Data Preprocessing for Energy Models

1. Introduction

The energy modelling for the CNC machines as well as the idle energy analytics requires the preprocessing of energy and production data. Table 1 gives the details of the data sources for energy and production data. It can be seen that the two data sources have different sampling rates. Therefore, preprocessing is required before the data can be used on the DENiM platform.

Table 1: Data sources for energy models and analytics

Data	Source	Sampling Rate	Remarks
Energy consumption of production assets	OSI Pi	1 min	Required for idle energy analytics and anomaly detection in energy consumption patterns. Data export from OSI Pi needs to follow certain constraints in order to be compatible with downstream services.
Production logs	OMS	N/A depends on production batches	Required for anomaly detection in energy consumption patterns.

2. Data

This section goes into the details of the data. The data dictionary is described which gives the details of the names of the various datapoints as well as the outputs defined within DENiM's semantic model.

2.1. Data Dictionary

The DENiM platform uses a complex semantic model to ensure accurate representation of the data within the platform. This also aids in the accurate implementation of the services that depend on the data. The following tables provide a complete set of data columns used in the platform with their descriptions.

Table 2 lists all the production assets within the scope of the project. Table 3 shows the related data fields on the platform and their description using LTHE0049 as an example. Note that while the data field `p1_time` is common to all assets, every asset has data fields corresponding to their energy usage, product names and quantities produced within one shift.

Asset Code in Platform	Type	Asset Code in Pilot System
p1_cpss0132	Cleanline	CPSS0132
p1_cpss0139	Cleanline	CPSS0139
p1_csaw004	Electric Saw	CSAW0004
p1_csaw006	Electric Saw	CSAW0006
p1_lthe0049	Lathe	LTHE0049
p1_lthe0067	Lathe	LTHE0067
p1_lthe0069	Lathe	LTHE0069
p1_lthe0070	Lathe	LTHE0070
p1_lthe0079	Lathe	LTHE0079
p1_lthe0086	Lathe	LTHE0086
p1_lthe0092	Lathe	LTHE0092
p1_lthe0096	Lathe	LTHE0096
p1_lthe0097	Lathe	LTHE0097
p1_lthe0116	Lathe	LTHE0116
p1_lthe0117	Lathe	LTHE0117
p1_lthe0130	Lathe	LTHE0130
p1_lthe0131	Lathe	LTHE0131
p1_lasr0031	Laser	LASR0031
p1_seal4004	Sealing Machine	SEAL4004

Table 2: Production assets within project scope

Data field in DENiM platform	Description
p1_time	Start time of the shift
p1_lthe0049_energy	Energy consumed by LTHE0069 in a shift in kWh
p1_lthe0049_prod_name	Name of the product(s) manufactured on LTHE0069 in a shift
p1_lthe0049_prod_qty	Quantity(-ies) of the product(s) manufactured on LTHE0069 in a shift

Table 3: A sample set of data fields related to the production assets

2. Data Preprocessing (Energy and Production)

As was seen in **Error! Reference source not found.**, the energy consumption and production logs have different sampling rates. Therefore, before the data can be used in the analysis, it needs to be pre-processed. Whereas the energy data is stored automatically at rate of 1 data point per minute, production records are logged by the operators manually within each shift. Thus, a time interval of one shift is used as the basis and the data are resampled to obtain the shift-wise aggregates. The following subsections detail the steps to process the production and energy data and merge them to obtain the required file which can then be uploaded to the DENiM platform for further analysis.

3.1. Production Data from OMS

The production data consists of logs made by the operators for each individual production operation. The records contain several fields of information. The preprocessing of this data requires the following steps:

a. The DENiM services require only the timestamps, product names, product quantities and the production assets are required. The other data fields need to be removed.
b. The production data logs are very voluminous as they contain data from all production assets, but the DENiM project has a limited number of assets within scope. The platform's services can only cater to these assets. Thus, data entries corresponding to other assets need to be removed.
c. Last but not the least, the data needs to be aggregated shift-wise so that it can be analysed to generate meaningful insights.

The jupyter python notebook titled *process_prod_data.ipynb* demonstrates the procedure to pre-process the production data from OMS. It uses the sample data file titled *prod_oms.csv*, however, any other file can be used in its place, provided that the correct filename is used in the code.

3.2. Energy Data from OSI Pi

The energy data file exported from OSI Pi consists of two columns of data corresponding to each asset within the scope of the project. Over the course of the project, it was found that the format of the data file has minor variations depending on the commands used to export the data from OSI Pi and generate the data file. Therefore, the script developed to preprocess the data was designed to be agnostic to the specific data format. The code uses the column names to automatically determine the columns corresponding to a particular asset.

N.B. The user MUST ensure that the data file contains all the production assets given in Table 2. If any assets are missing, it may cause errors in the downstream data processing scripts and/or digital services.

The jupyter python notebook titled *process_energy_data.ipynb* demonstrates the procedure to process the energy data file exported from OSI Pi. It uses the sample data file titled *energy_osi_pi.xlsx*

3.3. Merging Energy and Production data

The above two sections described the procedure to pre-process the production and energy data individually. However, the platform requires the data to conform to the semantic model. As per this model, the energy and production data share for all production assets share a common timestamp. Furthermore, duplication of timestamps is not allowed. This applies to certain assets where multiple types of products are produced within the same shift. In such cases, the entries corresponding to different product types need to be merged into one list for each timestamp. Therefore, the energy and production data need to be merged accordingly.

The jupyter python notebook titled *merge_energy_production.ipynb* demonstrates this procedure. It uses the

processed files generated in the above two steps described in subsections 3.1 and 3.2.