1 About us

Company overview:

- Name: Atomic Intelligence d.o.o.
- Address: Bencekovićeva 33, 10000 Zagreb

Atomic Intelligence d.o.o. is company with headquarters in Zagreb (HR) with primarily focus on EU and USA markets and specialization in areas of data governance, data integration, data warehousing and business intelligence, implementation of data application, data mining, machine learning and artificial intelligence in traditional analytical systems but also in systems using Big Data environments and platforms. Beside services of implementing solution within before said subject areas, Atomic Intelligence d.o.o. (abbreviated AI) is offering services from area of strategic IT consulting, development of custom solutions tailored to specific user needs and expertise in areas of Enterprise and Solution architecture. AI employees have vast and multi-year/decades of practical experience in different industries which can be backed by relevant certificates and customer references.

2 Challenge Use-Case

We are proposing name of this challenge: **Use IoT to enable Predictive Maintenance**

3 Challenge Use-Case description

Apple's Siri, Microsoft's Cortana and Amazon's Alexa are first things which come to mind when we are considering how devices are communicating with us (people). This is all about to change with the wider adoption of the Internet of Things (IoT). The IoT "is the network of physical devices, vehicles, home appliances, and other items embedded with electronics, software, sensors, actuators, and connectivity which enables these things to connect and exchange data." (Wikipedia)

The IoT allows even big and dull machines like forklifts to give you feedback from their sensors. This feedback is stored in a database and is used not only to predict when the machine is about to fail, but also why.

We will use dataset which enables us the unique opportunity to listen to the "voice" of automated storage and retrieval systems used in some of distribution centers. You will explore their sensor data and investigate how it relates to machine failures.

3.1 Business problem

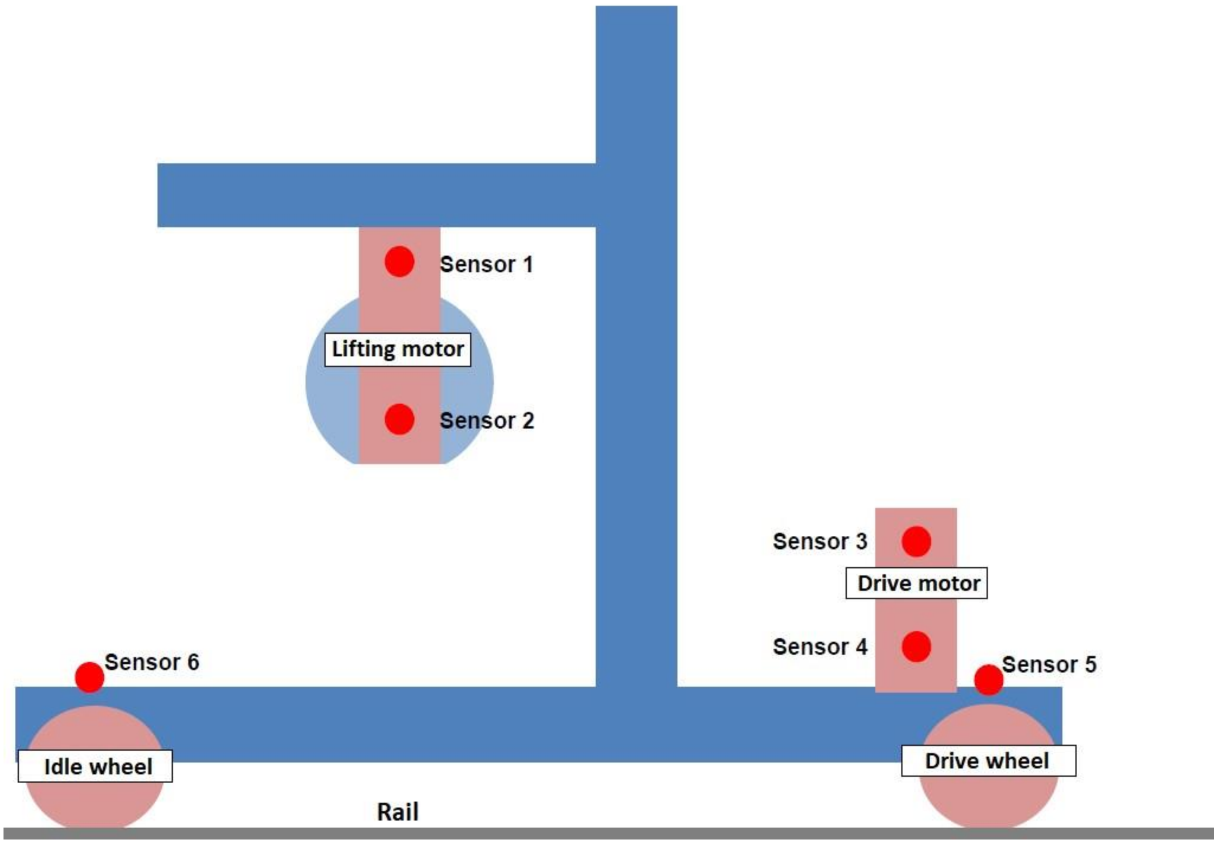
Failures of small parts in a machine can lead to a costly breakdown of the whole machine. Therefore, it is important to perform maintenance. The usual approach is to do it in fixed intervals. This leaves two risks: If the interval has been set too short, parts could fail just before maintenance is due, if it has been set too long, the maintenance does not find anything wrong. Furthermore, different parts have different lifespans, so a fixed interval for the whole machine is also not the most efficient way.

The approach that solves these issues is predictive maintenance. Here, historical data for a component is used to predict when it will fail. Therefore, the maintenance can be done focused on only the part that is about to fail and only just before it will do so, greatly reducing cost.

3.2 Challenge problem specification

The goal of this Challenge is to analyze data coming from seven automated storage and retrieval systems, large machines equipped with 6 different sensors.

The schema of the machine is as follows:



These sensors are located at the following positions:

- Lifting mechanism:
  - Sensor 1: Lifting motor
  - Sensor 2: Lifting gear
- Drive mechanism:
  - Sensor 3: Drive motor
  - Sensor 4: Drive gear
  - Sensor 5: Drive wheel
- Sensor 6: Idle (non-powered) wheel

On following picture, we see how the machine looks like in real life.

Figure 1 Warehouse forklift

All these sensors measure both the effective velocity of vibrations and the maximum acceleration of shocks. Each sensor collects data under controlled conditions three times a day. Two of the machines have collected data for almost two years, the other five for five months.

Suggestions for the analysis would be:

- Do you see any trends over the timespan?
- Are sensors correlated?
- Do they show similar trends?
- Is there evidence for disruptive events in the data?
- Could these events have been predicted with the data?
- Which sensor is the best to predict these failures?

We will also provide a few events, where large-scale maintenance was applied, like grinding of the rails or exchange of a whole wheel.

- Can we see in the data some trend that made this event necessary?
- Is something similar also visible for other machines?
- Do they need to be repaired as well?

3.3 Educative segment of this challenge

Developing solution for this challenge, you will have opportunity to understand and learn how different hardware components work and interact between themselves in large industrial warehouses. You will learn how to prepare data, see if there is need to create calculated/inferred features which are not directly present in provided dataset (like rolling averages of behavior, new calculations from few combined measurements, etc.), discard any unrelated piece of information which can complicate your modeling and output result.

This specific challenge task focuses on area where companies have huge revenue loss in terms of equipment malfunction and outage of services. The most expensive situation for company is when some equipment fails, then if they do regular scheduled maintenance (like yearly or after some mileage, car maintenance) because sometimes you replace completely correct (i.e. not broken or wear down) part, and most affordable is when you predict failure of part in advance and exchange it before failure happens.

There are multiple areas of applying solutions like this which are not tied to specific industry:

- Predicting failure of manufacturing equipment
- Predicting ware down of car parts (like windshield wipers, brake calipers, tires, etc.)
- Predict outage of computer/server parts,
- Predict outage of your end device (either IoT, mobile phone, or Internet device any Telco provider offers to you)
- etc.

4 Recommended programming languages

R & Python as Open Source solutions which can do whole end-to-end processing.

You can find all information for installation of selected programming language online because those depends on your local environment and OS. Some guidelines which can be helpful are:

- R: https://rstudio-education.github.io/hopr/starting.html
- Python: https://realpython.com/installing-python/

We recommend also to read CRISP-DM methodology ((http://www.sv-europe.com/crisp-dm-methodology) to prepare yourself with Data Science approach for gathering information and data, preparing them, modeling, evaluating and deploying.

First two phases of CRISP-DM methodology would be completed by us (**1. Business Understanding** and **2. Data Understanding**) while other steps (except last**: 6. Deployment**) are focus of this challenge.

5 Required Computing Resources

Minimum System Requirements

- Processors: Intel® Core™ i3 processor
- Disk space: 1 GB
- RAM: 4GB
- Operating systems: Windows* 7 or later, macOS, and Linux

Recommended System Requirements:

- Processors: Intel® Core™ i7 processor
- Disk space: 2-3 GB
- RAM: 8-16GB
- Operating systems: Windows* 7 or later, macOS, and Linux

Additional recommendation is to open Cloud trial (AWS, Azure, Oracle Cloud, Google Cloud) and use it for testing and building models if more computation power is required (our case will not require such computation requirements, but there is option to use Cloud environments)

6 Dataset example

Top 10 records for dataset created from sensor devices are as follows:

machine_name	sensor_type	date_measurement	start_timestamp	end_timestamp	realvalue	unit
RBG1	drive_gear_V_eff	2.9.2016	2.9.2016 15:26	2.9.2016 15:26	0.395	mm/s
RBG1	drive_gear_V_eff	2.9.2016	2.9.2016 15:26	2.9.2016 15:26	0.577	mm/s
RBG1	drive_gear_V_eff	2.9.2016	2.9.2016 15:26	2.9.2016 15:26	0.717	mm/s
RBG1	drive_gear_V_eff	2.9.2016	2.9.2016 15:26	2.9.2016 15:26	0.832	mm/s
RBG1	drive_gear_V_eff	2.9.2016	2.9.2016 15:26	2.9.2016 15:26	0.941	mm/s
RBG1	drive_gear_V_eff	2.9.2016	2.9.2016 15:26	2.9.2016 15:26	1.042	mm/s
RBG1	drive_gear_V_eff	2.9.2016	2.9.2016 15:26	2.9.2016 15:26	1.106	mm/s
RBG1	drive_gear_V_eff	2.9.2016	2.9.2016 15:27	2.9.2016 15:27	1.176	mm/s
RBG1	drive_gear_V_eff	2.9.2016	2.9.2016 15:27	2.9.2016 15:27	1.245	mm/s
RBG1	drive_gear_V_eff	2.9.2016	2.9.2016 15:27	2.9.2016 15:27	1.258	mm/s

7 Definition of dataset logical structure

The data will be provided as a structured CSV file, with about few million rows and 7 columns per row.

The columns are:

Name	Data type	Contents
machine_name	String	The name of the machine, ranging from "RBG1" to "RBG7".
sensor_type	String	The type of the sensor.
		Sensors are:
		drive_gear_V_eff
		drive_gear_a_max
		drive_motor_V_eff
		drive_motor_a_max
		drive_wheel_V_eff
		drive_wheel_a_max
		idle_wheel_V_eff
		idle_wheel_a_max
		lifting_gear_V_eff
		lifting_gear_a_max
		lifting_motor_V_eff
		lifting_motor_a_max
date_measurement	Date	The date of the measurement.
start_timestamp	Timestamp	The timestamp at the start of the measurement.

Name	Data type	Contents
end_timestamp	Timestamp	The timestamp at the end of the measurement.
realvalue	Decimal(18,6)	The measured value.
unit	String	The unit of the measurement. For all sensors measuring, it is “mg(milli-g)” (one thousandth of gravitational acceleration), for all sensors measuring, it is “mm/s”.

Besides dataset containing sensor event, we have data of events when large-scale maintenance was applied to specific machine. It is structured CSV file with following column description:

Name	Data type	Description
machine_name	String	The name of the machine.
date	String	The date or the range of dates when the repair event occurred.
repair_description	String	A description of the repair event.

8 Previous experience over proposed Challenge

We as Atomic Intelligence d.o.o. (company providing Challenge Use-Case) implemented and deployed project which had same or even more complex requirement like described in this Challenge Use-Case.

9 Additional information

To be provided after cases are announced….

10 Mandatory parts of completed solution

Mandatory parts of completed and delivered solution would be:

- Source code for building models, evaluating them and running them – raw code in selected programming language which allows execution and repeatability of delivered solution on different environment (all dependencies should be clearly stated in code)
- Project documentation in form of populated chapters as described within CRISP-DM methodology with focus on what was project understanding, issues with data, premise for modeling (and obstacles if any), selection criteria for training and evaluating models, output results, etc. – please refer to CRISP-DM for full description of process and required steps. – in PDF format
- Technical documentation in form of Run Book which contains information how should delivered solution be run to provide output result as described in Project documentation – in PDF format
- Final presentation in PPT or PDF format

Note:

- All code stays propriety of team (person or persons) who developed it and we as company providing Challenge Use-Case will use it only to validate results of delivered solution (e.g. can we repeat same results as described in delivered solution and user documentation)
- Data used for implementation of project is property of **Atomic Intelligence d.o.o.** and it can only be used for duration of this Challenge.
- Any unauthorized use of data would be strictly illegal and basis for private prosecution under Croatian laws.
- For use of data after Challenge is completed, please contact **Atomic Intelligence d.o.o.** for written permission.