Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Manufacturing and Civil Engineering

Torkild Alstad

Development of Machine Learning Models for Soil Predictions based on CPT Measurements and Preliminary Research and Creation of Framework for Assessing Machine Learning Projects in AEC

In a Perspective of mulidisiplinary consultancies and Change management in AEC

Master's thesis in Civil and Environmental Engineering/Digital Building Processes

Supervisor: Eilif Hjelseth

NTNU
Norwegian University of Science and Technology

PREFACE

The research in this master's thesis was carried out during the spring semester 2020. This document houses the work done during this period and includes two research articles. The first article concerns work on the development and exploration of predictive machine learning models for geotechnical ground surveys. The second article explores and proposes a theoretical framework for identifying, describing, ranking feasible problems in the industry of architecture, engineering and construction for machine learning. Note that this document only houses the articles that are the main independent deliverables

I would also like to thank my supervisors as facilitators in the research done; Herman Bjørn Smith in Multiconsult Norway for the opportunities to work up to the resulting articles and the easy accessibility when guidance was needed. Eilif Hjelseth for being available when needed and a resource for both academic and business perspectives. Lastly and not directly related to the work done here I would like to thank Cathrine Mørch as a motivational force and an enabler in 2019 leading up to this work.

ABSTRACT

Strategic digital transformation of civil engineering (CE) consulting firms in architecture, engineering, and construction (AEC) requires the implementation of business innovation and execution models more than exciting new technology (Kane, Palmer, Phillips, Kiron, & Buckley, 2015). Which models and how to employ them involves understanding the niche industry in question, the potential trajectory of the industry, current tools and methods, and how people and resources apply. Today a gap exists between civil engineers and cutting-edge technology and knowledge management. Newer technology does not allow for civil engineers to sit idly by as it once did. Instead, they must adapt and be open to educating themselves as the industry progresses. Through development in this thesis of a machine learning model for predicting soil based on data from the equipment used in ground surveys with lab reports as ground truth labels and the start of a preliminary theoretical framework to identify and rank the feasibility for potential machine learning problems. This thesis will develop and propose a substantial step forward for AEC multidisciplinary consulting firms that navigates to potential desired outcomes by providing a deeper understanding of the worth of data and what is leading in the implementation of new technology such as artificial intelligence.

SAMMENDRAG

For å lykkes med digital transformasjon i rådgivende ingeniørfirmaer innenfor bygg- og anleggsbransjen, kreves det økt strategisk fokus på verdiskapning gjennom virksomhetsinnovasjon og endring av gjennomføringmetodikk, mer enn ensidig fokus på ny teknologi (Kane, Palmer, Phillips, Kiron, & Buckley, 2015). Valg og bruk av aktuelle rammeverk, prosesser og metoder for implementering av ny teknologi, innebærer forståelse for næringen i seg selv og de fundamentale endringsdriverne den utsettes for i konstellasjon med de teknologiske utfordringene. I nåværende situasjon eksisterer det et gap mellom ingeniørenes kapabiliteter og de mulighetene avansert teknologi og kunnskapsstyring gir. Den teknologiske utviklingen krever en proaktiv tilnærming, som utfordrer hver enkelt til å tilpasse seg og være åpne for ny kunnskap og nye muligheter i sitt daglige arbeid etter hvert som næringen endres. Arbeidet med denne masteravhandlingen har utnyttet ny teknologi, ved å utvikle en maskinlæringsmodell for å kunne forutsi jordtyper basert på data fra grunnundersøkelser utført med trykksonderinger, med tilhørende laboratorieundersøkelser der jordtypene er verifisert. Labratorieundersøkelsene er brukt som fasit i utvikling og trening av maskinlærings-algoritmen. Erfaringene fra utviklingen av maskinlæringsløsninger la videre grunnlaget for etableringen av et teoretisk rammeverk for identifisering, beskrivelse og rangering av egnethet for mulige maskinlæringsproblemer. Betydningen av løsningene og rammeverket presentert i dette arbeidet, har som mål å tilby et potensielt betydelig skritt fremover for hvordan rådgivende ingeniørfirmaer i bygg- og anleggsbransjen kan realisere forretningsmessig verdi og jobbe med digital transformasjon, i arbeidet med forståelse av verdiene som ligger i tilgjengelige data og premissgiverne for implementeringen av ny teknologi som maskinlæring.

ABBREVIATIONS

AEC Architecture, Engineering and Construction

AI Artificial Intelligence

ANN Artificial Neural Network

CPT Cone Penetration Tests

CSV comma-separated values

ML Machine Learning

ROI Return on investment

SHAP SHapley Additive exPlanations

THESIS GLOSSARY AND MEANINGS

Accuracy: Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right.

Business Model Canvas: Business Model Canvas is a strategic management and lean startup template for developing new or documenting existing business models.

Cone penetration test: The cone penetration or cone penetrometer test is a method used to determine the geotechnical engineering properties of soils and delineating soil stratigraphy.

CatBoost: CatBoost is a machine learning algorithm that uses gradient boosting on decision trees.

Comma-separated values file: A comma-separated values (CSV) file is a delimited text file that uses a comma to separate values.

Correlation: In statistics, correlation or dependence is any statistical relationship, whether causal or not, between two random variables or bivariate data

Categorical Data: Categorical variables represent types of data which may be divided into groups.

Correlation graph: A scatterplot is used to represent a correlation between two variables. There are two types of correlations: positive and negative.

Data cleaning: Data cleaning is the process of ensuring that your data is correct, consistent and useable by identifying any errors or corruptions in the data.

Data lake : A data lake is usually a single store of all enterprise data including raw copies of source system data and transformed data used for tasks such as reporting, visualization, advanced analytics and machine learning.

F_1 score: In statistical analysis of binary classification, the F_1 score is a measure of a test's accuracy.

Feature values: When representing images, the feature values might correspond to the pixels of an image, while when representing texts the features might be the frequencies of occurrence of textual terms.

GPU engine: GPU engine represents an independent unit of silicon on the GPU that can be scheduled and can operate in parallel with one another.

Hyperparameter: In machine learning, a hyperparameter is a parameter whose value is used to

control the learning process.

Iterative design: Iterative design is a design methodology based on a cyclic process of prototyping, testing, analyzing, and refining a product or process.

LightGBM: LightGBM is a gradient boosting framework that uses tree based learning algorithms

Latency: Latency is a networking term to describe the total time it takes a data packet to travel from one node to another.

Python: Python is an interpreted, high-level, general-purpose programming language.

Pipeline: In computing, a pipeline, also known as a data pipeline, is a set of data processing elements connected in series, where the output of one element is the input of the next one.

SHAP: SHAP is a popular Python libraries for model explainability. SHAP (SHapley Additive exPlanation) leverages the idea of Shapley values for model feature influence scoring.

Training set: a subset to train a model

Test set: a subset to test the trained model.

Text segmentation: Text segmentation is the process of dividing written text into meaningful units, such as words, sentences, or topics.

Value Proposition Canvas: The Value Proposition Canvas is a tool which can help ensure that a product or service is positioned around what the customer values and needs.

1 THESIS INTRODUCTION

The AEC industry today is like the aftermath of a technical tidal wave with the threat of more to come and as one of the largest industries in the world. However, most of the early software dinosaurs were scrapped and simple desktop programs were the norm through 2000. Spans between software updates were not counted in weeks or months, but years. This pace no longer suits the AEC industry and preparing to stay relevant now requires a proactive approach to what is now a very technically disruptive stage (Day, 2019). How an AEC firm copes with this revolution will directly determine its viability and longevity. Larger AEC firms that are succeeding, employ IT or other digital directors whose primary directive is to strategically keep their firm ahead of the technological changes that are now coming

The initiating factor of reasearch article 1 exploring possiblities of machine learning in geosience was when the Department of Geo, Water and the Environment initiated a meeting. The agenda was to look at the opportunities in the subsection Ground Surveys on the possibilities to utilize machine-learning on their historical data from Cone Penetration Tests(CPT). The basis for looking at this type of data was that it had large amounts associated with CPT, questioning how to utilize the data to gain insights. (Alstad, 2020a)

The problems and difficulties found in developing a machine learning model conducted by Alstad (2020a) led to the insight into the necessity of a framework that can describe and rank the feasibility of potential problems solved by machine learning, which are the research explored in article two by Alstad (2020b)

2 DISCUSSION AND SUMMARY REMARKS

The past decade has seen an exponential growth of the application of machine learning techniques and deployment of machine learning projects in different fields of science. Soil science investigation, has employed statistical models to “learn” or comprehend from data the distribution of soils in time and space (Padarian & Minasny, 2019). The increasing availability of soil data that can be effectively attained proximally and remotely, and easily accessible open-source algorithms, has resulted in an enhanced implementation of machine learning techniques to examine soil data. The same effort was conducted in the present study where the author employed the use of machine-Learning on historical ground survey data from Cone Penetration Test (CPT) retrieved from project servers owned by Multiconsult Norway.

However, such research would indeed be counterproductive without the development of framework as it would lead to wastage of resources (human capital and finance) and time. Agrawal, Gans, and Goldfarb (2018) maintained that for a high impact machine learning project, cost of prediction and prediction is crucial for decision making. Not only that, cheap prediction is universal for problems across various business domains. This requires looking for complex parts of the pipeline and places where cheap prediction is valuable.

As far as the first scenario is concerned, the most fundamental cause behind the failure of machine learning project is the insufficient quantity of data which deters training precise models (Wang & Ji, 2015). This leads to the usage of the small percentage of the data. For example, in the machine learning project the author developed on soil predictions, only a 27 percent of the data initially collected was used in the research by Alstad (2020a). Likewise, in big data analytics, it is estimated that only 30 percent of collected data is of value by Walker (2012).In some cases, inaccurate models produce completely randomize classification results or prediction and the entire business functionality may be questionable.

3 CONCLUDING WORDS

Technology is only as good as it's application. Companies are only as successful as their tools and culture and people. Neglecting any element of this combination may not be detrimental to the success of a digital transformation, but it will leave a pain point or vulnerability that a new or transforming company cannot tolerate. Therefore, validation that empowers the stakeholders, culture, and optimizes the tools and values of innovation, is an optimal for resilience and success.

4 PROPOSAL FOR FURTER RESEARCH

For future research in the research of predicting soils the work will revolve around gathering more data closely with the geological department in Multiconsult, and training the algorithm potentially until it reaches results worth value in production.

The theoretical framework will require operational experience for further development and will be used in practice to assess potential machine learning problems, it will also be developed to include modules to define scalability and plan of development in later stages

REFRENCES

Agrawal, A., Gans, J., & Goldfarb, A. (2018). *Prediction Machines: The Simple Economics of Artificial Intelligence*: Harvard Business Review Press.

Alstad, T. (2020a). *EXPLORING AND DEVELOPMENT OF SOIL PREDICTION MODELS WITH GRADIENT BOOSTED MACHINE LEARNING ALGORITHMS* (Master). NTNU,

Alstad, T. (2020b). PRELIMINARY FRAMEWORK DEVELOPMENT FOR ASSESSING HIGH IMPACT AND LOW RISK POTENTIAL MACHINE LEARNING PROJECTS IN THE AEC SECTOR

Day, M. (2019). The Generation Game. Retrieved from https://www.aecmag.com/technology-mainmenu-35/1885-the-generation-game?fbclid=IwAR2_67BgxLDs-e1Oh_iAqAc74NLaAXvD4zH0QmtkckxREeDiAytB8l3L2HE.

Kane, G. C., Palmer, D., Phillips, A. N., Kiron, D., & Buckley, N. (2015). Strategy, not technology, drives digital transformation. *MIT Sloan Management Review and Deloitte University Press, 14*(1-25).

Padarian, J., & Minasny, B. (2019). Using deep learning for digital soil mapping. *Soil, 5*(1), 79-89.

Walker, M. (2012). Big Data Analytics Infrastructure.

Wang, Z., & Ji, Q. (2015). *Classifier learning with hidden information.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

SCIENTIFIC RESEARCH PROFILES

Research paper 1

1. **Title**
   Exploring and Development of Soil Prediction Models with Gradient Boosted Machine Learning Algorithms
2. **Author**
   Torkild Alstad
3. **Potential journal profile**
   Automation in Construction
4. **Research area**
   Geoscience in combination with machine learning, data analytics and civil engineering
5. **Background and enabler**
   Vast amount of available historical data from ground surveys
6. **Goal**
   Gaining insight into soil properties and using the insights to predict soil behaviors and phenomena, with the focus enabling machine learning usage with business value as determinator for decisions.
7. **Keywords**
   Cone Penetration Test, Machine Learning, Lightgbm, Catboost, Soil Prediction, Data pipeline, Model pipeline

Research article 2

1. **Title**
   Preliminary Framework Development for Assessing High Impact and Low Risk Potential Machine Learning Projects in The AEC Sector
2. **Author**
   Torkild Alstad
3. **Potential journal profile**
   International Journal of Project Management
4. **Research area**
   Process and innovation management in the construction industry in constellation with data science
5. **Background and enabler**
   Few existing framworks and missing methods for assessing feasibility and describing machine learning problems
6. **Goal**
   Development of an framework for assessing high impact and low-risk machine learning problems fast to assist in choosing the project which will yield most business value.
7. **Keywords**
   Keywords: Machine Learning, Framework, Problem Definition, Feasibility Definition, Business models, innovation models

EXPLORATION AND DEVELOPMENT OF SOIL PREDICTION MODELS WITH GRADIENT-BOOSTED MACHINE LEARNING ALGORITHMS

Torkild Alstad – tork.alstad@gmail.com – 10.06.2020
Norwegian University of Science and Technology – Multiconsult Norway

ABSTRACT

The research presented in this paper focuses on the development of a soil prediction model with machine learning and vast data processing. The paper examine the problem and process of identifying and analyzing data from Cone Penetration Tests (CPT) in ground surveys with corresponding laboratory reports. The developed soil prediction model was able to predict soils using the machine-learning approach using two fast-processing boosting tree algorithms (Lightgbm and CatBoost). Recent literature did not present the use of the chosen algorithms, and therefore an exploration into the algorithms for an academic outlook was worth the contributions and valuablel . The developed scripts used the Python programming language and consist of several modules for data extraction, data cleaning, and the data modeling module. It was a clear overrepresentation of clay in the data, which poses a possible cause for the presented 97% accuracy in the results. The machine-learning model verified overfitting by cross-validating the score on ten different segments, which led to a cross-validated accuracy of 73%. In terms of the algorithms, Catboost took a longer time to train than LightGBM and was a lot faster while providing a comparable result, thus making it a better choice for production. The data insights provided by the algorithms suggested that soil pressure, depth, and height above sea were the most significant factors to the features researched in soil prediction based on the given CPT data. The predictions could provide a significant advantage in future identification of soils and gaining insights into geoscience, but the model may improve performance and reduce overfitting using a larger dataset.

Keywords: Cone Penetration Test, Machine Learning, Lightgbm, Catboost, Soil Prediction

1 INTRODUCTION

There has not been a vast amount of research conducted on the applications and utilization of machine learning in geoscience, but some similarities have been found. In recent years, the research has varied by using models like k-nearest neighbor, distance-weighted nearest-neighbor, support vector machines, decision trees, random forest and, most popularly, artificial neural networks for various tasks. One common conclusion of most of this research is the time-consuming nature of the learning processes of the machine-learning models used (Carvalho & Ribeiro, 2019; Alavi, Gandomi, & Lary 2016; Finnie & Kortekaas, 2017; Karpatne et al., 2019; Caté et al., 2017; Ghaderi et al., 2018).

The development of machine learning in recent years has been proposed for use in exploring multiple layers or structures in soils, thus increasing the classification, predictive process and proper capabilities of the use of multi-feature datasets and big datasets (Alavi, Gandomi, & Lary, 2016). Compared to traditional systems, it indicates more durable computing powers and success when applied to geoscience and other fields. Because of the extraordinarily complex and nonlinear scope of soil, it has overcome potential problems, namely low prediction accuracies (Ivanova et al., 2019).

The research described in this paper aims to develop a soil prediction model through machine learning and vast data processing. The prediction

model will use two fast-processing boosting tree algorithms to predict soils. The algorithms chosen have not been used in soil predictions before, and are therefore worth exploring from an academic perspective.

1.1 Related research

Alavi, Gandomi, and Lary (2016) note that novel algorithms, methodologies, and models have been created to predict rare phenomena, and can be used in all settings to obtain high-quality data, especially from small sample sets. Machine-learning systems have shown massive potential in some geoscience applications (Lary et al., 2017) because the processes involve learning very complex nonlinear data representations and usually need extensive data to be labeled. Despite the challenges of being costly, it has significantly been effectively utilized for better prediction results (Caté et al., 2017). This is why this research will shed some light on the data management practices of multidisciplinary AEC consultancy firms at both the project-level and the human understanding level.

Carvalho and Ribeiro (2019) presented a research problem in which the data in use only included two-dimensional charts. Thus, only a limited amount of soil properties could be determined. Each region had its own set of data, meaning that there were different soil types observed in each region. Again, with the available methods, it was difficult to draw a distinct line between stiff soil and overconsolidated soil types. A distance-based algorithm was used with two available data sets as a reference point. To discuss the geotechnical aspects of soil classification systems, up to five input features were used. The results after using the machine-learning algorithm were positive both when the data sets were substituted and when incomplete values were used (Carvalho & Ribeiro, 2019)

Bhattacharya and Solomatine (2006) reviewed various methods and discussed that the basic classification methods may be inconsistent, especially when maintaining continuity is key. Based on the collected data, classifiers were built where some of the techniques were decision trees (Bhattacharya & Solomatine, 2006). The topic concluded that the use of the Support Vector Machines was most efficient, largely reaching above 90%+ and 100% in some soils. However, in this research, there is no review of overfitting results or use of elements in complexity theory to combat overfitting results and a rather small dataset. The paper does, however, provide insights into feature correlations. Green and Naeini (2019) pointed out that this determines topography, geomorphology, lithology, hydrogeological conditions, and geological structure. This accurate scientific evaluation is obtained to ensure that the solutions to be shared will be reliable and useful. Soil analysis involves complicated meteorological factors and structural effects. It is also challenging to form the right mathematical prediction model. The traditional algorithms and structure characteristics cannot adequately process big data, and therefore improving prediction further is difficult (Finnie & Kortekaas, 2017).

1.2 Problem definition

This technical research paper will dissect the solution and process of identifying and analyzing data from Cone Penetration Tests (CPT) in ground surveys with corresponding laboratory reports. The end goal is the prediction of soil based on CPT data, which will help immensely in identifying soils. With the generation of strong prediction outcomes, the potential of this research involves reductions in risk and costs for future AEC developments.

Multiconsult Norge has one of Norway's largest historical documentations of geotechnical surveys performed within Norway. This research consists of creating a system that houses data from CPT data and corresponding laboratory reports from projects in the Oslo region. A script was created to locate the files from servers containing project data and information, extract the relevant information from it, and create a dataset. The dataset would then be cleaned and feature engineered before a predictive analysis was completed with state-of-the-art machine-

learning algorithms. These predicted soils based on data from the CPT (soil pressure, height from ground and above sea, coordinates and the correlated impact from flushing pressure and torque). The machine-learning model's hyperparameters (algorithm "settings") were then adjusted with a search algorithm to find the best parameters after an evaluation was made to fit the problem and the dataset. Along the way, several business analytical representations and methods were used to evaluate the problem, data quality, and insights.

The question then remained how could this stored data be mined in order to understand how soil type related to other data and how this other data could "tell" what type of soil a sample was without having to do a potentially expensive and time-consuming laboratory test.

1.3 Research scope

The program developed in Python for this project will be able to access the directory where the data files are stored and match each drill data file (text file) with its sibling laboratory report file (Excel file format) in the same sub-directory. it would then extract the important data points from each file, merge them together, and repeat for all files in all sub-directories. Also, it is pertinent to note that only spreadsheets named with integer numbers are considered in each workbook.

The output dataset will be cleaned, and feature engineered. Then, it will split the cleaned set into a training set and test/validation set and then, finally pass the training set through the Lightgbm and CatBoost machine-learning algorithms to predict soil type on the test set and validate these predictions. It will also present detailed model metrics and results.

The scope of the research consists of five phases given in Figure 1 with the related tasks relevant to the research problem. The strategy is provided in the introduction chapters and is an iterative process between the other phases. The data preparation and preprocessing phase consists of researching the data used, collecting it, selecting the relevant information, assigning labels and translating the data into uniform categories for consistency. Several techniques for data-cleaning, and visualizing were used for gaining insights and present the data and their correlation were used. The modelling and dataset splitting phase is based heavily on the insights gained, and with the chosen algorithms, an iterative process was executed by trying out different parameters to achieve the best predictive model possible given the authors' knowledge and research. The model metrics were then presented and analyzed for insights that were debated in the results and discussion chapter. The fifth phase of model deployment was only given a recommendation due to the maturity of the amount and data quality.

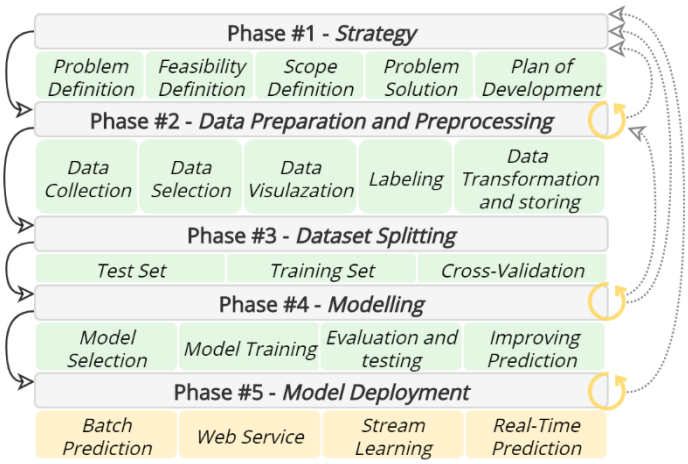

Figure 1 - Project phases

2 DEVELOPMENT

The method for executing the task relies on quantitative data collection to classify continuous and categorical outcomes, based on exploration and relationships given in the data with state-of-the-art machine-learning algorithms and techniques. Thus, it is also worth mentioning that this is an experimental research in a field of study where not much research has been done and relies on knowledge of the combination of information technology and civil engineering.

A machine-learning approach was considered to solve this problem where data would be fed into a suitable classification

machine-learning algorithm (Lightgbm and CatBoost), which would learn the relationship between the dependent variable (soil type) and other variables (drill depth, pressure, etc.) to predict the soil type of new unseen data quickly and accurately, given the variables.

1. The first step was data collection, which heavily relied on analyzing the project servers and gaining domain knowledge on mapping, project, and data structure.
2. The second step was to create a dataset from the vast sets of drill data and laboratory report files by extracting important data points in the drill data file and merging it with equally important data points in its sibling laboratory report.
3. The next step was to clean the new dataset by removing empty and incomplete rows and performing feature engineering through exploratory data analysis by balancing the dataset classes.
4. The final step was then passing this clean and feature engineered dataset into an appropriately tuned machine-learning algorithm that would be able to predict soil type given the other indicators (variables).

The project was developed modularly using the Python programming language. The modules include the data extraction module, the data cleaning module, and the data modelling module. It made use of library packages such as Pandas for data manipulations, text parsing and cleaning, Openpyxl for parsing .xlsx and .xlsm documents from Excel and data extraction, Matplotlib and Seaborn for data visualizations and data analytics.

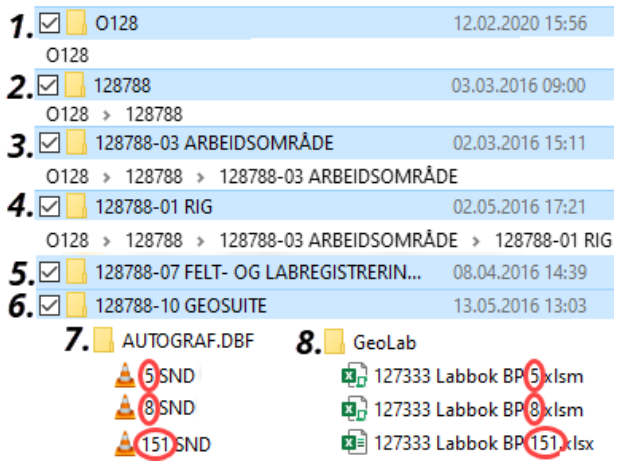

Figure 2 - Project repositories with CPT and laboratory data. Red circle shows the corresponding drill hole reference.

After developing the data model using the files available, the final data model was tested and fitted with more data until a desirable performance metric was achieved; it is recommended to deploy the model for use outside of the local machine and be used in real-time prediction while learning new data in streams on the cloud.

2.1 Data collection and selection

The first step is to locate and search all types of projects to locate relevant projects in which the program can identify and extract the desired files and information needed, which will be the basis for the machine-learning models to execute their algorithms. The data obtained was searched through project repositories hosted on Multiconsult fileservers and downloaded to create a local test environment for fast and flexible prototyping and troubleshooting of code. The folder file structure is shown in Figure 2.

An automated program was created with Python to work on the local project repositories. The first step was to import libraries and frameworks, as well as a helper function to make sure nested lists were converted to flat lists. The automated program in Python located the files, matched them up with their siblings, and housed them in a comma-separated values (CSV) data file which was a tabular data format.

In order to build a useful dataset from these data files, the data extraction module was built with the Python program to first find all laboratory reports in the project folder, then match them with their corresponding drill data file and pair both and add to a list.

From the drill data files, the data selected were the X and Y coordinates of the drill point, height above sea level and the table data that

contained the drill depth, drill pressure, flushing pressure and torque for each drill step in the file. From the laboratory report workbooks, the range of drill depths in each spreadsheet in the workbook were obtained alongside the soil type for those ranges in spreadsheets named with integer numbers.

2.2 Data transformation and labeling

The data is housed in a data-lake, which needs to be transformed and labeled. The first step was to do a translation of the soil names from Norwegian to English, as given in Table 1. To identify other given names, the script was developed to list all incompatible pairs that were dropped from the dataset. Drill data files in the pair list were parsed, and important data points such as drill point location, height above sea level, soil pressure, drill depth, etc. were obtained and put into a data-frame. The laboratory workbooks were also parsed and the drill range and soil type for each spreadsheet was obtained for spreadsheets named with numerical integers within the workbook. This was also put into a data frame. The final operation was merging the two data-frames together to produce a dataset for that pair list. This entire operation was then repeated for all pairs in the list, thereby producing a full dataset. Incompatible pairs with missing columns or empty values were saved into a list for later analysis.

Table 1 - Soil Translations

Index	*Norwegian*	*English*
0	LEIRE	CLAY
1	KVIKKLEIRE	QUICK CLAY
2	TØRRSKORPELEIRE	WEATHERED CLAY
3	SILT	SILT
4	TØRRSKORPESILT	WEATHERED SILT
5	SAND	SAND
6	GRUS	GRAVEL
7	TORV	PEAT
8	GYTJE	GYTJA
9	ORG. MATR.	ORG. MAT.
10	MATJORD	TOPSOIL
11	DY	DY
12	MATERIALE	MATERIAL
13	FYLLMASSE	FILL SOIL

Table 2 - Dataset Features

Index	*Feature*
0	Drill Depth (m)
1	Soil Type
2	X-Coordinate
3	Y-Coordinate
4	Height Above Sea Level (m)
5	Drill Pressure (kN)
6	Flushing Pressure (kN)
7	Torque

Table 3 - Dataset Tail

Index	*Drill Depth (m)*	*Soil type*	*X-coordinate*	*Y-coordinate*	*Height Above Sea Level*	*Drill Pressure (kN)*	*Flushing Pressure (kN)*	*Torque*
8543	5.500	CLAY	6647716.096	612212.117	137.279	3083.0	5.0	0.0
8544	5.525	CLAY	6647716.096	612212.117	137.254	3158.0	5.0	0.0
8545	5.550	CLAY	6647716.096	612212.117	137.229	3478.0	5.0	0.0

2.3 Data visualization and analytics

For exploratory data analysis purposes, some of the features in the dataset were visualized to determine their relationship to the soil type and gain extra insight in the problem and data. Figure 4 is a correlation plot of the features to each other exclusive of the dependent variable (soil type). This gives a breakdown of the relationship between the features, where a positive score means there is a degree of correlation, and a negative score means there is a degree of uncorrelation. Many features from the dataset are minutely correlated. The only majorly correlated features from the plot are the torque and drill pressure. The torque and drill pressure were suspected to have no correlation because they are manually started and regulated by the operator in CPT. The subsequent task was to consider the relationship between some features and the dependent variable. The first were the drill depth and pressure against the various soil types. The results show that soil types such as weathered clay and fill soil span over a large range of drill pressure and hence are found over a wide depth of drill pressure, whereas in surveys done at higher levels above sea, they are found at usually less than 10m of depth. Quick clay is also found at lower drill pressure at a relatively wide range of drill depth.

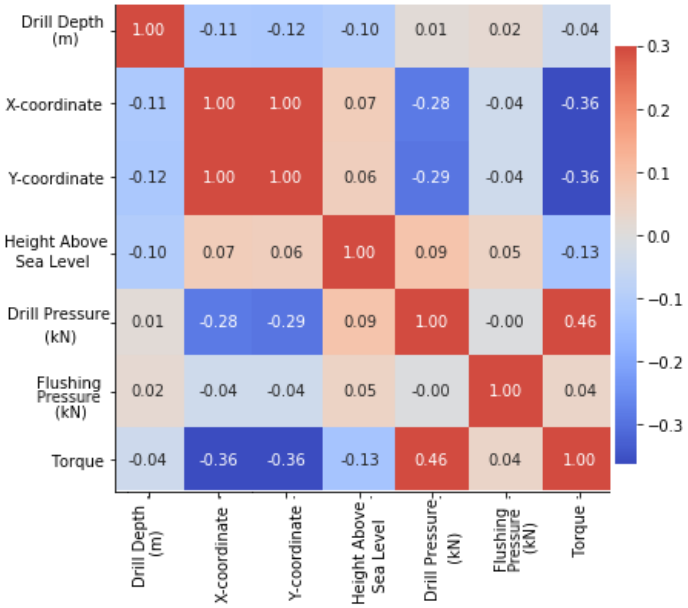

Figure 4 - Correlation map of the dataset

2.4 Dataset splitting

The pre-processed dataset was thereafter split into a train set and test/validation set using a random split. The aim of this was to construct a part of the data the ML algorithm could learn from, which was also non-biased in the data selection, hence the random selection of data points. The test/validation set could also be used to test the developed model and determine its performance.

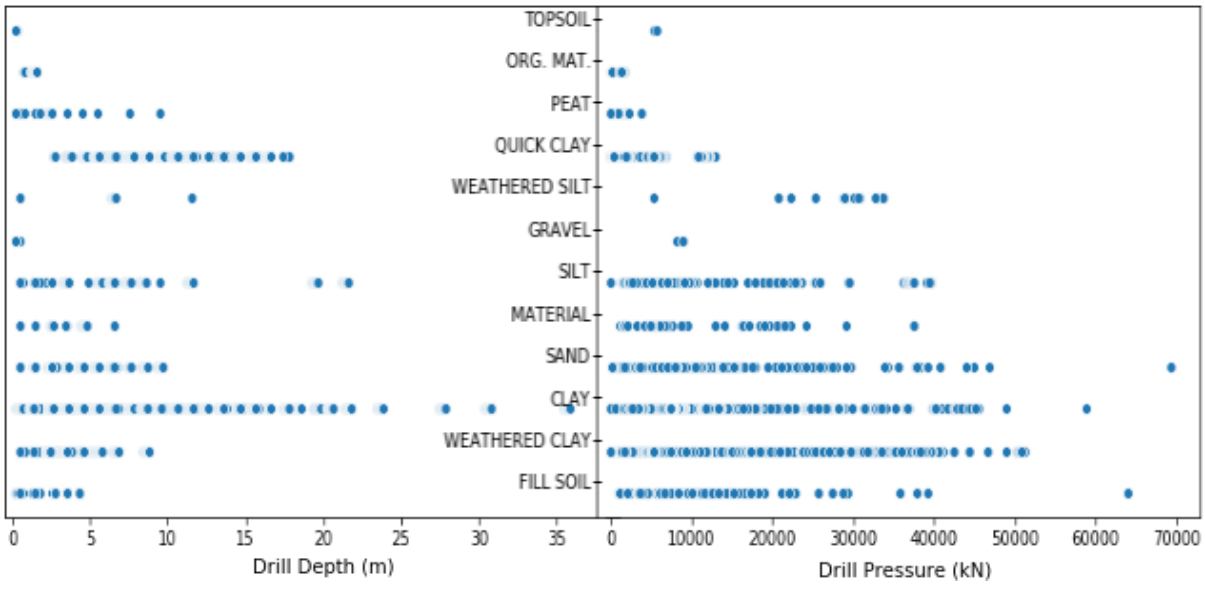

Figure 3 - Dataset feature graph

Due to the small dataset and to maximize the ML algorithms' capability to learn, the dataset was split into the train and test set only, which also was used for validation. The split was set to 70/30 where 70% of the dataset was used as a train set, and the remaining 30% was used as the test set.

2.5 Modelling

The ML algorithms chosen were Lightgbm and later CatBoost which both is gradient boosted trees that combine weak "learners" into a single strong learner in an iterative fashion (Ke et al., 2017). They have the advantage of being one of the fastest gradient boosted tree methods as well as being able to balance the soil type classes, which was heavily skewed, the results of the comparison is given in chapter 3.

The models were trained on the training set using default parameters to predict the validation set. Here, the dependent column was dropped (i.e., soil type) and fed into the model for prediction. The result was checked against the actual soil type column. Figure 5 shows the estimation of feature importance by the trained model to the dependent variable (soil type). It believes drill depth is the most important feature, closely followed by the height above sea level, and the least important being the flushing pressure.

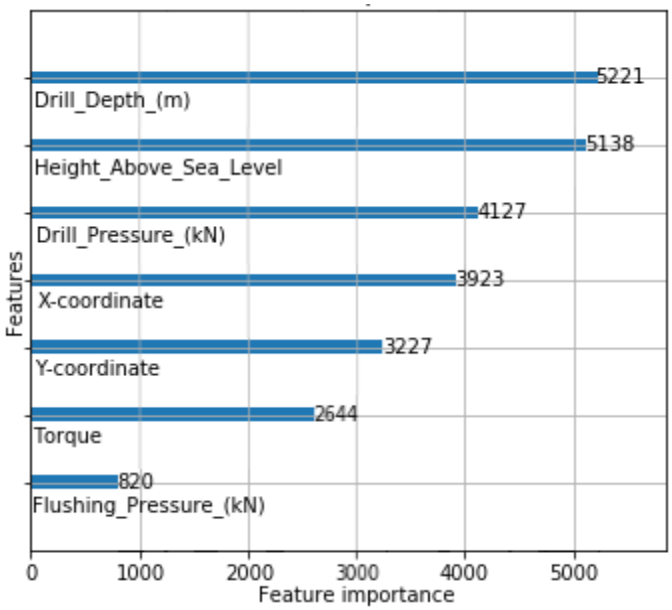

Figure 5 - Feature Importance Graph

The reason why the coordinates in the dataset had high importance is because of the amount of data given in the dataset and most of the projects were in the Oslo area. This leads to a low variation of range in coordinates, and one project can have around 1-10 CPT tests in one area.

The developed models were tuned for performance optimization. This was done using GridSearchCV from the scikit-learn library to search for the best parameters that could lead to an improvement in accuracy as well as other metrics. Its lists of parameters and hyperparameter values were fed into the search algorithm for the machine-learning algorithm, each was tested iteratively to obtain the best set of parameters and hyperparameters for both models. The model parameters searched through and chosen in showed in Table 4 and Table 5.

The model was then fitted and tested using the test set first with Lightgbm, which led to a precision score of 92%, recall of 88%, and F-1

Table 5 – Gridsearched and applied hyperparameters (LightGBM)

Index	Gridsearch	Best performing
Learning Rate	0.1, 0.05	0.1
Estimators	40, 200, 400	200
Leaves	20, 40	40
Boosting Type	gbdt	gbdt
Objective	multiclass	multiclass
Random State	42	42
Subsample Ratio of Columns	0.75, 1	0.75
Subsample	0.75,1	0.75
Alpha	1,0.5	1
Lambda	1,0.5	1

Table 4 – Gridsearched and applied hyperparameters (CatBoost)

index	Gridsearch	Best performing
Depth	4, 7, 10	10
Learning Rate	0.03, 0.1, 0.15	0.15
l2_leaf_reg	1, 4, 9	1
Iterations	300	300

score of 89%. This means the model performed very well with a good ratio of True Positives and True Negatives to False Positives and False Negatives. However, the model scored an accuracy of 97.7%, which was quite distant from the other classification scores. The model was suspected to be overfitting on the dataset. The assumed overfitting was then tested and confirmed by cross-validating the accuracy of the model on 10 randomly sampled sets of the data and found the mean score to be 72%. CatBoost gave an accuracy of 97.923% and cross-validated score of 73.628%. The same dataset was imported with the same test and training splits. The main difference in implementation was the hyperparameter grid search with other parameters to search through (Dorogush et al., 2018).

The models were evaluated with the analysis library SHAP in Python, and the trained and tested sets were analyzed on the function's impact on model performance. This shows the model's understanding of the dependence of soil pressure feature on drill depth in the test set. The analysis is shown in Chapter 3.

3 Results and discussion

The data preprocessing phase consists of a local test environment and preprocessing the data for dataset creation. A total of 18 632 project files were extracted and copied at the start, and in the end a total of 129 project file pairs remained after several data cleaning operations. The statistical results of the collection, selection, and cleaning are summarized in Table 7, Table 8, and Table 6.

The data collection and selection phase was the most time-consuming one in this project. Thus, it was also one of the most important ones due to the fact that the quantity and quality of data plays a big role in the learning process of the ML algorithms. With a total of 27% of the collected projects being selected, a large amount of data was resultantly excluded. This highlights the importance of the human factor related to understanding the quality and usability of data in the documentation of engineering works. It is estimated that only 30 percent of collected data is of value in big data analytics by Walker (2012), who elaborates further on the importance to having a strategic plan for information management by involving systems which detail the collection, storage, analysis and distribution of data created in data structures.

The dropped projects found in manual searches or classified as incompatible by the automated script were due to:

- Different project numbers on laboratory reports and CPT projects. In most instances, there were no references to each other either.
- Wrong drill-hole number given to either of the pairs.
- Typing errors in soil names in laboratory reports, which led to the dictionary not being able to forward information to the dataset.
- Typing errors in depths, which led to poor data quality.
- Naming conventions not being standardized both at the file and folder level. Projects made after 2016 seemed to be substantially more standardized.
- Incomplete data in CPT data files, such as coordinates in horizontal plane not given. The

Table 7 - Data pairs collected, selected and cleaned statistics in data preprocessing phase

	Files	Data Points
Compatible Pairs	137	8547
Cleaned Pairs	129	7655
*Removed Pairs**	247	892

Table 6 - Folder and File statistics in the data preprocessing phase

	CPT	Laboratory Reports	Total
Collected Folders	543	467	1010
Collected Files	18134	498	18632
Selected Folders	91	91	182
Selected Files	3736	298	4034
Removed Folders	452	376	828
Removed Files	14398	200	14598

CPT data with no given depth usually meant it was a planned CPT instead of an executed one.

The final dataset used in the machine-learning models with the soil percentage is given in Table 8. Here, it is clear that clay is overrepresented and a cause of the verified overfitting in the first place, especially when the split used in the training and test set was 70/30.

LightGBM and CatBoost scored relatively closely with the highest difference being in the F1 score. As is evident, the cross-validation and accuracy score differential is negligible. Overall, CatBoost takes much more time to successfully train, while LightGBM is much faster and provides a very similar result.

The importance of the dependent variables graph in Figure 6 on the next page shows on both models' outputs that the algorithms boost the soil types in a few instances very differently, while the three soils with most instances (clay, weathered clay, and quick clay) have the least varience in their impact on the model output in the different models.

Model performance was also analyzed with all features except coordinates' impact on the models' outcomes due to the coordinates most likely being of less importance in larger datasets. Torque and flushing pressure are controlled by the CPT operator and do not show a high impact in the feature importance graph (Figure 5). Figure 7 shows that torque has a small impact, mostly in low values, which most of data consists of, while flushing pressure shows an impact when the values are high in both models, but in Lightgbm this shift is very clear. The feature values' impact on model output shows that soil pressure has a high negative feature value and outliers on soil type in the CatBoost algorithm, while with Lightgbm there are less negative values, and one outlier are on the positive axis, though both have the majority of low feature values on the positive axis. Overall the importance of soil pressure is in favor of Lightgbm. Drill depth importance is similar in both models, with the majority of high feature values on the positive axis. Height above sea level is the feature with the least similarity. Lightgbm has more scattered and a higher amount of high feature values on the positive axis. The overall analysis of the plot in Figure 7 of the two features with lowest impact(flushing pressure and torque) analyzedCatBoost treat them as lower impact features towards the dependent value, thus it is only minor changes and mainly both algorithms treat the low impact features similar. Summerized, the plots are in favor of Lightgbm.

Table 8 - Summary of datapoints in the final dataset

Soil	Datapoints	%
Clay	5960	71.38
Weathered Clay	959	9.47
Quick Clay	764	9.11
Sand	348	4.13
Silt	253	2.81
Fill Soil	100	1.24
Material	56	0.69
Peat	50	0.62
Organic Material	38	0.29
Weathered Silt	15	0.19
Gravel	3	0.04
Topsoil	2	0.02

Table 9 - Model performance comparison

Index	Lightgbm	CatBoost
Accuracy	97.851%	97.923%
Precision	87.568%	94.400%
Recall	81.619%	87.402%
F1 Score	84.255%	90.263%
Cross Validation Score	73.567%	73.628%
Execution Time (CPU)	24.5 min	48.6 min

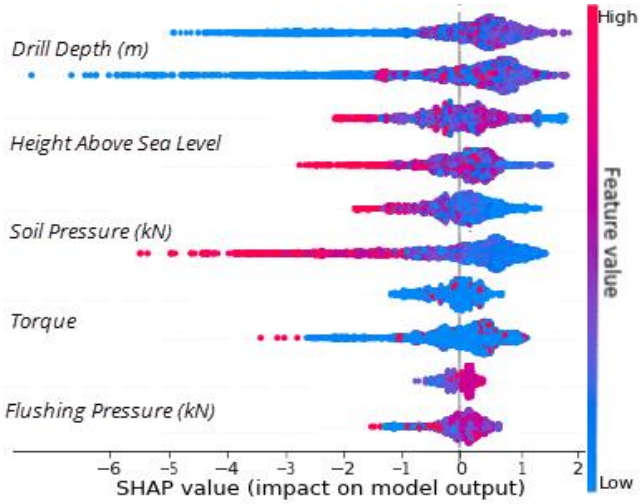

Figure 6 - SHAP Values, Impact of models features on the evaluated models

Upper values are from lightGBM and lower CatBoost

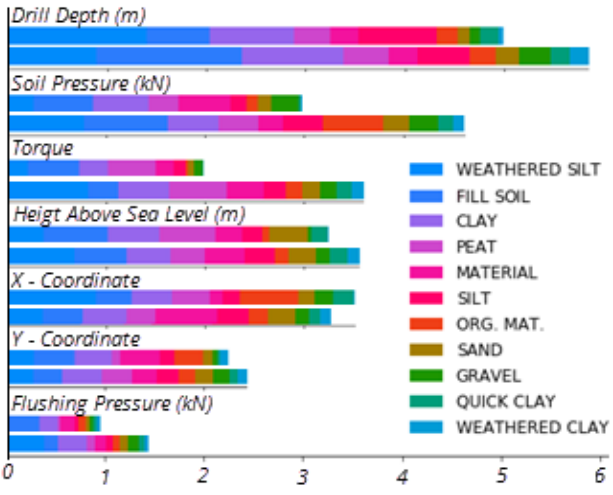

Figure 7 - Dependent feature value importance on model output plot.

Upper values are from lightgbm and lower CatBoost

4 CONCLUSION, RECOMMENDATIONS, AND FURTHER WORK

The research shows that the dependence assumed in the dataset and soil predictions will work based on soil pressure and depth data. The work around data preprocessing is highly important in terms of how data should be handled going forward for both this project and other developments, considering only 23% of the collected projects were selected. When cross-validated, both model predictions reached 73% which is not sufficient for real world geotechnical projects as of today. However, the results lay the groundwork for further data collection and model improvements.

The main conclusions are:

- Lightgbm is preferable to CatBoost due to a faster speed and negligible differences in the other results.
- Data and information management face problems in naming conventions, and quality control at the file, folder, and file content levels.
- The data preprocessing and preparation phase is the most important.
- Soil Pressure, depth and height above sea are the largest contributing factors to the features researched in soil prediction based on CPT.

In order to build a better model, many more data points need to be provided by using a much larger dataset. This means more CPT files and laboratory reports will need to be parsed and extracted, as this would help improve performance and reduce overfitting. Also, easily obtainable features given in the workbooks should be added to the dataset by a geotechnical expert. The program working on the present data repository takes approximately 20 minutes to run. This could be greatly improved and should be considered by making use of GPU engines to multi-thread the process and make efficient use of memory bandwidth, thereby improving speed, especially on a much larger dataset.

When the model has reached an adequate score to be used in predictions as a tool in projects, several model deployment methods should be considered for scaling outside the local computer that the program is run on. With batch prediction, the model can be used to generate predictions for new sets of data all at once and then act on a certain percentage or size of the data. This will typically have a high latency requirement, given that the size of observation sets passed into it may be large or at high speed and would require an equally quick response from the deployed model. Also, web services and real-time predictions should be considered, which would move the model to a cloud service where it would then access the necessary resources to update itself and deliver predictions in real time. It will then be made accessible through the web, which will make it available in locations where it may be difficult to connect to a local server. Some cloud services that can make this possible include Amazon Web Services (AWS), Microsoft Azure, Digital Ocean, Alibaba Cloud, and Google Cloud Platform (GCP). Stream learning is also an alternative in which the deployed ML model would update itself with new streams of data to improve predictions and output. This would help ensure that the model is not stagnant (concept drift) and predicts based on only developmental data. The model would be exposed to a data source that will supply the new training data, which will also contain the target variable. This process will be automated and occur a selected number of times using limited computing and storage capabilities.

5 ACKNOWLEDGMENTS AND AVAILABILITY

We would like to acknowledge Multiconsult Norway for enabling the use of their project file server for both data collection and running ML analysis on their data.

6 REFERENCES

Abaturova, I. V., Zudilin, A. E., Savintsev, I. A., Storozhenko, L. A., & Koroleva, I. A. (2017, April). Analysis of the degree of fracturing of the rock during engineering-geological surveys. In *13th Conference and Exhibition Engineering Geophysics 2017* (Vol. 2017, No. 1, pp. 1-7). *European Association of Geoscientists & Engineers.* Retrieved 5 April 2020, from https://doi.org/10.3997/2214-4609.201700374

Alavi, A. H., Gandomi, A. H., & Lary, D. J. (2016). Progress of machine learning in geosciences: Preface. *Geoscience Frontiers, 7*(1), 1-2. Retrieved 5 April 2020, from https://doi.org/10.1016/j.gsf.2015.10.006

Bhattacharya, B., & Solomatine, D. P. (2006, March). Machine learning in soil classification. Neural Networks, 19(2), 186-195. Retrieved April 7, 2020, from https://www.sciencedirect.com/science/article/abs/pii/S0893608006000116

Carvalho, L. O., & Ribeiro, D. (2019, August). Soil classification system from cone penetration test data applying distance-based machine learning algorithms. *Soils and Rocks 42*(2):167-178. Retrieved 5 April 2020, from. https://www.researchgate.net/publication/335733378_Soil_Classification_System_from_Cone_Penetration_Test_Data_Applying_Distance-Based_Machine_Learning_Algorithms

Caté, A., Perozzi, L., Gloaguen, E., & Blouin, M. (2017). Machine learning as a tool for geologists. *The Leading Edge, 36*(3), 215-219. Retrieved 5 April 2020, from https://doi.org/10.1190/tle36030215.1

Dorogush, A. V., Ershov, V., & Gulin, A. (2018, October 24). CatBoost: gradient boosting with categorical features support. *LearningSys.* Retrieved April 7, 2020, from https://learningsys.org/nips17/assets/papers/paper_11.pdf

Finnie, I., & Kortekaas, S. (2017). Integrated Geophysical and Geotechnical Planning: Through Use of Integrated Geoscience Techniques. *Encyclopedia of Maritime and Offshore Engineering,* 1-18. Retrieved 5 April 2020, from https://doi.org/10.1002/9781118476406.emoe519

Gil, Y., Pierce, S. A., Babaie, H., Banerjee, A., Borne, K., Bust, G., ... & Horel, J. (2018). Intelligent systems for geosciences: an essential research agenda. *Communications of the ACM, 62*(1), 76-84. Retrieved 5 April 2020, from https://www.isi.edu/~gil/papers/gil-tal-cacm19.pdf

Ghaderi, A., Shahri, A. A., & Larsson, S. (2018). An artificial neural network based model to predict spatial soil type distribution using piezocone penetration test data (CPTu). *Bulletin of Engineering Geology and the Environment, 78*(6), pp.4579-4588. Retrieved April 7, 2020, from https://link.springer.com/article/10.1007/s10064-018-1400-9

GitHub SHAP. (2020, February 28). Slundberg/shap. GitHub. Retrieved April 7, 2020, from https://github.com/slundberg/shap

GitHub. (2020, April 5). Microsoft/LightGBM. Retrieved April 7, 2020, from https://github.com/microsoft/LightGBM

Github. (2020, April 7). Catboost/catboost. GitHub. Retrieved April 7, 2020, from https://github.com/catboost/catboost

Green, S., & Naeini, E. Z. (2019, May). 3D Pore Pressure and Geomechanics: Work Smarter and Faster Integrating Geoscience with Machine Learning. In *Second EAGE Workshop on Pore Pressure Prediction* (Vol. 2019, No. 1, pp. 1-5).

Retrieved 5 April 2020, from https://doi.org/10.3997/2214-4609.201900520

Heath, P. (2019). Update on geophysical survey progress from geoscience Australia and the geological surveys of Western Australia, South Australia, Northern Territory, Queensland, New South Wales, Victoria and Tasmania (information current on 23 January 2019). Yvette Poudjom Djomani, geological survey of South Australia: Geophysical plans for 2019. *Preview, 2019*(198), 15-17. https://doi.org/10.1080/14432471.2019.1570802

Hughes, R. (2011). Geoscience data and derived spatial information: Societal impacts and benefits, and relevance to geological surveys and agencies. Geological Society of America Special Papers, 35-40. https://doi.org/10.1130/2011.2482(04)

Hunter, J. D. (2007). Matplotlib:A 2D Graphics Environment, Computing in Science & Engineering, vol. 9, no. 3, pp. 90-95, 2007. *PyPI*. Retrieved April 7, 2020, from https://pypi.org/project/matplotlib/

Ivanova, A., Roslyakov, A., Terekhina, Y., & Tokarev, M. (2019). Assessment of the hazards of faults in the offshore during engineering-geological surveys. Engineering and Mining Geophysics 2019 15th Conference and Exhibition. Retrieved April 7, 2020, from https://doi.org/10.3997/2214-4609.201901730

Karpatne, A., Ebert-Uphoff, I., Ravela, S., Babaie, H. A., & Kumar, V. (2019). Machine Learning for the Geosciences: Challenges and Opportunities. IEEE Transactions on Knowledge and Data Engineering, 31(8), 1544-1554. [8423072]. Retrieved April 7, 2020, from https://doi.org/10.1109/TKDE.2018.2861006

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T. (2017). LightGBM: A highly efficient gradient boosting decision tree. In Advances in neural information processing systems, pp. 3146-3154. Retrieved April 7, 2020, from https://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree

Lary, D. J., Alavi, A. H., Gandomi, A. H., & Walker, A. L. (2016). Machine learning in geosciences and remote sensing. Geoscience Frontiers, 7(1), 3-10. Retrieved April 7, 2020, from https://doi.org/10.1016/j.gsf.2015.07.003

Lib Warnings Python. (2017). warnings — Warning control. Lib/warnings.py. Retrieved April 7, 2020, from https://stackoverow.com/questions/47722353/how-to-install-warnings-package-in-python

LightGBM Classifier. (n.d.). Lightgbm.LGBMClassifier — LightGBM 2.3.2 documentation. Welcome to LightGBM's documentation! Retrieved 5 April 2020, from https://lightgbm.readthedocs.io/en/latest/pythonapi/lightgbm.LGBMClassifier.html

Ludden, J. N., & Smith, M. (2018). The Role of Geological Surveys in Addressing Societal Challenges Through Science Diplomacy. AGUFM, 2018, PA12A-02. Retrieved 5 April 2020, from https://ui.adsabs.harvard.edu/abs/2018AGUFMPA12A..02L/abstract

Maniar, H., Ryali, S., Kulkarni, M. S., & Abubakar, A. (2018). Machine-learning methods in geoscience. In SEG Technical Program Expanded Abstracts 2018 (pp. 4638-4642). Society of Exploration Geophysicists. Retrieved 5 April 2020, from https://doi.org/10.1190/segam2018-2997218.1

Pisetski, V., Abaturova, I., Storozhenko, L., Savintsev, I., & Petrova, I. (2017). Solving the problems of obtaining geological information with using geophysical methods of research during engineering-geological surveys. 23rd European Meeting of Environmental and Engineering Geophysics. https://doi.org/10.3997/2214-4609.201701984

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2017, July 28). CatBoost: Unbiased boosting with categorical features. arXiv.org. Retrieved April 7, 2020, from https://arxiv.org/abs/1706.09516

Python Foundation. (2019, November 29). Lightgbm 2.3.1. PyPI. Retrieved April 7, 2020, from https://pypi.org/project/lightgbm/

Python Software Foundation. (2017, July 14). Scikitlearn. PyPI. Retrieved April 7, 2020, from https://pypi.org/project/scikitlearn/

Python Software Foundation. (2020, April 2). Regex. PyPI. Retrieved April 7, 2020, from https://pypi.org/project/regex/

Python Software Foundation. (2020, February 27). OS-win. PyPI. Retrieved April 7, 2020, from https://pypi.org/project/os-win/

Python Software Foundation. (2020, January 10). Openpyxl. PyPI. Retrieved April 7, 2020, from https://pypi.org/project/openpyxl/

Python Software Foundation. (2020, March 17). Numpy. PyPI. Retrieved April 7, 2020, from https://pypi.org/project/numpy/

Python Software Foundation. (2020, March 18). Pandas. PyPI. Retrieved April 7, 2020, from https://pypi.org/project/pandas/

Python Standard Library. (2020, April 7). Time — Time access and conversions — Python 3.8.2 documentation. 3.8.2 Documentation. Retrieved April 7, 2020, from https://docs.python.org/3/library/time.html

Robertson, P. K. (2016). Cone penetration test (CPT)-based soil behaviour type (SBT) classification system—an update. *Canadian Geotechnical Journal, 53*(12), 1910-1927. Retrieved April 7, 2020, from https://doi.org/10.1139/cgj-2016-0044

Scikitlearn Resources and Information. (n.d.). Stable Modules Generated Sklearn.model selection GridSearchCV. scikitlearn.org - This website is for sale! - scikitlearn Resources and Information. Retrieved April 7, 2020, from https://scikitlearn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

seaborn.heatmap — seaborn 0.10.0 documentation. (2020). Retrieved 5 April 2020, from https://seaborn.pydata.org/generated/seaborn.heatmap.html

Sklearn.model_selection.train_test_split. (2020). Sklearn.model_selection.train_test_split — scikit-learn 0.22.2 documentation. scikit-learn: machine learning in Python — scikit-learn 0.16.1 documentation. Retrieved April 7, 2020, from https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

Swalin, A. (2018). CatBoost vs. Light GBM vs. XGBoost. Retrieved 5 April 2020, from https://towardsdatascience.com/catboost-vs-light-gbm-vs-xgboost-5f93620723db

Taluja, C., & Thakur, R. (2018, November 1). (PDF) Recent trends of machine learning in soil Classification:A review. ResearchGate ISSN (e): 2250 – 3005, Volume, 08, Issue, 9, *International Journal of Computational Engineering Research (IJCER)*. Retrieved April 7, 2020, from https://www.researchgate.net/publicatio

n/328927674_Recent_Trends_Of_Machine_Learning_In_Soil_ClassificationA_Review

Tavakoli, V. (2018). Geological core analysis: application to reservoir characterization (p. 99). Cham, Switzerland: Springer.

Trugman, D. T., Beroza, G. C., & Johnson, P. A. (2019). Machine Learning in Geoscience: Riding a Wave of Progress. Eos, Transactions American Geophysical Union (Online), 100(LA-UR-19-22852). Retrieved April 7, 2020, from https://www.researchgate.net/publication/332884055_Machine_Learning_in_Geoscience_Riding_a_Wave_of_Progress

PRELIMINARY FRAMEWORK DEVELOPMENT FOR ASSESSING HIGH IMPACT AND LOW RISK POTENTIAL MACHINE LEARNING PROJECTS IN THE AEC SECTOR

Torkild Alstad – tork.alstad@gmail.com – 09.06.2020
Norwegian University of Science and Technology – Multiconsult Norway

ABSTRACT

This study focuses on the enabling strategies for value creation and risk reduction in architecture, construction and engineering (AEC) sectors in correlation with taking advantage of the full potential in potential machine-learning projects. The author's background experience in developing machine learning solutions subsequently made the argument for providing a framework to approach potential machine-learning projects. The framework intends to describe a potential problem and assess the feasibility of potential machine learning projects based on risk reduction and business value as decision-making factors. The study uses secondary research and past experiences as a basis for creating the proposed framework. The method used for creating the framework was based on segmenting information from earlier research obtained by identifying correlations and relevancy to descriptions of the problem and feasibility definitions. The segmentation made the information clearer by organizing, summarizing, and categorizing before being analyzed. The frameworks will enable investigating potential problems and feasibility of solutions to be described correctly. The first part of the framework intends to define the problem, provide background and purpose and to describe the business value the possible solutions could have if implemented. The second part focuses on the use of the three-dimensional problem feasibility model and considers three dimensions; practical feasibility, business impact, and human factors. In order to assess the arising challenges with implementation, the proposed framework is further discussed through exemplification of implementation on existing project. The framework is established based on a theoretical basis, and concluding elements thus suggest it to be put in practice for further development and operationalization.

Keywords: Machine Learning, Framework, Problem Definition, Feasibility Definition, Business models, innovation models

1 INTRODUCTION

Machine learning is playing a major part in the Fourth Industrial Revolution (Industry 4.0), that is, the digitalization age, wherein intelligent technologies and systems are employed to generate an active collaboration between the digital (virtual) and physical spheres(Botu, Batra, Chapman, & Ramprasad, 2017). In the view of Rafiei and Adeli (2018), machine learning offers massive prospects for substantial productivity improvements by means of examining huge data volumes accurately and quickly. Furthermore, machine learning technologies and systems can tackle nonlinear, complicated practical problems and once trained, could carry out generalizations and predictions at increased speed.

Owing to these advantages, machine learning has gained considerable attention within an extensive range of industries, counting AEC (Architecture, Engineering and Construction) which is full of difficult and complex problems. Machine learning signifies potential influential methods and processes to help in addressing these problems(Kulesza & Taskar, 2012). Consequently, over the years, scholars have been

carrying out research on machine learning in the AEC industry.

1.1 Background and goal

The authors experience from previous work on development of a machine learning model (Alstad, 2020), and past studies suggests for any machine learning project, a framework defining the problem under consideration and the possible solutions and then the most solution could possibly be defined(Padarian & Minasny, 2019; Rossiter, 2018). A well-defined framework can help in quick understanding of the elements and motivation for the problem and whether machine learning is suitable or not.

The machine-learning case developed by Alstad (2020) on soil predictions has no prepared data in advance, and comprised of the whole process from identifying the case definition, finding relevant data, collecting, selecting, preprocessing and transformation of the data. Further, the data was divided into groups of training test and validation sets. Then the modeling of the Machine Learning model was conducted, where the right model was chosen with evaluation and testing. In the end the deployment and scalability were evaluated. The research done by Alstad (2020) was taken from the absolute start and the data was highly unstructured, and identified several underlying issues on using machine-learning, and provided insights and reasoning for this research, with the goal of enabling use of machine learning in the sector of AEC.

The goal of this research is to explore and lay the foundation for providing a common framework for entering and approaching potential machine-learning projects in the industry of AEC. In the present research, it is proposed a framework comprised of two phases and consisting of several layers: defining the problem and assessing the feasibility of the problem under consideration. Defining the framework can enable cost-effective and risk reduced for decisions making for potential machine learning projects and help businesses to generate required value.

1.2 Problem and scope

The question is how a generic process and method framework could be defined, detailed and generalized for machine learning projects, in order to successfully be used as a tool for describing a problem and determine the feasibility of a problem to be solved with machine learning technology. This can be used as an indicator if the project should be initiated and to identify the scope and breakdown of a project.

The scope of this research is limited to the creation of a theoretical framework for potential machine-learning problems, and consisting of the elements: (1) problem definition, which will describe the machine learning problem, its value, purpose, goal and background. This can be put in context and used alongside models such as Business Model Canvas (BMC) and Value Proposition Canvas (VPC) from Alex Osterwalder, Pigneur, Bernarda, Smith, and Papadakos (2014). Nevertheless, this paper aim at machine-learning in general and gives methods for describing machine-learning related subjects. (2) feasibility definition, which will describe the feasibility of solving a problem with a proposed method for ranking the feasibility in a 3D graph based on the projects practical feasibility, the impact on the business and human factors. The ranking should be used as an indicator for assessing if the machine learning project should be developed.

Outside the scope of this research paper, but an element to be developed within the framework are: (3) scope definition, which focuses on assessing the requirements of the developed product a solution want to create and the work required to develop the project in terms of the product requirements. The scope definition on product and project is also split to be explained in three dimensions, the data, model and production pipeline. When the scope definition is described, assessment of the feasibility definition can be reevaluated for assessing if the project can be executed or not. The last part of the framework

(4) is the plan of development which will take the project scope described on the basis for a work breakdown structure. The initial work of setting up required code bases and exploration of the data to understand its requirements will be carried out.

1.3 Contributing research

A machine learning project without a predefined framework poses high risk to fail and could result in loss of revenue. The formation of appropriate machine learning business models and problem assessment methods is paramount as many studies have found that machine learning techniques can assist in business and scientific processes(Gil, Greaves, Hendler, & Hirsh, 2014; Willcock et al., 2018). Furthermore, to identify the relevant challenges, processes, and methods, a machine learning project with corresponding detailed paper was executed, literature review on AI in the AEC sector. contribute to the final results in this research under the belief that those topics will highlight, identify, and provide essential questions and answers for utilizing and scaling machine learning in large AEC consultancies

The framework proposed is under the influence of ten research articles by information and data technology professionals and one white paper by Amazon Web Services on machine learning workflow and management. The papers are presented in Table 1.

Table 1 – Influenced research.

No	*Authors*	*Year*	*Profile*
1)	Andrej Karpathy	2017	Web paper
2)	Agrawal, Goldfarb, Gans	2018	Journal Paper
3)	Andrej Karpathy	2018	Web paper
4)	Eric Breck Shanqing Cai Eric Nielsen Michael Salib D. Sculley	2017	Conference paper
5)	Jordan	2018	Journal Paper
6)	Le	2019	Journal Paper
7)	Pant	2019	Conference paper
8)	Schmitt	2019	Journal Paper
9)	Stephen Merity	2019	Conference paper
10)	V.M. Megler	2019	White paper

2 METHOD ANALYSIS

The research can be classified as an iterative review and design process method by creating a theoretical framework and testing the conceptual framework on how machine learning project strategies can be defined. The framework is developed alongside researching the use of machine-Learning on historical ground survey data from Cone Penetration Test (CPT) retrieved from project servers owned by Multiconsult Norway (Alstad, 2020).

2.1 Analyzes of machine learning project strategies

The analysis of information and data from earlier research is a procedure used by scholars to gain knowledge and reduce the gathered information by segmenting it to interpret crucial insights. Text segmentation helps to shrink huge data chunks into tinier fragments that make more sense to both the researchers and their audience(Koshorek, Cohen, Mor, Rotman, & Berant, 2018). During analysis, three vital things take place. Firstly, information from research is organized, then it is summarized and categorized before being analyzed. The first two steps help with the identification of themes and patterns within the information for natural linking. In brief, the text segmentation can be summarized as an application of inductive and deductive logic to gain insight into general and specific information.

The method consists of segmenting parts of article texts by categorizing their topics, subjects and corresponding descriptions and potential outcomes relevant to machine-learning project strategies, with yellow, green and cyan markings in the text. The method used for segmenting texts is inspired by the methods developed by Beeferman, Berger, and Lafferty (1999); Hjelseth (2015). Exemplification of the segmentation process from the article post written by Le (2019) is given in Figure 2.

Further analysis and grouping of the segmentation of articles, the different color marking labels used on the text segmented represent column headings in a spreadsheet where the rows consist of the marked text belonging together. Columns were added holding the reference article and proposed belonging strategy phase, which will work as a grouping element. The range of strategy phases proposed to containerize the text elements was; (1) problem definition, (2) feasibility definition, (3) scope definition, (4) plan of development and (5) excluded for segments not used in further analysis.

The statistics over segmented parts in the worksheet is listed up in Table 2 and 3. Thus it should be mentioned that some cells in the worksheet contains data subjected to be divided into several cells, Figure 1 exemplifies that possibility. The whole work sheet is presented in Appendix A

Table 2 – Number of hits collected in worksheet, in terms of research reference

Reference	*Hits*
(Le, 2019)	12
(Alake, 2020)	32
(Karpathy, 2017)	3
(Agrawal et al., 2018)	1
(Altexsoft, 2018)	80
(Breck, Cai, Nielsen, Salib, & Sculley, 2017)	1
(Jordan, 2018)	19
(Pant, 2019)	37
(Schmitt, 2019)	16
(Merity, 2017)	4
(Megler, 2019)	40
Undefined	3

Table 3 – Number of hits collected in worksheet, in terms of strategy categorization

Strategy phase	*Hits*
Proflem definition	26
Feasibility definition	28
Scope definition	17
Plan of development	120
**Excluded parts*	26
Total	298

Description or example
1.Increase model capacity 2.Reduce regularization 3.Error analysis 4.Choose a more advanced architecture (closer to state of art) 5.Tune hyperparameters 6.Add features

Figure 1 – Showing one cell holding multiple points

Phase 4 — Model Deploying and Model Testing: At this phase, we want to pilot the model in a constrained environment (i.e., in the lab), write tests to prevent regressions, and roll the model into production. We may see that the model does not work well in the lab, so we want to keep improving the model's accuracy (thus go back to phase 3). Or we may want to fix the mismatch between training data and production data by collecting more data and mining hard cases (thus go back to

Figure 2 - Examplification of the text segmentation used to analyze articles on machine-learning project processes. Yellow is for the topic, green the subject and for cyan the description. The text example is from Le (2019) article on phases in machine learning projects

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Phase 4 — Model Deploying and Model Testing	Pilot the model in a constrained environment	tests to prevent regressions, and roll the model into production, keep improving the model's accuracy, fix the mismatch between training data and	le (2019)
Feasability Definition	feasibility and impact of your projects	prioritizing projects	build projects with high impact and high feasibility (aka, low cost).	le (2019)

Figure 3 - Analyzed text from Figure 1 structured in a spreadsheet

3 RESULTS

The following chapter aims to propose the results for the developed theoretical framework intended for examining the relevance and effectiveness of a machine learning project and help the businesses in creating the value for them. The results are divided in two sections. The first section elucidates two major stages: defining the problem and then conducting the feasibility study of the problem under consideration. The second section inclines to explore the effectiveness of the proposed framework by applying it to the case study, i.e., cost estimation in the tendering process by Matel et al. (2019).

3.1 Problem definition

The first part of the framework is defining the problem, laying the groundwork for later stages and ensures that the involved parties have a proper understanding of the problem and the desired outcome. Four dimensions are proposed to document when a case for machine learning is identified, the entire process of deliveries, which includes the following:

1. Goal definition (Figure 4)
2. Purpose definition (Figure 5)
3. Value proposition (Figure 6)
4. Problem information (Figure 7)

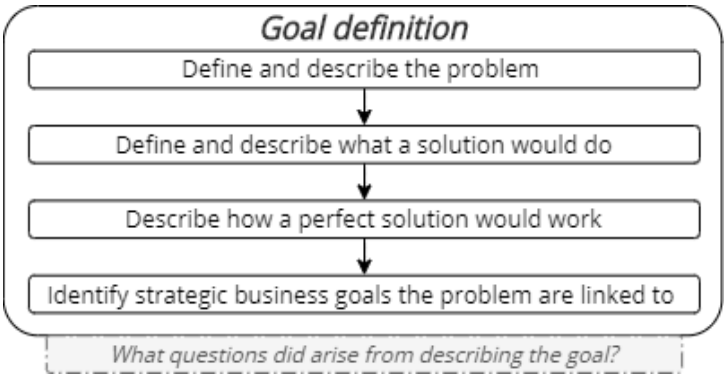

Figure 4 - Process of describing the goal

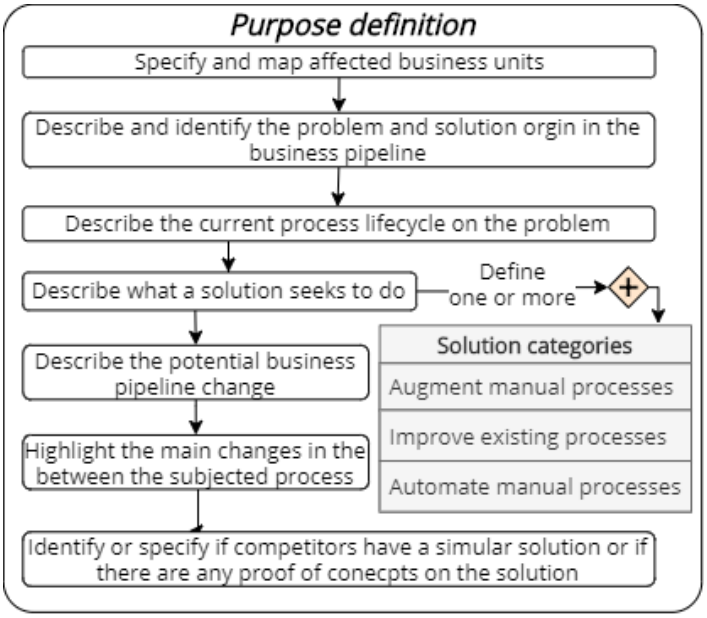

Figure 5 - Process of describing the purpose

3.1.1 Goal definition

Here, the problem will be presented along with a description of the solution and how it would work. Also, the strategic business goals linked to the solution should be identified to ensure that the solution is compatible with the company's strategic direction.

3.1.2 Purpose definition

The purpose defines the problem in more detail and identifies the business unit(s) that the problem affects. It also identifies the origin of the problem and which part of the business pipeline a solution could potentially affect. Also, the potential processes the solution seeks to change and the specific changes that would be made in the pipeline if the solution is implemented, will be described in the purpose definition.

3.1.3 Value proposition

The proposition will provide an overview of the value associated with the development and implementation of a solution for the problem. The key elements to be described are:

- Internal values and pains affecting the customers.
- The identification of the types of solutions that will be used
- The identification of the available revenues at this stage and
- Whether the solution will result in a competitive advantage in the market.

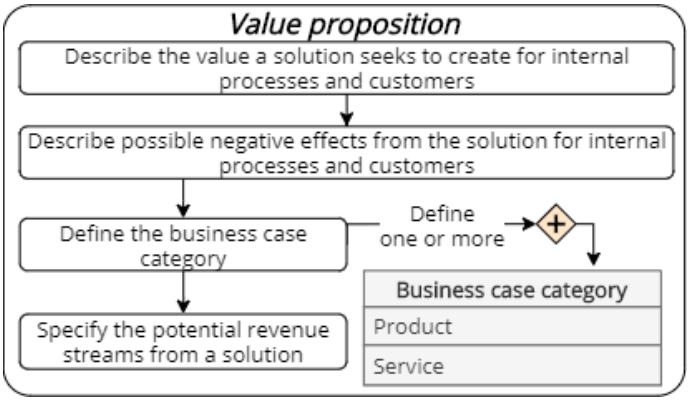

Figure 6 - Process of describing the value

3.1.4 Problem information

This last part aims to identify different aspects in order to determine whether the project is relevant to or suitable for machine-learning. To do so, both important elements in the data and the type of machine-learning project must be defined. To identify the type of machine learning problem, Table 4 can be used by describing the data vertically and moving horizontally for specifying the type of problem. Also, a problem descriptor should be specified in the form of a domain expert on the problem, and data should be specified to describe the problem

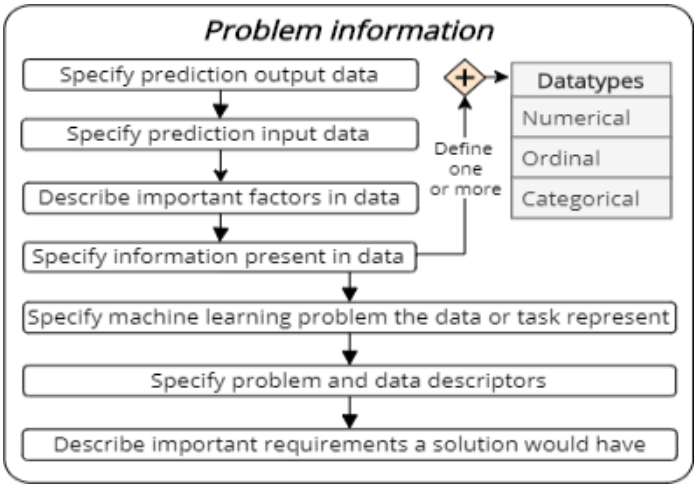

Figure 7 - Process of describing problem information

Table 4 - Helper table for identifying type of machine-learning problem (Bishop, 2006; Goodfellow, Bengio, & Courville, 2016; Hastie, Tibshirani, & Friedman, 2009; Russell & Norvig, 2009; Sutton & Barto, 2018).

Data	Type of problem		Descriptions
Input and output data are known	Supervised learning	Classification	Predicting a class label
		Regression	Predicting a numeric label
Input data known but not output	Unsupervised learning	Clustering	Finding groups in data
		Density estimation	Summarizing the distribution of data
		Visualization	Creating plots of data
		Projection	Creating lower-dimensional representations of data
Input and output data are not known	Reinforcement learning		Goal system, system learn based on feedback

3.2 Feasibility Definition

The project definition supplies the first descriptions, and the feasibility definition supplies the first analyses at an overall level. In this section, general descriptions are specified that, in conjunction with the definition, will help to rank different questions in a three-dimensional model. The score for each dimension is determined by ranking each dimensional question from 0-10, and the score along each axis is calculated by the percentage of points in each dimension and plotting it in the model. If the intersection of all dimensions ends up in the green square, it suggests the project to be very suitable for implementation.

Figure 8 shows the proposed model, with the three dimensions. The dimensional ranking questions are presented in Table 5-7. Both descriptions and questions were based on the work of Agrawal et al. (2018); Jordan (2018); Karpathy (2017); Le (2019); Megler (2019); A. Osterwalder et al. (2020); Schmitt (2019). The three dimensions are listed below for clarification:

1. Business impact (Table 5)
2. Practical feasibility (Table 6)
3. Human factors (Table 7)

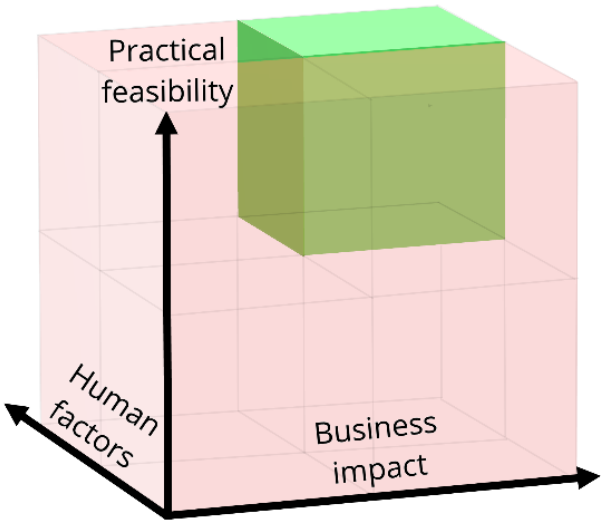

***Figure 8 - Three dimensional project feasibility model* - inspired by Agrawal, Gans, and Goldfarb (2018) definition on high impact projects, Karpathy (2017) definition on high feasibility projects and important human factors specified by A. Osterwalder, Pigneur, Smith, and Etiemble (2020)**

3.2.1 Business impact

Assesses and verifies the business impact on cost, and risks related to the business for solving the problem. The following parts related to the business impact should be described to make a manageable ROI., i.e.,

- The best- and worst-case scenarios of a potential solution,
- The minimum accuracy requirements of the predictions, and
- The potential costs and economic values with various success criteria. Also, other factors must be specified, i.e.,
- How the solution will affect business decisions
- Whether the input data will include private information
- The possibility of a discriminating solution and how consequential the results of a given the solution could potentially be.
- Determining alternative scenarios if a solution is could potential be discriminating.

Table 5 - Business impact ranking questions

Positive effects in current business pipeline
Potential economic value
Cost of wrong predictions
Cost of data acquisition
Expected ROI
Impact of inferences in model
Complications of bad press
Ethical biases in predictions
Solution liability

3.2.2 Practical feasibility

The practical aspect assesses the states of quality, obtainability, and security. Practical questions related which should be addressed are:

- Who can grant access to the data by domain experts on the problem in question?
- How and where are the data stored and the format used to store the data should be described.
- The availability of the data and their quantities should be described,
- Data relevancies of to the resultant prediction should be assessed
- Does the data have expiration dates, the quality and authenticity of the data must be described in detail?
- The importance whether the input and, or the output information in the data hold anonymous or sensitive information.
- Is data labeling required, how will the machine read the data.
- Assessing the availability and quality of related research on the problem.
- How and whether a simple or more advanced baseline can be set up for benchmarking
- Whether the data can evolve or change logic.
- List up the various machine learning algorithms identified in the problem information in chapter 3.1.4 to assess the different criteria and applicability of the algorithms related to the problem in question.

Table 6 - Practical feasibility ranking questions

Data obtainability
Data sufficiency
Data conformability
Data feasibility
Data quality
Data fairness
Research and proof of concept availability

3.2.3 Human factors

The human aspect of technology and adaptability address and assess the company and team ability for solving the problem and the feelings by potential users, customers and the team affected by a solution and its development. The company digital strategy and willingness to change is essential for reaching high scores on the human factor with data-driven decision-making processes. Looking at a firm as a network sets up a more collaborative team mentality. Interviews conducted by the most elaborated and reviewed research on digital transformation in multidisciplinary design firms from Bonanomi (2019) experts implies the importance to see the essential nature of shifting to collaboration and iteration to achieve cutting edge outcomes to tackle technological adaptation. Thus, the human factors to be assessed should involve:

- Assessment and descriptive company knowledge matrix for assessing if development can be done in house or must use external partners, if so, how should a potential partnership work?
- Does the problem have domain experts solving it traditionally to gather insights from or are the problem outside the firm's traditional practices?
- A firm may have domain expertise on the problem but how the structure, format or source of the data is will determine the potentially insights gained from the data itself
- How does the people feel about solving the problem, it is essential to have a plan and assess how to enable and account for how the end user or customer will use a potential solution.
- The team feeling about the problem should be assessed in various aspect for solvability, value, impact, accessibility.

Table 7 - Human factors ranking questions

Development expertise sufficency
Problem expertise suffiency
Data expertise suffiency
Team feeling
Customer feeling
User feeling

3.3 Example of using the framework

One case has been used for exemplification of the framework, which is the works done by For the analyzed research of cost estimation in the tendering process by Matel et al. (2019), the problem is defined as a lack of capacity and methodology to fully utilize earlier experiences of cost estimations for tender phase in quantifiable ways for ongoing and future cost estimates in tenders. Existing methods are also time-consuming.

3.3.1 Problem definition example

The defined problem is how to streamline and automate the time spent and improve the quality of cost estimations carried out in contradiction to traditional ways using machine learning. A solution will use existing knowledge and experience in data from earlier tender projects to calculate cost estimations in new projects using artificial neural networks. A perfect solution will supply better, more precise cost estimates faster than traditional methods.

The affected business units are sales, project management, and portfolio management units with relevant domain experts associated to tenders and cost estimations. In the business pipeline, the problem arises from early to the end of the tendering phase. Where coarse project details about the size, purpose, and requirements are the basis for preparing competing offer presentations between actors reflecting the project requirements given. The solution looks to partially automate and improve the traditional methods and processes used for cost estimates during the tender phase. Which can enhance the process cost estimates traditionally calculated. The main differences are potentially creating a faster and more exact cost estimation in tenders. No similar solutions have been found in consulting firms in the AEC industry Matel et al. (2019) implied; however, some were found from contractors point of perspective.

A solution looks to provide accurate cost estimation in tenders and provide more estimates based on frequent changes in project properties. The internal business values a solution would impact relates to a deeper understanding of the characteristics that affect bidding processes, potential time savings in the tendering phase, higher accuracy in cost estimates in the bidding process. Adverse effects internally and externally would be if the cost estimates used are inferior to what they would be using traditional methods and the consequences it entails.

The defined business case category will be a mix of service for use in internal processes, which has potential as a product used by customers. The revenue stream for a given solution will result in savings in tenders' time spent, leading to fewer employees on the payroll, or the opportunity to evaluate higher amounts of tenders at the same time. A later effect of the solution is the higher chance of winning tenders.

The values and properties in the system are factors found in the literature and critical factors used by the company when performing traditional cost estimations of engineering services. The evaluations will be a suggested choice of costs for project engineering works. The essential factors will be the weighting of the values to be used. Information in the data scope of the project in question requires the length, which disciplines will be involved, type of contract, among other things. The data properties of the values in the system will be of both ordinal and numeric character since the values can increase on an importance scale.

The machine learning problem can be classified as a supervised regression problem since it is used actual cost values as ground truth to train the algorithm. Table 4 in chapter 3.1.4 it can be identified as a supervised problem since both in and output data are known, and we are predicting a numerical label. In this issue, people who manage projects, calculate and work on the bidding process are essential to describe the data and the execution of tenders. Essential requirements of a potential solution will be the ability to provide reasonable cost estimates and insight into important factors that affect cost estimation in tendering processes

3.3.2 Feasibility definition example

Data were acquired through various streams for different purposes. Features properties and their ranking were obtained through earlier research and surveying domain experts. Surveys are scalable to the goal up to the needed amount. The dependent variable costs and project information depended on the amount of documented projects in the firm's project portfolio. The conformability and standardization of data are contingent on the firm's quality and data management policies.

The availability of proof of concept or similar solutions is mainly towards construction contractors (Arafa & Alqedra, 2011; Arage & Dharwadkar, 2017; Cheng, Tsai, & Sudjono, 2010; Emsley, Lowe, Duff, Harding, & Hickson, 2002; Günaydın & Dogan, 2004; Hyari, Tarawneh, & Katkhuda, 2016; Mahamid, 2013).

The business pipeline's positive effects in cost estimations in early-stage tender phases have great potential value in both time savings used on tenders and the fact that the consequences of incorrect calculations can be significant. However, the reliability of the solution should carefully individually comparing the predictions in each case to relevant baselines and actual costs. The cost of retrieving data is considered low due to requires answering surveys and examining the literature on essential factors and their importance. Model impacts can potentially have significant implications for the results; however, good quality checking procedures must be in place. If predictions cause substantial errors, and if used, it can cause bad press, yet, when using baselines as a quality check of the application, it should not be a significant problem. The same applies to the solution's stability, using earlier project estimates, time spent, and actual costs as baselines, and supplies a basic model for calculating the return of interest (ROI).

The knowledge to develop the solution should be present in Matel et al. (2019) case firm, there were also good amount of methods from reference cases and research, though limited to contractors. The data retrieved from the company with a determined format decided in advance; thus, the knowledge for developing the solution is simple. The same applies to the problem; the company often delivers offers and should have a good understanding of the process. The team, user, and customer feelings about the project are difficult to explore since the author only analyzes the methodology for the solution provided by the article from Matel et al. (2019)

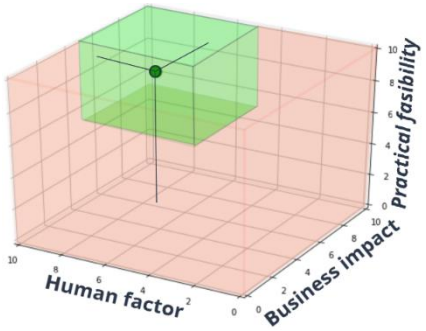

Figure 9 - Feasibility ranking exemplified on the research by Matel et al. (2019)

Table 10 – Human factors ranking exemplified on the research by Matel et al. (2019)

Questions	Ranking
Development expertise sufficiency	*7*
Problem expertise sufficency	*8*
Data/information expertise sufficency	*7*
Team feeling	*-*
Customer feeling	*-*
User feeling	*-*
Total	***7,33***

Table 9 – Practical feasibility exemplified on the research by Matel, Vahdatikhaki, Hosseinyalamdary, Evers, and Voordijk (2019)

Questions	Ranking
Data obtainability	*10*
Data sufficiency	*10*
Data conformability	*-*
Data feasibility	*10*
Data quality	*5*
Data fairness	*-*
Research and proof of concept available	*5*
Technological infrastructure sufficency	*10*
Total score	***8,33***

Table 8: Business impact ranking exemplified on the research by Matel et al. (2019)

Questions	Ranking
Positive effects in current business pipeline	*5*
Potenital economic value	*10*
Cost of wrong predictions	*2*
Cost of data acqusitions	*10*
Expected ROI	*-*
Impact of inferences in model	*4*
Complications of bad press	*3*
Ethical biases in predictions	*-*
Solution liability	*10*
Total	***6,28***

4 DISCUSSION

Machine Learning projects and their potential problems are being assessed using various methods with no single enabling outline for the user to arrive at the desired solutions. The process can be time-consuming and require the user to identify patterns when a solution is required. The problem descriptors would also have to consider all the various obstacles that one has to face during the steps leading to the outcome. Attending to obstacles is a time-consuming process and can be required to as numerous issues arise. A tried and tested framework would reduce the time taken to achieve the desired outcome.

The framework is a detailed outline that allows the problem to be defined, described and broken into smaller pieces to be studied in-depth, as mentioned by Alex Osterwalder et al. (2014). This framework also considers the parties that may be affected in obtaining the desired outcome and tries to establish the source of the issue to understand the affecting factors. It also considers the changes necessary to the existing structure in place if the solution is to be integrated into the system. These steps provide any business with a thorough examination of the problem and potential solution. The factors described above can be crucial and advantageous for businesses to understand their particular solvable machine learning problems and how to handle them in the future.

Within the problem definition stage, the proposed framework would also provide the business with a look into the wanted or needed value achievable from the final solution. The problem definition would allow the business to evaluate the value of several identified problems, which enables the opportunity to prioritize efforts between several identified problems available as an enabling force to choose the ones with the lowest risk and highest impact. Various problems would be evaluated in terms of its practical success and its outcome in the market. It would also provide the business with a priceless benefit that could allow the business to avoid issues for later discovery under developmental stages that may otherwise be evident.

The Three-dimensional project feasibility model provides the user with an analysis based on the definition of the problem that was previously. The business impact dimension is where the possible costs and risks are considered that would have to be taken up by the business to implement the solution, as mentioned by Karpathy (2017).This dimension provides the business with the best and worst possible outcomes that would not only prepare the business for any outcome but also help the business consider the worth of the solution with regards the possible consequences. It highlights the requirements that are important to understanding the success of the solution that would enable the business to focus on the necessary factors. The economic outcome of the solution would also be analyzed to again understand the worth of the solution. All the discussed factors would be of significance to any business when trying to extract the possible success rate of the available options.

The practical dimension provides the business with a breakdown of the practical aspects of the solution. This part considers the data available to the user at the beginning stages and tries to comprehend the information within the data, the format and type of data available as well as the validity of that data as stated by A. Osterwalder et al. (2020). A crucial step for businesses needs to have access to up-to-date information to be able to extract information relevant to the time and situation. Recent data enables the business to tackle the current issues at hand and face them on time.

5 CONCLUSION

The proposed framework, or any other framework for evaluating potential machine learning problems in the AEC field, takes the necessary steps to understand how to implement machine learning. With a focus on the business value and not the technology with lowering the risks and cost of developing solutions with the highest impact and lowest. When defining the problem in detail, and all the inter-related factors are studied to recognize the impact that they would have on the outcome. The value lies in considering a considerable amount of machine learning problems quickly and determining the defined problems on which one to develop when it is a benefactor for prioritized the right solution to develop. Figure 9 provides an overview of the framework and its processes.

Any business would be able to benefit from the proposed frameworks, as it enables smart decisions with a look into the repercussions of their decisions. Comprehending the consequences of a proposed solution is necessary to avoid any surprises in the future and be necessarily prepared to face any adverse outcomes. The framework can effectively help determine whether a machine learning project would be high-impact and cost-effective; however, further development is acknowledged and required by putting the framework into practice.

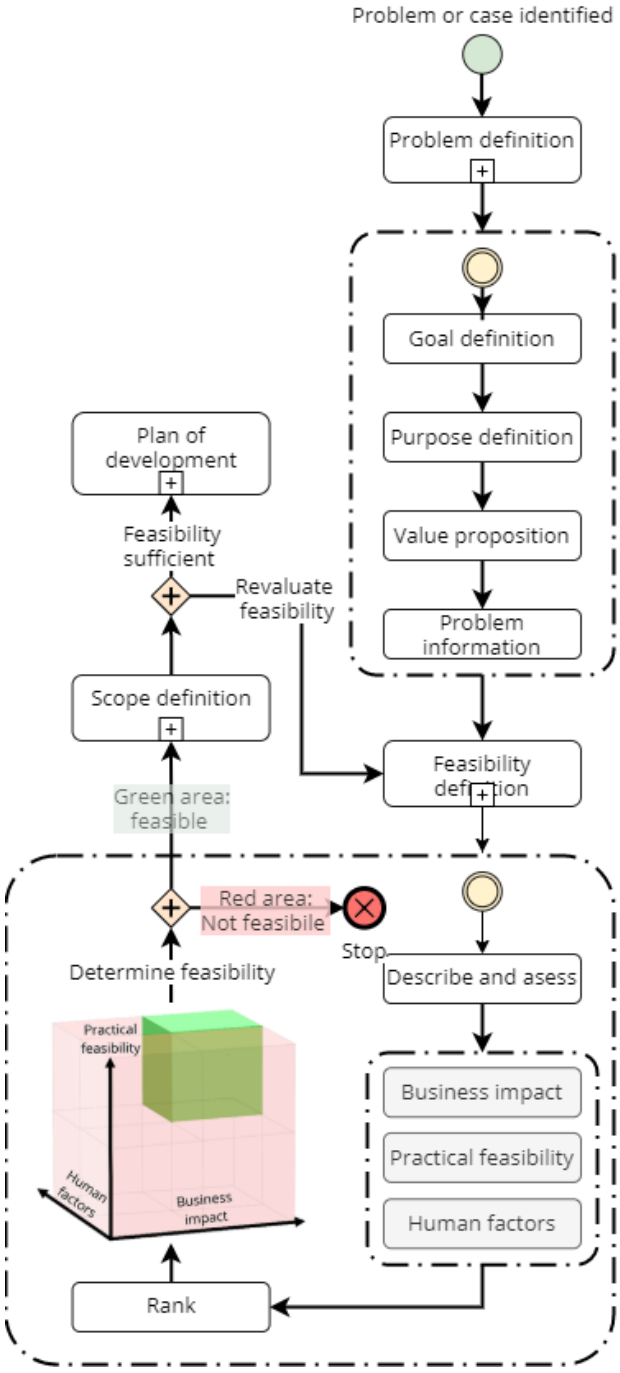

Figure 10 - Overview of framework

REFRENCES

Alstad, T. (2020). *EXPLORING AND DEVELOPMENT OF SOIL PREDICTION MODELS WITH GRADIENT BOOSTED MACHINE LEARNING ALGORITHMS* (Master). NTNU,

Arafa, M., & Alqedra, M. (2011). Early Stage Cost Estimation of Buildings Construction Projects using Artificial Neural Networks. *Journal of Artificial Intelligence, 4*. doi:10.3923/jai.2011.63.75

Agrawal, A., Gans, J., & Goldfarb, A. (2018). *Prediction Machines: The Simple Economics of Artificial Intelligence*: Harvard Business Review Press.

Arage, S., & Dharwadkar, N. (2017). *Cost estimation of civil construction projects using machine learning paradigm*.

Beeferman, D., Berger, A., & Lafferty, J. (1999). Statistical models for text segmentation. *Machine learning, 34*(1-3), 177-210.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*: Springer-Verlag.

Bonanomi, M. M. (2019). *Digital Transformation of Multidisciplinary Design Firms: A Systematic Analysis-Based Methodology for Organizational Change Management*: Springer International Publishing.

Botu, V., Batra, R., Chapman, J., & Ramprasad, R. (2017). Machine learning force fields: construction, validation, and outlook. *The Journal of Physical Chemistry C, 121*(1), 511-522.

Cheng, M.-Y., Tsai, H.-C., & Sudjono, E. (2010). Conceptual cost estimates using evolutionary fuzzy hybrid neural network for projects in construction industry. *Expert Systems with Applications, 37*(6), 4224-4231. doi:https://doi.org/10.1016/j.eswa.2009.11.080

Emsley, M., Lowe, D., Duff, A., Harding, A., & Hickson, A. (2002). Data modeling and the application of a neural network approach to the prediction of total construction costs. *Construction Management & Economics, 20*, 465-472. doi:10.1080/01446190210151050

Gil, Y., Greaves, M., Hendler, J., & Hirsh, H. (2014). Amplify scientific discovery with artificial intelligence. *Science, 346*(6206), 171-172.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*: The MIT Press.

Günaydın, H., & Dogan, S. (2004). A neural network approach for early cost estimation of structural systems of buildings. *International Journal of Project Management, 22*, 595–602. doi:10.1016/j.ijproman.2004.04.002

Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*: Springer.

Hjelseth, E. (2015). Foundations for BIM-based model checking systems: Transforming regulations into computable rules in BIM-based model checking systems.

Hyari, K., Tarawneh, Z., & Katkhuda, H. (2016). Detection Model for Unbalanced Pricing in Construction Projects: A Risk-Based Approach. *Journal of Construction Engineering and Management, Just Released*. doi:10.1061/(ASCE)CO.1943-7862.0001203

Jordan, J. (2018). Organizing machine learning projects: project management guidelines. Retrieved from https://www.jeremyjordan.me/ml-projects-guide/

Karpathy, A. (2017). Software 2.0. Retrieved from https://medium.com/@karpathy/software-2-0-a64152b37c35

Koshorek, O., Cohen, A., Mor, N., Rotman, M., & Berant, J. (2018). Text segmentation as a supervised learning task. *arXiv preprint arXiv:1803.09337*.

Kulesza, A., & Taskar, B. (2012). Determinantal Point Processes for Machine Learning. *Foundations and Trends® in Machine Learning, 5*. doi:10.1561/2200000044

Le, J. (2019). The 5 Steps to Set Your Machine Learning Projects Up for Success. Retrieved from https://medium.com/cracking-the-data-science-interview/the-5-steps-to-set-your-machine-learning-projects-up-for-success-885588fec3be

Mahamid, I. (2013). Common risks affecting time overrun in road construction projects in Palestine: Contractors' perspective. *Australasian Journal of Construction Economics and Building, 13*, 45. doi:10.5130/ajceb.v13i2.3194

Matel, E., Vahdatikhaki, F., Hosseinyalamdary, S., Evers, T., & Voordijk, H. (2019). An artificial neural network approach for cost estimation of engineering services. *International Journal of Construction Management*, 1-14. doi:10.1080/15623599.2019.1692400

Megler, V. M. (2019). Managing Machine Learning Projects.

Osterwalder, A., Pigneur, Y., Bernarda, G., Smith, A., & Papadakos, T. (2014). *Value proposition design : how to create products and services customers want*. Hoboken, N.J: Wiley.

Osterwalder, A., Pigneur, Y., Smith, A., & Etiemble, F. (2020). *The Invincible Company: How to Constantly Reinvent Your Organization with Inspiration From the World's Best Business Models*: Wiley.

Padarian, J., & Minasny, B. (2019). Using deep learning for digital soil mapping. *Soil, 5*(1), 79-89.

Rafiei, M. H., & Adeli, H. (2018). Novel Machine-Learning Model for Estimating Construction Costs Considering Economic Variables and Indexes. *Journal of Construction Engineering and Management, 144*. doi:10.1061/(ASCE)CO.1943-7862.0001570

Rossiter, D. G. (2018). Past, present & future of information technology in pedometrics. *Geoderma, 324*, 131-137.

Russell, S., & Norvig, P. (2009). *Artificial Intelligence: A Modern Approach*: Prentice Hall Press.

Schmitt, M. (2019). The Machine Learning Project Checklist. Retrieved from https://towardsdatascience.com/the-machine-learning-project-checklist-d9ee6e33a2b2

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction*: MIT Press.

Willcock, S., Martínez-López, J., Hooftman, D. A., Bagstad, K. J., Balbi, S., Marzo, A., . . . Voigt, B. (2018). Machine learning for ecosystem services. *Ecosystem services, 33*, 165-174.

Appendix A

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Feasibility definition	feasibility and impact of your projects	High feasibility projects	look for complicated rule-based software where we can learn the rules instead of programming them.	"Software 2.0," Andrej Karpathy
Feasibility definition	feasibility and impact of your projects	High Impact projects	cost of prediction and prediction is central for decision making, cheap prediction would be universal for problems across business domains. look for complex parts of your pipeline and places where cheap prediction is valuable.	Agrawal, Goldfarb, Gans (2018)
Plan of Development	Data Aggregation / Mining / Scraping	Data aggregation	sets a precedent for the effectiveness and performance of the trained model. The output of the agreed-upon solution defines the data aggregated. Data understanding is paramount and any sourced data should be examined and analyzed utilizing visualization tools or statistical methods. Data examination promoted data integrity and credibility by ensuring the data sourced is the expected data.	Alake (2020)
Plan of Development	Data Aggregation / Mining / Scraping	Data analysis and exploration	• The data gathered needs to be diverse enough to ensure that the model predictions capabilities accommodate a variety of possible scenarios. • The data gathered needs to aspire to be unbiased to ensure that the model can generalize appropriately during inference. • The data gathered needs to be abundant.	Alake (2020)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Data Preparation / Preprocessing / Augmentation		Preprocessing steps for data are based mainly on the model input requirements. Refer back to the research stage and recall input parameters and requirements that the selected model / neural network architecture requires.	Alake (2020)
Plan of Development	Data Preparation / Preprocessing / Augmentation	Data preprocessing	• Data Reformatting (resizing images, modification to color channels, noise reduction, image enhancement) • Data Cleaning • Data Normalisation	Alake (2020)
Plan of Development	Data Preparation / Preprocessing / Augmentation	Data augmentation	• Rotation of an image by any arbitrary degrees • Scaling of an image either to create zoomed in/out effects • Cropping of an image • Flipping (horizontal or vertical) of an image • Mean Subtraction	Alake (2020)
Plan of Development	Model Implementation		leveraging exiting models that are available from a variety of online sources. Most ML/DL framework such as PyTorch or TensorFlow, have pre-trained models that are leveraged to speed up the model implementation stage.These pre-trained models have been trained on robust datasets and mimic the state of the art neural network architectures' performance and structure.	Alake (2020)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Model Implementation	expected to be conducted during the model implementation stage	• Removal of last layers within a neural network to repurpose models for specific tasks. For example, removing the last layer of a Resnet neural network architecture enables the utilization of a descriptor provided by the model within an encoder-decoder neural network architecture • Fine-tuning pre-trained models	Alake (2020)
Plan of Development	Training		model training involves passing the refined aggregated training data through the implemented model to create a model that can perform its dedicated task well.	Alake (2020)
Plan of Development	Training	Hyperparameters	These are values that are defined before the training of the network begins; they are initialized to help steer the network to a positive training outcome. Their effect is on the machine / deep learning algorithm, but they are not affected by the algorithm. Their values do not change during training. Examples of hyperparameters are regularization values, learning rates, number of layers, etc.	Alake (2020)
Plan of Development	Training	Network parameter:	components of our network that are not manually initialized. They are embedded network values that are manipulated by the network directly.	Alake (2020)
Plan of Development	Training	metrics	• Training accuracy • Validation accuracy • Training Loss • Validation Loss	Alake (2020)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Training	Underfitting	This occurs when a machine learning algorithm fails to learn the patterns in a dataset. Underfitting can be fixed by using a better algorithm or model that is more suited for the task. Underfitting can also be adjusted fixed by recognizing more features within the data and presenting it to the algorithm.	Alake (2020)
Plan of Development	Training	Overfitting	This problem involves the algorithm predicting new instances of patterns presented to it, based too closely on instances of patterns it observed during training. This can cause the machine-learning algorithm to not generalize accurately to unseen data. Overfitting can occur if the training data does not accurately represent the distribution of test data. Overfitting can be fixed by reducing the number of features in the training data and reducing the complexity of the network through various techniques.	Alake (2020)
Plan of Development	Evaluation		utilize a partition of the refined data, usually referred to as the 'test data'. The test data have not been seen during the model during training. They are also representative of examples of data that are expected to be encountered in practical scenarios.	Alake (2020)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Evaluation	Confusion matrix (error matrix):	Provides a visual illustration of the number of matches or mismatches the annotation of the ground truth to the classifier results. A confusion matrix is typically structured in tabular form, where the rows are filled with the observational results from the ground-truth, and the columns are filled with inference results from the classifier.	Alake (2020)
Plan of Development	Evaluation	Precision-Recall	Provides a visual illustration of the number of matches or mismatches the annotation of the ground truth to the classifier results. A confusion matrix is typically structured in tabular form, where the rows are filled with the observational results from the ground-truth, and the columns are filled with inference results from the classifier.	Alake (2020)
Plan of Development	Parameter tuning and Inference	Parameter tuning	the process of model refinement that is conducted by making modifications to hyperparameter values. The purpose of parameter tuning is to increase the model performance, and this correlates to improvements in evaluation results. Once hyperparameters are tuned and new values are selected, training and evaluation commence again.	Alake (2020)
Plan of Development	Parameter tuning and Inference	Inference	real-world test of our model. It involves utilizing real-world data that have been sourced from applicable environments. At this stage, we should be confident in our model performance.	Alake (2020)
Plan of Development	Model Conversion to appropriate mobile format		Model conversion is a step that is required when developing models that are to be used within edge devices such as mobile phones or IoT devices.	Alake (2020)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Model Conversion to appropriate mobile format	Core ML	This is a framework released by Apple to create iOS only dedicated models. CoreML provides some models for common machine learning tasks such as recognition and detection. It's an iOS-only alternative to TensorFlow Lite.	Alake (2020)
Plan of Development	Model Conversion to appropriate mobile format	PyTorch Mobile	PyTorch is a popular machine learning framework and is used extensively in machine learning-related research. PyTorch mobile can be compared to TensorFlow Lite, as it enables the conversion of PyTorch trained model to a mobile-optimized version that can be leveraged on iOS and Android devices. Although, PyTorch Mobile is still in its infancy and currently in experimental release status.	Alake (2020)
Plan of Development	Model Conversion to appropriate mobile format	TensorFlow Lite	takes existing TensorFlow models and converts them into an optimized and efficient version in the form of a .tflite file. The streamlined model is small enough to be stored on devices and sufficiently accurate to conduct suitable inference.	Alake (2020)
Plan of Development	Model Deployment	Deploying	Integrating our model within a broader ecosystem of application or tool, or simply building an interactive web interface around our model, is an essential step of model deployment.	Alake (2020)
Plan of Development	Model Deployment	monitoring responsibility	assess the performance of the model while in a production environment. This is to ensure that the model is performing sufficiently well, and it still fit for purpose.	Alake (2020)
Plan of Development	Model Deployment	Model retraining and updating	Model updating ensures the credibility and reliability of our model for the desired task.	Alake (2020)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Model Deployment	deliverables	1. Model performance monitoring system 2. Web UI Interface to access model functionalities 3. Continuous integration pipelines that enable model redeployment	Alake (2020)
Excluded	Problem Definition		initial stage of a Computer Vision/ML project, and it focuses on gaining an understanding of the problem poised to be solved by applying ML.	Alake (2020)
Problem Definition	Problem Definition	problem descriptor	in a selected form, a scenario-based description of first-hand experience of an encounter of the problem to be solved.	Alake (2020)
Problem Definition	Problem Definition	problem descriptor	A problem descriptor can be clients, customers, users or colleagues.	Alake (2020)
Problem Definition	Problem Definition	deliverables	1. Problem Statement 2. Ideal Problem Solution 3. Understanding and insight into the problem 4. Technical requirements	Alake (2020)
Scope Definition	Research	foundation for later stages	An exploration into the form a solution will take is conducted, along with information into the data structures, formats, and sources. an understanding of the problem, unified with proposed solutions, and available data, will enable a suitable ML model selection process to achieve the ideal solution result.At this stage, it is helpful to research the hardware and software requirements for the algorithms and model implementation; this saves a lot of time in later stages.	Alake (2020)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Scope Definition	Research	deliverables	1. Data Structure and Source 2. Solution form 3. Neural Network / Model Architecture 4. Algorithm Research 5. Hardware Requirements 6. Software Requirements	Alake (2020)
Excluded	Strategy	chief analytics officer	Defines if the project should og ahead	AltexSoft (2018)
Excluded	Strategy	business analyst	defines the feasibility of a software solution and sets the requirements for it	AltexSoft (2018)
Excluded	Strategy	solution architect	organizes the development	AltexSoft (2018)
Excluded	Strategy	solution architect	make sure these requirements become a base for a new solution	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing		Data is the foundation for any machine learning project. The second stage of project implementation is complex and involves data collection, selection, preprocessing, and transformation	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data collection	find ways and sources of collecting relevant and comprehensive data, interpreting it, and analyzing results with the help of statistical techniques.	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data collection tools	Mixpanel, Hotjar, CrazyEgg, well-known Google analytics	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data visualization	large amount of information represented in graphic form is easier to understand and analyze.	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data visualization tools	Visualr, Tableau, Oracle DV, QlikView, Charts.js, dygraphs, D3.js	AltexSoft (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Dataset preparation and preprocessing	Labeling	An algorithm must be shown which target answers or attributes to look for. Mapping these target attributes in a dataset is called labeling.	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Labeling: Outsourcing	outsourcing it to contributors from CrowdFlower or Amazon Mechanical Turk platforms if labeling requires no more than common knowledge.	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Labeling: Domain experts	But in some cases, specialists with domain expertise must assist in labeling	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Labeling: Transfer learning	repurpose labeled training data with transfer learning. This technique is about using knowledge gained while solving similar machine learning problems by other data science teams. A data scientist needs to define which elements of the source training dataset can be used for a new modeling task	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Labeling Tools	crowdsourcing labeling platforms, spreadsheets	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data selection	subgroup of data to solve the defined problem	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data selection tools	spreadsheets, MLaaS	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data preprocessing	convert raw data into a form that fits machine learning	AltexSoft (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Dataset preparation and preprocessing	Data preprocessing: Data formatting	standardize recorded formats, checks whether variables representing each attribute are recorded in the same way	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data preprocessing: Data cleaning	removing noise and fixing inconsistencies in data	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data preprocessing: Data anonymization	anonymize or exclude attributes representing sensitive information	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data preprocessing: Data sampling	technique to select a smaller but representative data sample to build and run models much faster, and at the same time to produce accurate outcomes.	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data preprocessing tools	spreadsheets, automated solutions (Weka, Trim, Trifacta Wrangler, RapidMiner), MLaaS (Google Cloud AI, Amazon Machine Learning, Azure Machine Learning)	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data transformation	transforms or consolidates data into a form appropriate for mining (creating algorithms to get insights from data) or machine learning.	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data transformation: Scaling	Data may have numeric attributes (features) that span different ranges, for example, millimeters, meters, and kilometers. Scaling is about converting these attributes so that they will have the same scale, such as between 0 and 1, or 1 and 10 for the smallest and biggest value for an attribute.	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data transformation: Decomposition	finding patterns in data with features representing complex concepts, converts higher level features into lower level ones. In other words, new features based on the existing ones are being added	AltexSoft (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Dataset preparation and preprocessing	Data transformation: Aggregation	combining several features into a feature that represents them all. applied techniques and the number of iterations depend on a business problem and therefore on the volume and quality of data collected for analysis.	AltexSoft (2018)
Plan of Development	Dataset preparation and preprocessing	Data transformation: Tools	spreadsheets, automated solutions (Weka, Trim, Trifacta Wrangler, RapidMiner), MLaaS (Google Cloud AI, Amazon Machine Learning, Azure Machine Learning)	AltexSoft (2018)
Plan of Development	Dataset splitting	Training set	training set to train a model and define its optimal parameters — parameters it has to learn from data.	AltexSoft (2018)
Plan of Development	Dataset splitting	Test set	needed for an evaluation of the trained model and its capability for generalization. The latter means a model's ability to identify patterns in new unseen data after having been trained over a training data	AltexSoft (2018)
Plan of Development	Dataset splitting	Validation set	validation set is to tweak a model's hyperparameters — higher-level structural settings that can't be directly learned from data	AltexSoft (2018)
Plan of Development	Dataset splitting	tools	MLaaS (Google Cloud AI, Amazon Machine Learning, Azure Machine Learning), ML frameworks (TensorFlow, Caffe, Torch, scikit-learn)	AltexSoft (2018)
Plan of Development	Modelling	Model training	Feeding the algorithm with training data. An algorithm will process data and output a model that is able to find a target value (attribute) in new data	AltexSoft (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Modelling	Model training: Supervised learning	attributes are mapped in historical data before the training begins. With supervised learning, a data scientist can solve classification and regression problems.	AltexSoft (2018)
Plan of Development	Modelling	Model training: Unsupervised learning	find hidden interconnections between data objects and structure objects by similarities or differences. Unsupervised learning aims at solving such problems as clustering, association rule learning, and dimensionality reduction	AltexSoft (2018)
Plan of Development	Modelling	Model training: Tools	MLaaS (Google Cloud AI, Amazon Machine Learning, Azure Machine Learning), ML frameworks (TensorFlow, Caffe, Torch, scikit-learn)	AltexSoft (2018)
Plan of Development	Model evaluation and testing		develop the simplest model able to formulate a target value fast and well enough.	AltexSoft (2018)
Plan of Development	Model evaluation and testing	Cross-validation	A given model is trained on only nine folds and then tested on the tenth one (the one previously left out). Training continues until every fold is left aside and used for testing. . The cross-validated score indicates average model performance across ten hold-out folds.	AltexSoft (2018)
Plan of Development	Model evaluation and testing	Improving predictions with ensemble methods	create and train one or several dozen models to be able to choose the optimal model among well-performing ones	AltexSoft (2018)
Plan of Development	Model evaluation and testing	Stacking	developing a meta-model or higher-level learner by combining multiple base models	AltexSoft (2018)
Plan of Development	Model evaluation and testing	Bagging (bootstrap aggregating).	training dataset is split into subsets. Then models are trained on each of these subsets. After this, predictions are combined using mean or majority voting	AltexSoft (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Model evaluation and testing	Boosting	uses subsets of an original dataset to develop several averagely performing models and then combines them to increase their performance using majority vote	AltexSoft (2018)
Plan of Development	Model evaluation and testing	Tools	MLaaS (Google Cloud AI, Amazon Machine Learning, Azure Machine Learning), ML frameworks (TensorFlow, Caffe, Torch, scikit-learn)	AltexSoft (2018)
Plan of Development	Model deployment		Once a data scientist has chosen a reliable model and specified its performance requirements, he or she delegates its deployment to a data engineer or database administrator. After translating a model into an appropriate language, a data engineer can measure its performance with A/B testing. Testing can show how a number of customers engaged with a model used for a personalized recommendation, for example, correlates with a business goal.	AltexSoft (2018)
Plan of Development	Model deployment	Batch prediction	appropriate when you don't need your predictions on a continuous basis. When you choose this type of deployment, you get one prediction for a group of observations. A model is trained on static dataset and outputs a prediction. You can deploy a model on your server, on a cloud server if you need more computing power or use MlaaS for it.	AltexSoft (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Model deployment	Web service	Such machine learning workflow allows for getting forecasts almost in real time. A model however processes one record from a dataset at a time and makes predictions on it. It's possible to deploy a model using MLaaS platforms, in-house, or cloud servers.	AltexSoft (2018)
Plan of Development	Model deployment	Real-time prediction (real-time streaming or hot path analytics)	analyze live streaming data and quickly react to events that take place at any moment. Real-time prediction allows for processing of sensor or market data, data from IoT or mobile devices, as well as from mobile or desktop applications and websites	AltexSoft (2018)
Plan of Development	Model deployment	Stream learning	dynamic machine learning models capable of improving and updating themselves. You can deploy a model capable of self learning if data you need to analyse changes frequently	AltexSoft (2018)
Plan of Development	Model deployment	Tools	MlaaS (Google Cloud AI, Amazon Machine Learning, Azure Machine Learning), ML frameworks (TensorFlow, Caffe, Torch, scikit-learn), open source cluster computing frameworks (Apache Spark), cloud or in-house servers	AltexSoft (2018)
Plan of development	Testing and evaluation	Training system	Test the full training pipeline (from raw data to trained model) to ensure that changes haven't been made upstream with respect to how data from our application is stored. These tests should be run nightly/weekly.	Andrej Karpathy (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of development	Testing and evaluation	Prediction system	Run inference on the validation data (already processed) and ensure model score does not degrade with new model/weights. This should be triggered every code push.	Andrej Karpathy (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Excluded	Evaluating production readiness		• Feature expectations are captured in a schema.• All features are beneficial.• No feature's cost is too much.• Features adhere to meta-level requirements.• The data pipeline has appropriate privacy controls.• New features can be added quickly.• All input feature code is tested.Model:• Model specs are reviewed and submitted.• Offline and online metrics correlate.• All hyperparameters have been tuned.• The impact of model staleness is known.• A simple model is not better.• Model quality is sufficient on important data slices.• The model is tested for considerations of inclusion.Infrastructure:• Training is reproducible.• Model specs are unit tested.• The ML pipeline is integration tested.• Model quality is validated before serving.• The model is debuggable.• Models are canaried before serving.• Serving models can be rolled back.Monitoring:• Dependency changes result in notification.• Data invariants hold for inputs.• Training and serving are not skewed.• Models are not too stale.• Models are numerically stable.• Computing performance has not regressed.• Prediction quality has not regressed.	Eric Breck Shanqing Cai Eric Nielsen Michael Salib D. Sculley (2017)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Applied machine learning	Starting point	ModelZoo.co, facebook AI research, spaCy, OpenAI, PyTorch, OpenCV, Sosephmisti, PyTorch, Ncidia ADLR, TensorFlow Hub, Uber AI	Helica Inc
Excluded	Project lifecycle	Team roles	• data engineer (builds the data ingestion pipelines) • machine learning engineer (train and iterate models to perform the task) • software engineer (aids with integrating machine learning model with the rest of the product) • project manager (main point of contact with the client)	Jordan (2018)
Feasibility definition	Determining feasibility	feasibility of a project	• Cost of data acquisition• Is there sufficient literature on the problem?• Computational resources available both for training and inference• Will the model be deployed in a resource-constrained environment?• How hard is it to acquire data?• How expensive is data labeling?• How much data will be needed?• Cost of wrong predictions• How frequently does the system need to be right to be useful?• Availability of good published work about similar problems• Has the problem been reduced to practice?	Jordan (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Project lifecycle	Data collection and labeling (Step 2)	• Define ground truth (create labeling documentation) • Build data ingestion pipeline • Validate quality of data • Revisit Step 1 and ensure data is sufficient for the task	Jordan (2018)
Plan of Development	Project lifecycle	Model exploration (Step 3)	• Establish baselines for model performance • Start with a simple model using initial data pipeline • Overfit simple model to training data • Stay nimble and try many parallel (isolated) ideas during early stages • Find SoTA model for your problem domain (if available) and reproduce results, then apply to your dataset as a second baseline • Revisit Step 1 and ensure feasibility • Revisit Step 2 and ensure data quality is sufficient	Jordan (2018)
Plan of Development	Project lifecycle	Model refinement (Step 4)	• Perform model-specific optimizations (ie. hyperparameter tuning) • Iteratively debug model as complexity is added • Perform error analysis to uncover common failure modes • Revisit Step 2 for targeted data collection of observed failures	Jordan (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Project lifecycle	Testing and evaluation (Step 5)	• Evaluate model on test distribution; understand differences between train and test set distributions (how is “data in the wild” different than what you trained on)• Revisit model evaluation metric; ensure that this metric drives desirable downstream user behavior• Write tests for:• Input data pipeline• Model inference functionality• Model inference performance on validation data• Explicit scenarios expected in production (model is evaluated on a curated set of observations)	Jordan (2018)
Plan of Development	Project lifecycle	Model deployment (Step 6)	• Expose model via a REST API • Deploy new model to small subset of users to ensure everything goes smoothly, then roll out to all users • Maintain the ability to roll back model to previous versions • Monitor live data and model prediction distributions	Jordan (2018)
Plan of Development	Project lifecycle	Ongoing model maintenance (Step 7)	• Understand that changes can affect the system in unexpected ways • Periodically retrain model to prevent model staleness • If there is a transfer in model ownership, educate the new team	Jordan (2018)
Plan of development	Setting up a ML codebase	data/	provides a place to store raw and processed data for your project.	Jordan (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of development	Setting up a ML codebase	data/README.md	file which describes the data for your project.	Jordan (2018)
Plan of development	Setting up a ML codebase	docker/	is a place to specify one or many Dockerfiles for the project. Docker (and other container solutions) help ensure consistent behavior across multiple machines and deployments.	Jordan (2018)
Plan of development	Setting up a ML codebase	api/app.py	exposes the model through a REST client for predictions. You will likely choose to load the (trained) model from a model registry rather than importing directly from your library.	Jordan (2018)
Plan of development	Setting up a ML codebase	models/	defines a collection of machine learning models for the task, unified by a common API	Jordan (2018)
Plan of development	Setting up a ML codebase	base.py	These models include code for any necessary data preprocessing and output normalization.	Jordan (2018)
Plan of development	Setting up a ML codebase	datasets.py	manages construction of the dataset. Handles data pipelining/staging areas, shuffling, reading from disk.	Jordan (2018)
Plan of development	Setting up a ML codebase	experiment.py	manages the experiment process of evaluating multiple models/ideas. This constructs the dataset and models for a given experiment.	Jordan (2018)
Plan of development	Setting up a ML codebase	train.py	defines the actual training loop for the model. This code interacts with the optimizer and handles logging during training.See other examples here, here and here.	Jordan (2018)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Problem Definition	Specifying project requirements	Examples:	• Optimize for accuracy • Prediction latency under 10 ms • Model requires no more than 1gb of memory • 90% coverage (model confidence exceeds required threshold to consider a prediction as valid)	Jordan (2018)
Strategy	Project lifecycle	Planning and project setup (Step 1)	• Define the task and scope out requirements • Determine project feasibility • Discuss general model tradeoffs (accuracy vs speed) • Set up project codebase	Jordan (2018)
Feasibility definition	feasibility and impact of your projects	prioritizing projects	build projects with high impact and high feasibility (aka, low cost).	le (2019)
Plan of Development	Phase 2 — Data Collection and Labeling	Collect training data	Collecting relevant data to the problem (images, text, tabular, etc.).	le (2019)
Plan of Development	Phase 3 — Model Training and Model Debugging	Implement baseline models quickly	find and reproduce state-of-the-art methods for the problem domain, debug our implementation, and improve the model performance for specific tasks	le (2019)
Plan of Development	Phase 4 — Model Deploying and Model Testing	Pilot the model in a constrained environment	tests to prevent regressions, and roll the model into production, keep improving the model's accuracy, fix the mismatch between training data and production data by collecting more data and mining hard cases.	le (2019)
Problem Definition	Problem Definition	Value proposition	What value the solved problem will amount to	le (2019)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
excluded	Phase 1 — Project Planning and Project Setup	Decide the problem	Determine the requirements and goals	le (2019)
Problem Definition	Project Archetype	improve an existing process	improving route optimization in a ride-sharing service, building a customized churn prevention model, building a better video game AI...Does performance improvement generate business value? Do performance improvements lead to a data flywheel?	le (2019)
Problem Definition	Project Archetype	augment a manual process	turning mockup designs into application UI, building a sentence auto-completion feature, helping a doctor to do his/her job more efficient... How well does the system need to be so that the prediction can be useful? How can you collect enough data to make it that good?	le (2019)
Problem Definition	Project Archetype	automate a manual process	developing autonomous vehicles, automating customer service, automating website design... What is an acceptable failure rate for the system? How can you guarantee that it won't exceed that failure rate? How inexpensively can you label data from the system?	le (2019)
Problem Definition	Problem Solution	Project Lifecycle	How would a potential lifecycle look like	le (2019)
Scope Definition	Project Metrics	prediction scores	simple average, F1 score, Precision and Recall, choose ones that are least sensitive to model choice and are closest to desirable values.	le (2019)
Feasibility definition	Project Baselines	Baseline models	baseline is a model that is both simple to set up and has a reasonable chance of providing decent results to compare against	le (2019)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Applied machine learning	Python Libraries	1. Numpy	Pant (2019)
Plan of Development	Applied machine learning	Python Libraries	2. Pandas	Pant (2019)
Plan of Development	Applied machine learning	Python Libraries	3. Sci-kit Learn	Pant (2019)
Plan of Development	Applied machine learning	Python Libraries	4. Matplotlib	Pant (2019)
Plan of Development	Project Strategy	define the machine learning workflow	1. Gathering data 2. Data pre-processing 3. Researching the model that will be best for the type of data 4. Training and testing the model 5. Evaluation	Pant (2019)
Plan of Development	Data pre-processing	Data collection	The process of gathering data depends on the type of project we desire to make	Pant (2019)
Plan of Development	Data pre-processing	preprocessing	Data pre-processing is a process of cleaning the raw data i.e. the data is collected in the real world and is converted to a clean data set	Pant (2019)
Plan of Development	Data pre-processing	messy datatypes	1. Missing data: Missing data can be found when it is not continuously created or due to technical issues in the application (IOT system).	Pant (2019)
Plan of Development	Data pre-processing	messy datatypes	2. Noisy data: This type of data is also called outliners, this can occur due to human errors (human manually gathering the data) or some technical problem of the device at the time of collection of data.	Pant (2019)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Data pre-processing	messy datatypes	3. Inconsistent data: This type of data might be collected due to human errors (mistakes with the name or values) or duplication of data.	Pant (2019)
Plan of Development	Data pre-processing	Ignoring the missing values	Whenever we encounter missing data in the data set then we can remove the row or column of data depending on our need. This method is known to be efficient but it shouldn't be performed if there are a lot of missing values in the dataset.	Pant (2019)
Plan of Development	Data pre-processing	Filling the missing values	Whenever we encounter missing data in the data set then we can fill the missing data manually, most commonly the mean, median or highest frequency value is used.	Pant (2019)
Plan of Development	Data pre-processing	Machine learning	If we have some missing data then we can predict what data shall be present at the empty position by using the existing data.	Pant (2019)
Plan of Development	Data pre-processing	Outliers detection	There are some error data that might be present in our data set that deviates drastically from other observations in a data set. [Example: human weight = 800 Kg; due to mistyping of extra 0]	Pant (2019)
Plan of Development	Dataset split	Training and testing the model on data	train the classifier using 'training data set', tune the parameters using 'validation set' and then test the performance of your classifier on unseen 'test data set'	Pant (2019)
Plan of Development	Dataset split	Training set	The training set is the material through which the computer learns how to process information. Machine learning uses algorithms to perform the training part. A set of data used for learning, that is to fit the parameters of the classifier.	Pant (2019)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Plan of Development	Dataset split	Validation set	Cross-validation is primarily used in applied machine learning to estimate the skill of a machine learning model on unseen data. A set of unseen data is used from the training data to tune the parameters of a classifier.	Pant (2019)
Plan of Development	Dataset split	Test set	A set of unseen data used only to assess the performance of a fully-specified classifier.	Pant (2019)
Plan of Development	confusion matrix		has 4 parameters, which are 'True positives', 'True Negatives', 'False Positives' and 'False Negative'. We prefer that we get more values in the True negatives and true positives to get a more accurate model. The size of the Confusion matrix completely depends upon the number of classes.	Pant (2019)
Plan of Development	confusion matrix	True positives	cases in which we predicted TRUE and our predicted output is correct.	Pant (2019)
Plan of Development	confusion matrix	True negatives	predicted FALSE and our predicted output is correct.	Pant (2019)
Plan of Development	confusion matrix	False positives	predicted TRUE, but the actual predicted output is FALSE.	Pant (2019)
Plan of Development	confusion matrix	False negatives	predicted FALSE, but the actual predicted output is TRUE.	Pant (2019)
Plan of Development	Evaluation		helps to find the best model that represents our data and how well the chosen model will work in the future.	Pant (2019)
Plan of Development	Evaluation	model improvement	tune the hyper-parameters of the model and try to improve the accuracy and also looking at the confusion matrix to try to increase the number of true positives and true negatives.	Pant (2019)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Problem Definition	Supervised Learning		In Supervised learning, an AI system is presented with data which is labelled, which means that each data tagged with the correct label.	Pant (2019)
Problem Definition	Supervised Learning	Classification	Classification problem is when the target variable is categorical (i.e. the output could be classified into classes — it belongs to either Class A or B or something else).	Pant (2019)
Problem Definition	Supervised Learning	most used classification algorithms	K-Nearest Neighbor, Naive Bayes, Decision Trees/Random Forest, Support, Vector Machine,Logistic Regression	Pant (2019)
Problem Definition	Supervised Learning	Regression	A Regression problem is when the target variable is continuous (i.e. the output is numeric).	Pant (2019)
Problem Definition	Supervised Learning	most used regression algorithms	Linear Regression, Support Vector Regression, Decision Tress/Random Forest, Gaussian Progresses Regression, Ensemble Methods	Pant (2019)
Problem Definition	Unsupervised Learning		In unsupervised learning, an AI system is presented with unlabeled, un-categorized data and the system's algorithms act on the data without prior training. The output is dependent upon the coded algorithms. Subjecting a system to unsupervised learning is one way of testing AI.	Pant (2019)
Problem Definition	Unsupervised Learning	Clustering	A set of inputs is to be divided into groups. Unlike in classification, the groups are not known beforehand, making this typically an unsupervised task.	Pant (2019)
Problem Definition	Unsupervised Learning	Methods used for clustering	Gaussian mixtures, K-Means Clustering, Boosting, Hierarchical Clustering, K-Means Clustering, Spectral Clustering	Pant (2019)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Problem Definition	Data pre-processing	datatypes	1. Numeric e.g. income, age	Pant (2019)
Problem Definition	Data pre-processing	datatypes	2. Categorical e.g. gender, nationality	Pant (2019)
Problem Definition	Data pre-processing	datatypes	3. Ordinal e.g. low/medium/high	Pant (2019)
Scope Definition	Data pre-processing	Conversion of data	Machine Learning models can only handle numeric features, hence categorical and ordinal data must be somehow converted into numeric features.	Pant (2019)
unsure	Project Strategy	Performance measurement	do you have benchmark to compare against	Schmitt (2019)
unsure	Project Strategy	Performance measurement	how will the accuracy be measured	Schmitt (2019)
Feasibility definition	Project Strategy	Performance measurement	what is the minimum accuracy for the project	Schmitt (2019)
Problem Definition	Project Strategy	Performance measurement	what would a perfect solution do	Schmitt (2019)
Feasibility definition	Project Strategy	Performance measurement	are there referencing solutions	Schmitt (2019)
Feasibility definition	Project Strategy	Contacts	Who can grant access to the datasets?	Schmitt (2019)
Feasibility definition	Project Strategy	Contacts	Who can help understand the current process and / or the simple benchmark (domain expert)?	Schmitt (2019)
Problem Definition	Project Strategy	Project Motivation	What is the problem you want to solve	Schmitt (2019)
Problem Definition	Project Strategy	Project Motivation	What strategic goal is this connected to	Schmitt (2019)
Problem Definition	Project Strategy	Problem Definition	What specific output do you want to predict	Schmitt (2019)
Problem Definition	Project Strategy	Problem Definition	what input data do you have for the algorithm	Schmitt (2019)
Problem Definition	Project Strategy	Problem Definition	what is the relevant factors in the data	Schmitt (2019)
Scope Definition	Project Strategy	Timeline	Are there any deadlines to be aware of?	Schmitt (2019)
Scope Definition	Project Strategy	Timeline	When do you need to see the first results?	Schmitt (2019)
Scope Definition	Project Strategy	Timeline	When do you want to have a finished solution?	Schmitt (2019)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
excluded	Project Strategy	Contacts	Who is responsible for the project (PM)?	Schmitt (2019)
Plan of development	Model refinement	Addressing underfitting	1. Increase model capacity 2. Reduce regularization 3. Error analysis 4. Choose a more advanced architecture (closer to state of art) 5. Tune hyperparameters 6. Add features	Stephen Merity (2017)
Plan of development	Model refinement	Addressing overfitting	1. Add more training data 2. Add regularization 3. Add data augmentation 4. Error analysis 5. Tune hyperparameters 6. Reduce model size	Stephen Merity (2017)
Plan of development	Model refinement	Addressing distribution shift:	1. Perform error analysis to understand nature of distribution shift 2. Synthesize data (by augmentation) to more closely match the test distribution 3. Apply domain adaptation techniques	Stephen Merity (2017)
Plan of development	Model refinement	Debugging ML projects	• Implementation bugs • Hyperparameter choices • Data/model fit • Dataset construction	Stephen Merity (2017)
Feasibility definition	Assessing Economic Value	project costs and risks	The economic value and risk analysis should include the end-to-end process.	V.M. Megler (2019)
Feasibility definition	Assessing Economic Value	Figure	This model will provide context to inform project decisions, moving the focus from the ML technology to its impact on the business. Model given p 17	V.M. Megler (2019)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
unsure	Manage and Mitigate Risk	Sample Scorecards	Project context – Addresses the social, business, and regulatory environment of the project	V.M. Megler (2019)
Feasibility definition	Financial	Financial model built	Model with anticipated ROI available for review	V.M. Megler (2019)
Excluded	Financial	Potential upside return	Increased customer retention of 5%, Decreased cost per transaction of 5%	V.M. Megler (2019)
Excluded	Financial	Potential downside risk	Decreased customer retention (10%), Increased cost per transaction (5%)	V.M. Megler (2019)
Feasibility definition	Financial	Worst-case downside	Automated trading algorithm causes Great Financial Crash	V.M. Megler (2019)
Feasibility definition	Financial	Liability	Self-driving car kills pedestrian	V.M. Megler (2019)
Scope definition	Financial	Cost of building model	6 months, team of 225	V.M. Megler (2019)
Scope definition	Financial	Cost of maintaining model	Ongoing, 10 hours/month	V.M. Megler (2019)
Excluded	Data Quality	Input data precision	Test & production data have same characteristics; outliers discarded for both model & production	V.M. Megler (2019)
Excluded	Data Quality	Production vs model data pipeline	Prod inferences will use separate data source than model trained on	V.M. Megler (2019)
Feasibility definition	Data Quality	Data change over time: processes considered	Upstream system changes logic & meaning of its input to model	V.M. Megler (2019)
Excluded	Project Processes	Research or Development	Research, leading to development if successful	V.M. Megler (2019)
Feasibility definition	Project Processes	Project Team Skills & Availability	Team is missing production application development skills	V.M. Megler (2019)
Scope definition	Project Processes	Project Timelines	Committed timelines assume data is available, appropriate, and sufficient for model	V.M. Megler (2019)
Feasibility definition	Project Processes	Tests for bias applied	Recidivism rates the same across all populations	V.M. Megler (2019)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Feasibility definition	Project Processes	Long tail analysis performed	Minority groups disadvantaged because system is trained on majority	V.M. Megler (2019)
Excluded	Project Processes	Statistical analysis validated	Incorrect statistical analysis shows correlation where none exists, leading to incorrect inferences	V.M. Megler (2019)
Feasibility definition	Project Processes	Economic analysis of model metrics	Results of model are still in range of project economic value estimates	V.M. Megler (2019)
Feasibility definition	Project Processes	Team temperature check	Team gut check: team willing to be a customer of the system	V.M. Megler (2019)
Excluded	Project Processes	Verification and validation procedures completed	Testing completed: privacy assurances, A/B testing	V.M. Megler (2019)
scope definition	Project Processes	Security assessment completed	Integration with existing processes & systems	V.M. Megler (2019)
Feasibility definition	Financial	Quality of model predictions vs expectations	Economic model assumes 100% correct predictions, but results 85% correct	V.M. Megler (2019)
Feasibility definition	Financial	Uncertainty in model predictions	Prediction might be accurate +/– 10% Extreme data points cause bad predictions	V.M. Megler (2019)
Scope definition	Project Context	Ethics	Weapons targeting systems, Predictive policing, AI imitating humans	V.M. Megler (2019)
Feasibility definition	Project Context	Model makes consequential decisions	Denying people entry to country, or loans. Criminal risk assessments for arrests, bail, sentencing	V.M. Megler (2019)
Excluded	Project Context	Privacy	HIPAA / GDPR applies	V.M. Megler (2019)
Feasibility definition	Project Context	Fairness, Bias	Race identified as loan risk factor. Displaced jobs disproportionately held by minorities	V.M. Megler (2019)
Feasibility definition	Project Context	Risk of bad press	Photo labeling app labels African-American as gorilla. Self-driving car kills pedestrian	V.M. Megler (2019)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
Scope definition	Project Context	Need for transparency & auditability	Recommendations must be independently verifiable	V.M. Megler (2019)
Scope definition	Project Context	Applicability/ success of ML for this application	Natural language assistant chat bots vs free-form conversational understanding	V.M. Megler (2019)
Scope definition	Project Context	Closed-world development/ testing vs open-world deployment	Robot in lab vs open house environment (children, pets, stairs)	V.M. Megler (2019)
Feasibility definition	Project Context	Impact of ML model inferences	Self-driving car sees pedestrian but ignores it as false-positive	V.M. Megler (2019)
Feasibility definition	Data Quality	Input data accuracy	Sensor values estimated to be +- 5% of actual	V.M. Megler (2019)
Feasibility definition	Data Quality	Data volumes & duration	Model data only available for 3 months (but business cycle is 1 year), 50% of source #3 data discarded	V.M. Megler (2019)
Excluded	Data Quality	Data sources & pre-processing validated	Data source #1 now undergoing additional quality checks. Data extract #2 discovered to be flawed; re-training required	V.M. Megler (2019)
Scope definition	Project Processes	New interfaces or APIs required	User process changes required	V.M. Megler (2019)
Scope definition	Project Processes	Instrumentation & monitoring in place	Model endpoint is instrumented to feed data back into training process. Monitoring process that identifies model drift is defined	V.M. Megler (2019)
excluded	Project Processes	Production model revisions	Plan to retrain, relaunch models in place & funded	V.M. Megler (2019)

Proposed strategy phase	Topic	Subject	Description or example	Referencing Article
unsure	Manage and Mitigate Risk	Sample Scorecards	Financial – Identifies the costs and benefits of the problem you are trying to solve with ML, and of the ML system you are developing	
unsure	Manage and Mitigate Risk	Sample Scorecards	• Data quality – Highlights areas that are frequently problematic in ML projects, and that can easily mislead the project—missing a signal that exists in the data or believing a signal exists where there is none—if not identified and addressed Page 21	
unsure	Manage and Mitigate Risk	Sample Scorecards	• Project processes – Addresses processes in the ML project that are easily overlooked in the excitement of developing and testing the algorithm and identifying promising results	
unsure	Manage and Mitigate Risk	Sample Scorecards	• Summary – Captures the key risk areas to bring to executive attention	

NTNU

Norwegian University of
Science and Technology