ReClean: Reinforcement Learning for Automated Data Cleaning in ML Pipelines

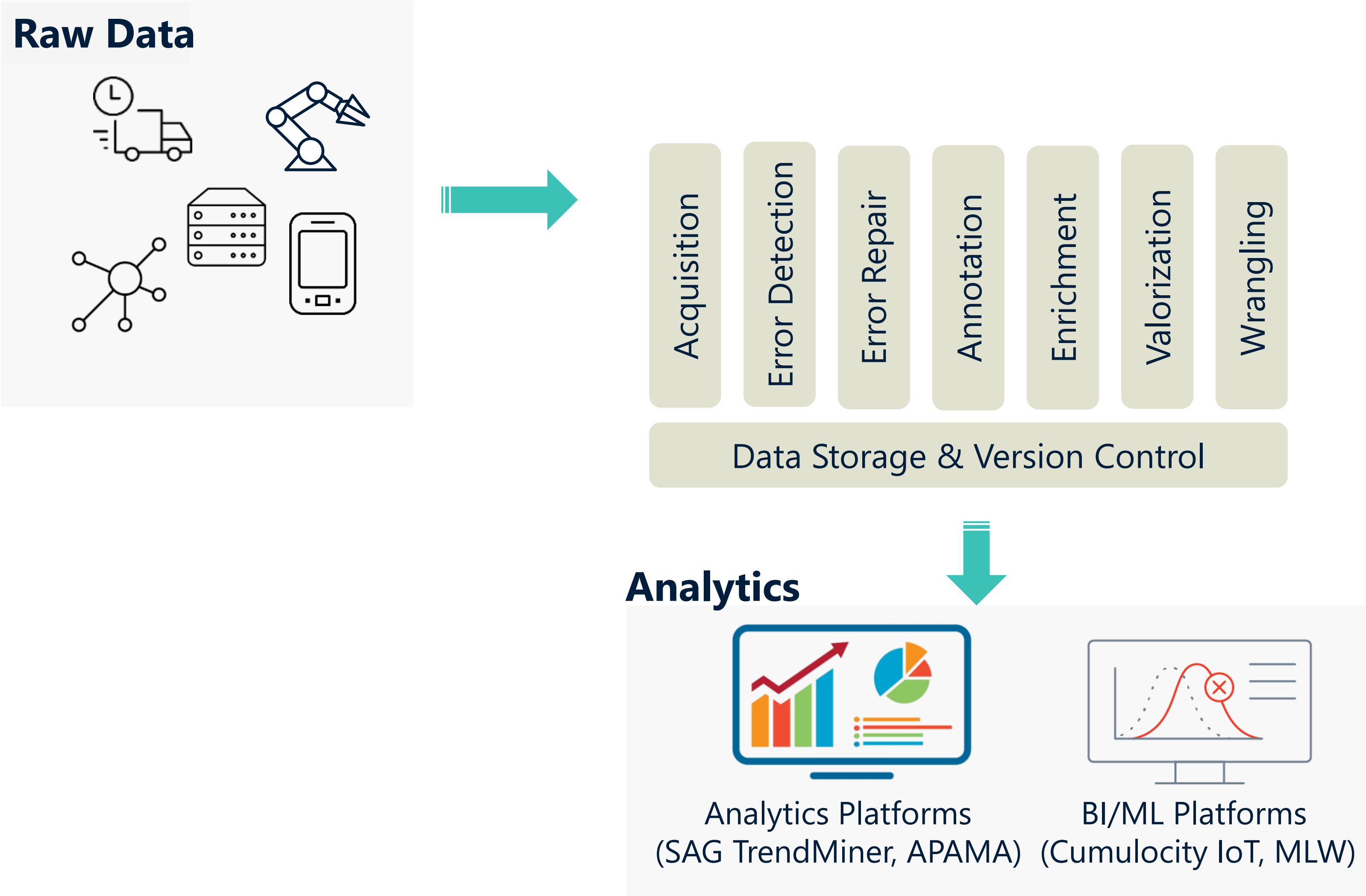
DBML@ICDE'24, 13rd May 2024, Utrecht, Netherlands

Mohamed Abdelaal, Anil Bora Yayak, Kai Klede, Harald Schöning

SPONSORED BY THE

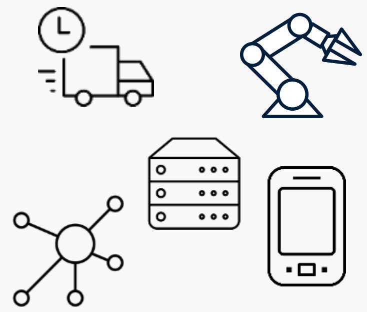
Federal Ministry
of Education
and Research

Automated Data Preparation



Automated Data Preparation

Raw Data



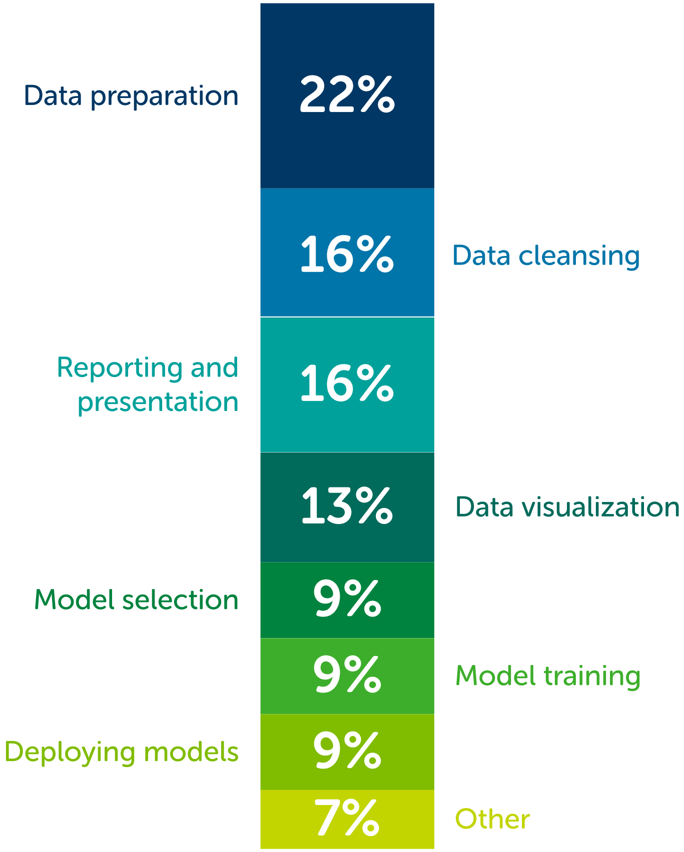
Acquisition | Error Detection | Error Repair | Annotation | Enrichment | Valorization | Wrangling

Data Storage & Version Control

Analytics

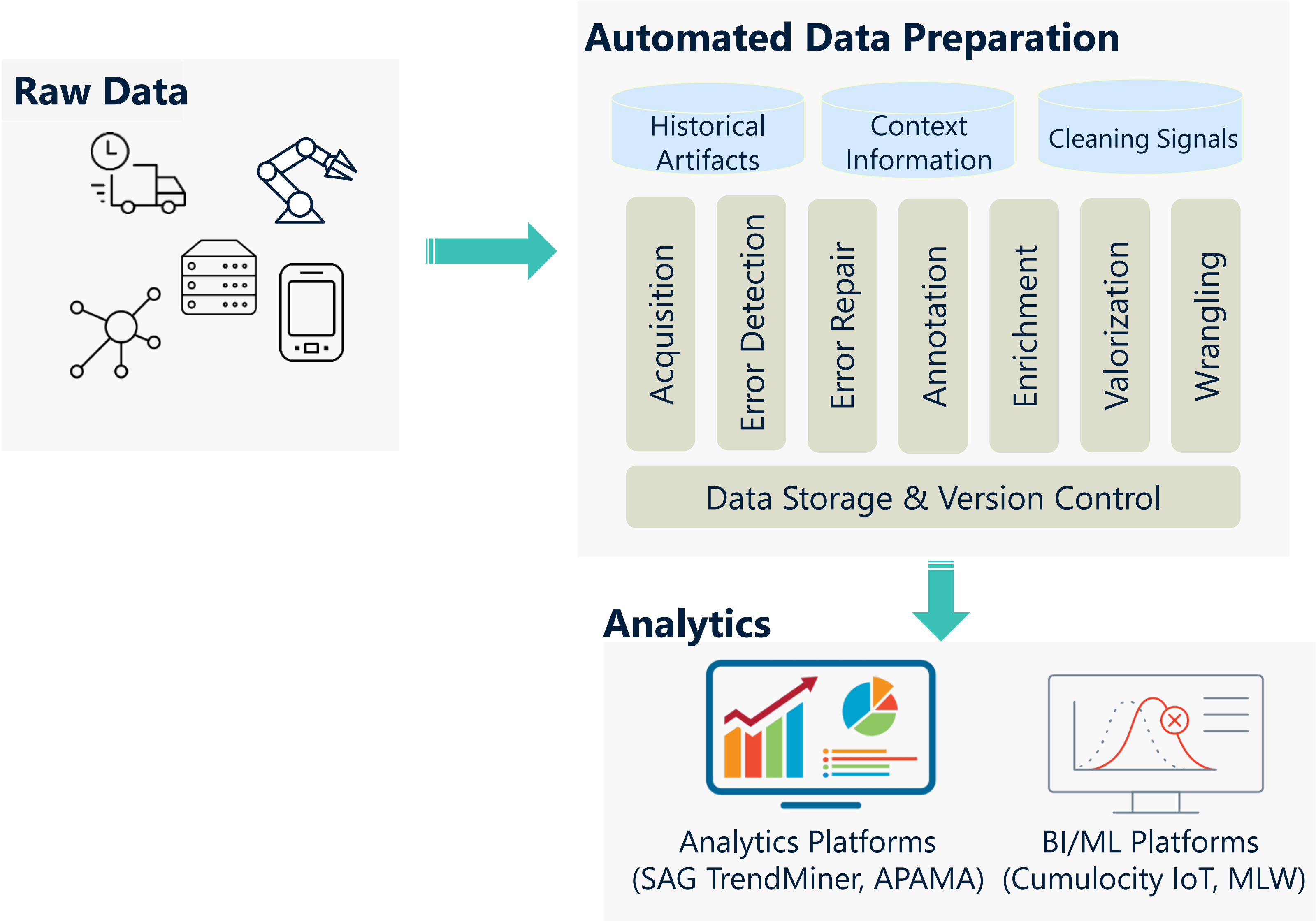
Analytics Platforms
(SAG TrendMiner, APAMA)

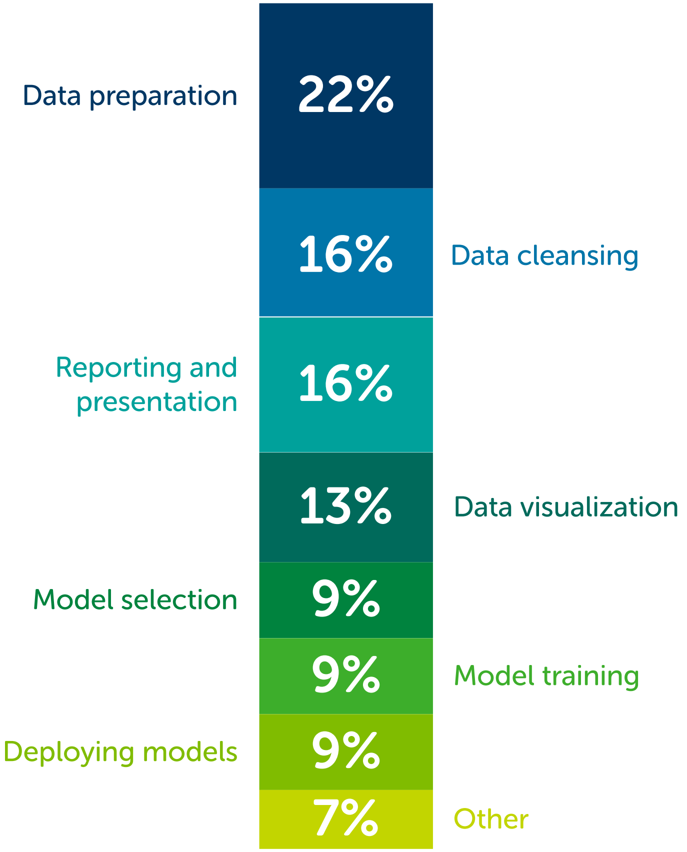
BI/ML Platforms
(Cumulocity IoT, MLW)



Anaconda: 2022 State of Data Science

Automated Data Preparation





Anaconda: 2022 State of Data Science

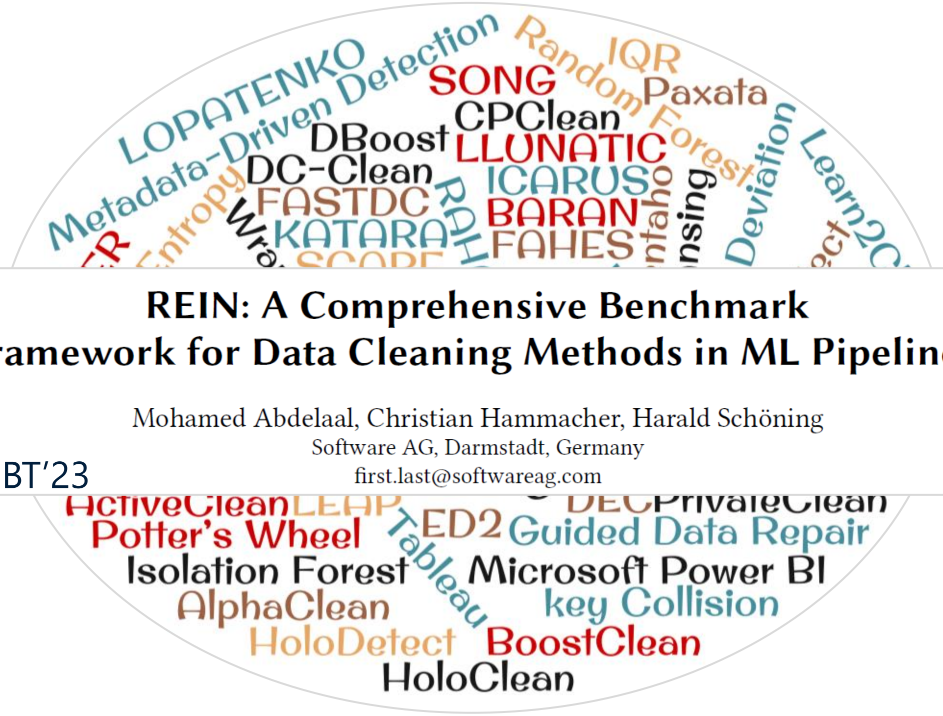
Data Cleaning for ML Pipelines

Challenges & Contributions

software AG

Data Cleaning for ML Pipelines

Challenges & Contributions



REIN: A Comprehensive Benchmark Framework for Data Cleaning Methods in ML Pipelines*

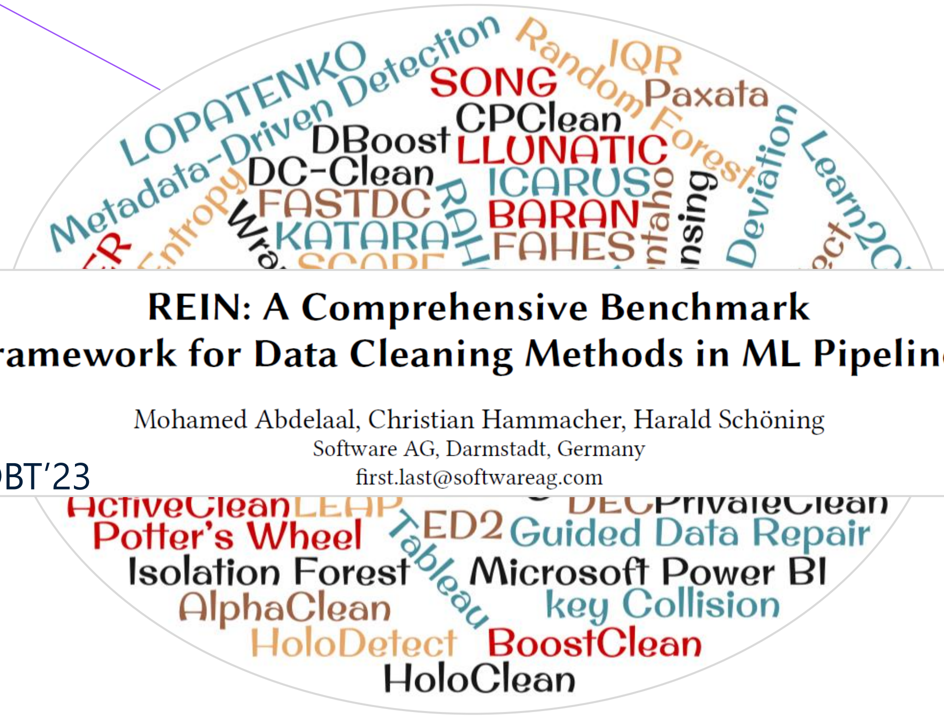
Mohamed Abdelaal, Christian Hammacher, Harald Schöning
Software AG, Darmstadt, Germany
first.last@softwareag.com

EDBT'23

Data Cleaning for ML Pipelines

Challenges & Contributions

Scalability problems



REIN: A Comprehensive Benchmark Framework for Data Cleaning Methods in ML Pipelines*

Mohamed Abdelaal, Christian Hammacher, Harald Schöning
Software AG, Darmstadt, Germany
first.last@softwareag.com

EDBT'23

Data Cleaning for ML Pipelines

Challenges & Contributions

Scalability problems

SAGED: Few-Shot Meta Learning for Tabular Data Error Detection

Mohamed Abdelaal
Software AG, Darmstadt, Germany
Mohamed.Abdelaal@softwareag.com

Tim Ktitarev
Software AG, Darmstadt, Germany
Tim.Ktitarev@softwareag.com

Daniel Städtler
Hochschule Darmstadt, Germany
Daniel.Staedtler@yahoo.com

Harald Schöning
Software AG, Darmstadt, Germany
Harald.Schoening@softwareag.com

EDBT 2024

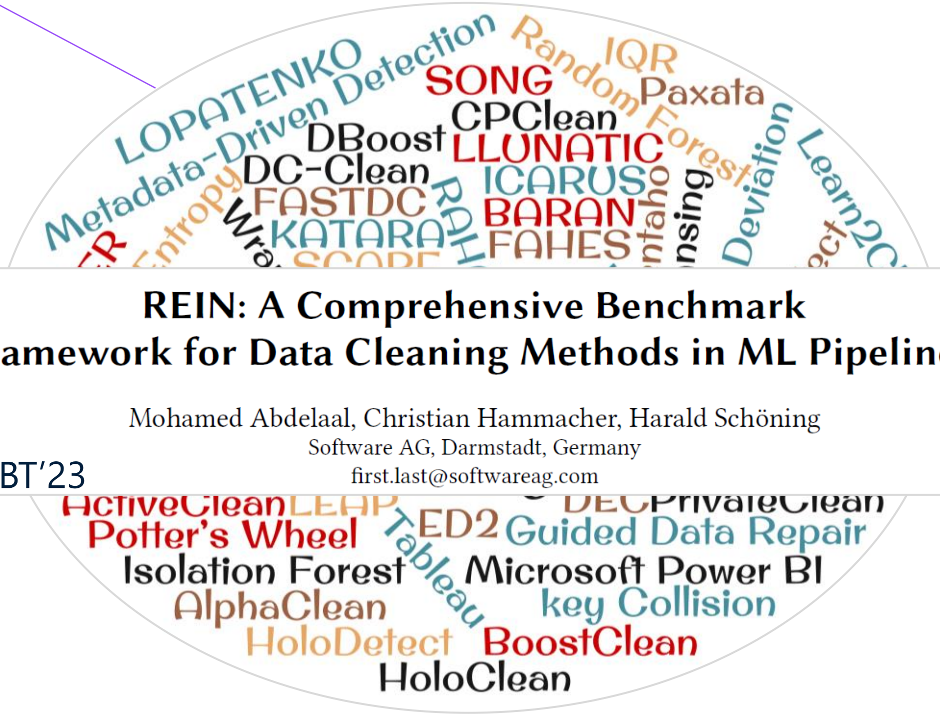
AutoCure: Automated Tabular Data Curation Technique for ML Pipelines

Mohamed Abdelaal
Software AG
Darmstadt, Germany
mohamed.abdelaal@softwareag.com

Rashmi Koparde
Otto von Guericke University Magdeburg
Magdeburg, Germany
rashmi.koparde@st.ovgu.de

Harald Schöning
Software AG
Darmstadt, Germany
harald.schoening@softwareag.com

aiDM@SIGMOD'23



REIN: A Comprehensive Benchmark Framework for Data Cleaning Methods in ML Pipelines*

Mohamed Abdelaal, Christian Hammacher, Harald Schöning
Software AG, Darmstadt, Germany
first.last@softwareag.com

EDBT'23

Data Cleaning for ML Pipelines

Challenges & Contributions

Scalability problems

SAGED: Few-Shot Meta Learning for Tabular Data Error Detection

Mohamed Abdelaal
Software AG, Darmstadt, Germany
Mohamed.Abdelaal@softwareag.com

Tim Ktitarev
Software AG, Darmstadt, Germany
Tim.Ktitarev@softwareag.com

Daniel Städtler
Hochschule Darmstadt, Germany
Daniel.Staedtler@yahoo.com

Harald Schöning
Software AG, Darmstadt, Germany
Harald.Schoening@softwareag.com

EDBT 2024

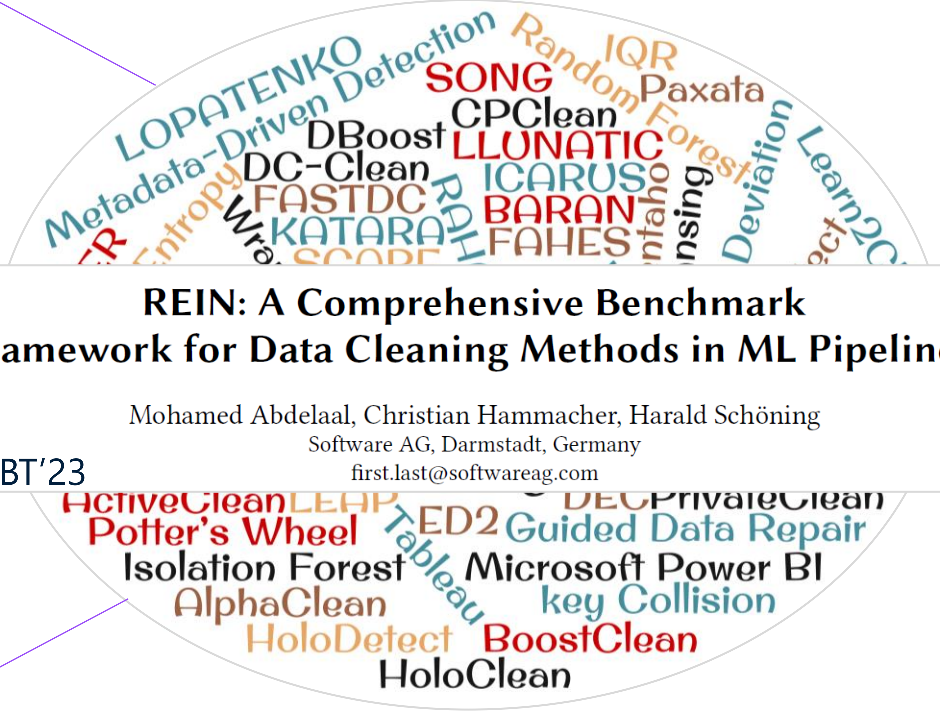
AutoCure: Automated Tabular Data Curation Technique for ML Pipelines

Mohamed Abdelaal
Software AG
Darmstadt, Germany
mohamed.abdelaal@softwareag.com

Rashmi Koparde
Otto von Guericke University Magdeburg
Magdeburg, Germany
rashmi.koparde@st.ovgu.de

Harald Schöning
Software AG
Darmstadt, Germany
harald.schoening@softwareag.com

aiDM@SIGMOD'23



REIN: A Comprehensive Benchmark Framework for Data Cleaning Methods in ML Pipelines*

Mohamed Abdelaal, Christian Hammacher, Harald Schöning
Software AG, Darmstadt, Germany
first.last@softwareag.com

EDBT'23

Ignoring Context Information

Data Cleaning for ML Pipelines

Challenges & Contributions

Scalability problems

SAGED: Few-Shot Meta Learning for Tabular Data Error Detection

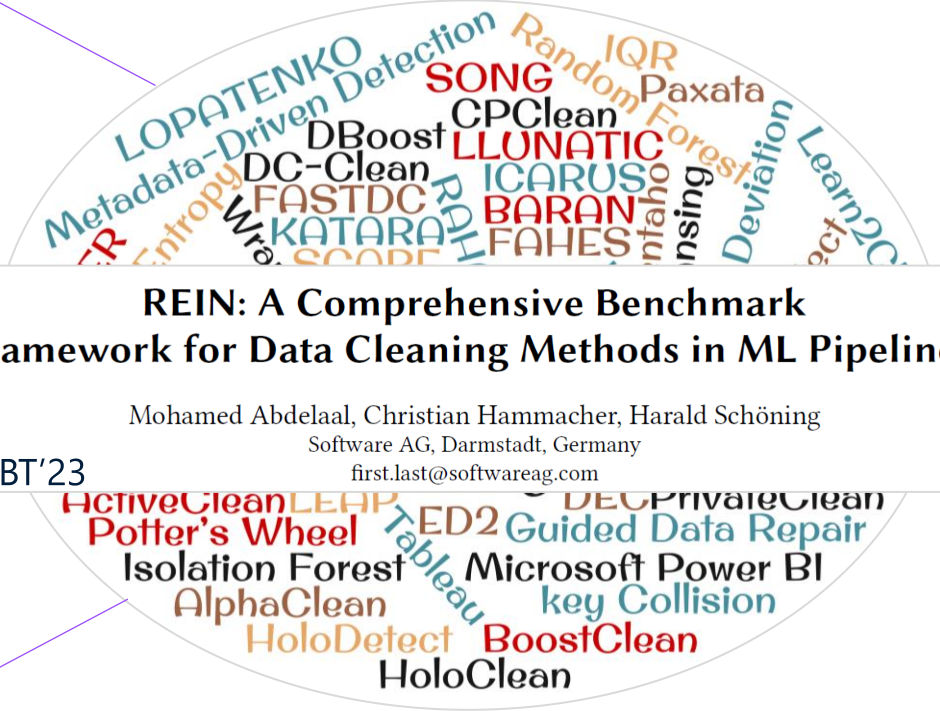
Mohamed Abdelaal, Software AG, Darmstadt, Germany, Mohamed.Abdelaal@softwareag.com
Tim Ktitarev, Software AG, Darmstadt, Germany, Tim.Ktitarev@softwareag.com
Daniel Städtler, Hochschule Darmstadt, Germany, Daniel.Staedtler@yahoo.com
Harald Schöning, Software AG, Darmstadt, Germany, Harald.Schoening@softwareag.com

EDBT 2024

AutoCure: Automated Tabular Data Curation Technique for ML Pipelines

Mohamed Abdelaal, *Software AG*, Darmstadt, Germany, mohamed.abdelaal@softwareag.com
Rashmi Koparde, *Otto von Guericke University Magdeburg*, Magdeburg, Germany, rashmi.koparde@st.ovgu.de
Harald Schöning, *Software AG*, Darmstadt, Germany, harald.schoening@softwareag.com

aiDM@SIGMOD'23



REIN: A Comprehensive Benchmark Framework for Data Cleaning Methods in ML Pipelines*

Mohamed Abdelaal, Christian Hammacher, Harald Schöning
Software AG, Darmstadt, Germany
first.last@softwareag.com

EDBT'23

Ignoring Context Information

RTClean: Context-aware Tabular Data Cleaning using Real-time OFDs

Daniel Del Gaudio, *University of Stuttgart*, Stuttgart, Germany, Daniel.Del-Gaudio@ipvs.uni-stuttgart.de
Tim Schubert, *University of Stuttgart*, Stuttgart, Germany, st148736@stud.uni-stuttgart.de
Mohamed Abdelaal, *Software AG*, Darmstadt, Germany, Mohamed.Abdelaal@softwareag.com

CoMoRea@PerCom'23

Data Cleaning for ML Pipelines

Challenges & Contributions

Scalability problems

SAGED: Few-Shot Meta Learning for Tabular Data Error Detection

Mohamed Abdelaal, Software AG, Darmstadt, Germany, Mohamed.Abdelaal@softwareag.com
Tim Ktitarev, Software AG, Darmstadt, Germany, Tim.Ktitarev@softwareag.com
Daniel Städtler, Hochschule Darmstadt, Germany, Daniel.Staedtler@yahoo.com
Harald Schöning, Software AG, Darmstadt, Germany, Harald.Schoening@softwareag.com

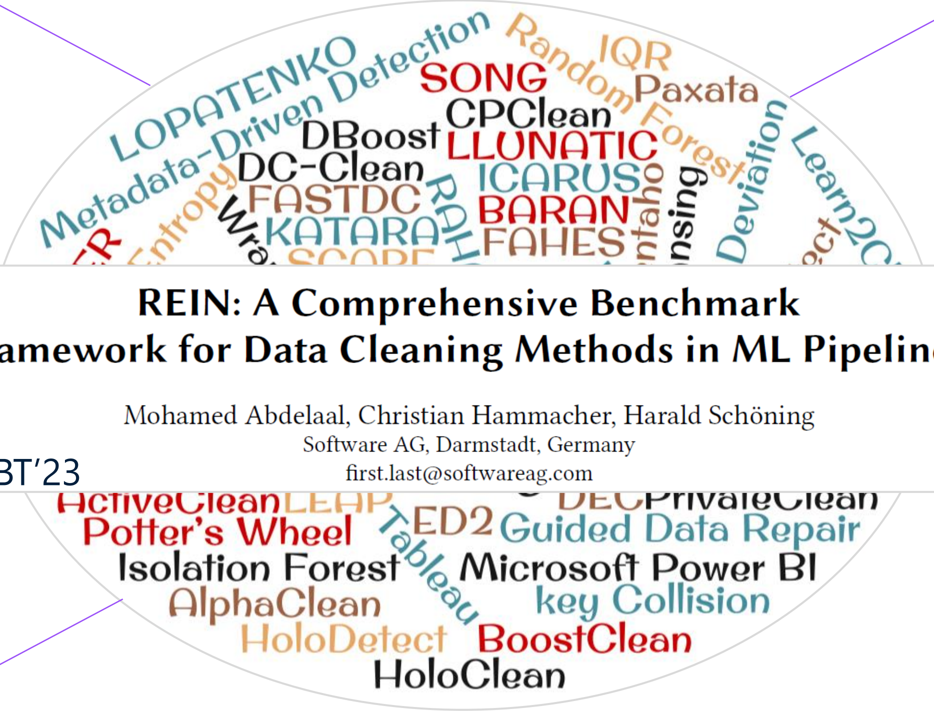
EDBT 2024

AutoCure: Automated Tabular Data Curation Technique for ML Pipelines

Mohamed Abdelaal, *Software AG*, Darmstadt, Germany, mohamed.abdelaal@softwareag.com
Rashmi Koparde, *Otto von Guericke University Magdeburg*, Magdeburg, Germany, rashmi.koparde@st.ovgu.de
Harald Schöning, *Software AG*, Darmstadt, Germany, harald.schoening@softwareag.com

aiDM@SIGMOD'23

Overlooking Downstream Tasks



REIN: A Comprehensive Benchmark Framework for Data Cleaning Methods in ML Pipelines*

Mohamed Abdelaal, Christian Hammacher, Harald Schöning
Software AG, Darmstadt, Germany
first.last@softwareag.com

EDBT'23

Ignoring Context Information

RTClean: Context-aware Tabular Data Cleaning using Real-time OFDs

Daniel Del Gaudio, *University of Stuttgart*, Stuttgart, Germany, Daniel.Del-Gaudio@ipvs.uni-stuttgart.de
Tim Schubert, *University of Stuttgart*, Stuttgart, Germany, st148736@stud.uni-stuttgart.de
Mohamed Abdelaal, *Software AG*, Darmstadt, Germany, Mohamed.Abdelaal@softwareag.com

CoMoRea@PerCom'23

software AG

Data Cleaning for ML Pipelines

Challenges & Contributions

Scalability problems

SAGED: Few-Shot Meta Learning for Tabular Data Error Detection

Mohamed Abdelaal, Software AG, Darmstadt, Germany, Mohamed.Abdelaal@softwareag.com
Tim Ktitarev, Software AG, Darmstadt, Germany, Tim.Ktitarev@softwareag.com
Daniel Städtler, Hochschule Darmstadt, Germany, Daniel.Staedtler@yahoo.com
Harald Schöning, Software AG, Darmstadt, Germany, Harald.Schoening@softwareag.com

EDBT 2024

AutoCure: Automated Tabular Data Curation Technique for ML Pipelines

Mohamed Abdelaal, *Software AG*, Darmstadt, Germany, mohamed.abdelaal@softwareag.com
Rashmi Koparde, *Otto von Guericke University Magdeburg*, Magdeburg, Germany, rashmi.koparde@st.ovgu.de
Harald Schöning, *Software AG*, Darmstadt, Germany, harald.schoening@softwareag.com

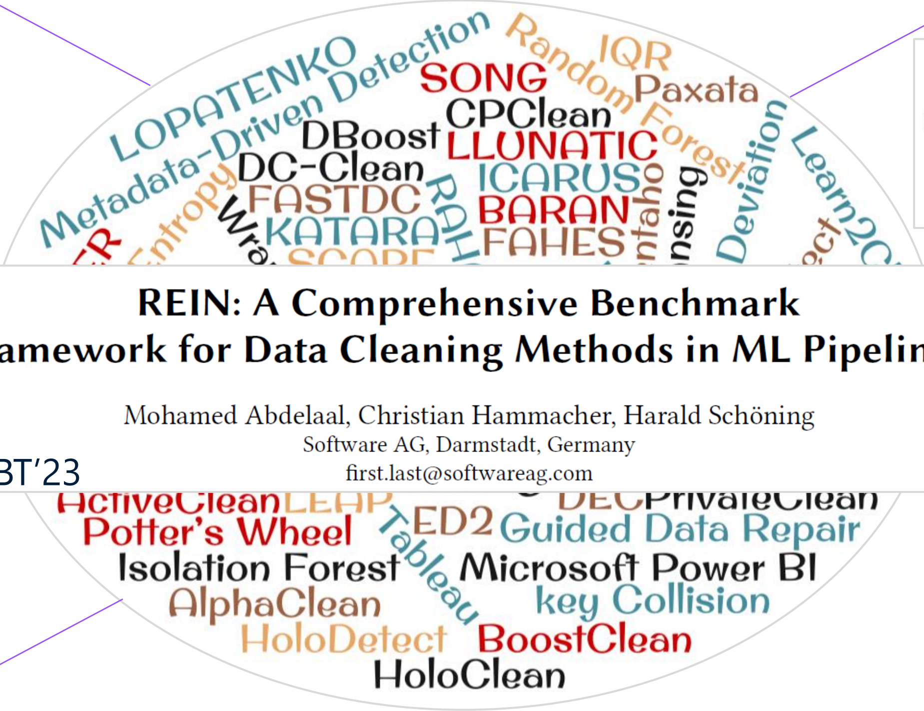
aiDM@SIGMOD'23

Overlooking Downstream Tasks

DiffML: End-to-end Differentiable ML Pipelines

Benjamin Hilprecht*, TU Darmstadt
Christian Hammacher*, Software AG
Eduardo Reis, TU Darmstadt
Mohamed Abdelaal, Software AG
Carsten Binnig, TU Darmstadt

DEEM@SIGMOD'23



REIN: A Comprehensive Benchmark Framework for Data Cleaning Methods in ML Pipelines*

Mohamed Abdelaal, Christian Hammacher, Harald Schöning
Software AG, Darmstadt, Germany
first.last@softwareag.com

EDBT'23

Ignoring Context Information

RTClean: Context-aware Tabular Data Cleaning using Real-time OFDs

Daniel Del Gaudio, *University of Stuttgart*, Stuttgart, Germany, Daniel.Del-Gaudio@ipvs.uni-stuttgart.de
Tim Schubert, *University of Stuttgart*, Stuttgart, Germany, st148736@stud.uni-stuttgart.de
Mohamed Abdelaal, *Software AG*, Darmstadt, Germany, Mohamed.Abdelaal@softwareag.com

CoMoRea@PerCom'23

software AG

Data Cleaning for ML Pipelines

Challenges & Contributions

Scalability problems

SAGED: Few-Shot Meta Learning for Tabular Data Error Detection

Mohamed Abdelaal
Software AG, Darmstadt, Germany
Mohamed.Abdelaal@softwareag.com

Tim Ktitarev
Software AG, Darmstadt, Germany
Tim.Ktitarev@softwareag.com

Daniel Städtler
Hochschule Darmstadt, Germany
Daniel.Staedtler@yahoo.com

Harald Schöning
Software AG, Darmstadt, Germany
Harald.Schoening@softwareag.com

EDBT 2024

AutoCure: Automated Tabular Data Curation Technique for ML Pipelines

Mohamed Abdelaal
Software AG
Darmstadt, Germany
mohamed.abdelaal@softwareag.com

Rashmi Koparde
Otto von Guericke University Magdeburg
Magdeburg, Germany
rashmi.koparde@st.ovgu.de

Harald Schöning
Software AG
Darmstadt, Germany
harald.schoening@softwareag.com

aiDM@SIGMOD'23

Overlooking Downstream Tasks

DiffML: End-to-end Differentiable ML Pipelines

Benjamin Hilprecht*
TU Darmstadt

Christian Hammacher*
Software AG

Eduardo Reis
TU Darmstadt

Mohamed Abdelaal
Software AG

Carsten Binnig
TU Darmstadt

DEEM@SIGMOD'23

REIN: A Comprehensive Benchmark Framework for Data Cleaning Methods in ML Pipelines*

Mohamed Abdelaal, Christian Hammacher, Harald Schöning
Software AG, Darmstadt, Germany
first.last@softwareag.com

EDBT'23

Ignoring Context Information

RTClean: Context-aware Tabular Data Cleaning using Real-time OFDs

Daniel Del Gaudio
University of Stuttgart
Stuttgart, Germany
Daniel.Del-Gaudio@ipvs.uni-stuttgart.de

Tim Schubert
University of Stuttgart
Stuttgart, Germany
st148736@stud.uni-stuttgart.de

Mohamed Abdelaal
Software AG
Darmstadt, Germany
Mohamed.Abdelaal@softwareag.com

CoMoRea@PerCom'23

Lack of Generalizability

software AG

Data Cleaning for ML Pipelines

Challenges & Contributions

Scalability problems

SAGED: Few-Shot Meta Learning for Tabular Data Error Detection

Mohamed Abdelaal, Software AG, Darmstadt, Germany, Mohamed.Abdelaal@softwareag.com
Tim Ktitarev, Software AG, Darmstadt, Germany, Tim.Ktitarev@softwareag.com
Daniel Städtler, Hochschule Darmstadt, Germany, Daniel.Staedtler@yahoo.com
Harald Schöning, Software AG, Darmstadt, Germany, Harald.Schoening@softwareag.com

EDBT 2024

AutoCure: Automated Tabular Data Curation Technique for ML Pipelines

Mohamed Abdelaal, *Software AG*, Darmstadt, Germany, mohamed.abdelaal@softwareag.com
Rashmi Koparde, *Otto von Guericke University Magdeburg*, Magdeburg, Germany, rashmi.koparde@st.ovgu.de
Harald Schöning, *Software AG*, Darmstadt, Germany, harald.schoening@softwareag.com

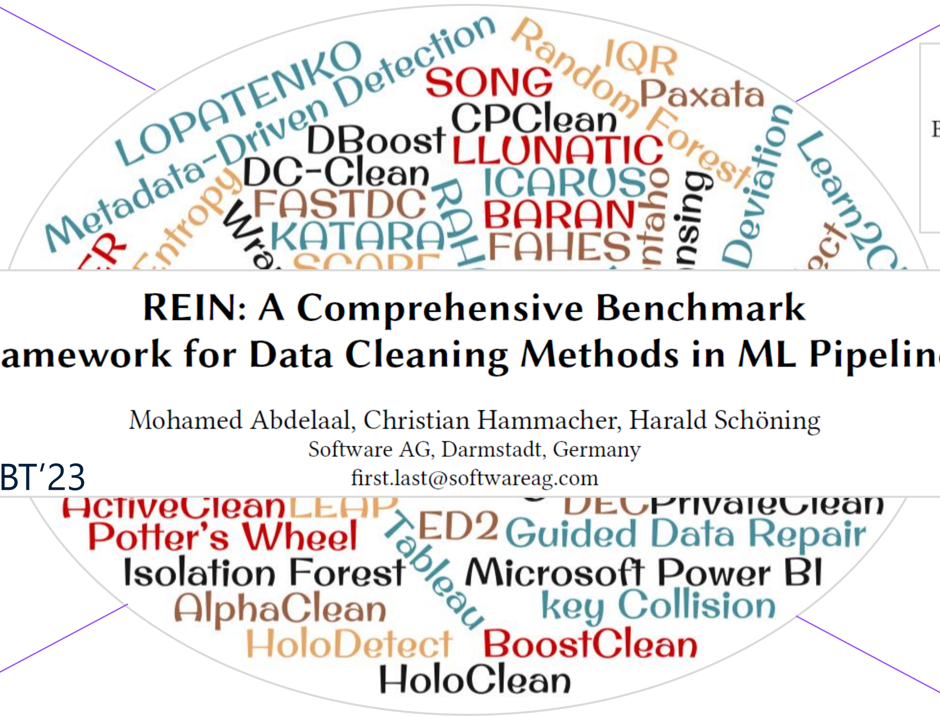
aiDM@SIGMOD'23

Overlooking Downstream Tasks

DiffML: End-to-end Differentiable ML Pipelines

Benjamin Hilprecht*, TU Darmstadt
Christian Hammacher*, Software AG
Eduardo Reis, TU Darmstadt
Mohamed Abdelaal, Software AG
Carsten Binnig, TU Darmstadt

DEEM@SIGMOD'23



REIN: A Comprehensive Benchmark Framework for Data Cleaning Methods in ML Pipelines*

Mohamed Abdelaal, Christian Hammacher, Harald Schöning
Software AG, Darmstadt, Germany
first.last@softwareag.com

EDBT'23

Ignoring Context Information

RTClean: Context-aware Tabular Data Cleaning using Real-time OFDs

Daniel Del Gaudio, *University of Stuttgart*, Stuttgart, Germany, Daniel.Del-Gaudio@ipvs.uni-stuttgart.de
Tim Schubert, *University of Stuttgart*, Stuttgart, Germany, st148736@stud.uni-stuttgart.de
Mohamed Abdelaal, *Software AG*, Darmstadt, Germany, Mohamed.Abdelaal@softwareag.com

CoMoRea@PerCom'23

Lack of Generalizability

ReClean: Reinforcement Learning for Automated Data Cleaning in ML Pipelines*

Mohamed Abdelaal*, Anil Bora Yayak†, Kai Klede†, Harald Schöning*
* *Software AG, Darmstadt, Germany*
† *University of Erlangen-Nuremberg, Erlangen, Germany*
* First.Last@softwareag.com, Abora.Yayak@gmail.com, kai.klede@fau.de

DBML@ICDE'24

ReClean

Proper selection of a well-suited data cleaning strategy requires data expertise

ReClean

Proper selection of a well-suited data cleaning strategy requires data expertise

Example Dataset: Customer Satisfaction

Customer ID	Age	City	Monthly Spend	Satisfaction Level
1	29	New York	200	High
2	35	Los Angeles	-1	Medium
3		Chicago	150	Low
4	42	San Francisco	220	Medium
5	31	San Diegeo	185	High

ReClean

Proper selection of a well-suited data cleaning strategy requires data expertise

Example Dataset: Customer Satisfaction

Customer ID	Age	City	Monthly Spend	Satisfaction Level
1	29	New York	200	High
2	35	Los Angeles	-1	Medium
3		Chicago	150	Low
4	42	San Francisco	220	Medium
5	31	San Diegeo	185	High

Mean of all Customers

Monthly Spend
200
188
150
220
185

Group-based Mean (City)

Monthly Spend
200
210
150
220
185

Regression Model

Monthly Spend
200
205
150
220
185

ReClean

Pains

Proper selection of a well-suited data cleaning strategy requires data expertise

Automated repair methods may harm downstream ML models

Example Dataset: Customer Satisfaction

Customer ID	Age	City	Monthly Spend	Satisfaction Level
1	29	New York	200	High
2	35	Los Angeles	-1	Medium
3		Chicago	150	Low
4	42	San Francisco	220	Medium
5	31	San Diegeo	185	High

Mean of all Customers

Monthly Spend
200
188
150
220
185

Group-based Mean (City)

Monthly Spend
200
210
150
220
185

Regression Model

Monthly Spend
200
205
150
220
185

software AG

ReClean

Pains

Proper selection of a well-suited data cleaning strategy requires data expertise

Automated repair methods may harm downstream ML models

ReClean

Select/combine cleaners based on their impact on downstream predictive tasks

Example Dataset: Customer Satisfaction

Customer ID	Age	City	Monthly Spend	Satisfaction Level
1	29	New York	200	High
2	35	Los Angeles	-1	Medium
3		Chicago	150	Low
4	42	San Francisco	220	Medium
5	31	San Diegeo	185	High

Mean of all Customers

Monthly Spend
200
188
150
220
185

Group-based Mean (City)

Monthly Spend
200
210
150
220
185

Regression Model

Monthly Spend
200
205
150
220
185

ReClean

Pains

Proper selection of a well-suited data cleaning strategy requires data expertise

Automated repair methods may harm downstream ML models

ReClean

Gains

Select/combine cleaners based on their impact on downstream predictive tasks

Avoid Model/Data dependency

Example Dataset: Customer Satisfaction

Customer ID	Age	City	Monthly Spend	Satisfaction Level
1	29	New York	200	High
2	35	Los Angeles	-1	Medium
3		Chicago	150	Low
4	42	San Francisco	220	Medium
5	31	San Diego	185	High

Mean of all Customers

Monthly Spend
200
188
150
220
185

Group-based Mean (City)

Monthly Spend
200
210
150
220
185

Regression Model

Monthly Spend
200
205
150
220
185

software AG

ReClean

Formulate the task of selecting the best repair tools as RL problem

ReClean

Formulate the task of selecting the best repair tools as RL problem

States are batches of feature vectors of the input dirty data

Policy selects an action (i.e., repair tool) for a given feature vector

Reward is accuracy of target predictor on a validation set

ReClean

Formulate the task of selecting the best repair tools as RL problem

States are batches of feature vectors of the input dirty data

Policy selects an action (i.e., repair tool) for a given feature vector

Reward is accuracy of target predictor on a validation set

REINFORCE algorithm to optimize a cleaner selector network

ReClean

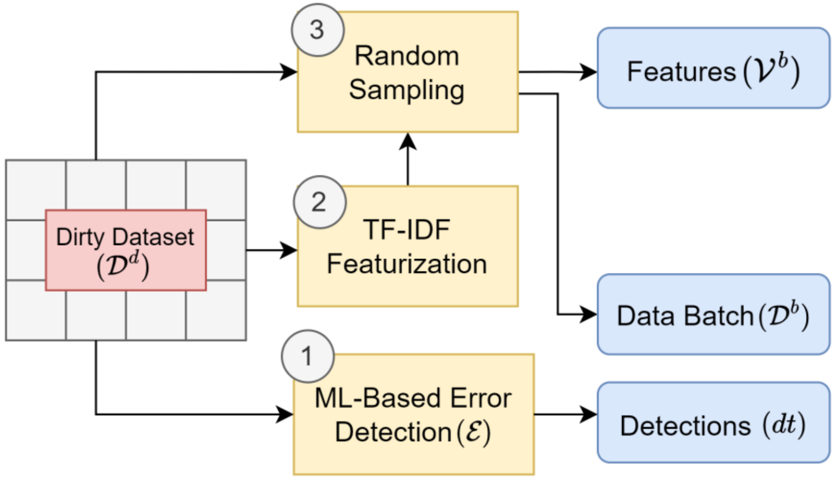
Formulate the task of selecting the best repair tools as RL problem

States are batches of feature vectors of the input dirty data

Policy selects an action (i.e., repair tool) for a given feature vector

Reward is accuracy of target predictor on a validation set

REINFORCE algorithm to optimize a cleaner selector network



ReClean

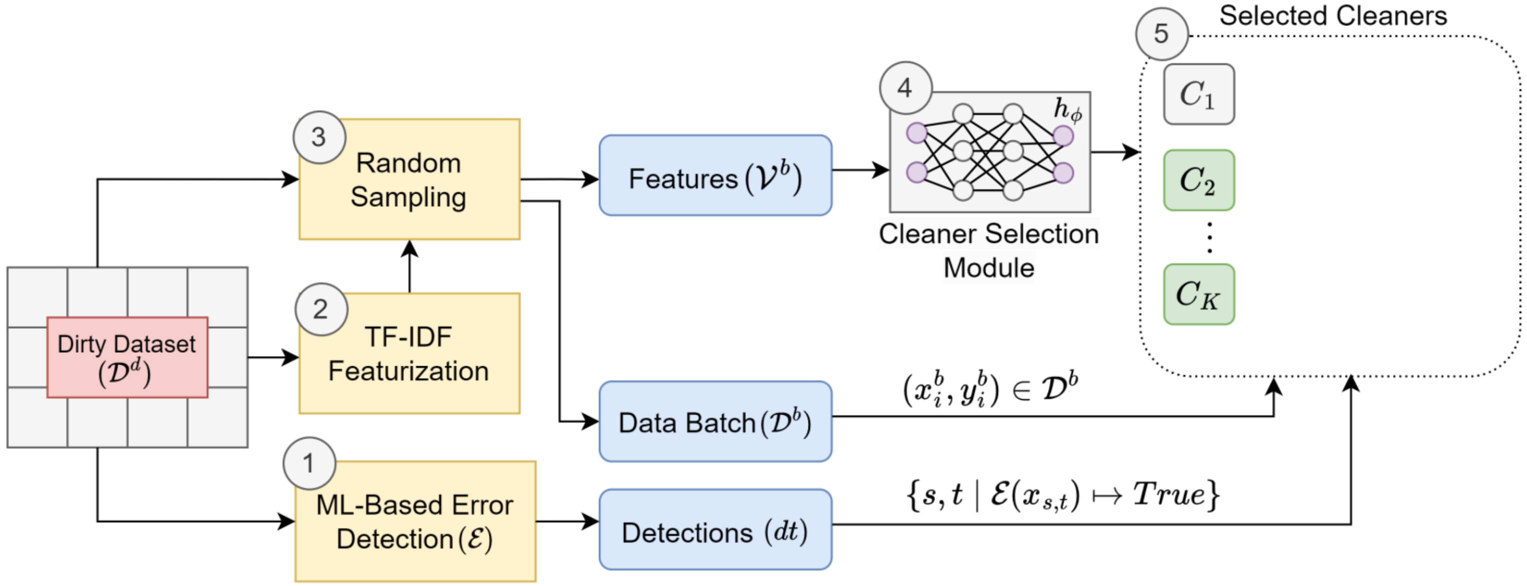
Formulate the task of selecting the best repair tools as RL problem

States are batches of feature vectors of the input dirty data

Policy selects an action (i.e., repair tool) for a given feature vector

Reward is accuracy of target predictor on a validation set

REINFORCE algorithm to optimize a cleaner selector network



ReClean

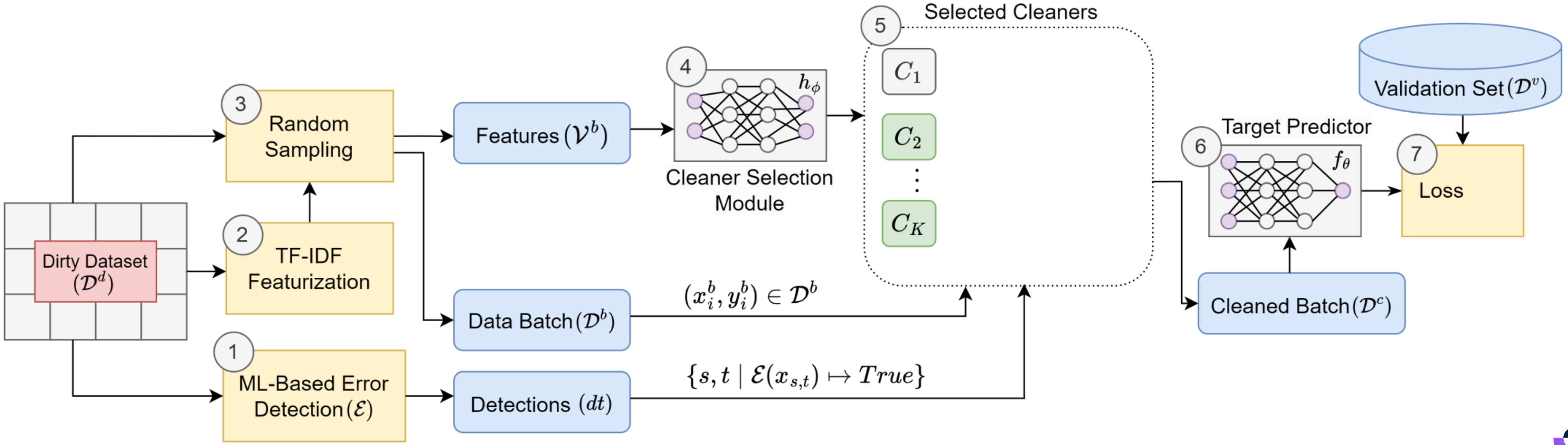
Formulate the task of selecting the best repair tools as RL problem

States are batches of feature vectors of the input dirty data

Policy selects an action (i.e., repair tool) for a given feature vector

Reward is accuracy of target predictor on a validation set

REINFORCE algorithm to optimize a cleaner selector network



ReClean

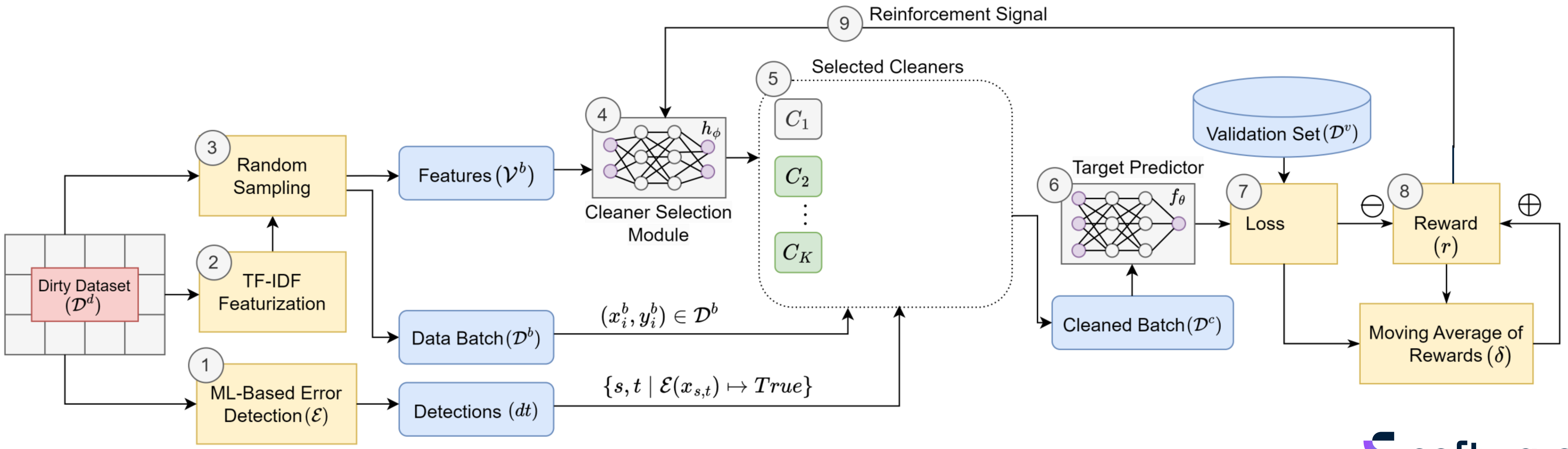
Formulate the task of selecting the best repair tools as RL problem

States are batches of feature vectors of the input dirty data

Policy selects an action (i.e., repair tool) for a given feature vector

Reward is accuracy of target predictor on a validation set

REINFORCE algorithm to optimize a cleaner selector network



ReClean

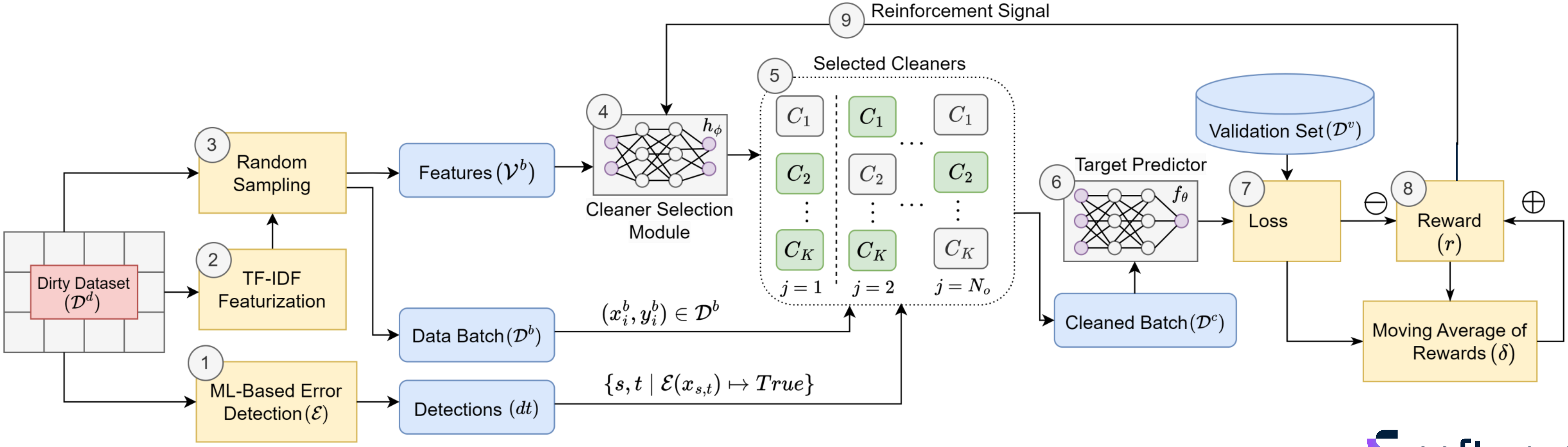
Formulate the task of selecting the best repair tools as RL problem

States are batches of feature vectors of the input dirty data

Policy selects an action (i.e., repair tool) for a given feature vector

Reward is accuracy of target predictor on a validation set

REINFORCE algorithm to optimize a cleaner selector network



ReClean

Optimization Problem

Minimize validation loss

$$\min_{h_\phi} \boxed{\mathbb{E}_{(\mathbf{x}^v, y^v)\sim P^t}\left[\mathcal{L}_h(f_\theta(\mathbf{x}^v), y^v)\right]}$$

$$\text{s.t. } f_\theta = \arg\min_{\hat{f}\in\mathcal{F}} \boxed{\mathbb{E}_{(\mathbf{x},y)\sim P^t}\left[\mathcal{L}_f\left(\hat{f}(\mathbf{x}_{h_\phi(\mathcal{V})}), y_{h_\phi(\mathcal{V})}\right)\right]}$$

Optimization of target model on repaired data

- Repair selection network h_ϕ
- Target model $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$
- Repaired data $\left(\mathbf{x}_{h_\phi(\mathcal{V}_i)}, y_{h_\phi(\mathcal{V}_i)}\right)$
- Validation set $(\mathbf{x}_i^v, y_i^v) \mid i \in \{1, 2, ..., K\} \sim \mathcal{P}^t$

ReClean

Reward Estimation

$$R = \boxed{\mathcal{L}_h\ \left(f_\theta(x^v), y^v\right)} - L_{movAvg}$$

Current validation loss

ReClean

Reward Estimation

$$R = \boxed{\mathcal{L}_h \ (f_\theta(x^v), y^v)} - \boxed{L_{movAvg}}$$

Current validation loss

Moving average
of previous losses

ReClean

Reward Estimation

$$R = \boxed{\mathcal{L}_h\left(f_\theta(x^v), y^v\right)} - \boxed{L_{movAvg}} + \boxed{\varepsilon_{explore}}$$

Current validation loss

Moving average
of previous losses

Regularization term
to force exploration

ReClean

Reward Estimation

$$R = \boxed{\mathcal{L}_h\ (f_\theta(x^v), y^v)} - \boxed{L_{movAvg}} + \boxed{\varepsilon_{explore}}$$

Current validation loss

Moving average of previous losses

Regularization term to force exploration

$$\boxed{y_{pred}} = \begin{bmatrix} p_{11} & \cdots & p_{1m} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nm} \end{bmatrix}$$

Predictions of the cleaner selector network

n, m denote the number of tuples within batch and the number of available repair tools

ReClean

Training Algorithm

Require: Mini-batch size B_s, number of iterations for RL agent N_O, number of iterations for predictor N_I, dirty training dataset $\mathcal{D}^d$, validation dataset $\mathcal{D}^v$, feature vectors $\mathcal{V}^d$, moving average window $T > 0$

Initialize parameters ϕ, θ, moving average δ = 0

for j = 1, ..., N_O **do**

  Sample a mini-batch of samples from the dirty training dataset and their corresponding feature vectors: $\mathcal{D}^b$ = $(\mathbf{x}_i, y_i)_{i=1}^{B_s}$ and V^b = $(\mathcal{V}_i)_{i=1}^{B_s}$

  Output cleaners $C_i = h_\phi(\mathcal{V})$

  Apply cleaners on the samples of $\mathcal{D}^b$: $\mathcal{D}^c = (\tilde{\mathbf{x}}_i, \tilde{y}_i)_{i=1}^{B_s}$

  for j = 1, ..., N_I **do**

    Update the parameters of the predictor network

$$\theta \leftarrow \theta - \alpha \frac{1}{B_s} \sum_{i=1}^{B_s} \nabla_\theta \mathcal{L}_f(f_\theta(\tilde{\mathbf{x}}_i, \tilde{y}_i))$$

  Update the parameters of the cleaner selector

$$\phi \leftarrow \phi - \beta \frac{1}{B_s} \left[\sum_{i=1}^{B_s} [\mathcal{L}_h(f_\theta(\mathbf{x}_i^v, y_i^v)) - \delta \right] \nabla_\theta \log \pi_\phi(\mathcal{V}^b)$$

  Update the moving average baseline: $\delta \leftarrow \frac{T-1}{T}\delta + \frac{1}{LT}\sum_{j=1}^{K}[\mathcal{L}_h(f_\theta(\mathbf{x}_j), y_j)]$

ReClean

Training Algorithm

Require: Mini-batch size B_s, number of iterations for RL agent N_O, number of iterations for predictor N_I, dirty training dataset $\mathcal{D}^d$, validation dataset $\mathcal{D}^v$, feature vectors $\mathcal{V}^d$, moving average window $T > 0$

Initialize parameters ϕ, θ, moving average $\delta = 0$

for j = 1, ..., N_O **do**

  Sample a mini-batch of samples from the dirty training dataset and their corresponding feature vectors: $\mathcal{D}^b = (\mathbf{x}_i, y_i)_{i=1}^{B_s}$ and $V^b = (\mathcal{V}_i)_{i=1}^{B_s}$

  Output cleaners $C_i = h_\phi(\mathcal{V})$

  Apply cleaners on the samples of $\mathcal{D}^b$: $\mathcal{D}^c = (\tilde{\mathbf{x}}_i, \tilde{y}_i)_{i=1}^{B_s}$

  for j = 1, ..., N_I **do**

    Update the parameters of the predictor network

$$\theta \leftarrow \theta - \alpha \frac{1}{B_s} \sum_{i=1}^{B_s} \nabla_\theta \mathcal{L}_f(f_\theta(\tilde{\mathbf{x}}_i, \tilde{y}_i))$$

  Update the parameters of the cleaner selector

$$\phi \leftarrow \phi - \beta \frac{1}{B_s} \left[\sum_{i=1}^{B_s} [\mathcal{L}_h(f_\theta(\mathbf{x}_i^v, y_i^v)) - \delta \right] \nabla_\theta \log \pi_\phi(\mathcal{V}^b)$$

  Update the moving average baseline: $\delta \leftarrow \frac{T-1}{T}\delta + \frac{1}{LT}\sum_{j=1}^{K}[\mathcal{L}_h(f_\theta(\mathbf{x}_j), y_j)]$

Sample a data batch

Predict probabilities of repair tools

Repair each tuple

Update target model parameters

Update cleaner selector network parameters

Update moving average loss

software AG

Performance Evaluation

Experimental Setup

What is the accuracy of ReClean compared to the baseline tools?

Performance Evaluation

Experimental Setup

What is the accuracy of ReClean compared to the baseline tools?

What is the impact of increasing the error rate on ReClean and the baselines?

What is the number of repair tools employed by ReClean?

Performance Evaluation

Experimental Setup

What is the accuracy of ReClean compared to the baseline tools?	What is the impact of increasing the error rate on ReClean and the baselines?	What is the number of repair tools employed by ReClean?

- Six real-world datasets with regression & classification tasks
  - errors injected with different rates (typos, missing values, Gaussian noise)
- ED2 has been used for detecting errors
- Cleaner-selection network is a four-layer feed-forward neural network with ReLU activation
  - # hidden units adjusted according to the dimensionality of the feature vectors
- Ubuntu 20.04 LTS machine with 16 2.60 GHz cores and 64 GB memory.

Performance Evaluation

Experimental Setup

What is the accuracy of ReClean compared to the baseline tools?

What is the impact of increasing the error rate on ReClean and the baselines?

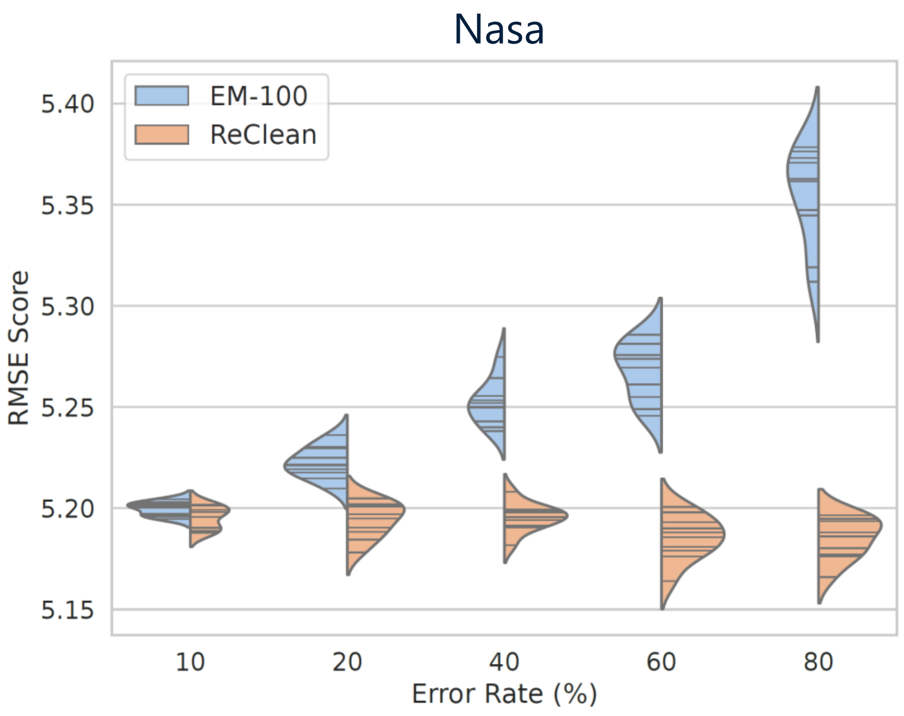
What is the number of repair tools employed by ReClean?

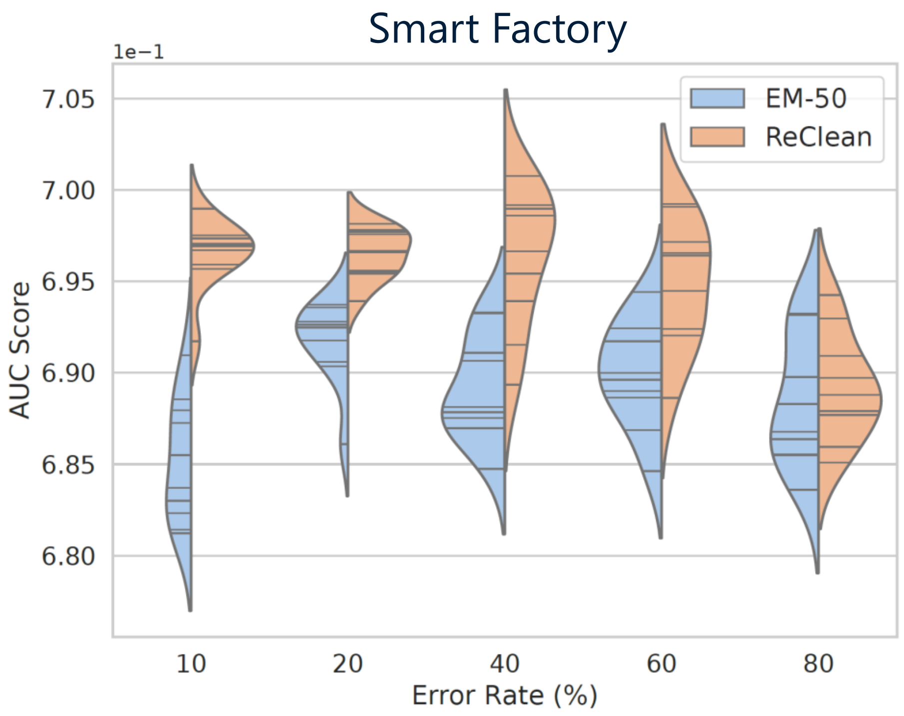
- Six real-world datasets with regression & classification tasks
  - errors injected with different rates (typos, missing values, Gaussian noise)
- ED2 has been used for detecting errors
- Cleaner-selection network is a four-layer feed-forward neural network with ReLU activation
  - # hidden units adjusted according to the dimensionality of the feature vectors
- Ubuntu 20.04 LTS machine with 16 2.60 GHz cores and 64 GB memory.

Baseline Method	Configured Parameter
Mean Imputer	-
Median Imputer	-
KNN Imputer (1)	number of neighbors
KNN Imputer (2)	number of neighbors
KNN Imputer (3)	number of neighbors
EM Imputer (1)	number of iterations
EM Imputer (2)	number of iterations
Bayesian Ridge Imputer	-
MissForest Imputer (1)	number of trees in the forest
MissForest Imputer (2)	number of trees in the forest
MissForest Imputer (3)	number of trees in the forest

Performance Evaluation

Accuracy

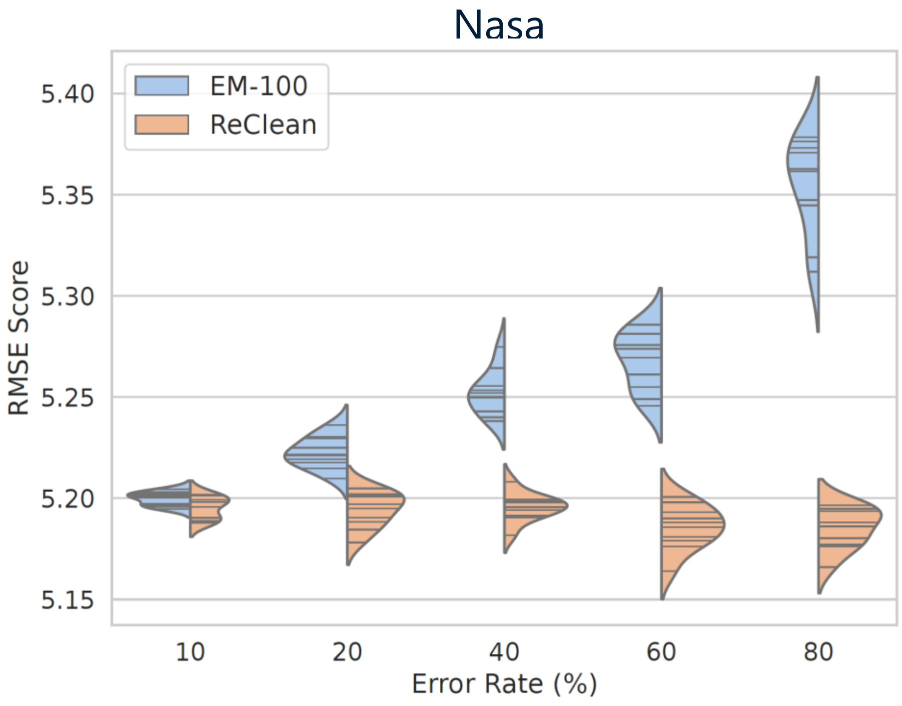


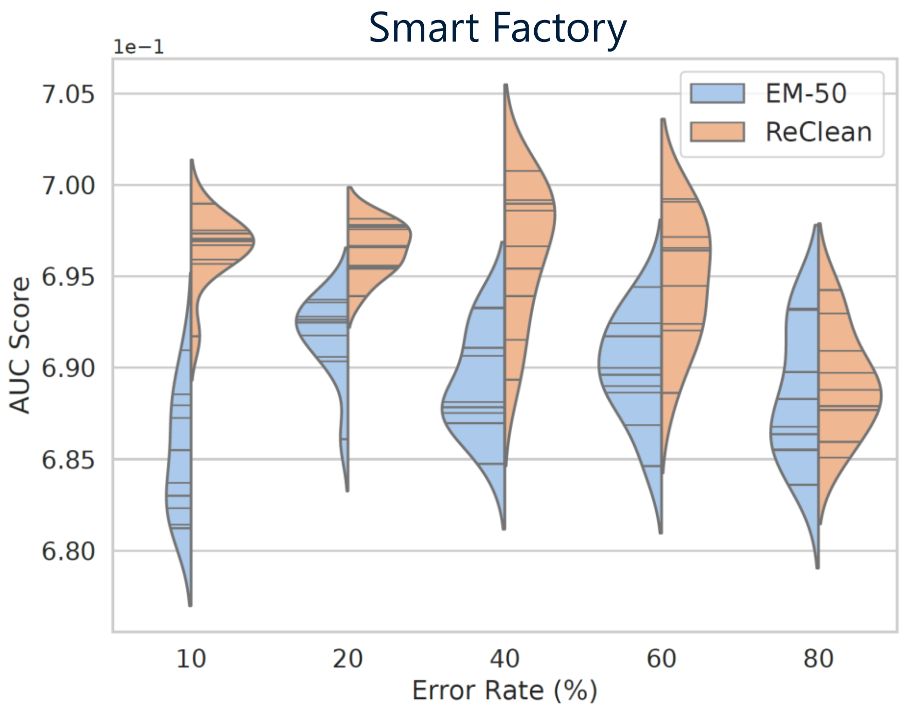


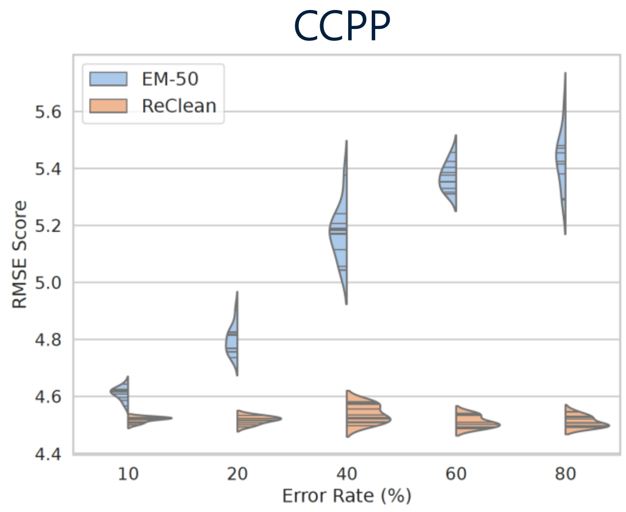
- ReClean consistently outperforms the leading repair tool across different datasets
- Increasing the error rate has no/slight influence on the performance of ReClean

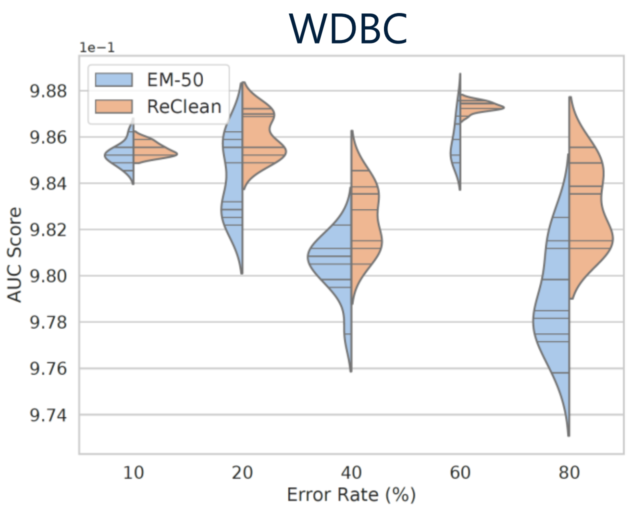
Performance Evaluation

Accuracy









- ReClean consistently outperforms the leading repair tool across different datasets
- Increasing the error rate has no/slight influence on the performance of ReClean

Performance Evaluation

Number of repair tools employed by ReClean

"M" and "Std" denote the mean and standard deviation of ten experiments and γ represents the error rate

	Smart Factory		WDBC		Nasa		Wine		CCPP		Retail	
$\gamma(\%)$	M	Std	M	Std	M	Std	M	Std	M	Std	M	Std
10	**3.2**	1.53	3.1	1.51	**2.9**	1.49	**4**	1.94	**3.2**	1.3	4.1	1.3
20	2.6	1.01	**3.5**	1.20	2.7	1.26	2.8	0.6	2.7	1.00	**4.4**	1.11
40	2.5	0.92	2.7	1.00	2.7	0.9	2.4	0.91	3	1.18	3.6	1.35
60	2.3	0.78	2.3	0.9	2.6	1.04	1.9	0.83	3.1	1.60	4.3	0.91
80	2.7	0.9	2.1	0.53	2.6	0.74	1.9	0.94	3	1.18	3.5	0.80

- Mean ranging from 1.9 to 4.4 across different datasets and error rates, suggesting a tailored approach to error correction for each scenario
- Highest average number of tools used tends to occur at the lowest error rate

Conclusion & Future Work

Conclusions

- ReClean is a RL-based method for jointly optimize data cleaning and downstream predictive tasks
- ReClean consistently outperforms baseline methods across various datasets

    - ReClean selects repair tools at the tuple level, improving the granularity and precision of data cleaning

    - ReClean requires 2.53 Min compared to 20.3 Min for DiffML for the Nasa data set

Conclusion & Future Work

Conclusions

- ReClean is a RL-based method for jointly optimize data cleaning and downstream predictive tasks
- ReClean consistently outperforms baseline methods across various datasets
  - ReClean selects repair tools at the tuple level, improving the granularity and precision of data cleaning
  - ReClean requires 2.53 Min compared to 20.3 Min for DiffML for the Nasa data set

Limitations

- ReClean relies on the performance of error detection tools
- REINFORCE algorithm has a relatively high variance, which makes the gradient estimates noisy

Conclusion & Future Work

Conclusions

- ReClean is a RL-based method for jointly optimize data cleaning and downstream predictive tasks
- ReClean consistently outperforms baseline methods across various datasets

  - ReClean selects repair tools at the tuple level, improving the granularity and precision of data cleaning
  - ReClean requires 2.53 Min compared to 20.3 Min for DiffML for the Nasa data set

Limitations

- ReClean relies on the performance of error detection tools
- REINFORCE algorithm has a relatively high variance, which makes the gradient estimates noisy

Future Work

- Explore other RL algorithms, e.g., Actor-Critic algorithm
- Extend the selection network to consider error detection and repair tools

Conclusion & Future Work

Conclusions

- ReClean is a RL-based method for jointly optimize data cleaning and downstream predictive tasks
- ReClean consistently outperforms baseline methods across various datasets

  - ReClean selects repair tools at the tuple level, improving the granularity and precision of data cleaning

  - ReClean requires 2.53 Min compared to 20.3 Min for DiffML for the Nasa data set

Limitations

- ReClean relies on the performance of error detection tools
- REINFORCE algorithm has a relatively high variance, which makes the gradient estimates noisy

Future Work

- Explore other RL algorithms, e.g., Actor-Critic algorithm
- Extend the selection network to consider error detection and repair tools

GitHub

software AG

ReClean

REINFORCE Algorithm

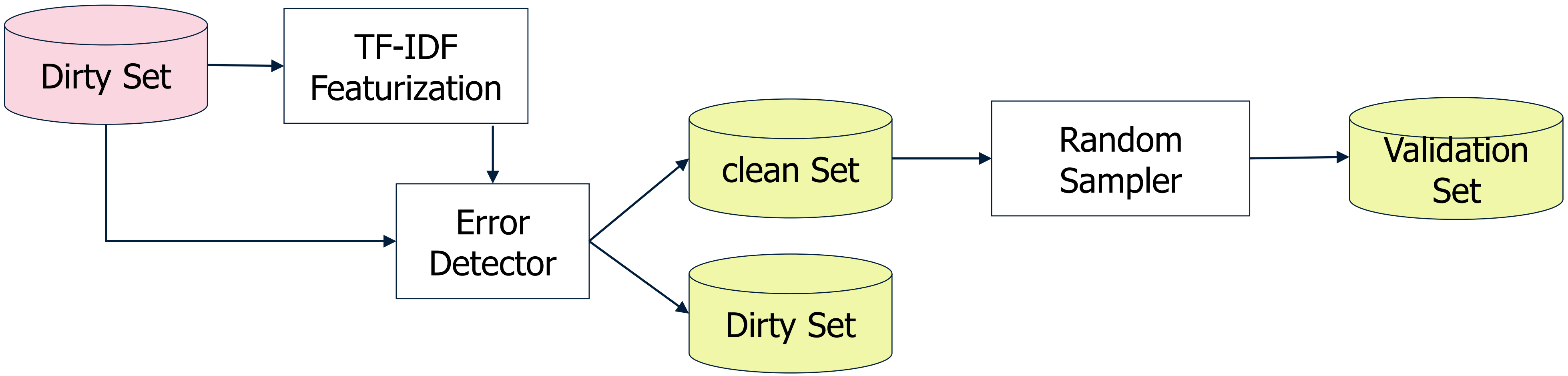
- It was introduced by Ronald Williams in 1992
- It is a **Monte Carlo** method for learning policies in environments with sparse, delayed rewards
  - It learns what actions will lead to the best outcomes through **trial and error**
    - trying out different actions & observing the outcomes, then using those observations to update the policy to choose better actions in the future

- 1. Initialize policy parameters θ
- 2. For each episode:
  - 1. Initialize state s
  - 2. While the episode is not over:
    - 1. Sample action a from the policy π(a|s; θ)
    - 2. Take action a and observe reward r and next state s'
    - 3. Store (s, a, r) in replay buffer
    - 4. Set s = s'
  - 3. Compute discounted return G for each time step t in the episode
  - 4. Calculate gradient of expected return with respect to policy parameters
  - 5. Update policy parameters θ
- 6. Return policy

ReClean

Validation Set

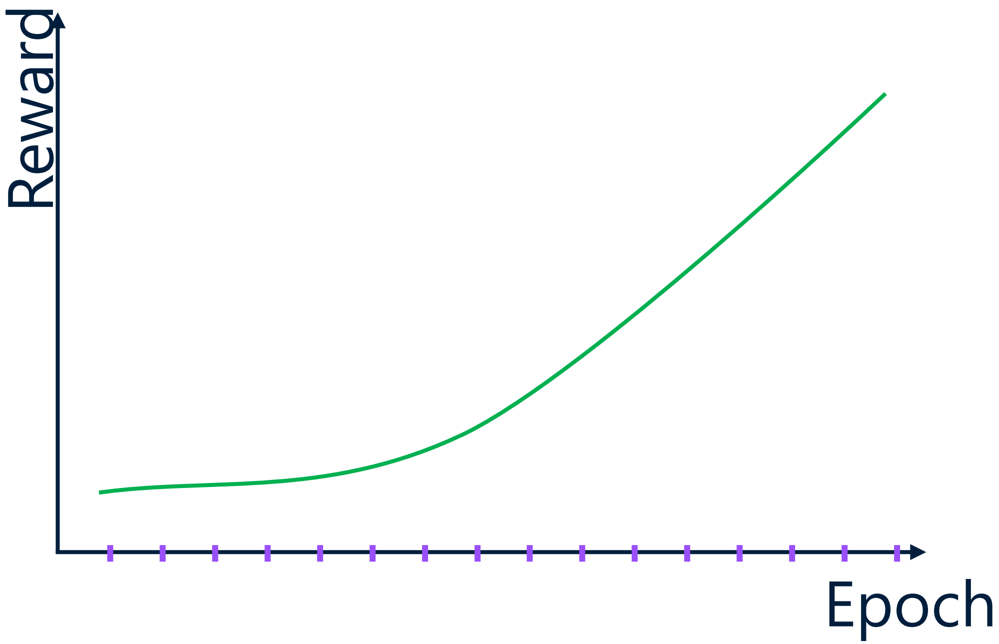
- Validation set is used to estimate the reward
- It is created through extracting a clean fraction from the dirty data and then randomly sample the clean fraction



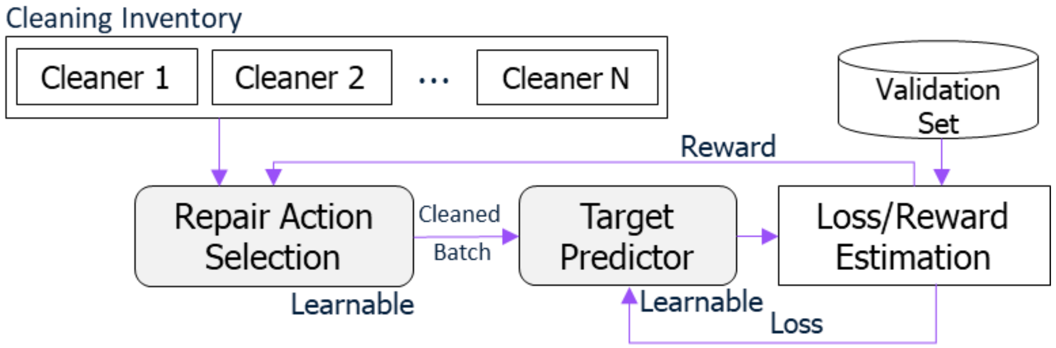
ReClean

Run-time Problem

- An "epoch" refers to one full pass through the entire training dataset
- To estimate the loss, we need to **use the repaired data** to train the target predictor
  - This implies executing all repair tools on the batches **at each epoch**
  - Number of epoch → at least 2000
  - Executing all repair tools in each epoch highly **increases the runtime** of the proposed invention



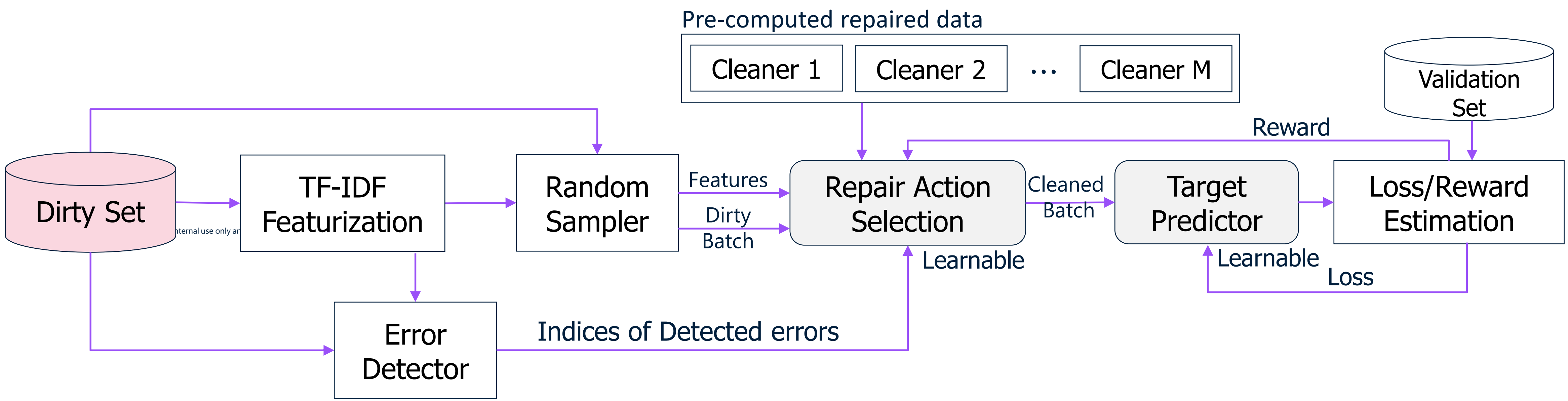
Stochastic Gradient Descent



ReClean

Run-time Trick

- Instead of executing repair tools on the batches at each epoch
  - we turn it to a simple assignment operation
  - Before training, a list of repaired datasets by each single cleaner
  - Based on the selected repair tools in a batch, we replace the dirty samples with their repaired versions obtained from the pre-prepared repaired datasets

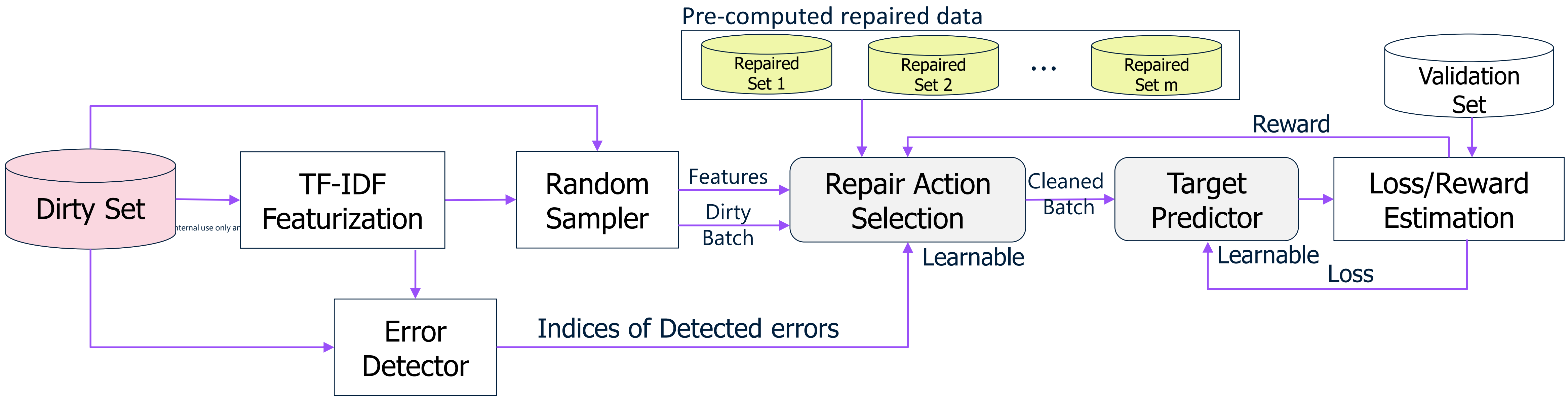


software AG

ReClean

Run-time Trick

- Instead of executing repair tools on the batches at each epoch
  - we turn it to a simple assignment operation
  - Before training, a list of repaired datasets by each single cleaner
  - Based on the selected repair tools in a batch, we replace the dirty samples with their repaired versions obtained from the pre-prepared repaired datasets



State of the Art

- ML-Oriented Data Cleaners

- Exploit ML model training and inference to select the best repair candidates
  - Employ a pool of already existing error detection and repair methods

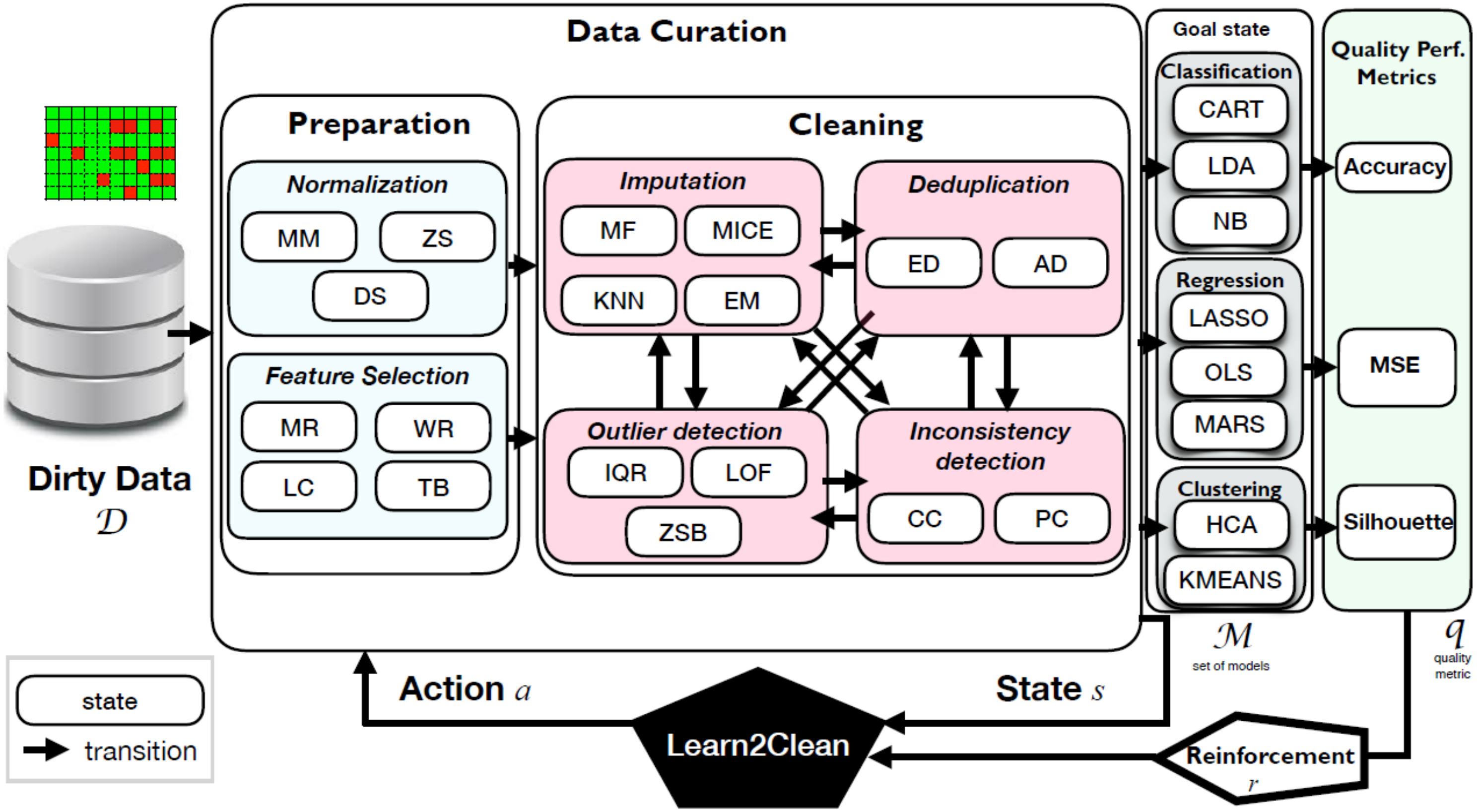
ActiveClean	BoostClean	CPClean
AL module to select data samples which help the ML model to converge	Ensemble learning based on models trained on different repaired versions of the data	Conditional entropy of training ML models using different repaired versions

- Challenges
  - Not able to combine repair candidates
  - No learnable modules which can be used after deployment
  - Complexity increased → additional method to select best repair candidates
  - Tailored to specific ML models, e.g., CPClean limited to KNN models

State of the Art

- RL-based Data Preparation

- Learn2Clean (automated sequencing)
  - model-free RL technique that selects a ML model, and a quality performance metric, the optimal sequence of tasks for preprocessing the data such that the quality of the ML model result is maximized



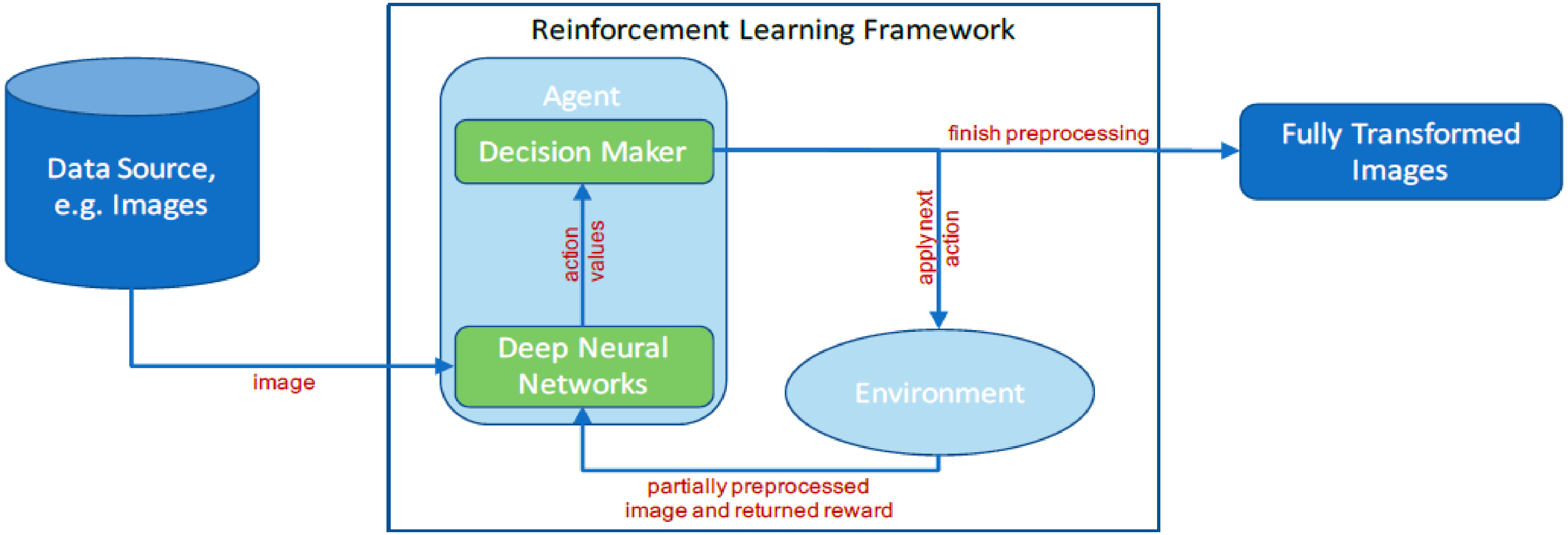
- Challenges
  - Limited to the available ML models
  - High time and computational complexity
  - Not able to combine repair candidates from multiple repair tools

State of the Art

- RL-based Data Preparation

- Automated Image Data Preprocessing with Deep Reinforcement Learning (2018)
  – transformations such as cropping, filtering, rotating or flipping images

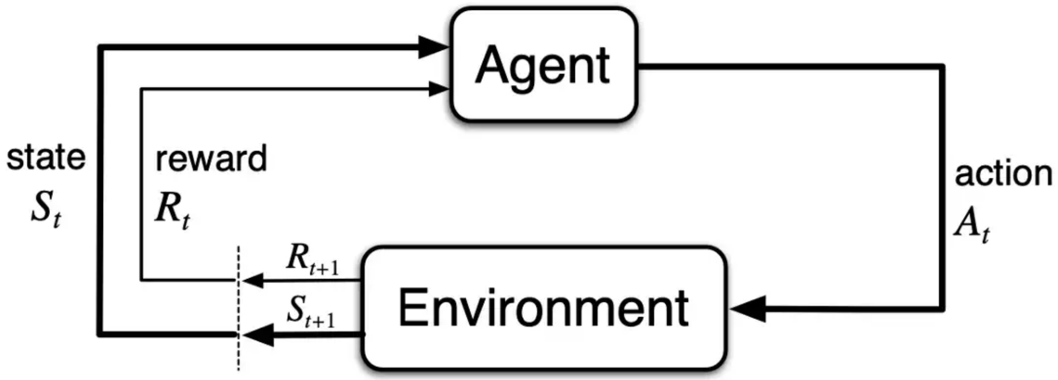
- Challenges
  - **Limited to images** & cannot be used with tabular data



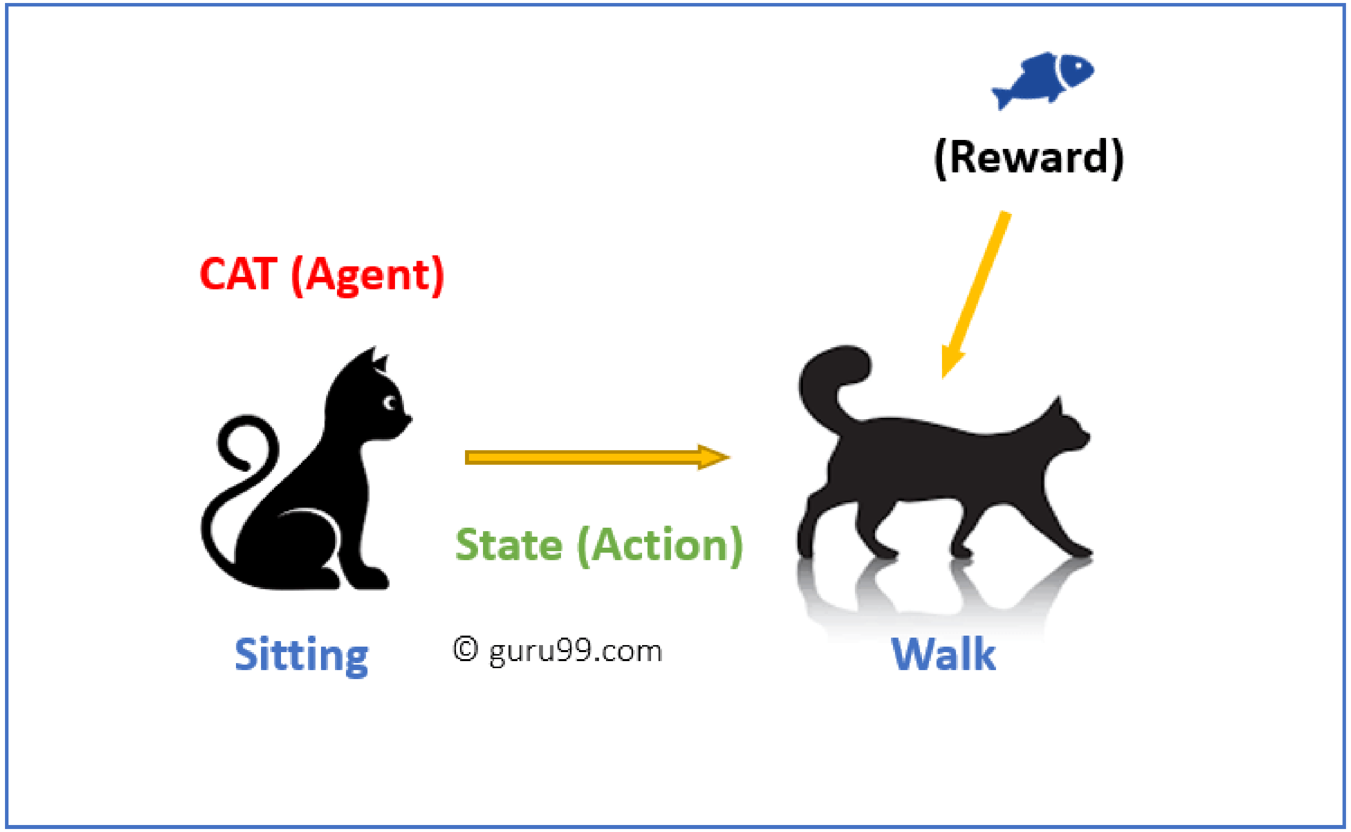
Introducing RL

RL Elements & Example

- To use RL, the following parameters need to be defined
  - **Set of actions**
  - **Set of states**
  - **Reward function:** function used to generate the reward at a certain state
  - **Policy:** method to decide the next action based on the current state
  - **Value:** It is expected long-term return with discount

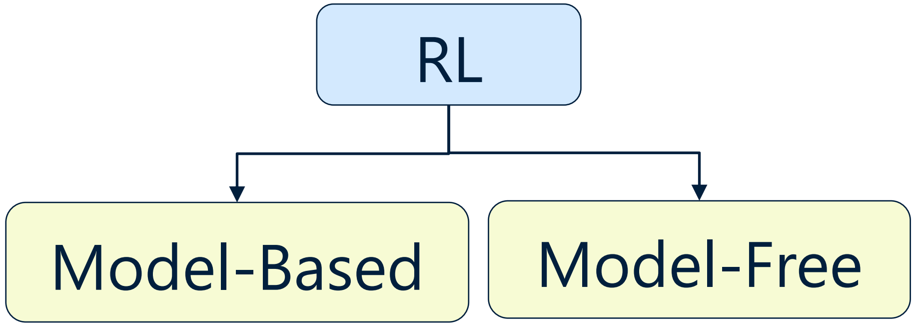


House (environment)



Introducing RL

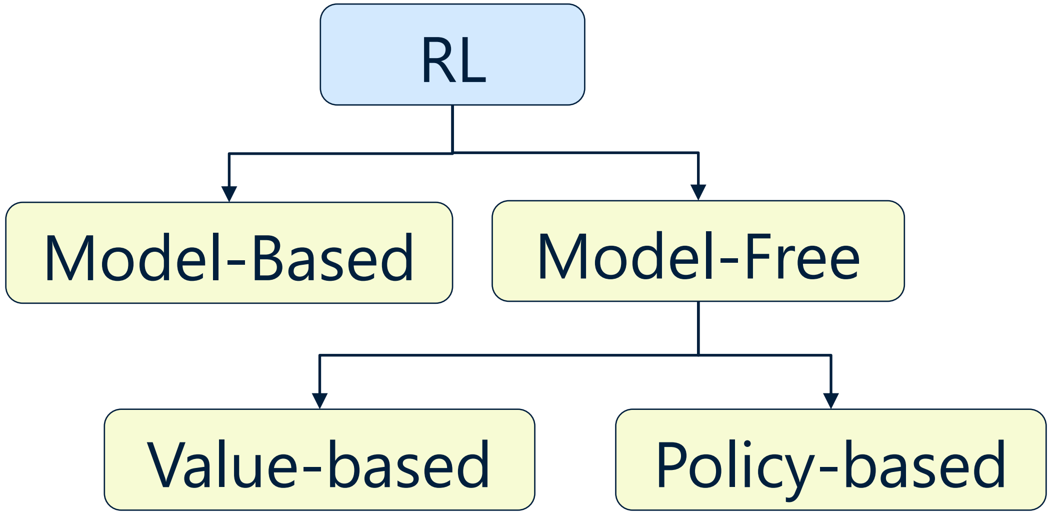
Introduction to Reinforcement Learning --- Comparison of RL Methods



Model-Based	Model-Free
The agent **learns a model** of the environment, including the dynamics and reward function.	The agent **does not learn a model** of the environment, but instead learns from experience.
The agent can use the model to **plan and evaluate** different actions and policies.	The agent learns directly from the rewards and transitions experienced during **interaction with the environment.**
The agent may require **more data and computational resources** to learn the model.	The agent may require **less data** and **computational resources** but may also be **slower to learn**.
The agent may be more **sample efficient**, as it can reuse the learned model for multiple tasks.	The agent may be **less sample efficient**, as it must learn a new policy for each task.

Introducing RL

Comparison of RL Methods



Criteria	Value-Based	Policy-Free
Nature of the learned function	**Value function** that estimates the expected return of each state or state-action pair.	**Policy function** that defines the action to take in each state.
How the learning process is performed	It involves estimating the value function, and then using it to derive the policy.	It involves directly learning the policy function.
How the policy is derived	**Indirectly** (from the value function by choosing the action with the highest value in each state)	**Directly** (without learning a value function)
Examples	**Q-learning**, SARSA, Deep Q-Network (DQN), Double Q-learning, Dyna-Q, Expected Sarsa, True Online Sarsa, Absolute Baseline	**REINFORCE**, actor-critic (e.g., A2C, A3C, PPO), Trust Region Policy Optimization (TRPO), Natural Policy Gradient (NPG), Soft Actor-critic
Suitability for continuous action spaces	- less suitable for continuous action spaces	+ more suitable for continuous action spaces
Sample efficiency	+ more sample efficient	- less sample efficient
Potential for instability or oscillation	+ more stable, as they rely on the value function, usually smoother than the policy	- less stable, as they directly optimize the policy, which may be more noisy
Applicability to partially observable environments	- less suitable for partially observable environments.	+ more suitable for partially observable environments.