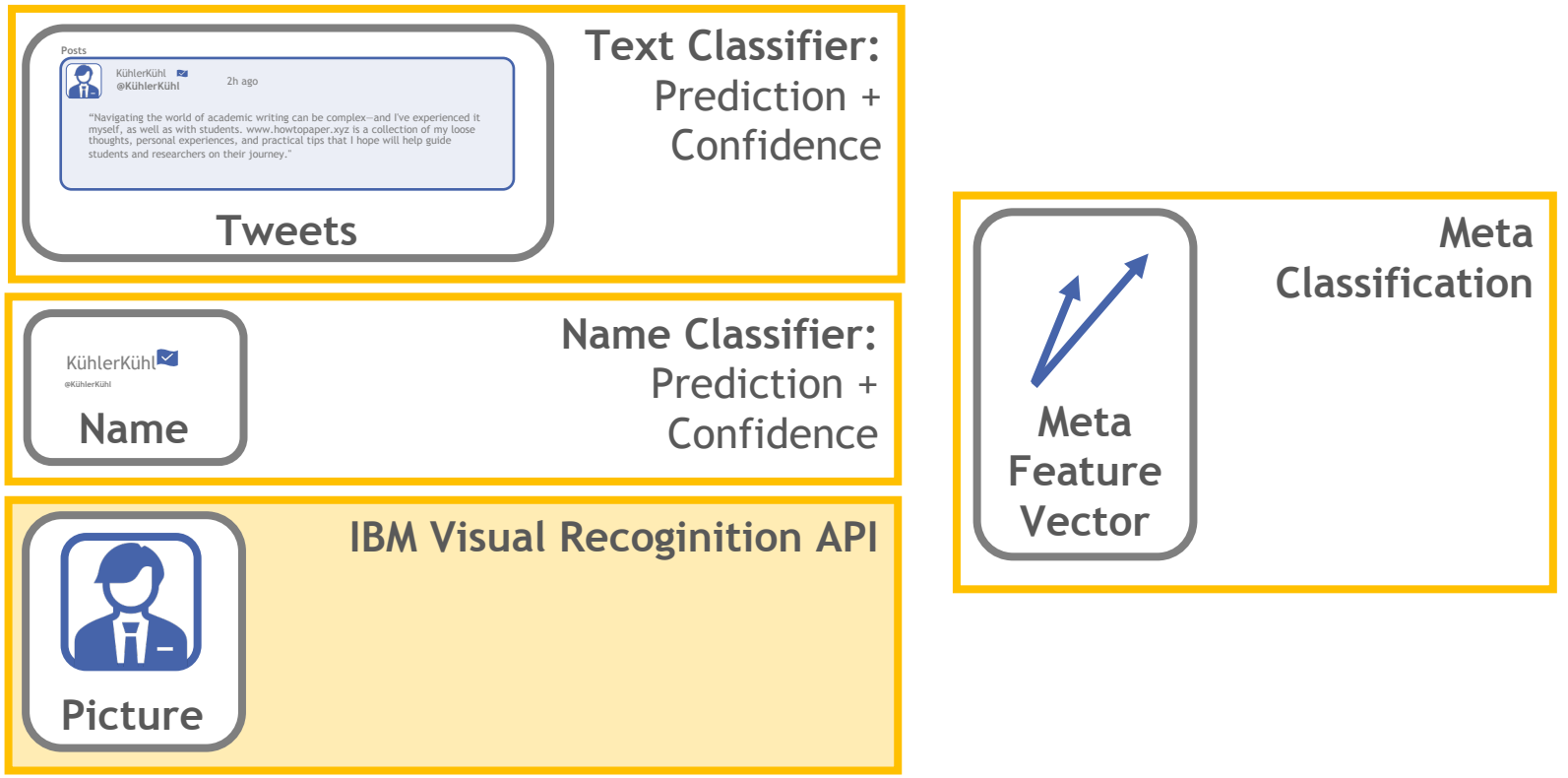
D. Hirt, R., Kühl, N., & Satzger, G. "Cognitive computing for customer profiling: meta classification for gender prediction" Electronic Markets, Springer, 2019

Meta Machine Learning | Example 2

A meta classification makes it possible to include 3rd party classifiers that can be remote.

2.
Data heterogeneity

- Include a 3rd party classifier:
  IBM Visual Recognition API
- The output of API serves as an additional input for meta classifier



We reach an F1-score of 81.47% (+2.30%), a recall of 81.53% (+2.48%) and a precision of 81,54% (+1.89%).

D. Hirt, R., Kühl, N., & Satzger, G. "Cognitive computing for customer profiling: meta classification for gender prediction" Electronic Markets, Springer, 2019

Meta Machine Learning | Example 2
Challenges of analyzing distributed data sources

	1. Data privacy 	2. Data heterogeneity 	3. Data velocity/volume 
How meta machine learning copes with challenge	**Solution:** • As only abstracted prediction outputs are being exposed outside an entities' border, data privacy and confidentiality is preserved. • The abstract prediction outputs can be processed.	**Solution:** • For each data source type, a specific model can be tailored and its performance optimized • For each entity: expert knowledge can be incorporated into a specific model that was otherwise not accessible.	**Solution:** • By only transferring prediction outputs, the amount of transferred data is significantly reduced. • Note: this principle is basis to edge or fog computing.

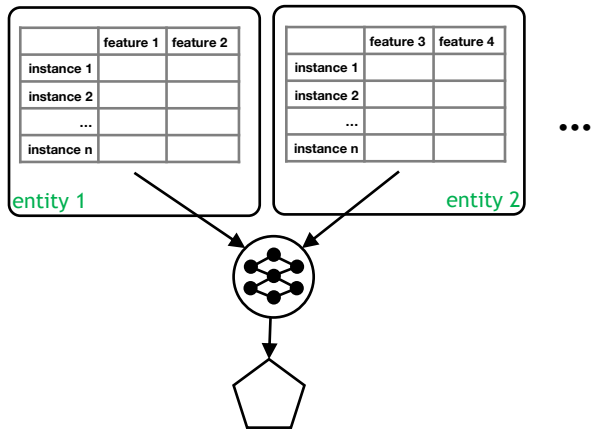
1. (AI) Service Systems
2. System-wide Learning
3. Meta Machine Learning
4. **Transfer Machine Learning**
5. Federated Machine Learning

Meta Machine Learning
Systemic challenges of ML in AI service systems

Aggregated data of a system (simplified)

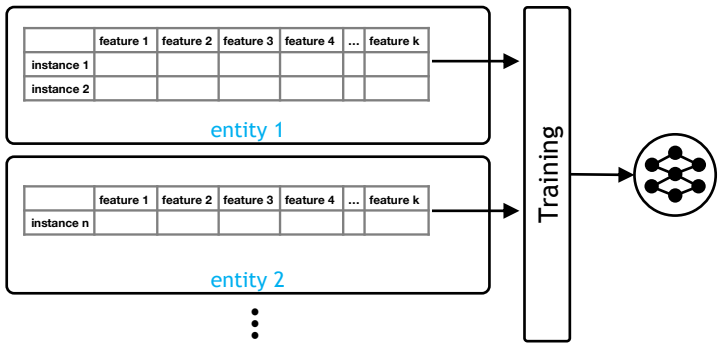
	feature 1	feature 2	feature 3	feature 4	...	feature k	
instance 1	x_{11}	x_{21}	x_{31}	x_{41}	...	x_{k1}	entity 1
instance 2	x_{12}	x_{22}	x_{32}	x_{42}	...	x_{k2}	
...	...	...	...	...	...	...	entity ...
instance n	x_{1n}	x_{2n}	x_{3n}	x_{4n}	...	x_{kn}	
	entity 1		entity 2			entity ...	

Challenge A:
Predicting outcomes based on distributed features



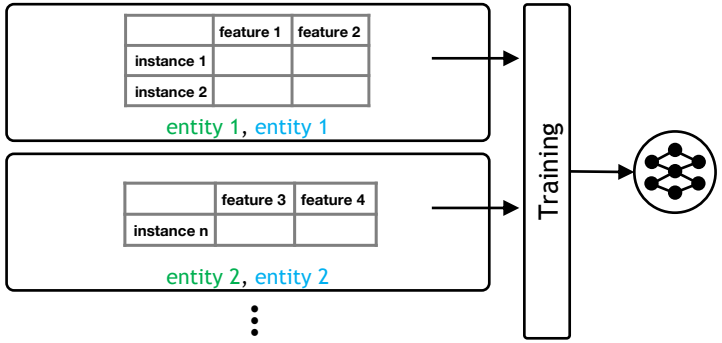
e.g., Demirkan & Delen (2013), G. Liu et al. (2015)

Challenge B:
Training models based on distributed instances



e.g. Brisimi et al. (2018), Moore et al. (2016)

Challenge C:
Training models based on distributed features and distributed instances

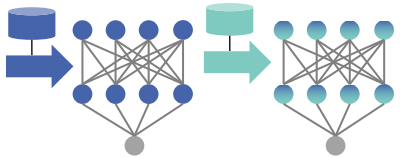


e.g. Brisimi et al. (2018), Moore et al. (2016)

Transfer Machine Learning

Transfer knowledge from a related task to a new task

Transfer learning is the improvement of learning in a new task through the **transfer of knowledge from a related task** that has already been learned.

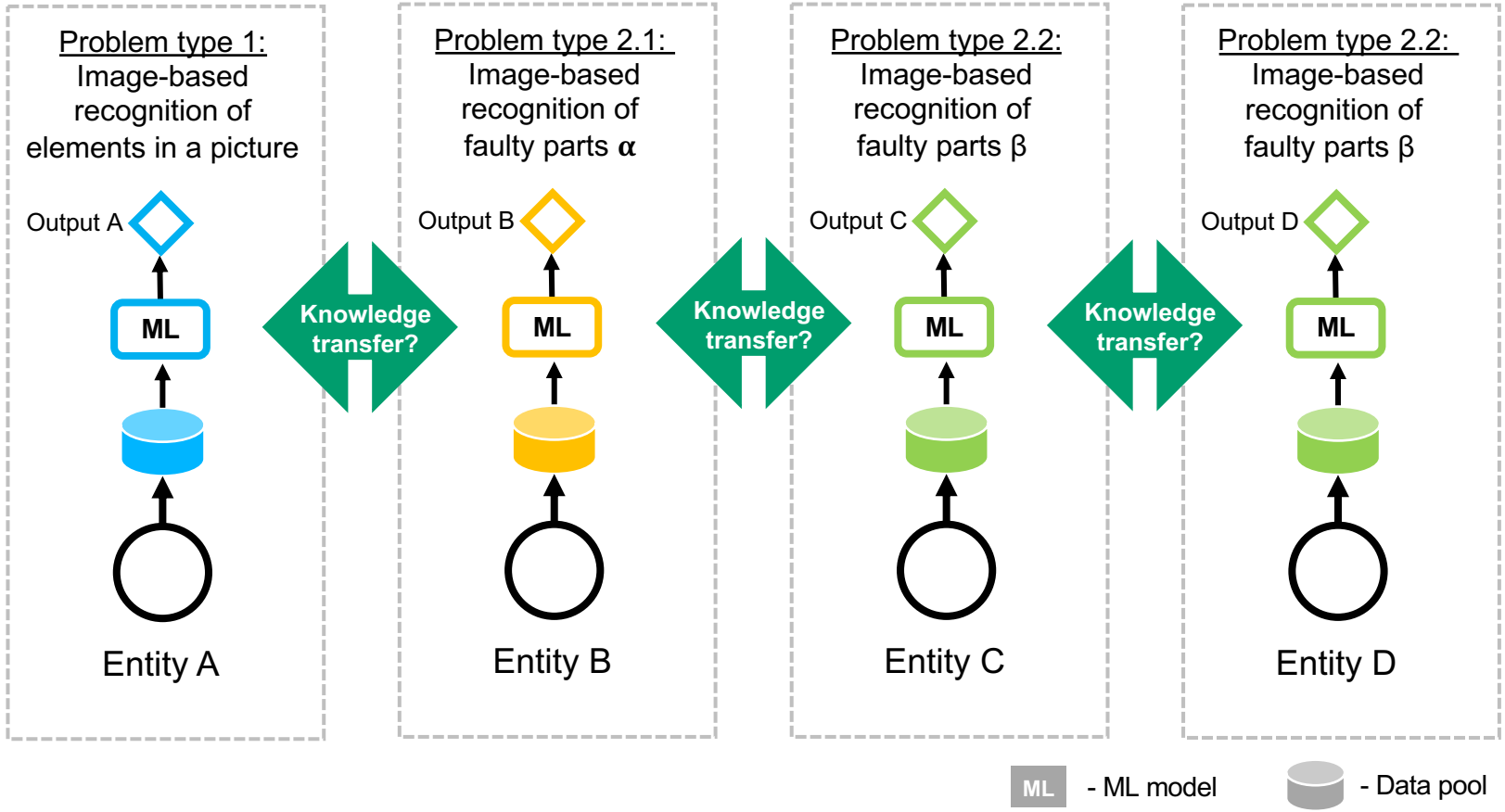


Transfer learning for object recognition

Transfer learning for natural language processing

...

Transfer Machine Learning
Those analytical tasks have different “levels” of similarity



How can analytical knowledge be exchanged across entities?

Transfer Machine Learning
Formal notation for transfer machine learning

Domain D

X — feature Space

$\mathrm{P}(X)$ — probability distribution

Learning Task T

Y — set of all labels

$f(\cdot)$ — predictive function

- Domain D consists of two components $D = \{X, \mathrm{P}(X)\}$:
  a feature Space X, and a marginal probability distribution $\mathrm{P}(X)$ where $X = \{x_1, \ldots, x_n\} \in \chi$
- Task $T = \{Y, f(\cdot)\}$ is composed of the set of all labels Υ and an objective predictive function $f(\cdot)$
- Predictive function $f(\cdot)$ is not observed but derived by training data $\{x_i, y_i\}$

Transfer Machine Learning

Transfer Learning describes a concept in machine learning where knowledge from one domain is transferred to another.

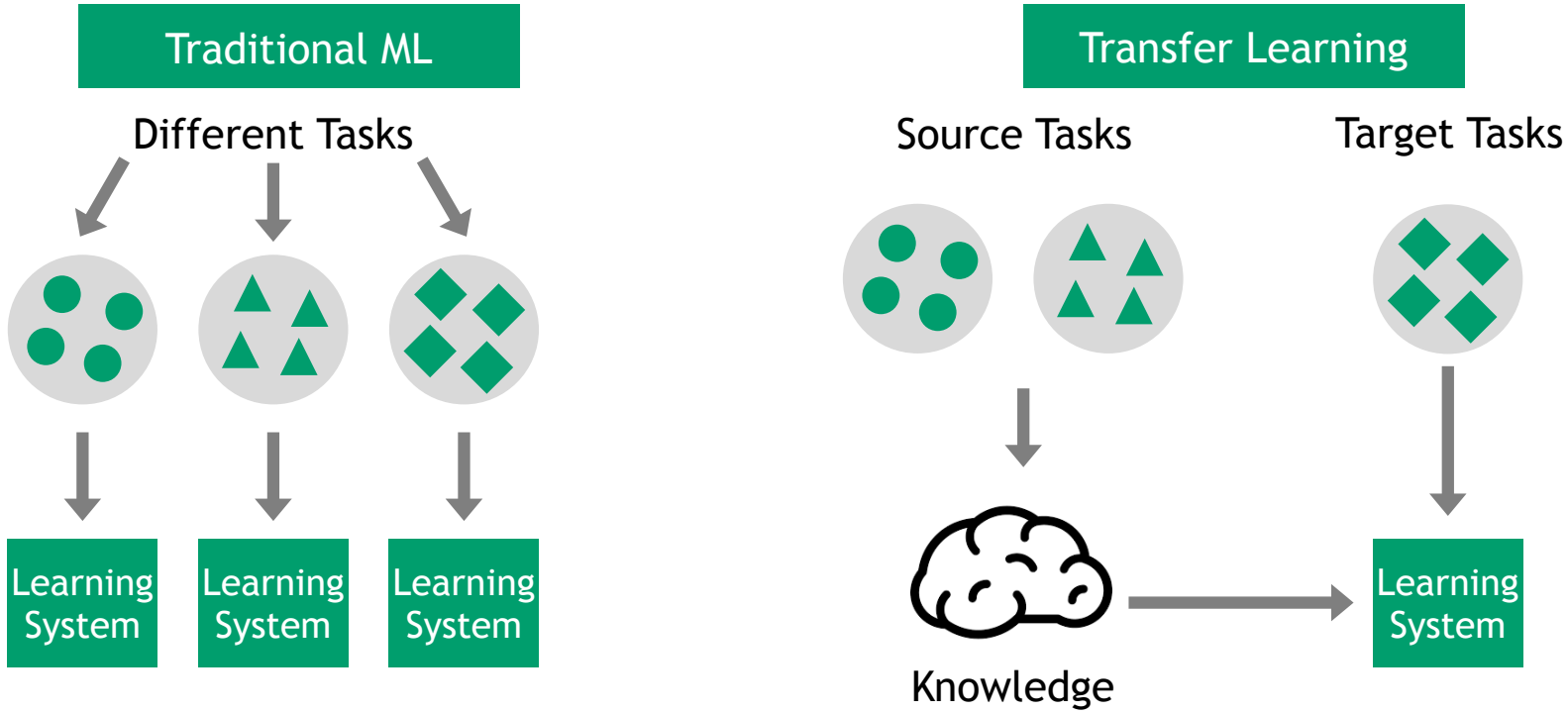
Definition: Transfer Machine Learning

Given a source domain D_S and learning task T_S, a target domain D_T and learning task T_T, transfer learning aims to help improve the learning of the target predictive function f_T in D_T using the knowledge in D_S and T_S, where $D_S \neq D_T$, or $T_S \neq T_T$ [1]

The insight behind *transfer learning* (TL) is that generalization may occur not only within tasks, but also *across tasks*. [2]

[1] Pan and Yang (2009)
[2] Taylor and Stone (2009)

Transfer Machine Learning
Traditional Machine Learning vs. Transfer Learning

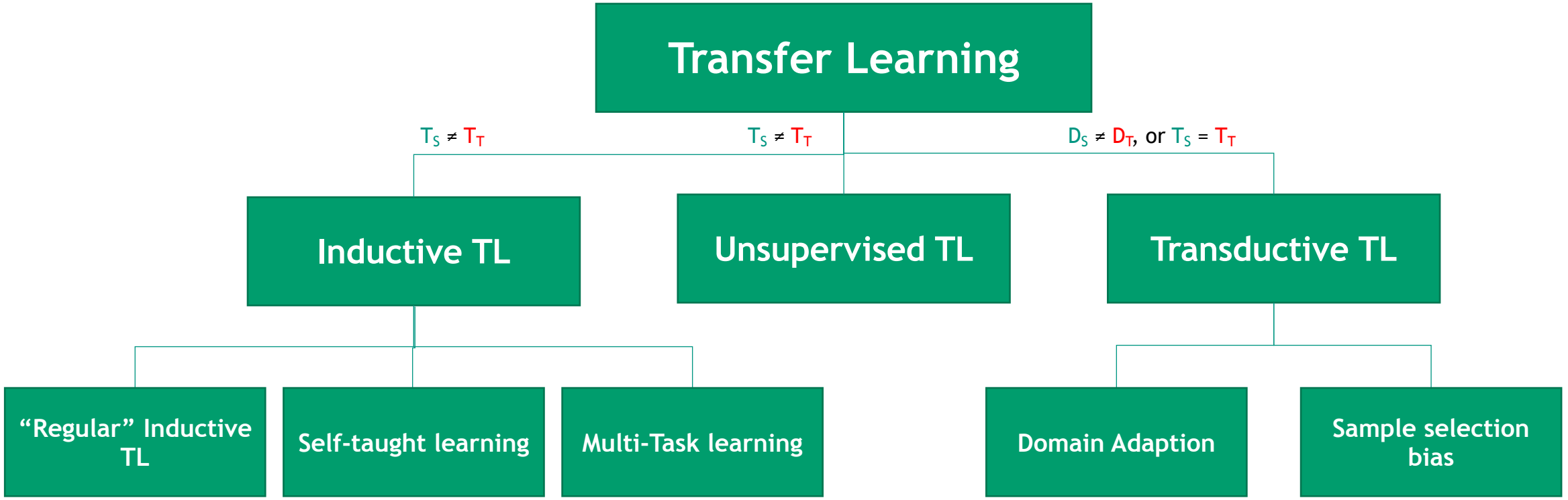


Transfer Learning Concepts try to improve a target learning task with knowledge from already known tasks from e.g. other domains

Sinno Pan and Qiang Yang (2009), A Survey on Transfer Learning

Transfer Machine Learning

Transfer Learning reutilizes knowledge from already performed learning processes



Based on: Sinno Pan and Qiang Yang (2009), A Survey on Transfer Learning

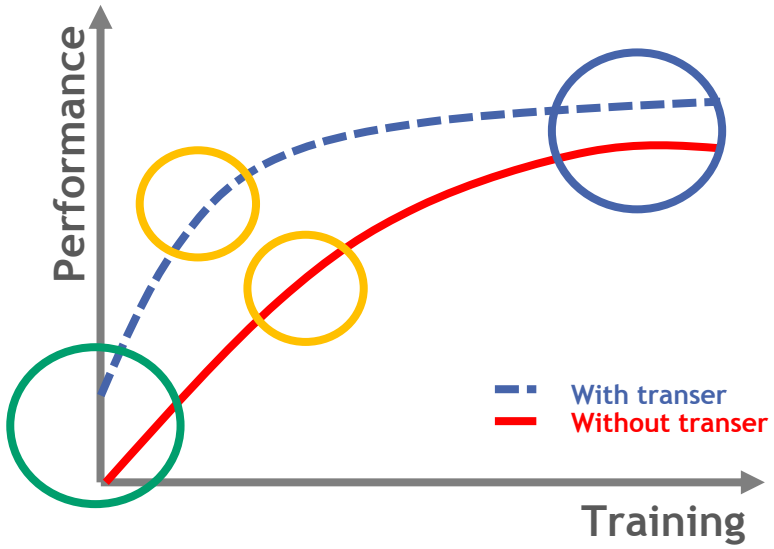
Transfer Machine Learning
Transfer Learning improves learning.

“Higher Slope“

The target task can be learned faster

“Higher Start“

Better initial performance through transferred knowledge



“Higher Asymptote“

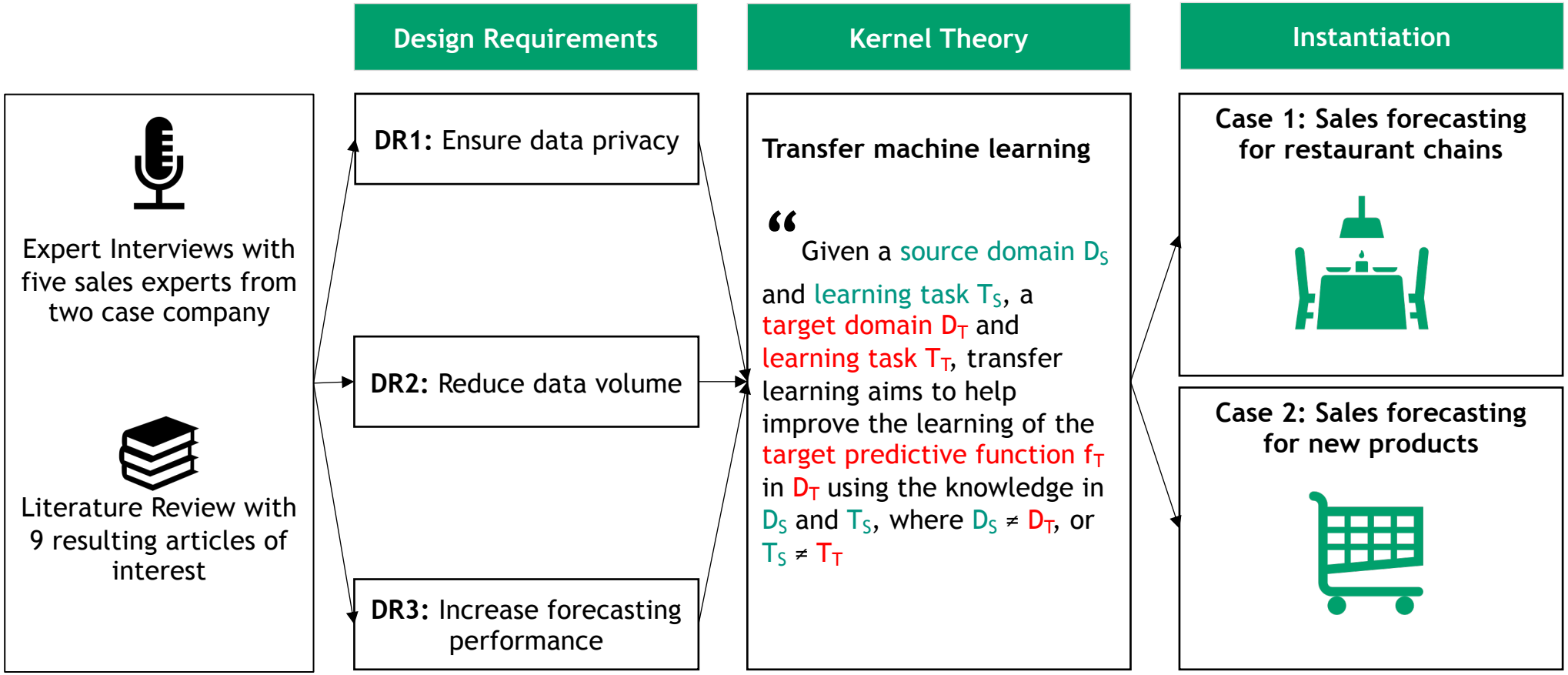
Better final performance

Torrey & Shavlik (2009)

Transfer Machine Learning
Challenges of analyzing distributed data sources

	1. Data privacy 	2. Data heterogeneity 	3. Data velocity/volume 
How meta machine learning copes with challenge	**Solution:** ▪ Just the models are transferred.		**Solution:** ▪ By using pretrained models the training is more efficient and needs less data.

Transfer Machine Learning | Example Sales Forecasting
Derive design requirements from expert interviews



Transfer Machine Learning | Example Sales Forecasting
Sales forecasts for restaurants as a basis for personnel planning

Task

Predict the daily net sales for six restaurant branches belonging to two different chains

Data

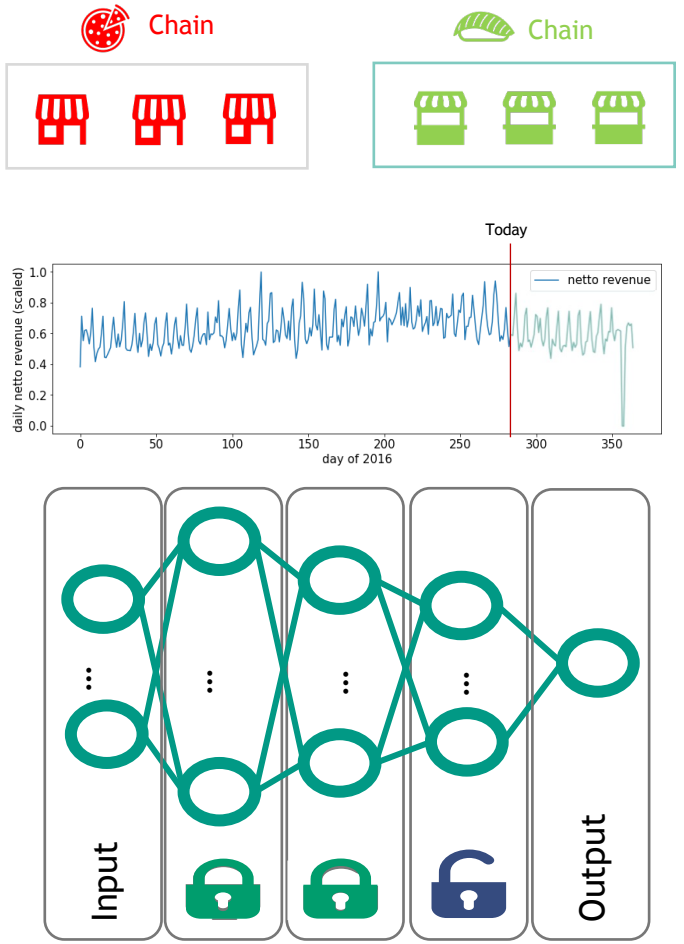
Daily net sales for each branch (2012 – 2017)

Algorithm

Deep Convolutional Neural Network (CNN)

Method: Sequential Transfer Learning

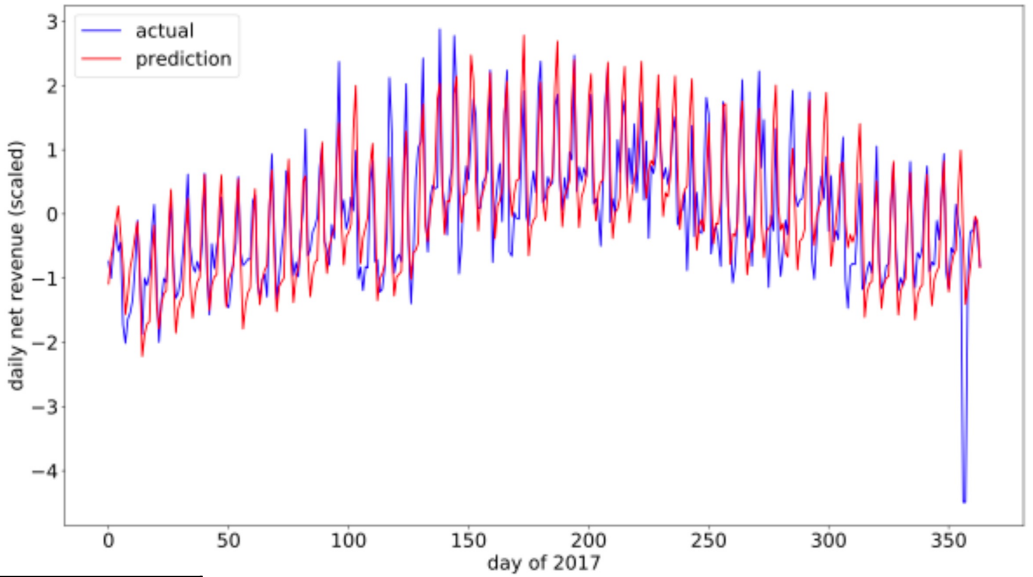
- A deep neural network is trained on restaurant 1.
- The model is then transferred to restaurant 2 (no exchange of raw data).
- The first layers of the neural network are frozen, the later layers can be refined. The model is continued to be trained for restaurant 2.
- This step is repeated.



Hirt, Robin ; Srivastava, Akash ; Berg, Carlos ; Kühl, Niklas: Sequential Transfer Machine Learning in Networks : Measuring the Impact of Data and Neural Net Similarity on Transferability. HICSS-54, 2021.

Transfer Machine Learning | Example Sales Forecasting
Base models perform reasonably well in forecasting the sales

- The results show that already the base models perform well.
- The transfer of models can yield significant performance increases in many cases. On average, transfer learning improves MAPE by 1.11%.
- However, there are cases where transfer learning significantly decreases the performance.



	Branch 1	Branch 2	Branch 3	Branch 4	Branch 5	Branch 6
Base model (MAPE)	9.56	12.78	14.04	10.18	24.95	13.31

Hirt, Robin ; Srivastava, Akash ; Berg, Carlos ; Kühl, Niklas: Sequential Transfer Machine Learning in Networks : Measuring the Impact of Data and Neural Net Similarity on Transferability. HICSS-54, 2021.

Transfer Machine Learning | Example Sales Forecasting
Transfer learning successfully fulfills the initial requirements

Design Requirements	Result	Requirement fulfilled
DR1: Ensure data privacy	No raw data exchanged and no inferences to raw data possible*	✅
DR2: Reduce data volume	Volume of machine learning model (~4 KB) significantly lower than transfer of raw data (~500 KB per restaurant)	✅
DR3: Increase forecasting performance	Average increase of MAPE of 1.11% compared to isolated cases	(✅)

Hirt, Robin ; Srivastava, Akash ; Berg, Carlos ; Kühl, Niklas: Sequential Transfer Machine Learning in Networks : Measuring the Impact of Data and Neural Net Similarity on Transferability. HICSS-54, 2021.

Transfer Machine Learning | Example II Sales Forecasting
Sales forecast of newly introduced products at grocery stores.

Problem

Sales prediction for new products in retail is a difficult task due to few training data and the lack of seasonal and promotional data.
At the supermarket chain, a sales expert currently predicts the sales forecast of new products. This results in multiple challenges in its current process:

- Human bias & problem of imperfect information
- Dependence of expert availability and qualification
- Manual labor with automation potential

We apply a transfer learning-based approach and focus on the analysis of DR3 — performance improvement. We use sales models from existing products and transfer them to newly introduced ones.

Data

14 different bakery goods, 3 years of sales, price and promo data

Algorithm

Long short-term memory neural network (LSTM)

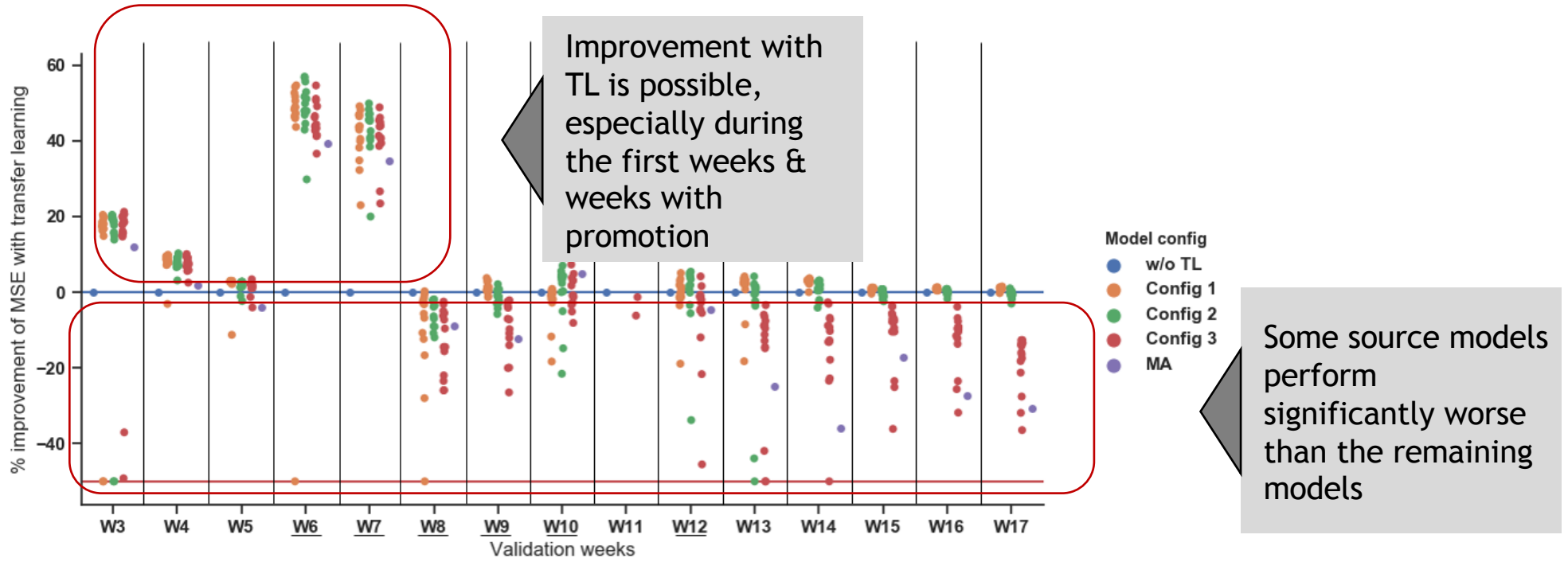
Sales prediction for new products

Few training data + No seasonal & promotional data = Poor prediction quality

Karb, Tristan ; Kühl, Niklas ; Hirt, Robin ; Glivici-Cotruță, Varvara, “A network-based transfer learning approach to improve sales forecasting of new products.” ECIS, 2020

Transfer Machine Learning | Example II Sales Forecasting
Transfer learning increases forecasting performance

Design Requirements	Result	Requirement fulfilled
DR3: Increase forecasting performance	Increase of MSE of up to 50% in the first weeks	(✅)



Karb, Tristan ; Kühl, Niklas ; Hirt, Robin ; Glivici-Cotruță, Varvara, "A network-based transfer learning approach to improve sales forecasting of new products." ECIS, 2020

Transfer Machine Learning | Example II Sales Forecasting
How do we know in advance if a transfer is meaningful?

In both use cases we could show how transfer machine learning enables:

privacy-preserving | **volume-reducing** | **high-performing**

inter-organizational machine learning

However

Transfer machine learning does not “per se” improve performance in comparison to an isolated example w/o transfer learning. This poses a computational challenge, as for N entities, there are $\sum_{k=0}^{n-1} \frac{n!}{k!}$ possible models / transfer paths. In the case of our restaurant example, this means $\sum_{k=0}^{5} \frac{6!}{k!}$ = 1950 possible models!

We must be able to identify the cases of a successful (=performance improving) transfer prior to the transfer itself.

Ongoing: We are investigating ways on how to determine the optimal transfer path ex ante:

- Way 1: Similarity of data
- Way 2: Similarity of meta data

Karb, Tristan ; Kühl, Niklas ; Hirt, Robin ; Glivici-Cotruță, Varvara, “A network-based transfer learning approach to improve sales forecasting of new products.” ECIS, 2020

1. (AI) Service Systems
2. System-wide Learning
3. Meta Machine Learning
4. Transfer Machine Learning
5. **Federated Machine Learning**

Federated Machine Learning
It can tackle multiple key properties

Federated learning must handle data with the following characteristics:

Massively distributed	Non-IID	Unbalanced	Sparse	Uncertain (Big Data)	Limited communication (Big Data)
The data is distributed over a large number of nodes K. K can be bigger than the data n.	Data on each node may be drawn from different distributions.	n is distributed unevenly over K.	Features are occurring on small subset of n.	Often created by users indirectly through their interaction with mobile devices (user interaction data).	Offline or slow, set of users is unpredictable.

Source: Kone, J., Mcmahan, H. B., Ramage, D., & Richt, P. (2016). *Federated Optimization : Distributed Machine Learning for On-Device Intelligence*. 1–38. Brendan McMahan, H., Moore, E., Ramage, D., Hampson, S., & Agüera y Arcas, B. (2017). Communication-efficient learning of deep networks from decentralized data. Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 54.;Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., … Seth, K. (2017). Practical secure aggregation for privacy-preserving machine learning. Proceedings of the ACM Conference on Computer and Communications Security, 1175–1191. https://doi.org/10.1145/3133956.3133982

Federated Machine Learning
To combine the updates of each client, a fusion model is needed.

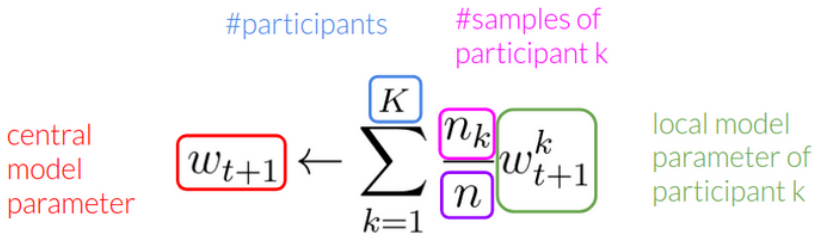
Federated Averaging

Federated Averaging combines local stochastic gradient descent (SGD) on each client with a server that performs model averaging

Remember data is imbalanced (distribution and size)

How it works

- Local models are aggregated by weighting each local model by the number of available training samples
- Clients with more data have more influence on aggregated model
- Other fusion models can be found under https://github.com/IBM/federated-learning-lib



Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., & Bacon, D. (2016). Federated learning: Strategies for improving communication efficiency. *ArXiv Preprint ArXiv:1610.05492*.

Federated Machine Learning
Plays a crucial role in tackling all defined challenges

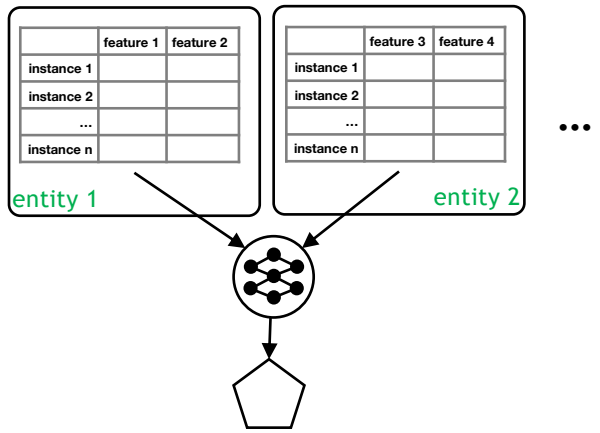
Aggregated data of a service system (simplified)

	feature 1	feature 2	feature 3	feature 4	...	feature k
instance 1	x_{11}	x_{21}	x_{31}	x_{41}	...	x_{k1}
instance 2	x_{12}	x_{22}	x_{32}	x_{42}	...	x_{k2}
...	...	...	...	...	...	...
instance n	x_{1n}	x_{2n}	x_{3n}	x_{4n}	...	x_{kn}

entity 1 (instance 1, instance 2) — entity 2 (instance n)

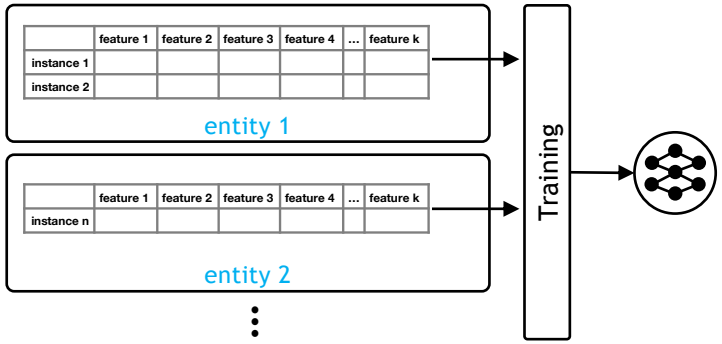
entity 1 (feature 1, feature 2) — entity 2 (feature 3) — entity 4 (feature k)

Challenge A (Vertical federated learning):
Predicting outcomes based on distributed features



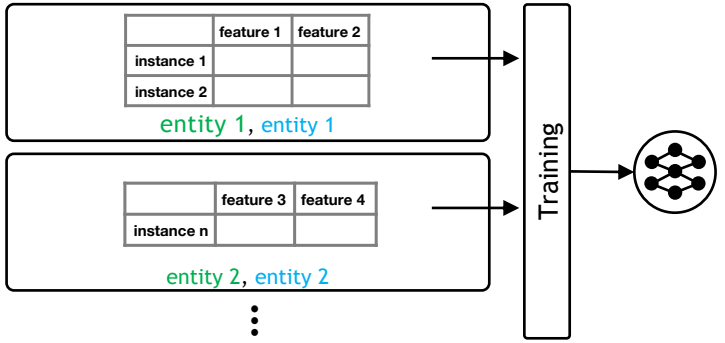
e.g., Demirkan & Delen (2013), G. Liu et al. (2015)

Challenge B (Horizontal federated learning):
Training models based on distributed instances



e.g. Brisimi et al. (2018), Moore et al. (2016)

Challenge C (Federated transfer learning):
Training models based on distributed features and distributed instances



e.g. Brisimi et al. (2018), Moore et al. (2016)

Federated Machine Learning
Challenges of analyzing distributed data sources

How meta machine learning copes with challenge	1. Data privacy 	2. Data heterogeneity 	3. Data velocity/volume 
	Solution: ▪ In federated learning just the local updates are transferred ▪ Additional security components can be implemented to strengthen the security of federated learning even further: ▪ Secure Aggregation ensures that the update transmission is secure		**Solution:** ▪ Because of the magnitude of edge devices, the data would have a huge volume ▪ In federated machine learning the data is never transferred to a central instance

1. (AI) Service Systems
2. System-wide Learning
3. Meta Machine Learning
4. Transfer Machine Learning
5. Federated Machine Learning

Summary

Meta machine learning

1. A meta-learning system must include a learning subsystem, which adapts with experience.
2. Experience is gained by exploiting metaknowledge extracted
   a) ...in a previous learning episode on a single dataset, and/or
   b) ...from different domains or problems.

Transfer machine learning

- Transfer learning is the improvement of learning in a new task through the transfer of knowledge from a related task that has already been learned.

Federated machine learning

- Distributed learning for decentralized data with a focus on privacy preserving (Google)

Summary

Systemic challenges of distributed ML can be addressed by different concepts and combinations of these concepts

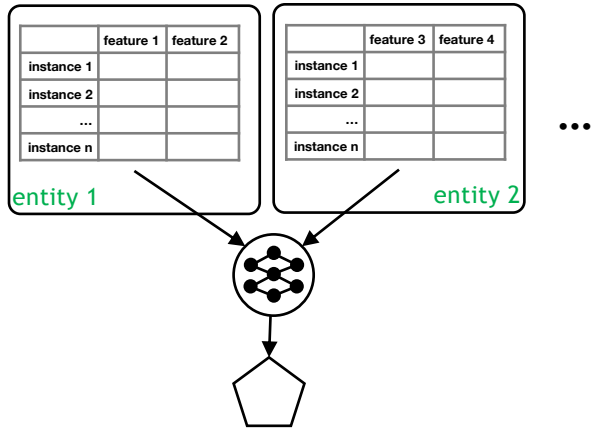
Aggregated data of a AI service system (simplified)

	feature 1	feature 2	feature 3	feature 4	...	feature k
instance 1	x_{11}	x_{21}	x_{31}	x_{41}	...	x_{k1}
instance 2	x_{12}	x_{22}	x_{32}	x_{42}	...	x_{k2}
...	...	...	...	...	...	...
instance n	x_{1n}	x_{2n}	x_{3n}	x_{4n}	...	x_{kn}

entity 1 (instance 1, instance 2) | entity 2 (instance n)

entity 1 (feature 1, feature 2) | entity 2 (feature 3, feature 4) | entity 4 (feature k)

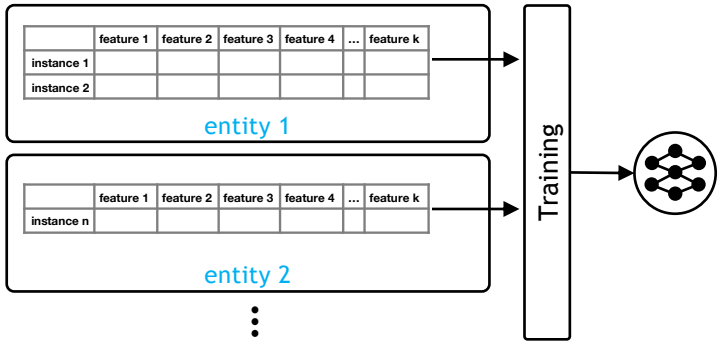
Challenge A:
Predicting outcomes based on distributed features



e.g., Demirkan & Delen (2013), G. Liu et al. (2015)

(Federated) Meta machine learning

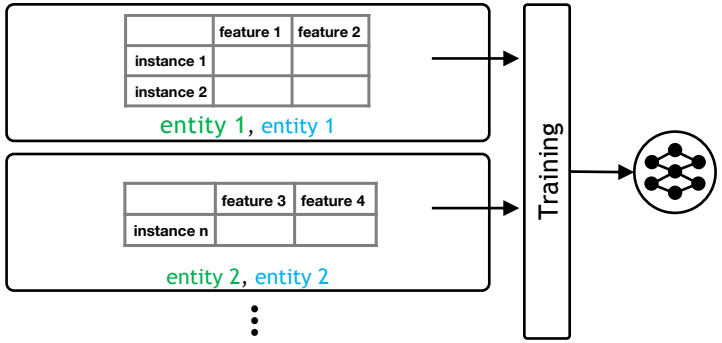
Challenge B:
Training models based on distributed instances



e.g. Brisimi et al. (2018), Moore et al. (2016)

(Federated) Transfer machine learning

Challenge C:
Training models based on distributed componants and distributed instances



e.g. Brisimi et al. (2018), Moore et al. (2016)

(Federated) Transfer machine learning

Summary
Depending on the challenges at hand different concepts are recommended to analyze distributed data sources

	1. Data privacy 	2. Data heterogeneity 	3. Data velocity/volume 
Meta machine learning	• As only abstracted prediction outputs are being exposed outside an entities' border, data privacy and confidentiality is preserved.	• Builds predictions for each data type separately	• Reduce the data volume to predictions
Transfer machine Learning	• Whole model is transferred	• /	• By using pretrained models the training is more efficient and needs less data.
Federated machine learning	• In federated learning just the local updates are transferred • Secure Aggregation ensures that the update transmission is secure	• /	• Because of the magnitude of edge devices, the data would have a huge volume • In federated machine learning the data is never transferred to a central instance

1