PMI – vision document

Version 15/12/2022

M.G. Cimino

The purpose of the project is the **development of a software factory** based on the "Machine Learning Ops" paradigm, a DevOps approach able to deploy and maintain machine learning models and IT operations in production reliably and efficiently.

1.PROCESS LANDSCAPE

What to provide: using diagrams.net, provide only the final image of the process landscape (like the telephone company)

Some **activities** of the software factory, for the purpose of determining the **process landscape**, are the following:

1.1) configure service: set up information systems, configure information systems, configure client-side applications

1.2) ingest data (process only data flow, not values): synchronize samples, join samples in session, dispatch session, mark missing samples, discard incomplete sessions (threshold), generate/deliver raw session;

1.3) prepare data: correct missing samples, detect and correct absolute outliers, extract features, generate/deliver prepared session;

1.4) segregate data: collect prepared sessions (threshold), check data balancing (torelance), check data quality (radar diagram), generate/deliver training, validation and test sets (percentage)

1.5) develop service: early training with average hyperparameters, adjust number of generations of the gradient descent plot (flat curve condition), validation via grid search (hyperparameter optimization), use validation set, select top 5 classifier (compare training and validation errors), select the winner classifier, asses testing error against validation error, deploy classifier

1.6) execute service: take raw session, prepared session, provide the class;

1.7) monitor service: compare the class provided by human expert and by the classifier, over consequent inputs (threshold), and decide to develop when accuracy is not sufficient)

GENERAL ADVICES ON PROCESS LANDSCAPE

1.8) the cardinality 1:1 means that, in the happy case, one execution of the activity on the left corresponds to one execution of the activity on the right

1.9) (development phase): data ingestion > 1:1 > data preparation > M:1 > data segregation > 1:M > architecture training (training set) > M:1 > model validation (validation set) > 1:1 > model testing (testing set) > 1:1 > model deployment

1.10) (execution phase): data ingestion > 1:1 > data preparation > 1:1 > model execution > M:1 > performance monitoring

1.11) vocabulary: use "architecture" instead of "model", because the concept of model is used when you have to compare different architectural models, in our case we develop one architecture whose model is already selected.

1.12) vocabulary: use business vocabulary e.g. not "data" or "time series" or "time window" but "current EEG activity". I suggest to assign to a member the role of "nomenclator" (person in charge of finding names), another member will study the process landscape methodology, and so on

1.13) meeting management: separate in talking session and thinking session (even if you are connected), e.g. 10 min. of individual activities (to find names etc.) and 5 minutes of discussion and review.

A POSSIBLE RESULTS OF PROCESS LANDSCAPE

1.14) the cardinality M:1 between training and validation is not significant for the business level, it is an algorithmic aspect => we could add in the same process

> PREPARE RECORD/SESSION
> GENERATE R/S LEARNING SETS
> DEVELOP CLASSIFIER
> CLASSIFY RECORD/SESSION
> CHECK CLASSIFIER PERFORMANCE

1.15) when systems are configured?

> CONFIGURE SERVICE (register, configure, …)

2.SAMPLE TRELLO TASKS

a) MANAGEMENT:
Title: *Setup and configure all tools*
Description: *Setup and configuration of MS Teams, MS Word, Trello, Clockify, Diagrams.net, Signavio*
Activity: *Created profiles on Trello and Clockify. Manual registration of the time already spent…*

b) ASSESSMENT
Title: *Create the process landscape*
Description: *All the core processes of the factory are determined via the process landscape method*
Activity: *Identified the core processes "Setup mental movement classification service", "Prepare mental movement session", "Generate learning sessions set" "Develop mental movement classifier", "Classify mental movement" and "Check classification quality".*

As a reference, consider about 30% of the total time on modeling, 30% on simulation, and 30% on mining

3.HANDOFF>SERVICE MODEL

What to provide: with signavio, a BPMN diagram for each process, start with handoff level and then modify it as a service level.

Separated cards for subprocesses, assigned to different members (except for the configure)

1.2) LANES are INFORMATION SYSTEMS, focus on server side

EXAMPLES OF CLIENT-SIDE SYSTEMS:
- wearable devices (e.g. brain helmet, smart watch, smart shoes, etc.)
- desktop applications (e.g. annotator)
- mobile app (e.g. emotion selection, music playlist)

EXAMPLES OF SERVER SIDE:
- CRM (e.g. user registration, customer profiles)
- ERP (e.g. for product category)
- Ingestion System
- Preparation System
- Segregation System
- Development System
- Execution System
- Monitoring System

1.3) HUMAN TASKS, name format "job profile: task name"

Assign a job profile according to the skills required by the task:

- Administrative
- Data Analyst
- Data Engineer
- Data Administrator
- Machine Learning Engineer
…

European ICT Professional role profiles

https://tinyurl.com/e-CF-methodology
https://tinyurl.com/ICT-profiles
https://tinyurl.com/ICT-salary
https://tinyurl.com/e-CF-explorer

4.TASK LEVEL MODEL

4.1) Model a detailed use case, in terms of steps, for each human task, and show **mockups** when plots are supposed or an **UML class** for data objects.

4.2) Associate a **cognitive effort** to each step of the use case, 1-to-4 values

4.3) Associate an **occurrence** to each step of the use case, i.e., a percentage of occurrence (in case of "If" branch) or an average number of occurrences (in case of iteration)

4.4) Associate a **normalized salary** to each step of the use case, according to the job profile (take all salaries and divide by the minimum, which will be 1)

4.5) Compute the cost of the human task as the summation, for all tasks, of cognitive effort x occurrence x normalized salary.

5.BIMP SIMULATION (AS IS and TO-be)

5.1) Create a collapsed version (one pool and no lanes) of the process including only human tasks and gateways.

5.2) it is important to define the token of the collapsed version

5.3) estimate the percentage of each gateway on the basis of domain application assumptions (e.g. 90/10, 70/30, 40/60, 99/1,…)

5.4) simulate the as-is model and take the heatmap, and all data provided by the BIMP

5.5) add three improvements to the processes:
- at handoff level, e.g. to balance classes by taking minor data from other customers;
- at service level, e.g. to skip hyper-parameterization by taking hyperparameters from another customer of the same category;
-at task level (i.e., to reduce the cognitive effort in a use case)

5.6) simulate the to-be model and take the heatmap, and all data provided by the BIMP

5.7) motivate the result by writing some sentences that can be verified from the simulation reports

6.PROCESS MINING

Another commercial process mining tool:

https://apromore.org

Use both Apromore and ProM for mining and compare models

Question: improve the normative model (to-be) to manage 3 new cases by using process mining instead of human design

6.1) simulate the normative process with BIMP by assigning a default 1 eur cost and 1 sec duration to each task, 10 resources per lane, 50% to each gateway, and 100 input tokens;

6.2) take the simulation MXML log, convert it using Disco into XES and mine a transition map and a BPMN model

6.3) calculate the four quality dimensions (using the plugins/formula described in the material)

6.4) export the log as a csv (disco)

6.5) edit the csv with Excel - modify some timestamps or remove some task to **3 cases** to obtain **3 realistic violations** to the model. Examples: to skip or to anticipate some task, and explain the reason of realism: "the hyperparameterization can be skipped when the error is similar to the previous net", "the training can be skipped by taking a trained network of another customer with similar profile") ;

6.6) for each new case, apply the same modifications on other 2 instances, to have **3 instances of each case**

6.7) calculate the fitnesses of all instances of the modified log via conformance checking

6.8) Using the modified log, mine the new model

6.9. calculate the four quality dimensions of the new model (using the plugins and the formula described in the material)

Notes:

a) for each step take pictures and results, insert them into the shared document, comment, and collect the original and the modified log, as well as intermediate data to be submitted as a separate zip file with the pdf report

b) add your consideration: report in a list the points of strength and weakness of the new model from the business perspective (i.e. considering the advantages in terms of better quality / costs for the customer)

c) the BPMN model created with Visual Paradigm might cause some issues with ProM; to solve any issue open the model with Apromore, export it in the same format

d) the MXML log generated by BIMP contains additional start/end events separated from the process, which cause issues for conformance; to solve those issues import the MXML in Disco, export as CSV, import the CSV in ProM (with the related plugin) and convert it in XES on ProM by selecting only the time of process end; the resulting XES can be used as a log in ProM.