Cloud solutions for predictive maintenance

Speaker: Dr. Volodymyr Andrushchak, Data Science Team Lead at Lemberg Solutions

Goal of the presentation

Predictive maintenance is a technique to detect abnormalities in the operation of your equipment so you can fix them before these deviations result in machinery failure.

Solve predictive maintenance tasks (to automate the detection of deviations in wind turbines' operation) using two approaches

- Fully automated (AWS Lookout)
- Manual (AWS EC2)

Compare by following parameters

- Accuracy
- Price
- Scalability
- Deployment and modification

Plan

- Goal of the presentation
- Dataset description
- AWS Lookout
- AWS EC2
- Comparison of the results
- Our investigation
- Conclusions

Specifics of predictive maintenance for wind turbines

- Collecting potential failure examples for each type of turbine is complicated as such cases are rare
- It is too expensive to simulate such cases for data collection
- Weather data plays a significant role in turbine failure prediction and should therefore be considered in the model
- Collecting data on turbine failures, even of one type, is impossible under all potential weather conditions

Dataset description

- Use dataset from wind turbines telemetry - **https://opendata-renewables.engie.com/**
- The dataset contains over 1 million of data samples and has information from 4 wind turbines
- It has 34 sensors such as torque, power, speed, temperature, frequency, voltage, etc.
- In this case, only one turbine was used for model training
- Also some sensors data, which contain more than 10% of missing values, were dropped (260K data samples and 112 sensors data are remaining)
- The dataset has about 7 years of telemetry, 80% of which were used for training and 20% for accuracy evaluation
- The model is configured to detect anomalies with a sampling rate of 10 minutes

```
Variable_name;Variable_long_name;Unit_long_name;Comment
Q;Reactive_power;kVAr;
Ws;Wind_speed;m/s;Average wind speed
Va2;Vane_position_2;deg;Second wind vane on the nacelle
Git;Gearbox_inlet_temperature;deg_C;
Ot;Outdoor_temperature;deg_C;
Ws2;Wind_speed_2;m/s;Second anemometer on the nacelle
Nf;Grid_frequency;Hz;
Nu;Grid_voltage;V;
Dst;Generator_stator_temperature;deg_C;
Wa_c;Absolute_wind_direction_corrected;deg;
DCs;Generator_converter_speed;rpm;
Yt;Nacelle_temperature;deg_C;
Na_c;Nacelle_angle_corrected;deg;
Ya;Nacelle_angle;deg;
Rm;Torque;Nm;
Gost;Gearbox_oil_sump_temperature;deg_C;
Rs;Rotor_speed;rpm;
Gb2t;Gearbox_bearing_2_temperature;deg_C;
Wa;Absolute_wind_direction;deg;
Ba;Pitch_angle;deg;
Ds;Generator_speed;rpm;
Va;Vane_position;deg;
Db2t;Generator_bearing_2_temperature;deg_C;
Cm;Converter_torque;Nm;
Rt;Hub_temperature;deg_C;
Ws1;Wind_speed_1;m/s;First anemometer on the nacelle
S;Apparent_power;kVA;Should be the square root of the su
P;Active_power;kW;
Cosphi;Power_factor;;Should equal P/S
Gb1t;Gearbox_bearing_1_temperature;deg_C;
Db1t;Generator_bearing_1_temperature;deg_C;
Va1;Vane_position_1;deg;First wind vane on the nacelle
Pas;Pitch_angle_setpoint;;
Rbt;Rotor_bearing_temperature;deg_C;
```

AWS Lookout for equipment

LEMBERG SOLUTIONS

Set up AWS Lookout tool

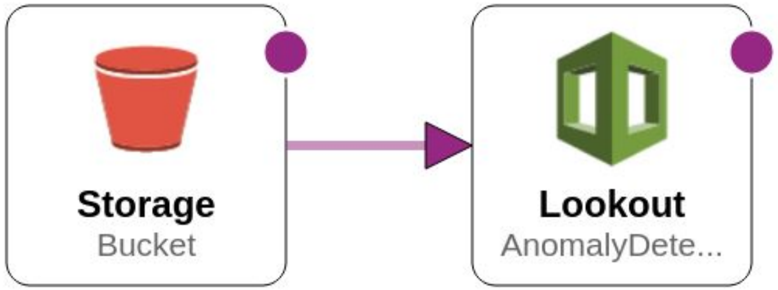
1. **Select necessary wind turbine dataset**
2. **Data cleaning (drop rows with missing data)**
3. **Feature engineering (work with sensor data)**
4. **Split dataset into training and test part**
5. **Model training**
6. **Run test data and evaluate results**

Dataset description

- 112 input parameters used
- Data preparation and feature engineering not needed
- Lookout examines appropriate AI model, selects needed parameters, and trains the model

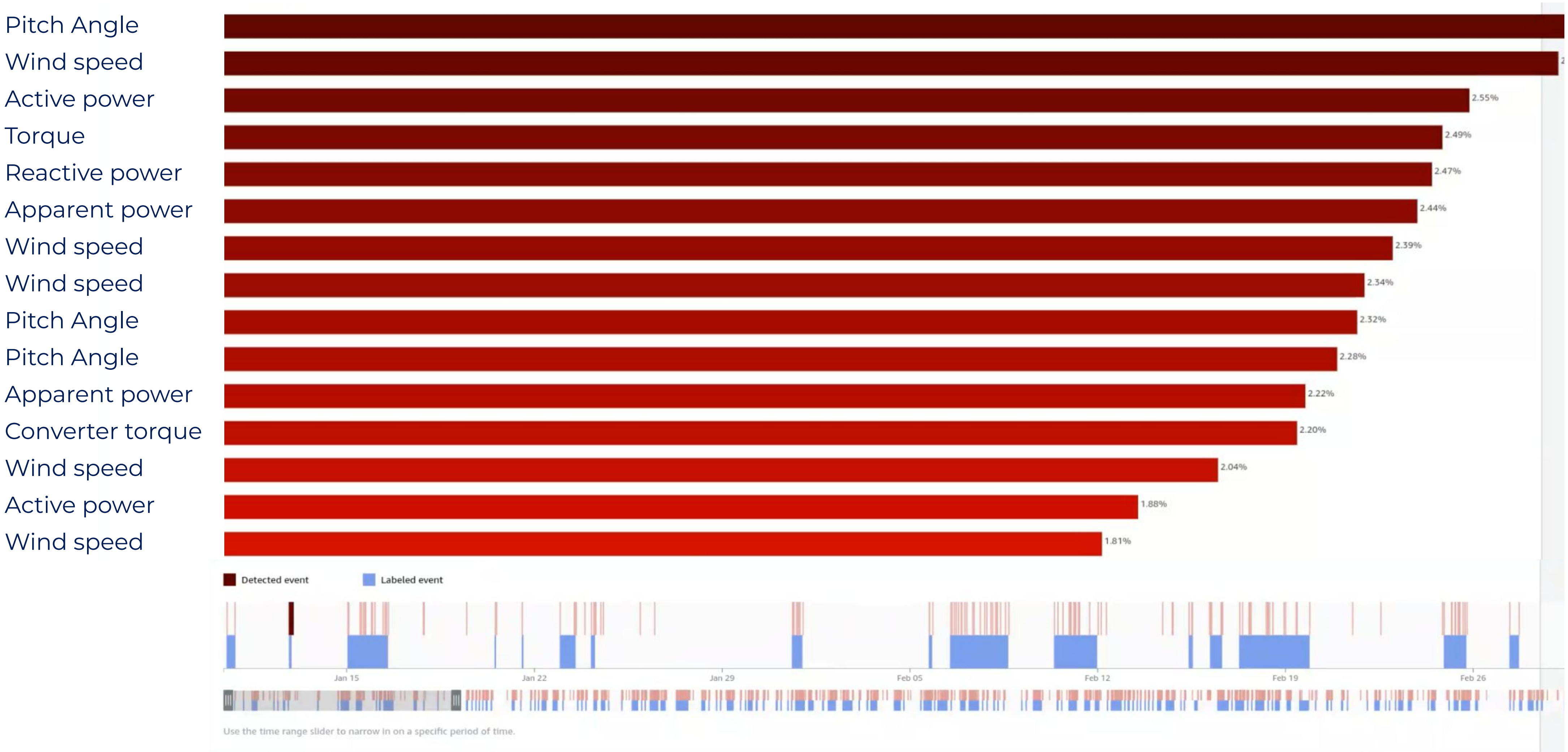
AWS Tools

- Lookout for Equipment
- S3



Visual result of AWS Lookout tool

Top 15 contributing sensors



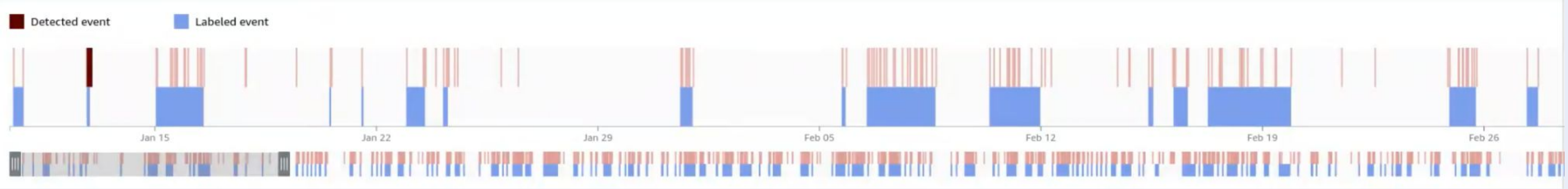
Evaluation of the results

- The dataset labels represent only one-class anomaly (binary classification)
- The dataset source does not explain the nature of deviations
- Even though the trained model produces a lot of fragmented detections, almost all labeled anomalies were successfully detected (92.7% accuracy)

- Data Ingestion - $0.20 per GB
- Model Training - $0.24 per training hour
- Scheduled Inference - $0.25 per inference hour

For our case:

- Data ingestion: 0.20 $ (in our case dataset size in 58MB)
- Model Training: 0.36$ (1.5 hours of training)
- Scheduled Inference 0.25 * 365 (days) * 24 (hours) = 2190 $



AWS virtual machine

Setting up EC2 and evaluating results

Dataset and model preparation

- Use all 112 input parameters
- Use all input parameters like in case with AWS Lookout
- Build custom NN (binary classification)
- Train it on Virtual Machine

input_1 (InputLayer)	[(None, 1, 112)]
lstm (LSTM)	(None, 1, 42)
lstm_1 (LSTM)	(None, 42)
signal_out (Dense)	(None, 1)
class_out (Dense)	(None, 2)

Results

- Data Ingestion - $0.20 per GB
- EC2 (2 vcpus, 4 GiB memory) $0.0416 per hour

For our case:

- Data ingestion: $0.20 (in our case dataset size is 58MB)
- Scheduled Inference 0.0416 * 365 (days) * 24 (hours) + 0.2 = $455

Accuracy: 86.75%

AWS Tools

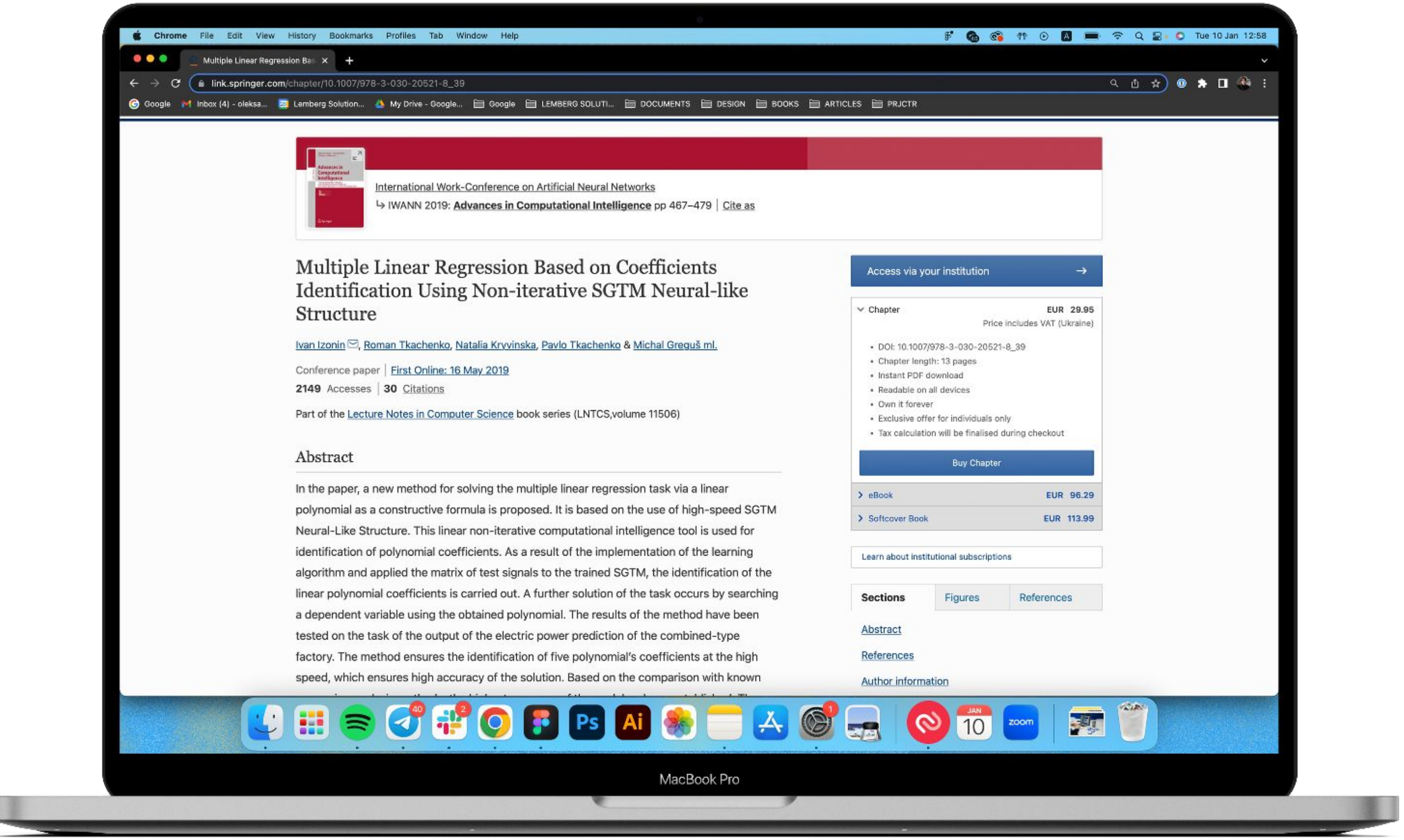
- EC2
- S3

Comparison of the results (one turbine)

Parameter	AWS lookout	AWS EC2
Accuracy, %	92.7	86.7
Price (one year)	2190$	455$ (full time VM)
Engineer’s time	24 hours	24 hours
Scalability (turbines)	Easy	Medium
Scalability (input data)	Easy	Medium
Manual or automated work	Automated (training and deploy model)	Manual (training and deploy)

How has it become possible? SGTM neuro-like structure

(Successive Geometric Transformation Machine)



- Non-iterative training analyzes large amounts of data at high speed
- Provides remarkable accuracy and complete repeatability of the results under the same conditions
- Establishes unambiguous relationships between input parameters (when they are present)
- The SGTM network can solve the dichotomy problem by only being trained on one class of objects.

It can detect even the most minor deviations in the dependencies between the input parameters

Problem is solved

Conclusion

- We compared two ways of implementing the task of predictive maintenance on AWS (manual and automated)
- AWS Lookout allows building a scalable and high-quality model and system quickly
- EC2 enables you to build a more optimal model for a specific case

About us

Lemberg Solutions is a tech consulting and engineering company that creates **IoT products, Digital experiences, AI solutions and provides SAP services.**

Our clients rely on our full suite of engineering services and industry experience to build new products and bring digital transformation to existing systems.

200
cross-functional technical experts

15
years on the market

Lviv HQ Office

72 Heroiv UPA street,
79018, Lviv, Ukraine

Hamburg Office

Am Sandtorkai 32, 20457,
Hamburg, Germany

OLGA LYSAK
CEO of Lemberg Solutions GmbH
olga.lysak@lembergsolutions.com

VOLODYMYR ANDRUSHCHAK
Data Science Team Lead
volodymyr.andrushchak@lembergsolutions.com