Making Maintenance Smart

Zurich, Switzerland

amplo;

Content

- Introduction
- Data-Centric AI
- Why MLOps
- How we use MLOps
- Our toolset and implementation
- Conclusion

INTRODUCTION

Amplo

Jun '20 Founded Amplo

Predictive Maintenance & manufacturing digitization projects
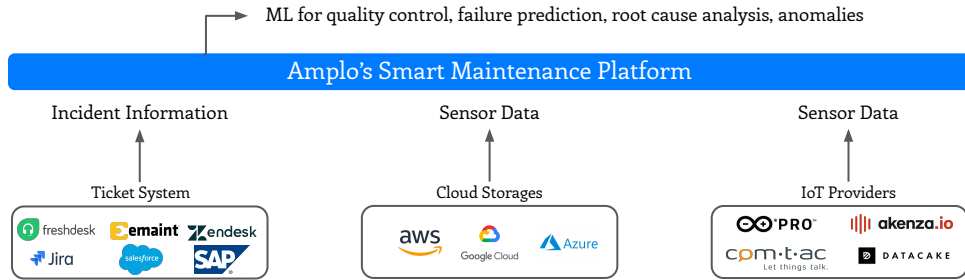
Feb '20 Acceleration program Bluelion

Jun '21 Platform launch

Dec '21 Pre-seed funding

Always Looking for pilots!

INTRODUCTION

Smart Maintenance Platform

No-code platform to connect any type of industrial machine data and develop & manage operational machine learning without any programming or maths

Smart Maintenance Platform

1. Connect Data

Using MQTT, HTTP, OPC UA, Blob storages, data dumps, custom adapters to allow binary data, text, images

2. Get Incident Labels

Extract automatically from ticket system, assisted with unsupervised ML or manually

3. Start our AutoML

In-house developed pipeline that automates the full cycle of model development.

4. Automatic Serving

An in-house developed framework with robust testing and monitoring systems allows for continuous deployment of new models.

Smart Maintenance Platform

Service Technicians
Know machines inside out
No ML or coding expertise

Amplo
No domain knowledge
Team of ML engineers

Amplo's Vision with Data-Centric AI

Expert in specific machine

Organise & label data

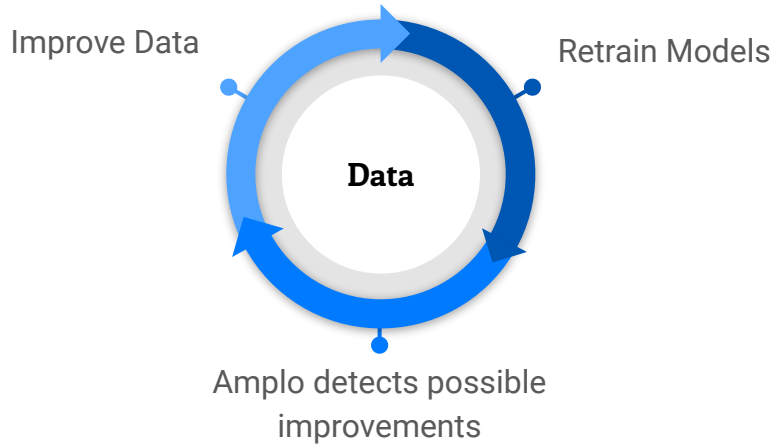Expert in ML(Ops)

Provide tools & framework

User

Amplo

Amplo's Vision with Data-Centric AI

Improve Data

Retrain Models

Data

Amplo detects possible improvements

Systems:
- Assisted data labeling
- Automated feature engineering
- Automated model training
- Automated deployment
- Integrated performance monitoring
- Integrated version control
- Automated Testing

User

Amplo

First Customer Story

Before Amplo:

Diagnosing issues takes ~12 working hrs

With Amplo:

Automated Diagnosis with Machine Learning
→ reassign 80% service engineers

TRITIUM

Manufacturer of EV chargers
Maintains 6.700 chargers worldwide

First Customer Story

Onboarding & testing phase:

Took nine months due to **slow iterations** and **unobserved** issues

Issues:

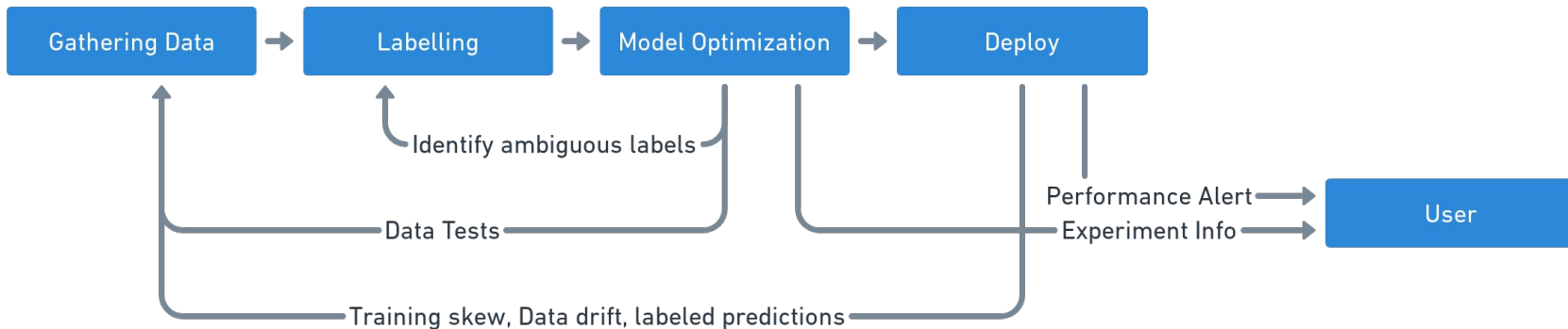|	
Too little data	→ Enforced & assisted labelling
CAN conversion	→ Automated data tests
US data unaccounted	→ Monitoring & inspection
Module naming	→ Prediction interpretability / accountability

Our Focus on MLOps

Our MLOps drives:

- Quicker model-to-market
    - And therefore shorter sales cycles
    - Quicker integration, adoption and value gain
    - Faster upsell
- Better ML models
    - Faster iteration cycles
    - Deep insight into ML
    - Robust framework
    - Continuous improvement

# Deploy first iteration within first week!

- Expected label
- Recommend unlabeled
- Enforce feedback
- Identify label mistakes

- Data cleaning
- Feature experimenting
- Model experimenting
- Hyperpar. Optimization
- Experiment Tracking

- Data monitoring
- Performance monitoring
- Main predictors analysis
- API endpoints
- Version control

Gathering Data	→	Labelling	→	Model Optimization	→	Deploy

Identify ambiguous labels

Data Tests

Performance Alert → User
Experiment Info →

Training skew, Data drift, labeled predictions

Assisted Labeling
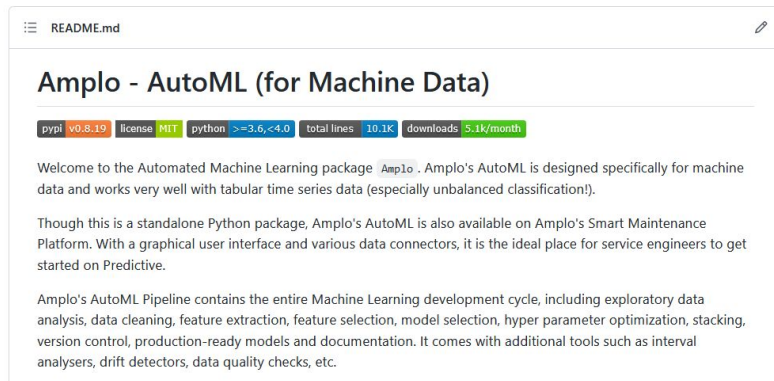
Overcoming time-consuming data gathering

- Exploit production & unlabeled data
- Fast label identification
  - Query engine to mark sequences
  - Clustering for recommended label
- Efficient attention
  - Recommend underrepresented data
  - Representations for specific selection
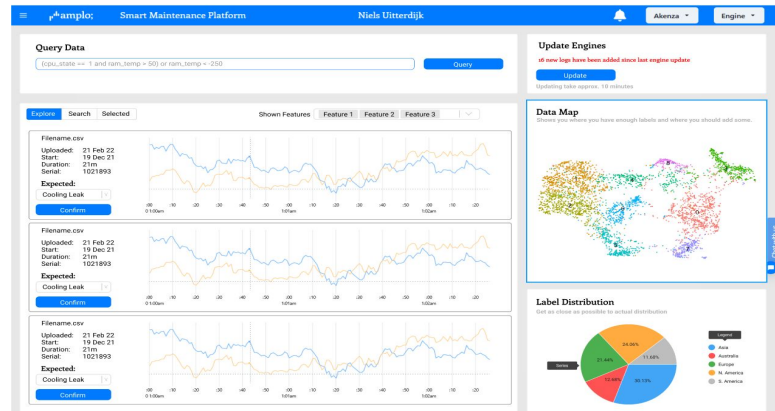
Automated Machine Learning Package

Puts model development on autopilot!

- Design steps identical
    - Feature engineering
    - Model selection
    - Parameter optimization
- Predefined steps & search space
- Robust framework with testing
- Built-in monitoring
- Built-in interpretability analysis
- Jobs deployed on self-destroying spot instance

README.md

Amplo - AutoML (for Machine Data)

`pypi v0.8.19` `license MIT` `python >=3.6,<4.0` `total lines 10.1K` `downloads 5.1k/month`

Welcome to the Automated Machine Learning package `Amplo`. Amplo's AutoML is designed specifically for machine data and works very well with tabular time series data (especially unbalanced classification!).

Though this is a standalone Python package, Amplo's AutoML is also available on Amplo's Smart Maintenance Platform. With a graphical user interface and various data connectors, it is the ideal place for service engineers to get started on Predictive.

Amplo's AutoML Pipeline contains the entire Machine Learning development cycle, including exploratory data analysis, data cleaning, feature extraction, feature selection, model selection, hyper parameter optimization, stacking, version control, production-ready models and documentation. It comes with additional tools such as interval analysers, drift detectors, data quality checks, etc.

AutoML - Testing

Data

- Collinearity
- Monotonically in/de-creasing
- Index leakage
- Minority sensitivity
- In-sample errors
- Regression tests
- Extreme values
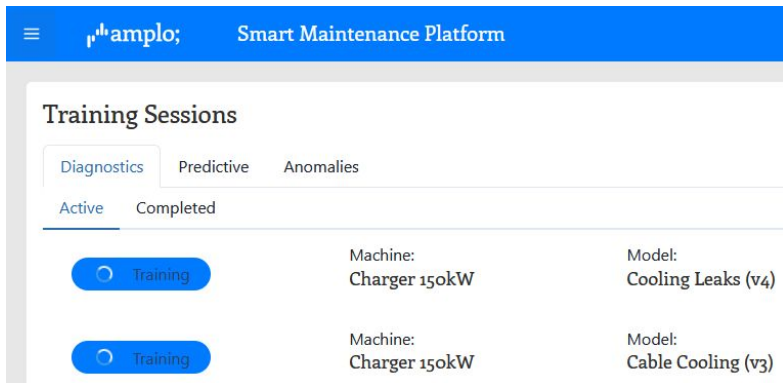- Odd statistics

Model

- Slice testing
- Reproducibility
- Better than linear model
- Numerical stability
- Serving latency
- RAM usage
- Invariance tests

And then of course pipeline integration & regression tests, code unit & integration tests, etc.
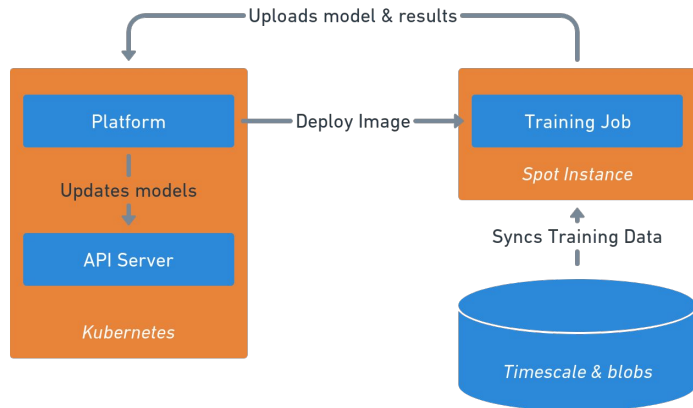
Automatic Deployment

Make continuous improvements as easy as possible!

- Results of AutoML directly uploaded to platform
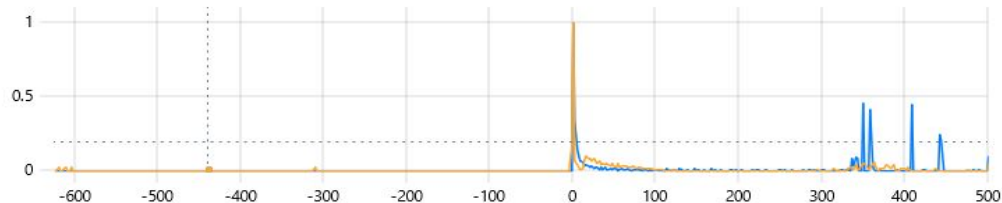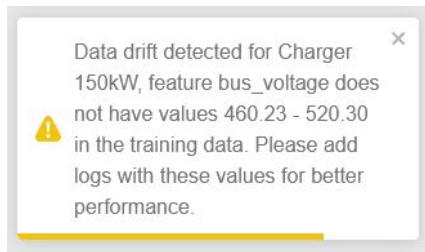- API endpoints check for possible updates

Performance Improvement

Notified when & instructed on how to improve:

- Data drift or training / production skewness is detected
- Ambiguous labels
- RAM / time increase
- Performance analysis

Did it help?

Onboarding speed
→ Two models deployed within first two weeks
→ Value and implementation quick and easily understood

Iteration speed
→ Customers enjoy unlimited re-training
→ Average ~5 iterations for robustness

Machine Learning without coding / knowledge!
→ Machine expert gathers / labels data
→ Rest on autopilot!
→ Our engineers can focus on infra, not customer specific problems!

Contact **Niels Uitterdijk**
Email **niels.uitterdijk@amplo.ch**
Mobile **+41 79 58 26 875**

Zurich, Switzerland

amplo;