

FMH606 Master's Thesis 2022

Process Technology

Foaming prediction in the post-combustion CO₂ capture plants (amine-based) by utilizing machine learning techniques

Nasir Niayifar

Faculty of Technology, Natural sciences and Maritime Sciences
Campus Porsgrunn

Course: FMH606 Master's Thesis, 2022

Title: Foaming prediction in the post-combustion CO₂ capture plants (amine-based) by utilizing machine learning techniques

Number of pages: 54

Keywords: Global warming, CO₂ Capture, Foaming prediction, Data-driven models, Artificial neural network, Decision tree

Student: Nasir Niayifar

Supervisors: Prof. Leila Ben Saad and Prof. Ru Yan

External partner: Technology Centre Mongstad (TCM)

Summary:

Human activities have increased the emitted greenhouse gases into the atmosphere. Among greenhouse gases, the excess of CO₂ in the atmosphere has caused severe environmental issues such as global warming and ozone depletion. Several international agreements, such as the Paris Agreement, have been signed to reduce CO₂ emissions and their impacts on the environment. These agreements aim to reduce the CO₂ footprint regarding all human activities (move toward CO₂ neutralization). For this reason, there have been substantial efforts to develop new methods and technologies applied to the exhaust gas from industrial activities to reduce the concentration of greenhouse gases (in a way that is not harmful for human and environment) prior to release them into atmosphere. Post-combustion flue gases contain significant amount of CO₂. One common method for capturing CO₂ released from post-combustion flue gases is to use an amine-based (i.e., solvent) CO₂ capture plant. Although these plants have demonstrated good performances, the problem of foaming within their columns (absorber and desorber) reduces the plant's efficiency. Due to the complexity of the process, there is no physical model that can simulate foaming within a post-combustion CO₂ capture (amine-based) plant. Therefore, the main goal of this report is to develop, for the first time, a data-driven model that can simulate and predict the mentioned undesirable characteristic. The data used in this report was provided by technology center Mongstad (TCM) that is the external partner of this project. The data includes the time series (10968 hourly time-steps) of 35 features (i.e., physical properties of the process) regarding the CO₂ capture process of TCM post-combustion CO₂ capture plant (amine-based). It is worthwhile to mention that the solvent used in the TCM plant was CESAR1. Comprehensive data preprocessing was done in order to tag foaming/non-foaming time-steps as well as to ensure the high data quality before feeding the data to models. Furthermore, a correlation-based method was used for feature selection in order to avoid feeding statistically similar features to the models (retained features are 14). An artificial neural network (ANN) was employed to build a predictive foaming model. The developed model showed a promising performance in predicting the foaming in the plant where the model also was validated by comparing the results with a decision tree model results. Overall, the findings of this report provide practical insights that may allow for the prediction of foaming occurrence within post-combustion CO₂ capture (amine-based) plants. The latter can lead to the implementation of preventive measures that can increase CO₂ capturing efficiency, and thus favoring the environmental sustainability.

Preface

This thesis is submitted to fulfill the graduation requirements of Master of Science at the University of South-Eastern Norway (USN). I have been always passionate by the topics that link data science and machine learning to real life issues. Climate change is one of the main challenges that humankind is facing now where its negative impacts have significantly affected our planet. This thesis focuses on improving methods that can decrease the emission of CO₂ into atmosphere and therefore minimizing the negative climate impacts. Concretely, a data-driven machine learning model is developed that can predict foaming in post-combustion CO₂ capture plants (amine-based). The latter can provide insights to improve the efficiency of CO₂ capture plants. The main target markets of this thesis are the research and industrial communities that aim to improve the CO₂ capture processes.

I would like to express my special gratitude to my supervisors, Prof. Leila Ben Saad and Prof. Ru Yan, for their excellent guidance and support during this process. I also wish to thank the external partner of this thesis, technology center mongstad (TCM), especially Mr. Rune Teigland, Mr. Blair McMaster and Mr. Koteswara Rao Putta for providing the data and also fruitful discussions.

Porsgrunn, 18.05.2022

Nasir Niayifar

Contents

1	Introduction	7
1.1	Background and objectives	7
1.2	Structure of the report.....	9
2	Foaming in CO2 capture and machine learning approaches.....	10
2.1	Foaming	10
2.1.1	<i>Foaming definition</i>	10
2.1.2	<i>Foaming structure</i>	11
2.1.3	<i>Forces acting on the foam</i>	11
2.1.4	<i>Effective mechanisms on foaming stability or rupture</i>	12
2.1.5	<i>Symptoms of foaming in the columns</i>	13
2.1.6	<i>Foam neutralization using chemical additives</i>	13
2.2	Literature review of machine learning methods applied to CO2 capture	14
2.2.1	<i>General concepts of artificial neural network (ANN)</i>	15
2.2.2	<i>Decision tree model principals</i>	18
3	Data preprocessing.....	19
3.1	Description of the provided data.....	19
3.1.1	<i>Foaming data extraction</i>	20
3.1.2	<i>Solvent density</i>	21
3.2	Cleaning offline plant data and data allocation to the strippers	21
3.3	Finalizing data related to each stripper	21
3.4	Finalizing the plant data	22
3.5	Foaming data labeling	23
3.6	Feature selection	25
3.7	Feeding data to machine learning models	27
4	Results and discussion	28
4.1	ANN model.....	28
4.1.1	<i>Performance of ANN model</i>	29
4.2	Performance of the decision tree model	31
4.3	Feature analysis based on labeling (foaming/ not-foaming).....	31
4.4	Discussion	34
5	Conclusion	36
5.1	Future studies	37
	Appendix A	42
	Appendix B	45
	Appendix C	53
	Appendix D	54

Nomenclature

Symbols	Explanation	Unit
Σ	Foaminess coefficient	[t]
V	Volume	[m ³]
\dot{V}	Volume flow rate	[$\frac{\text{m}^3}{\text{t}}$]
φ	Gas fraction	[-]
\dot{Q}	Reboiler duty	[W]
\dot{m}	Reboiler mass flow rate	[$\frac{\text{Kg}}{\text{t}}$]
T	Temperature	[°C]
c	Specific heat capacity	[$\frac{\text{J}}{\text{Kg} \cdot \text{°C}}$]

1 Introduction

Climate change as a consequence of greenhouse gases emission has been in the limelight in the recent years [1-3]. Among greenhouse gases, there always has been a huge concern regarding CO₂ emissions which in turn has accelerated global warming. Technologies regarding CO₂ capture at the source points (e.g., industrial emissions) have been applied efficiently in practice [4]. More specifically, the post-combustion CO₂ capture methods applied to power plants and chemical process factories have been in the center of attention in the last two decades as crucial high-techs in the CO₂ neutralization [5, 6]. Many methods have been implemented to capture post-combustion emitted CO₂ from large scale plant, while the most technologically mature and commercially viable one appears to be amine-based chemical absorption [7].

1.1 Background and objectives

The post-combustion CO₂ capture technology (amine-based) consists of two main steps: absorption and desorption (see Figure 1.1). Post-combustion flue gas that possesses an amount of CO₂ enters at the bottom of absorption column. The flue gas is in contact with a solvent, which is mainly amine-base solvent like monoethanolamine (MEA), methyldiethanolamine (MDEA) or diethanolamine (DEA) [8, 9], within the absorber in the presence of packing or tray. Since the CO₂ is transferred from flue gas to the solvent, the cleaned gas exits from the top of column into atmosphere with less amount of CO₂. The rich solvent (rich with the respect of having CO₂) contaminated with CO₂ exit from the bottom of column and enter the stripper [10]. In the stripper, the heat required for thermal stripping process is supplied by a reboiler. When stripping process is completed, the lean solvent is regenerated and recycled into the absorber, again. Furthermore, the CO₂ leaves the stripper as the top product and it is prepared (condensed and stored) for further usage [10].

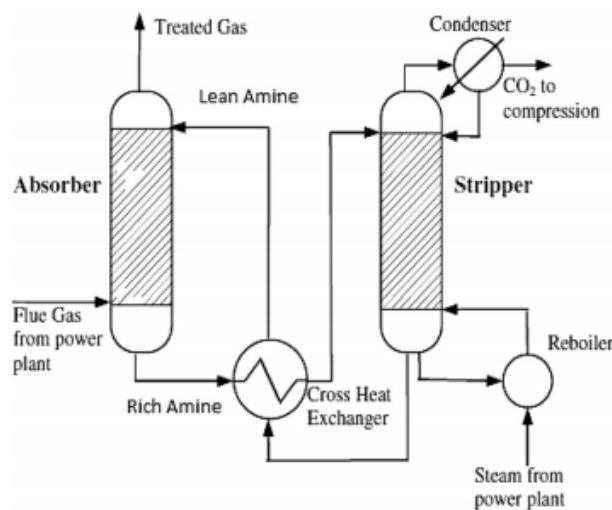


Figure 1.1: Process flow diagram of post-combustion CO₂ capture plant [10].

The post-combustion CO₂ capture plants experience some challenges such as solvent degradation [11], corrosion [7], cost of solvent [12] and foaming [13] that leads to negative effects on the process efficiency [14]. In this report, the problem of foaming in the post-combustion CO₂ capture plants (amine-based) is investigated.

A big challenge in the post-combustion CO₂ capture plants is foaming phenomena which happens mostly in the water based amine solvent and results in poor efficiency [15]. There have been some plant experiences [16-23] and research works [4, 24, 25] in the literature regarding foaming problem in the acid gas plants that use amine-based aqueous solutions. There are actions like solution reclamation, solvent filtration and introducing anti-foam to the column that can mitigate the foaminess in columns. However, the latter can overcome foaminess temporarily [17, 20, 21, 24].

Although there is hardly extracted specific quantitative relationship between physical process properties of plant and foaminess, Bikerman, J.J [26] introduced a foaminess coefficient Σ that is interpreted as average lifetime of a bubble (foam) before rupture and it has the dimension of time (seconds). Furthermore, Robin Thiele and his team [27] had an experimental study regarding foaminess coefficient. They used a method suggested by Bikerman, J.J [28] in which passed nitrogen through sintered frit is injected in the liquid and the height of created foam is measured. Afterwards, the height of foam and the area of container were used to calculate the volume V of created foam and from equation 1.1 foaminess coefficient was calculated as:

$$\Sigma = \frac{V}{\dot{V}}, \quad 1.1$$

where \dot{V} is volume flow rate of nitrogen. However, this method works only for tray columns and has some difficulties in measuring needed parameters in real plants.

Generally, there are two major types of columns: tray and packed columns where the tray columns are more susceptible to foaming due to the nature and shape of them [29]. Two main types of packed columns are structured and unstructured packed. When the liquid rate is low (vapor rate is high), structured packing can face foaming more effectively than unstructured packing. This is due to the fact that when the liquid rate within the packing is high, the liquid can bridge over porosities and the vapor flowing upward causes foaming. On the other hand, when the liquid rate is high unstructured packing is less prone to foaming since its interior shape is more open laterally [29].

In another systematic study [30], foaminess of packed column (structured and unstructured) with different solvent such as Methyl diethanolamine (MDEA), water and butanol was evaluated and it was revealed that the foaminess occurs in the different places in the absorber and desorber (packed columns) such as the liquid entrance (shows the importance of distributor type), in the base, within the reboiler and inside the packing. Experiment showed that the more void fraction within packing prones the packed column to more severe foaminess.

It is worth mentioning that about half of foaming occurrence within columns in the industry is reported to be in the acid gas treating units [31]. Although there have been several studies in the literature [10, 32-37] regarding implementation of data-driven models in order to simulate relevant properties of CO₂ capture plants using amine-based solvent (interested properties w.r.t output of model), there are a few scientific works to represent data-driven model that can be useful in the simulation and prediction of foaming occurrence in the post-combustion CO₂ capture plants using amine-based solvent. For instance, Nwaoha, et al. [36] implemented data-driven methods to model the process. The created model set the CO₂ capture, foaming tendency and amine vaporization as outputs. In this study, the flow rate of absorbed pentane (liquid hydrocarbon) by the rich amine-based solvent is chosen as the foaminess tendency criteria (more absorbed pentane, more foaming tendency). Eventually, the sensitivity analysis revealed that the most effective physical properties on the foaming tendency was lean amine flow rate that was reported previously in a study by Bullin and Brown [38].

As discussed earlier, since there is no physical based model that is able to simulate stable foaminess (stable foams (i.e., the foams with lifetime more than a few seconds) can affect the performance of plant) [30], this report studies and proposes a data-driven model by applying machine learning, for the first time, to real data (CO₂ capture utilizing amine-based solvent in the fractionation columns) provided by Technology Center Mongstad (TCM). More information about TCM is provided in the Appendix D. The ultimate goal of this report is to create a model that can predict the foaminess occurrence based on the given process properties. The latter can improve the efficiency of post-combustion CO₂ capture plants (amine-based) by adopting preventive actions before foaming occurs. For this sake, the data will be preprocessed, and then fed into an artificial neural network (ANN) and also a decision tree model. The extracted results of both models will then be compared to select the proper model for foaming prediction in the post-combustion CO₂ capture plant (amine-based). Finally, there'll be a discussion about the parameters which mostly affect the occurrence of foaming in such plants.

1.2 Structure of the report

This report is structured as follows: The chapter 2 explains the foaming problem and how it impacts post-combustion CO₂ capture processes and also appropriate machine learning approaches to model foaming. Chapter 3 describes the data used in this study as well as data preprocessing. Next, results and discussions are presented in Chapter 4. Finally, a conclusion is made in Chapter 5.

2 Foaming in CO2 capture and machine learning approaches

The definition of the foaming phenomenon and how its negative aspects can affect the columns and therefore CO2 capture are discussed at first. The second section discusses the concepts of artificial neural network (ANN), which is the main machine learning method used in this report to simulate the problem of foaming in the post-combustion CO2 capture plant (amine-based solvent). In addition, another machine learning method known as decision tree is described at the end of the second part.

2.1 Foaming

Post-combustion CO2 capture processes suffer from the problem of foaming in the absorber and stripper that decreases absorption efficiency, mass transfer area between solvent and flue gas (i.e., containing CO2) [39] and also has negative effects in the stripper column.

2.1.1 Foaming definition

There are several definitions in the literature that describe how foams can be created. One definition of foaming creation describes its process as rising the gas phase within the liquid phase up without rupture [40]. Another definition describes foam as vapor or gas which is encapsulated within a thin wall of liquid [29]. The systems with the probability of foaming are facing two conversely sub-processes; the tendency to create the bubbles and tendency to rupture them. Once the foam creation tendency win the competition and foam is stabilized, the system efficiency will drop down [29]. In [27], foam is divided in three layers based on the gas fraction φ (see Figure 2.1). If the foams in the Kugelschaum layer coalesce and create bigger bubbles, Kugelschaum will be converted to Polyederfoam. On the other hand, if the gravity forces which tries to make the wall of bubble thinner (drainage effect) win, the bubble rupture occurs and Kugelschaum will be converted to Gasdispersion [27].

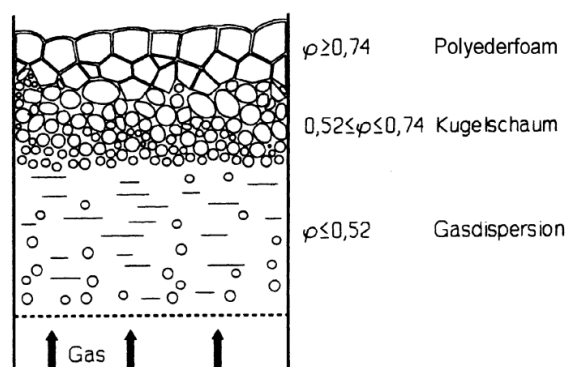


Figure 2.1: Foam types based on the gas fraction [41].

The lifetime of a foam varies from seconds to years, but generally foams have limited lifetime. A quick rupture occurs after 5 second, while an approximately stabilized foam can tolerate up to three minutes [42].

2.1.2 Foaming structure

When a bubble is located within the Polyederfoam layer, three liquid films (lamellae) intercept with angle of 120° in a plateau border that is illustrated in the Figure 2.2 [43].

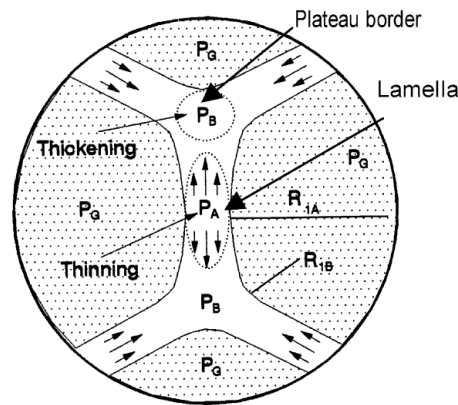


Figure 2.2: Foams interception structure in the polyederfoam [43].

It has been seen that in the most cases, three items are needed for the foam generation; gas, liquid and different types of foam agents such as impurities, macromolecules and surfactants. The latter, naturally has low surface tension and therefore decreases the required energy to expand gas inside the liquid capsule [43].

2.1.3 Forces acting on the foam

Across their lifetime, foams are subjected to forces acting on their walls, and this will result in rupturing or stabilizing [40]. These forces are listed in the Table 2.1 and are as follow:

- Gravity (drainage): Gravity force acts on the liquid film and force it to move from the wall into the base where the wall continues to become thinner until rupture happens [40].
- Interfacial tension: Surface tension creates a pressure gradient across the bubble wall with higher pressure inside the foam. Different radiuses inside the bubble result in different pressures in the lamellae and plateau borders leading to flowing liquid toward low-pressure points (plateau borders), thinning foam wall and eventually rupture (see Figure 2.2) [43].
- Capillary: Sucks liquid from the base into the expanded surface of the bubble and help in foam stabilization [40].
- Viscosity: Increasing viscosity opposes liquid drainage and result in the foam stabilization [29].

Table 2.1: Forces acting on the foam.

Forces result in rupture	Forces result in stabilizing
Gravity (drainage)	Capillary
Interfacial tension	Viscosity

2.1.4 Effective mechanisms on foaming stability or rupture

There are four mechanisms that can affect stabilization of a foam which are listed below and will be discussed shortly in the following.

- Marangoni effect
- Mass-transfer induced marangoni effect
- Ross-type foaming
- Gelatinous surface layer

Marangoni effect: When there is a surface tension gradient in the liquid phase, the liquid is sucked from the points with lower surface tension into higher surface tension. For instance, whenever an additive that has lower surface tension (surfactant) is introduced to the liquid phase, the concentration of surfactant in the expanded area of the foam decreases and as a result, the surface tension of the expanded part will be higher than the base of foam. At this stage, marangoni effect causes that the liquid flows from bulk liquid base toward expanded wall and after a while the foam will be thicker and stabilized [29].

Mass-transfer induced marangoni effect: In the distillation columns, when there is no Marangoni effect (lack of surface-active component or impurities), a foam can be stabilized as a result of Mass-transfer induced marangoni effect. If the more volatile component of foam has higher surface tension, and the fact that the more volatile component of the foam locates in the expanded wall of the foam, the liquid is sucked into expanded wall of foam due to surface tension gradient and foam stabilizes eventually [40].

Ross-type foaming: Sometimes a weak solution of liquid may lead into creation of second liquid phase that is susceptible to foaming [44]. This counts for equilibrium effects and may be mitigated by increasing the temperature [45].

Gelatinous surface layer: This phenomenon occurs when a chemical or intermolecular interaction happens in the expanded liquid part of foam (due to impurities presence). Gelatinous surface layer results in stationary liquid film which in turn mitigates drainage and capillary effect (motionless layer) [40].

2.1.4.1 Role of particulates in foaming

One interesting fact about particulates is that they are not able to convert a non-foaming system to foaming system. Instead, once the foam is created, depend on particulates shape, composition and size, they can help to stabilize foaming [29]. One negative aspect that makes

it difficult to deal with foaming stabilization due to particulate in the fractionation columns is the size of particulate. While the smaller particulates are more effective in foam stabilization, it is more difficult to remove them from the liquid [29].

While the foams created solely by marangoni effect (known as physical foam) are susceptible to drainage and are not considered as stable foam [40], Mass-transfer induced Marangoni effect causes more stable foams and can lead to severe foaming in the columns [46]. The formation of very stable foams includes two steps: At first, the foam is created by marangoni effect or mass-transfer induced Marangoni effect, and secondly the foam will be stabilized by the help of gelatinous surface layer [40].

2.1.5 Symptoms of foaming in the columns

Many foaming symptoms have been reported in the literature that are described in this section. However, the foaming can also occur in the process with one, some or sometimes without any symptoms (i.e., known up to present) [40], which are counted below:

1. Premature flooding and massive entrainment accompanied with pressure drop. The latter can be interpreted as below [40]:
 - Intrusive increasing in differential pressure [30].
 - A differential pressure exceeding of 40% to 50% of tray spacing or exceeding pressure drop 1 inch of liquid per foot of packed bed.
 - Erratically variation in the differential pressure.
2. Steady-state condition dominates the system when flooding starts [45].
3. Abnormal temperature profiles. As an example, in the amine-base absorbers, difference between inlet and outlet gas temperature decreases significantly along with dropping in temperature difference between rich and lean solution [40].
4. High probability of losing solvent in the columns [14].
5. Decrease in the process efficiency [15].
6. Efficiency will increase by adding anti-foam [45, 47-49].
7. The solvent is regenerated incompletely in the columns [14].

The symptoms mentioned above can be measured in post-combustion CO₂ capture columns (amine-based) to see if there is a relationship between these physical column characteristics and foaming occurrence.

2.1.6 Foam neutralization using chemical additives

The simplest way to mitigate the foaming effect is to add foam inhibitors that are usually liquid. Their ability to neutralize the foaming arises from the fact that they intend to spread naturally over the expanded area of the foam and flow the liquid from the foam expanded part into the base and eventually thinning and rupture of the foam [40].

Since prediction of foaming in Co₂ capture is considered as a difficult task, the next section investigates the potential use of machine learning methods to solve this problem.

2.2 Literature review of machine learning methods applied to CO₂ capture

There are several criteria that we can use to categorize data-driven models. Frequency domain models with dead time, data mining algorithms, fuzzy logic, stochastic models, statistical models, state-space models, case-based reasoning models, geometric models, and instantaneous models are the main categories [50]. Furthermore, each of the categories can be subdivided into one or more specific techniques, as shown in Figure 2.3.

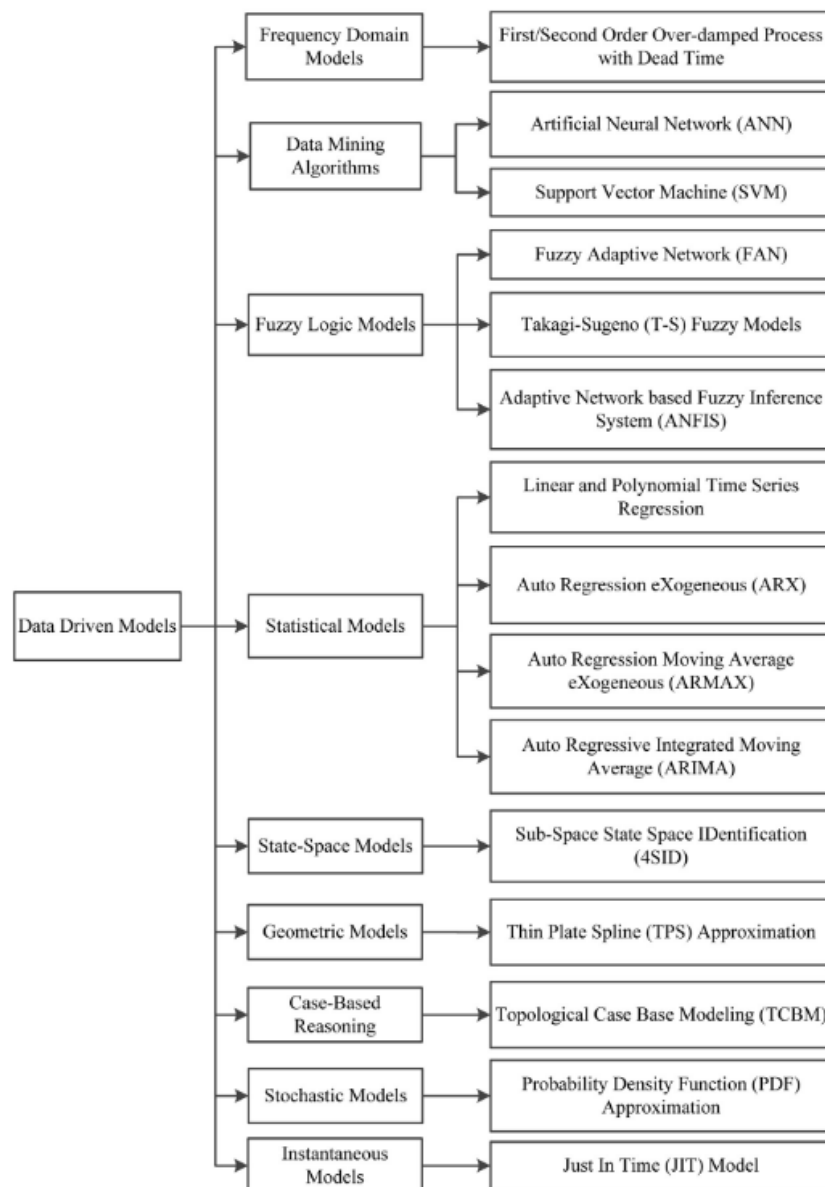


Figure 2.3: Data-driven methods [50].

Among the whole data-driven models, this report focuses on the machine learning techniques. As seen from the Figure 2.4, the machine learning procedure can be interpreted in 7 steps.

Except for the first step, that is handled by external partner of this project (TCM), the other steps will be implemented and discussed in this report.

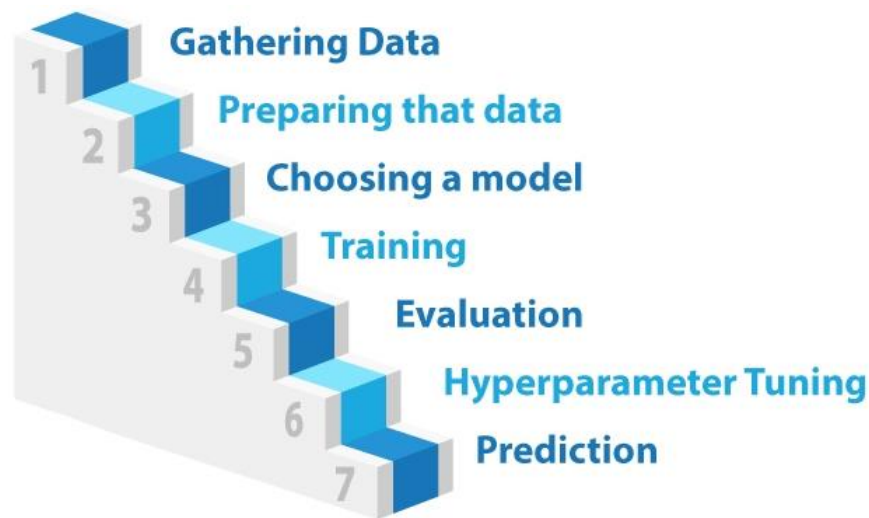


Figure 2.4: Machine learning procedure [51].

All types of machine learning methods can take discrete or continuous values as input, but regarding the type of output one can divide them into two main groups: regression and classification. The output part of examples, fed into classification patterns for training only include discrete values. As a result, the model in these patterns is capable of predicting discrete values (e.g., labeling or categorizing). While in the regression patterns, the output part of examples, fed into the model during training procedure is continuous and the model can predict continuous values [52]. Since artificial neural networks (ANN) have a high potential for predicting classification and regression patterns [53], this machine learning method has been selected, a model is created, trained and tested using preprocessed data. In addition, a decision tree model is implemented on the preprocessed data, and the results are compared to the ANN results.

2.2.1 General concepts of artificial neural network (ANN)

The primary structure of a neural network is similar to that of the human brain which is illustrated in the Figure 2.5. The network takes the input portion of preprocessed data, processes it, computes the output, and finally calculates the difference between measured output and predicted output using a well-defined cost function. Several tunable parameters such as the number of hidden layers, the number of neurons in each hidden layer, the kind of activation function, the quality and quantity of input features, optimizer type and learning rate can influence how data is processed within the ANN as well as the quality of prediction.

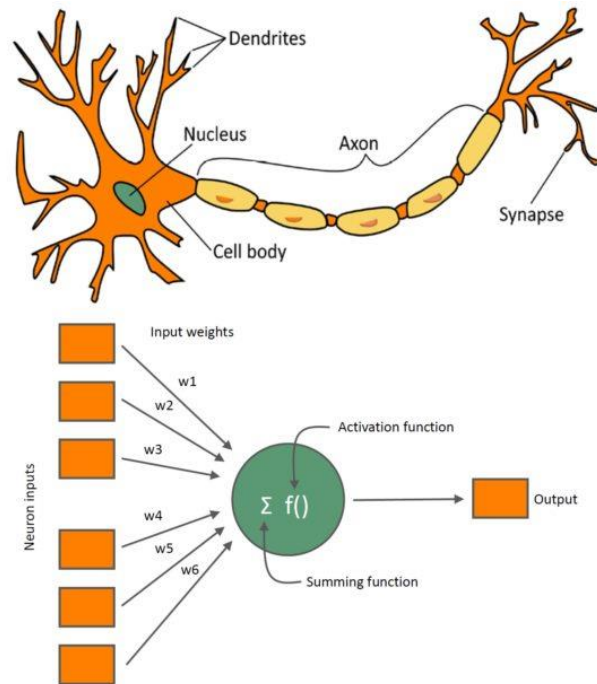


Figure 2.5: Similarities between ANN and biological neuron [54].

The ANN architecture is shaped by the number of hidden layers, the number of neurons in each hidden layer, the number of applied features and the number of predicted outputs. The number of neurons in the final layer or output layer is determined by the number of items to be predicted, and if this is the case in the classification patterns, the output layer can be defined using the ‘softmax’ function. It is worth noting that ‘softmax’ is a function that converts a vector of values to a vector of probabilities.

2.2.1.1 Neuron structure and performance

The preliminary part of calculations within a neuron is a matrix multiplication plus a bias term. Matrix multiplication is performed between the output of previous layer neurons and the weight matrix interfaced between the previous layer neurons and the neuron in the current layer. Afterward, an activation function such as ‘sigmoid’, ‘tanh’, or ‘ReLU’ (see Figure 2.6) can be used to obtain the neuron's output, which is then used as input to calculate the next layer neurons (see Figure 2.7).

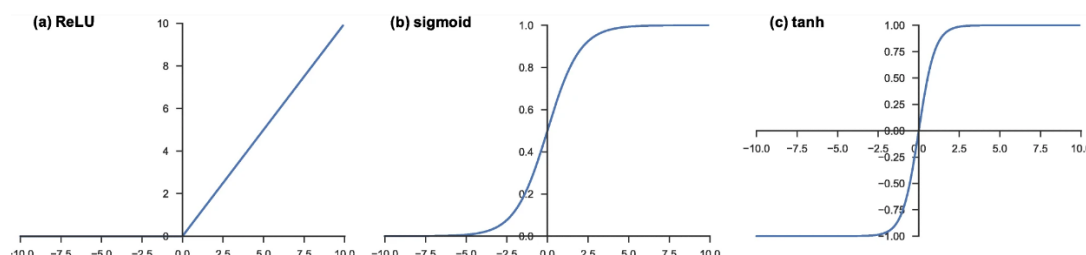


Figure 2.6: Typical activation functions used in neurons [55].

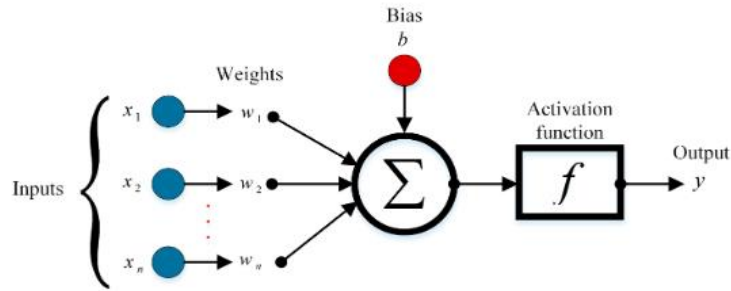


Figure 2.7: Performance of single neuron [56].

2.2.1.2 Types of feeding the data into the ANN model (Batch vs. mini-batch)

One iteration is needed to optimize the model's weights one time. The typical method consists of passing the whole training set (batch) through the model in one iteration (epoch). According to the novel technique, the training set is broken into smaller sets (mini-batch), and weights will be optimized when a mini-batch passes through ANN (one iteration). As a result, when all mini-batches (the entire training set) are run through ANN, the weights will be optimized many times rather than once and this helps to speed up the optimization process.

2.2.1.3 Optimization process

When the model predicts outputs in a single iteration (forward propagation), the predicted outputs are compared to the real output values, and the cost function, is calculated. It is worth noting that the ultimate objective of the optimization procedure is to minimize the cost function. To reach this goal, the weights and the bias terms are updated on each iteration using the optimization algorithm and the learning rate (backward propagation). The whole training set is divided into mini-batch groups and the weights and bias terms are updated after passing a mini-batch through the model. Number of epochs is another parameter that must be fine-tuned in a way that be kept as low as possible while still providing the best accuracy. Furthermore, the number of neurons should be tuned to identify the minimum number of neurons necessary to achieve acceptable accuracy during training and test procedure. Figure 2.8 represents a schematic view of the optimization process.

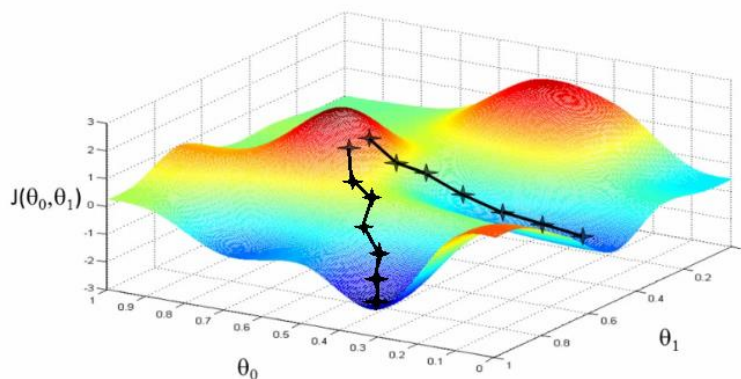


Figure 2.8: Optimization process ($J(\theta_0, \theta_1)$: Cost function, θ_0 and θ_1 : Weights) [57].

2.2.1.4 Underfitting and overfitting

Underfitting occurs when the model's accuracy on the training and test sets is poor. There are some ways to avoid this undesirable outcome, such as selecting more features [58]. On the other hand, if the model fits the training set with good accuracy but fails to generalize to new samples (i.e., test set), it is considered overfitting and can be alleviated by using regularization, drop-out or early stopping [58, 59]. Figure 2.9 depicts underfitting and overfitting in a simple and clear manner.

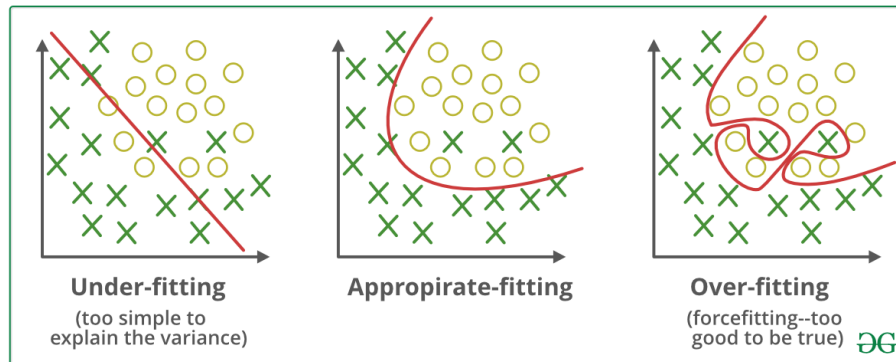


Figure 2.9: Underfitting, appropriate fitting and overfitting [60].

2.2.2 Decision tree model principals

Decision tree algorithm is constituted of different paths. Each path's logic is that it begins with a root and then continues with some decision nodes. A Boolean within the decision node divides the path into new paths, and each path ends if it reaches a leaf node [61].

Furthermore, the longest path between the decision node and the leaf node is known as max depth. Figure 2.10 depicts a typical decision tree schematic algorithm.

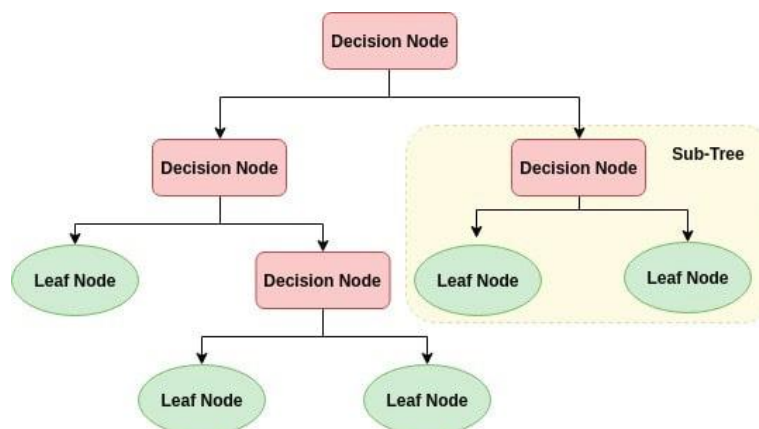


Figure 2.10: Typical decision tree algorithm [62].

3 Data preprocessing

When it comes to using data-driven models to extract valuable information from a system, one of the most important steps is data preprocessing. The reason is that the physical relationships and equations are not known at all or the fact that finding physical relationships between desired inputs and outputs are very rigorous work. Since data-driven models trust on the given data to extract relationship between inputs and outputs, the given data must be accurate and clean as much as possible.

3.1 Description of the provided data

A series of data in different excel files were given by TCM. The 'USN - Foaming Data Extract' excel file includes three sheets consisting of 'Flow Pressure & Temp', 'CHP Stripper Profiles' and 'RFCC Stripper Profiles'. Furthermore, the lean and rich density of solvent are given in the 'Density_include' excel file. The whole given data include hourly time-steps from '01.09.2019 00:00' to '30.11.2020 22:00' (10968 samples).

Figure 3.1 shows TCM post-combustion CO₂ capture (amine-based) plant. The overall objective of the plant is to capture the CO₂ of the post-combustion flue gas before the gas is released to the atmosphere in order to minimize the carbon emissions. The core elements of this plant are: direct contact coolers (DCC), absorber, reboilers and two strippers that are based heat and power plant (CHP) and the residual fluid catalytic cracker (RFCC). The solvent used in this plant is CESAR1, a non-proprietary solvent developed as part of the EU CESAR project and originally intended for use in the carbon capturing process. This mixture (CESAR1) contains AMP and piperazine (both are amines) as well as water, and the goal of creating this solvent is to replace conventional MEA [63]. The process begins when post-combustion flue gas is introduced into the CHP and RFCC direct contact coolers, respectively. Once the flue gas is cooled off, it enters the absorber and flows upward. On the other hand, the lean solvent streams enter the absorber from the points that is higher than flue gas entrance (orange lines at the left of absorber) and flows downward. When the flue gas comes into contact with the lean solvent flow in the presence of structured packing (yellow rectangles within the absorber), the CO₂ in the flue gas transfers to the lean solvent flow and converts it to the rich solvent flow (rich in terms of CO₂ possession). Depleted flue gas (Cleaned gas) exits from the top of absorber and the rich solvent flow exits from the bottom of absorber into strippers. When the rich solvent flow is entered in the stripper from the point above the structured packing (light brown rectangles within the strippers), it is warmed up by the reboiler stream and solvent decomposition occurs. The released CO₂ from the rich solvent exits the stripper from the top and is stored for future use. The solvent, which now is known as lean solvent, again, exits the stripper from the bottom and proceeds to the next cycle into absorber. It is worth noting that the strippers do not operate simultaneously, and therefore only one of them operates on a single time-step of the provided data. In general, all of the features (columns in the excel sheets) in the provided data (all of the files provided by TCM) consist of the entire time-steps, and due to the fact that both strippers cannot be online at the same time, the entire time-steps must be tagged by one of the CHP or RFCC strippers.

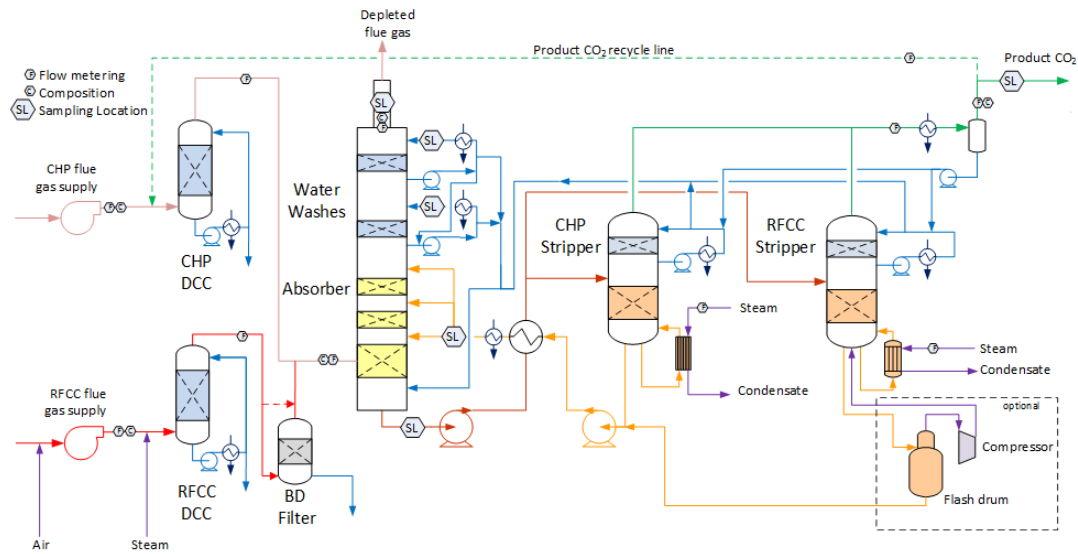


Figure 3.1: Schematic drawing of TCM post-combustion CO₂ capture plant [64].

3.1.1 Foaming data extraction

‘USN - Foaming Data Extract’ file includes three sheets which are described in the following subsections.

3.1.1.1 Flow, Pressure and Temperature

The ‘Flow, Pressure and Temperature’ table includes detailed information regarding solvent flow rates, temperatures and pressures. Furthermore, the flow rate and temperature of reboilers and also CO₂ production rate in the strippers can be found in this excel sheet (see Figure 3.2). The important aspect of this table that must be considered is that the data regarding the temperature, pressure and flow rate of solvent are not tagged by strippers and must be treated before any further usage.

	Common Flows		CHP Stripper Flows		RFCC Stripper Flows		CHP Rblr Temp	RFCC rblr temp	Rich Solvent Temp	Lean Solvent Temp	Rich Solvent Pressure	Lean Solvent Pressure
	Lean Solvent Flow	Rich Solvent Flow	Rblr stream	CO ₂ Prod	Rblr stream	CO ₂ Prod						
	kg/h	kg/h	kg/h	kg/h	kg/h	kg/h	oC	oC	oC	oC	barg	barg
	8611-FT-2112	8611-FIC-2004	8655-FI-2386B	8615-FT-2391	8655-FI-2158B	8615-FT-2167	8611-TT-2385	8611-TT-2157	8611-TIC-2111	8611-TT-2256	8611-PT-2043	8611-PT-2257
01.09.2019 00:00	0	0	0	835.601861	0	0	17.93689588	17.96499327	17.87964971	17.89524807	0	0.112344197
01.09.2019 01:00	0	0	0	847.659196	0	0	17.04428624	17.55592831	16.91000518	17.32245867	0	0.10978869
01.09.2019 02:00	0	0	0	859.032703	0	0	15.65329817	16.94560471	15.6119669	16.64576676	0	0.108837071
01.09.2019 03:00	0	0	0	860.955027	0	0	14.54348874	16.46488714	14.41720959	16.07262684	0	0.109330007
01.09.2019 04:00	0	0	0	864.293677	0	0	13.78945178	16.11330119	13.56811277	15.61487451	0	0.109056475
01.09.2019 05:00	0	0	0	871.497724	0	0	13.01826161	15.7578749	12.79695093	15.1546853	0	0.109659498
01.09.2019 06:00	0	0	0	871.748297	0	0	12.42691084	15.44470378	12.17752018	14.77792824	0	0.110926698
01.09.2019 07:00	0	0	0	870.267724	0	0	12.14546785	15.20836774	11.81192137	14.42024461	0	0.111442944
01.09.2019 08:00	0	0	0	868.162613	0	0	12.18232655	14.99979093	11.68866785	14.13242457	0	0.112000813
01.09.2019 09:00	0	0	0	864.738557	0	0	12.46301049	14.89773865	11.81226489	13.98772168	0	0.113209257
01.09.2019 10:00	0	0	0	849.489942	0	0	13.03944967	14.9167825	12.2809839	14.09970264	0	0.1136149
01.09.2019 11:00	0	0	0	858.224309	0	0	13.16897416	14.90712097	12.4090645	14.24757344	0	0.113617012
01.09.2019 12:00	0	0	0	840.64384	0	0	13.24905997	15.01557183	12.66210739	14.69968722	0	0.114073739
01.09.2019 13:00	0	0	0	845.865087	0	0	13.559171	15.23896439	12.96881058	14.93722346	0	0.114749447
01.09.2019 14:00	0	0	0	848.548975	0	0	13.82887695	15.491424	13.23294587	15.14245402	0	0.113898582
01.09.2019 15:00	0	0	0	846.450602	0	0	14.02539438	15.67290922	13.3646035	15.15044911	0	0.113840386

Figure 3.2: Sample of provided data by TCM (‘USN - Foaming Data Extract’ file/ ‘Flow, Pressure and Temp’).

3.1.1.2 CHP Stripper Profiles

Temperature profiles within the CHP stripper are given in this datasheet; temperatures at the bottom of the stripper, at various points along the height and radius of the packing, above the packing and at the stripper's outlet. Furthermore, the measured pressure drops in the demister, water wash and packing are presented in this datasheet.

3.1.1.3 RFCC Stripper Profiles

The same temperature profiles (e.g., temperatures at the bottom of stripper) and pressure drops profiles as CHP Stripper described above (section: 3.1.1.2) within the RFCC stripper are provided in the data.

3.1.2 Solvent density

Lean and rich solvent densities are given in the 'Density_include' excel file. This file also is not tagged by the strippers and the time-steps of given data must be tagged before using in the model.

3.2 Cleaning offline plant data and data allocation to the strippers

The first step in the cleaning of the given data is to determine in which time-steps the plant was offline and then removing all data associated with those time-steps. Based on the external partner, TCM, plant's knowledge, the time-steps where both strippers are offline can be detected, as described below:

- Both rich and lean solvent flow rates are less than 10000 [kg/h]
- Both strippers (i.e., CHP and RFCC) produce CO₂ less than 1000 [kg/h]
- The temperature in the reboilers of CHP and RFCC strippers are lower than 33 (°C)

Furthermore, if the reboiler temperature of one stripper is less than 33 (°C), that stripper is offline in the corresponding time-step, and the data must be deleted. Two separate Python scripts were encoded to clean the data for strippers that are available in the Appendix B.

3.3 Finalizing data related to each stripper

For each stripper, the data is first read from the 'USN - Foaming Data Extract' and 'Density_include' excel files. The files are then stacked and the columns '8611-PDT-2442' and '8611-PDT-2441' which represent the water wash pressure drop and demister pressure drop, respectively, are removed (there are very few usable data in these columns). Following that, the columns in which the number of 'NaN' values exceeds 40% of the total in that column are deleted and afterwards, the time-steps that are tagged plant offline are removed from each stripper data frame. Finally, after removing the remaining time-steps containing 'NaN' values or zeros, the total usable data for the strippers is obtained. It is worth noting that in a time-

step, if any of the columns contains a 'NaN' or a zero, the entire data belong to that time-step (row) is removed (TCM plant knowledge).

3.4 Finalizing the plant data

The final step in the data preprocessing in this report is to stack the data from both strippers and convert it to an excel file for later use in the machine learning models. Usable number of time-steps for each stripper and total number are shown in Table 3.1.

Table 3.1: Usable data (Usable time-steps).

Stripper	# Usable data (time-steps)
CHP	1106
RFCC	3049
Total	4155

Besides, the total number of features after data preprocessing is 35, as shown in Figure 3.3.

Lean solvent flow rate	Rich solvent pressure	8611-TT-2401C	8611-TT-2403B	8611-TT-2446B
Rich solvent flow rate	Lean solvent pressure	8611-TT-2401D	8611-TT-2403C	8611-TT-2446D
Reboiler stream	8611-TIC-2379	8611-TT-2402A	8611-TT-2404C	8611-TT-2382
CO ₂ production	8611-TT-2399C	8611-TT-2402B	8611-TT-2405A	8615-TT-2390
Reboiler stream temperature	8611-TT-2399D	8611-TT-2402C	8611-TT-2405B	8611-PDT-2383
Rich solvent temperature	8611-TT-2401A	8611-TT-2402D	8611-TT-2405C	Lean amine density
Lean solvent temperature	8611-TT-2401B	8611-TT-2403A	8611-TT-2446A	Rich amine density

Figure 3.3: Total features after data preprocessing (complete explanation regarding feature's name is available in Appendix C).

3.5 Foaming data labeling

All the remaining time-steps must be tagged with ‘foaming’ / ‘not foaming’ classification so that this information can be used as the output of the model (y) for the training and test procedures. The procedure begins with analyzing the figure provided by TCM, which is presented in Figure 3.4. For the period from ‘02.10.2019 10:00’ to ‘22.10.2019 23:00’ that ‘foaming’ / ‘not foaming’ classification is provided by TCM, the data regarding CO2 production rate, pack pressure drop, lean and rich solvent density and temperatures in different positions within the CHP stripper were plotted and screened.

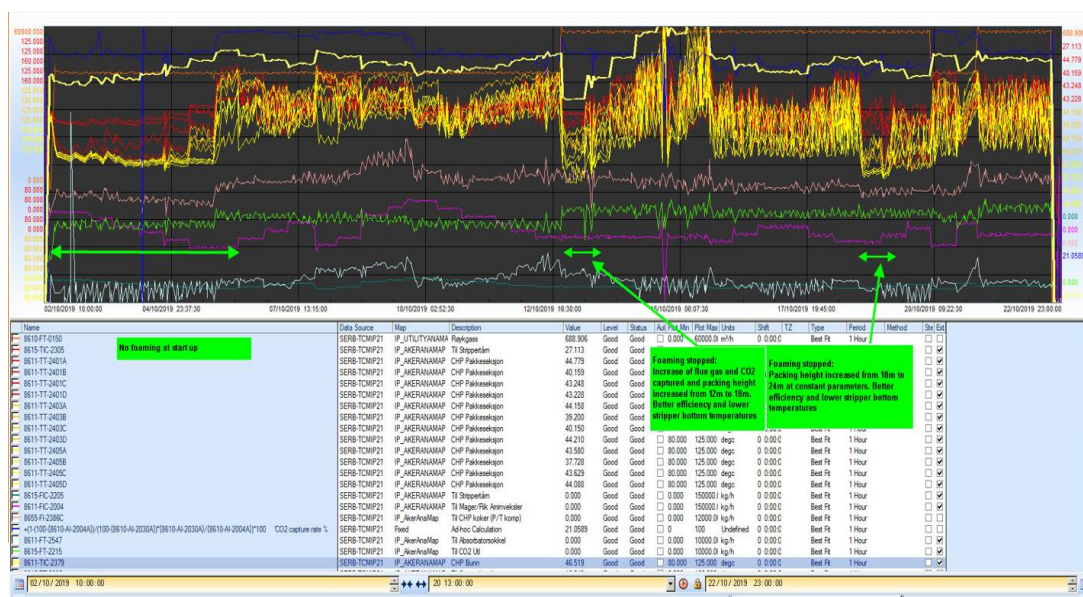


Figure 3.4 : Labeled ‘foaming’ / ‘not foaming’ period for CHP stripper by TCM.

After analyzing the mentioned data features over the period of Figure 3.4, it was discovered that foaming occurs in a time-step when all the following criteria are met simultaneously.

- Temperature at the bottom of packing is upper than 109 [°C].
- Temperature at the middle of packing is upper than 109 [°C].
- Lean solvent density is less than 1039 $\frac{kg}{m^3}$.
- Pack pressure drop is higher than 1 [mbarg] in the CHP and 0.1 [mbarg] in RFCC.

It is vital to notice that these special characteristics of the plant at the time of foaming are included in the foaming symptoms mentioned in the foaming chapter (i.e., section 2.1.5). For the sake of better understanding of the concept, Figure 3.5 (from a to c) depicts the mentioned temperature, lean solvent density profiles and pack pressure drop. Regarding the last criteria (i.e., pack pressure drop), the CHP pack pressure drop (see Figure 3.5d) higher than 1 [mbarg] represents the foaming with sufficient accuracy. However, the mentioned criterion for CHP seems to be too high for RFCC pack pressure drop (see Figure 3.6). As a result, a scaling between the maximum value and the foaming criterion for pack pressure drop was performed (for CHP, the maximum pack pressure drop is 10 [mbarg]). Since the maximum

pack pressure drop for RFCC is 1 [mbarg], the pack pressure drop criterion was found to be 0.1 [mbarg] by scaling.

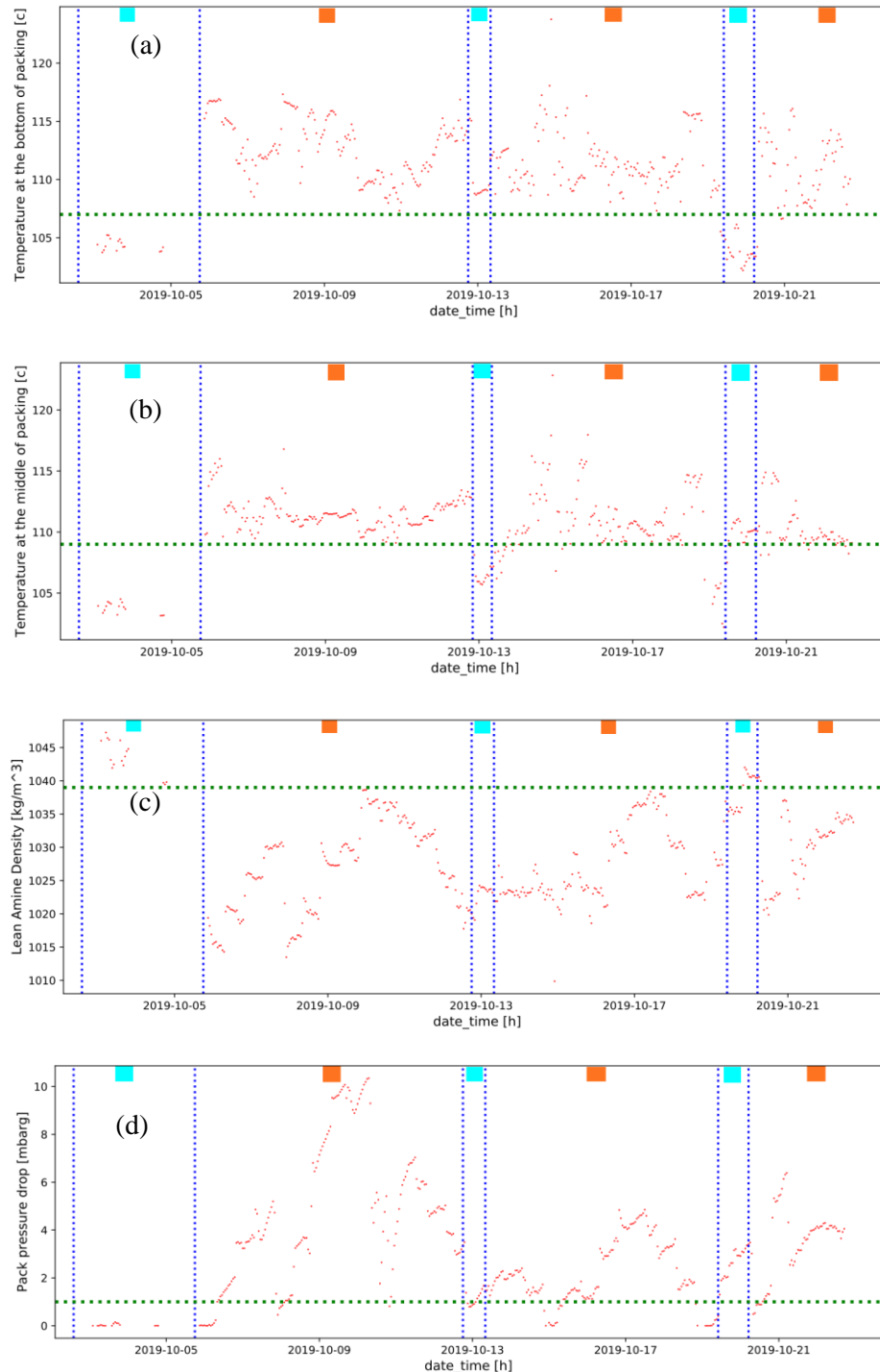


Figure 3.5 : Temperature profile at the packing's bottom (a) and packing's middle (b) as well as the lean solvent density profile (c) and pack pressure drop profile (d) of CHP stripper (Turquoise square: not foaming, red square: foaming).

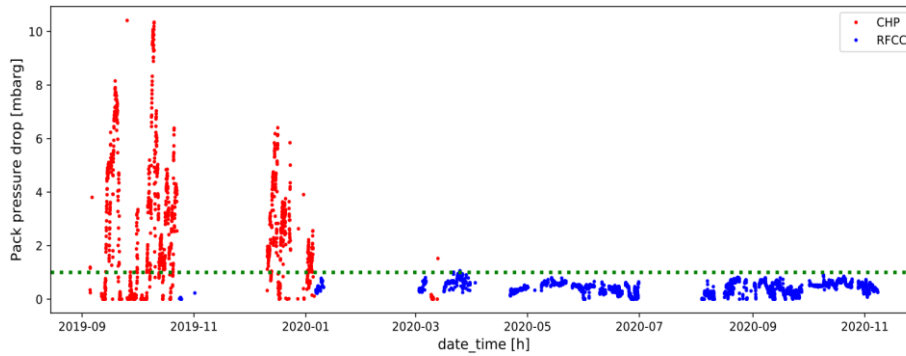


Figure 3.6: Pack pressure drop of both strippers (i.e., CHP and RFCC).

Once all of the foaming criteria are applied to the entire time-steps and both strippers, there are 2450 time-steps that are labeled as foaming and the results are shown in Figure 3.7.

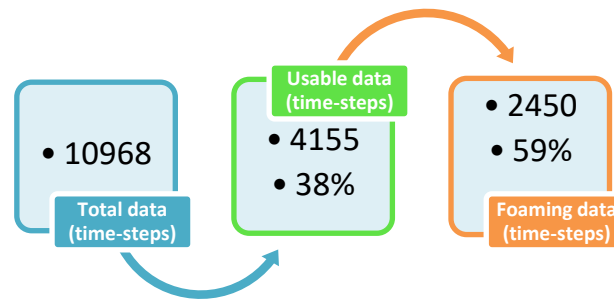


Figure 3.7: Data cleaning and labeling results.

3.6 Feature selection

As technology advances, it becomes easier to collect large amounts of data with a high dimensionality. On the other hand, analyzing and extracting useful information from big data with many features faces some challenges, such as high computational costs or a higher risk of overfitting. Feature selection can be applied to data in machine learning approaches to select the most optimal features out of all of them. The goal is to simplify the model, make it more understandable and improve its performance [65]. As discussed earlier, the total number of features after data preprocessing is 35 and it seems reasonable to implement a correlation-based feature selection to see if there are any similar statistical behaviors between features. The result of correlation-based feature selection is illustrated in the Figure 3.8. The most noteworthy outcome of the feature selection is that the number of temperature features has decreased drastically from 23 to 8 (not surprisingly). Furthermore, CO₂ production was deleted from feature's list since it exhibited more than 0.9 similar behavior to the reboiler stream, which seems to be reasonable. For instance, once foaming occurs within the stripper and the CO₂ production rate tends to decrease, the reboiler stream rate increases to compensate for the foaming impact.

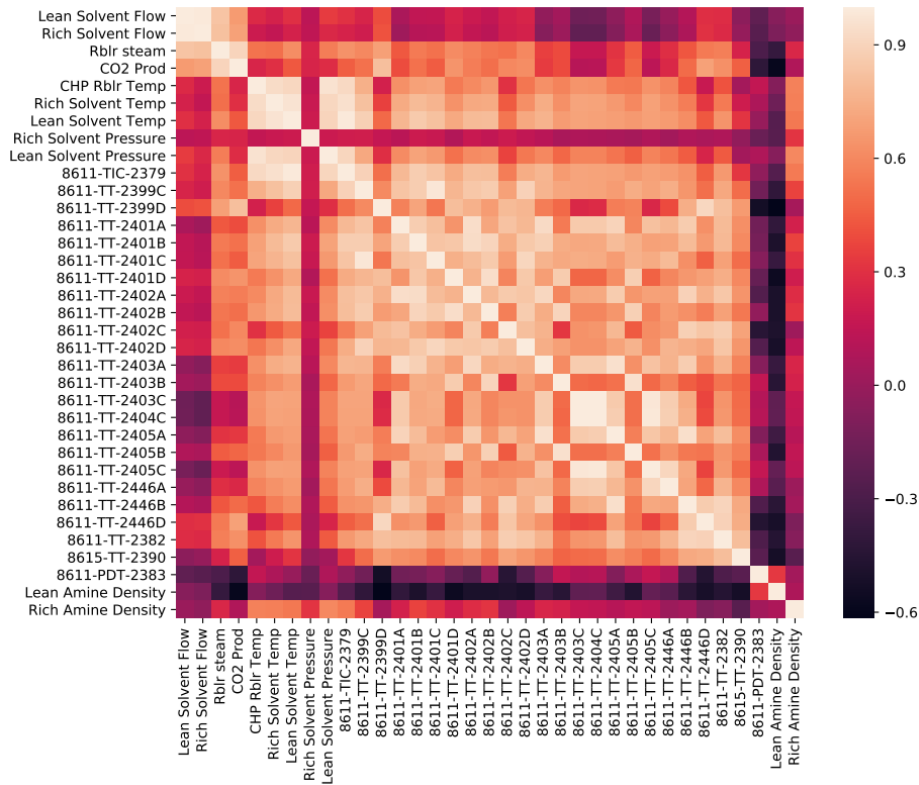


Figure 3.8: Correlations among features (color bar denotes the statistical similarity between features).

The final result is given in Figure 3.9 after applying the outcome of correlation-based feature selection to the 35 existing features.



Figure 3.9: Deleted and kept features after feature selection (Green: retained & Red: deleted).

3.7 Feeding data to machine learning models

The data fed into the machine learning models after feature selection consists of 4155 rows (i.e., time-steps) and 15 columns (i.e., features). The first 14 columns are considered as input (x), whereas the last column (foaming:1 and not-foaming:0) is considered as output (y). Next, the data is shuffled and after that these input and output are assigned to x and y variables, respectively. Afterwards, 80 percent of the data is allocated to the train set and the remaining 20 percent is maintained to test the model. At the next stage, input (training and test) is normalized in order to prevent gradient exploding and also help to speed up the optimization process (for ANN model). Once all of the preceding stages have been completed, the data will be fed into the designed models.

4 Results and discussion

The results of applying machine learning methods (ANN and decision tree) to predict the foaming occurrence within post-combustion CO₂ capture plant is presented, discussed and compared in this chapter. Furthermore, an exploratory analysis is made over the selected features to see their behavior when foaming/ not-foaming. Finally, a discussion is provided to present insights that can be helpful in predicting and preventing foaminess within the post-combustion amine-based CO₂ capture plants.

4.1 ANN model

The ANN model includes 1 hidden layer with 60 neurons. The number of neurons was tuned by encoding a for loop over entire model to determine how many neurons are required to obtain an acceptable accuracy for both training and test set, as illustrated in Figure 4.1.

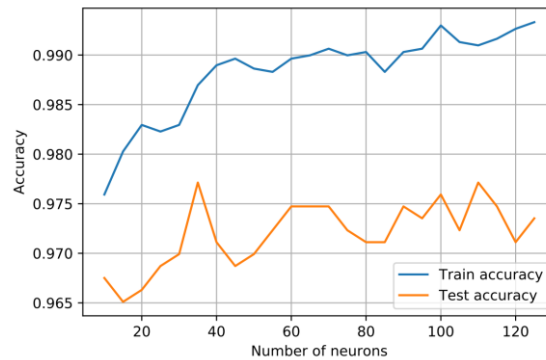


Figure 4.1: Tuning the number of neurons in ANN model.

The type of layer is fully connected and the activation function used within the hidden layer neurons is ‘ReLU’. Weights are assigned random values and the bias term is set to zero in the first iteration. Since the model's output is a classifier (foaming/ not-foaming), its value can be zero or one. As a result, the output activation function is set to ‘sigmoid’. The Figure 4.2 depicts a schematic architecture of adopted artificial neural network in this report.

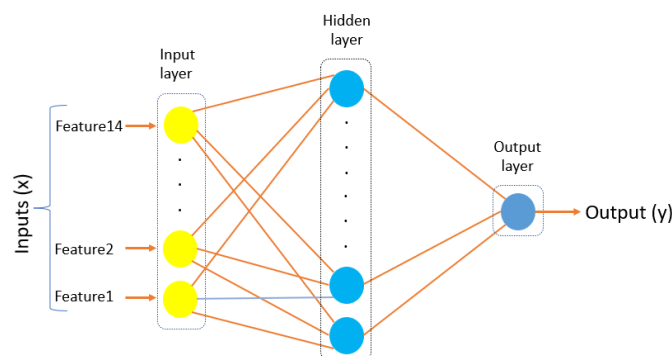


Figure 4.2: Schematic architecture of adopted ANN (number of neurons in the hidden layer are 60).

The data allocated to training is split to groups of 64 examples (i.e., mini-batches) and the cost function, 'BinaryCrossentropy', is calculated after running a mini-batch through the model. The backward propagation technique is then initiated, and an 'Adam' optimizer is applied to update the weights and bias terms where the learning rate is set to $1e^{-3}$. The number of iterations (epochs) across the entire training set is initially set to 1000, and based on the results shown in Figure 4.3, 400 epochs seem to be a suitable selection since the performance of the model does not improve with increasing epochs.

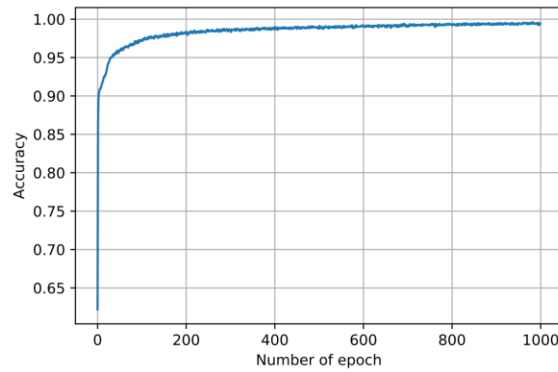


Figure 4.3: Number of epochs vs training set accuracy.

4.1.1 Performance of ANN model

The model achieves promising results in both the training and testing processes, which are shown in the Table 4.1. Since the accuracies in both training and test are high, it is concluded that the data is not underfitted by the final model. Furthermore, the test accuracy is slightly lower than the training accuracy, indicating that the data is not overfitted by the final model. The loss in the test is roughly as twice as training process, which seems reasonable. Regarding the false negative results (i.e., predicted foaming while true labeling is not-foaming), it can be confusing at first to understand why the test result outperforms the training result. However, when the number of test and training sets are considered (train: 3324, test: 831), it seems logical to have more false negative during the training process (i.e., 0.5 % and 1.7 % false negative in train and test, respectively).

Table 4.1: Results of ANN model in training and test with one hidden layer.

Procedure type	Accuracy	Loss	False negative
Training	98.5%	0.0357	15
Test	97.5%	0.0618	14

In order to better characterize the performance of the adopted ANN model, the results of a confusion matrix on the test set are shown in Figure 4.4.

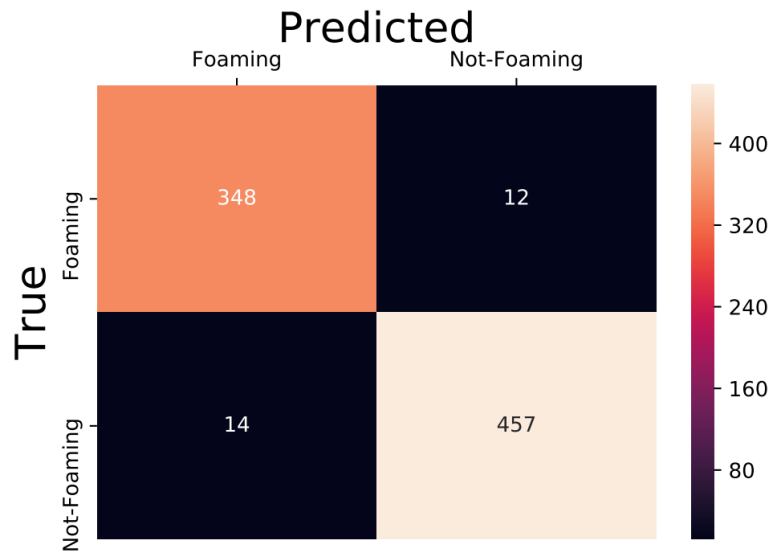


Figure 4.4: Confusion matrix indicating the performance of the ANN model on the test set.

To achieve the best possible results, another hidden layer with the same settings (fully connected layer, 60 neurons and activation function ‘ReLU’) was added to the model, and the results are shown in Table 4.2. Although the accuracy in the training set increased by 1%, the test accuracy dropped by 0.5%. Moreover, the number of false negative cases in the test set increased in comparison with the number of false negative cases in the model with one hidden layer (from 14 to 16). Considering these two changes in the results, as well as the fact that the total number of neurons/hidden layers was doubled (resulting in a model with higher order), it is likely that the model is slightly overfitting the data in the training set and cannot generalize in the test procedure as good as the model with one hidden layer did. Therefore, the model with one hidden layer seems to have a better performance on the data.

Table 4.2: Results of ANN model in training and test with two hidden layers.

Procedure type	Accuracy	Loss	False negative
Training	99.4%	0.0156	9
Test	97%	0.1465	16

Besides, the split ratio of training and test data sets was tuned, but there was not a notable change in the results.

4.2 Performance of the decision tree model

A decision tree model was used in order to validate the results obtained from the ANN model. The input data and features to the decision tree model were identical to the ones to the ANN model. Furthermore, the data was shuffled and 80 percent of the data was allocated to the train set and the remaining 20 percent was kept to test the model (same procedure as ANN). It is worthwhile to mention that the max depth (i.e., the longest path between decision node and leaf node (see Figure 2.10)) was initially set to be flexible (i.e., depending to the performance of the training set, the max depth is tuned), which caused the model to overfit the data. Therefore, the max depth was tuned to find the optimal number that can prevent overfitting, and it was revealed that using a max depth of 4, the model performs well in the training and generalizes with sufficient accuracy in the test procedure. The obtained results are shown in the Table 4.3.

Table 4.3: Results of decision tree model in training and test using a max depth of 4.

Procedure type	Accuracy
Training	96%
Test	94.5%

The obtained results in the training and test are worse than the ANN results (2.5% and 3% worse than ANN results in the training and test, respectively), indicating that the ANN model offers promising accuracy.

4.3 Feature analysis based on labeling (foaming/ not-foaming)

An exploratory search was performed on the all of retained features after feature selection to determine if there is a threshold that can separate the foaming and not-foaming data. For this reason, the entire time-steps of all 14 features were plotted separately where the foaming and not foaming time-steps were categorized by color (green: not-foaming, red: foaming; e.g., see Figure 4.5). Following, an attempt was made on all of the created figures to find out if a horizontal line (i.e., threshold) can classifies with sufficient accuracy the not-foaming labeled time-steps (see the blue dash line in the Figure 4.5). Out of the 14 retained features, 7 feature's plots (see Figure 4.5 to Figure 4.7) have the specified attribute; meaning that a threshold can classify the foaming/ not-foaming condition.

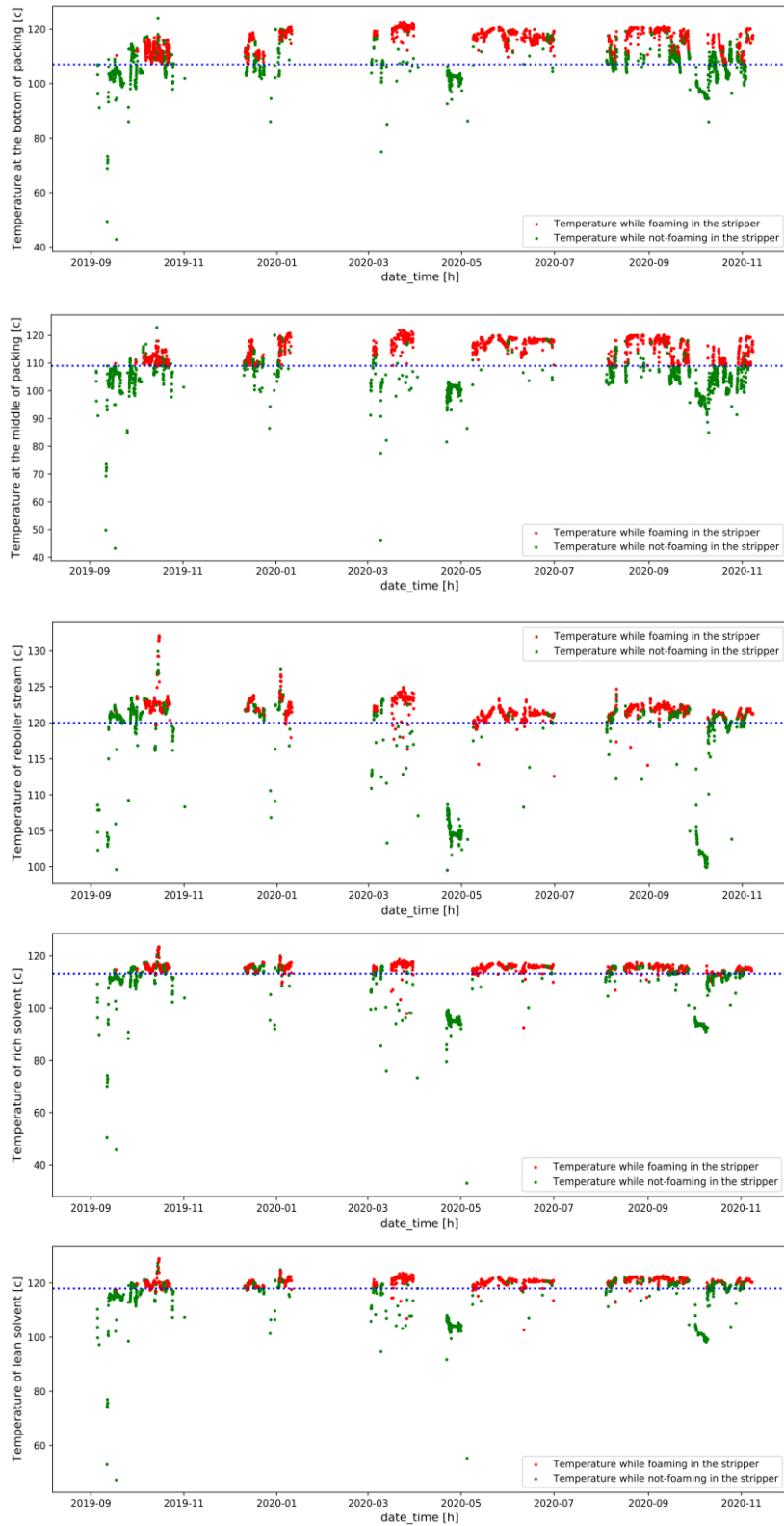


Figure 4.5: Temperature profiles belong to the packing's bottom (a), packing's middle (b), reboiler stream (c), rich solvent (d) and lean solvent (e) while foaming/ not-foaming in the stripper. Blue dotted lines represent the thresholds (see Table 4.4 for the exact value) of the foaming alarm.

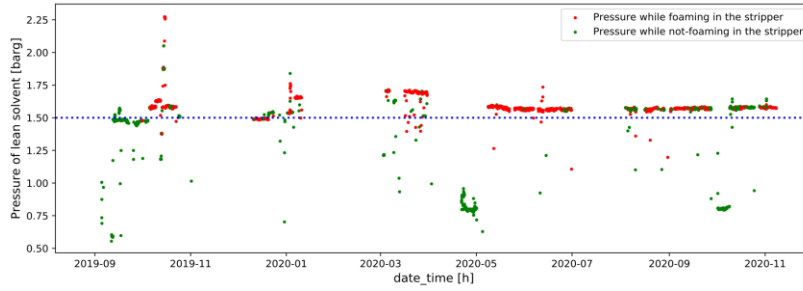


Figure 4.6: Pressure profile of lean solvent while foaming/ not-foaming. Blue dotted line represents the thresholds (see Table 4.4 for the exact value) of the foaming alarm.

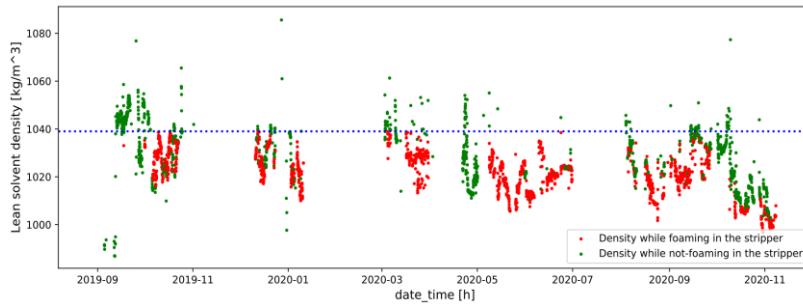


Figure 4.7: Density profile of lean solvent while foaming/ not-foaming. Blue dotted line represents the thresholds (see Table 4.4 for the exact value) of the foaming alarm.

These features and the determined thresholds are presented in the Table 4.4.

Table 4.4: Feature's foaming thresholds of TCM strippers.

Feature (physical property)	Foaming threshold
Temperature at the bottom of packing	107 °C
Temperature at the middle of packing	109 °C
Temperature of lean solvent	118 °C
Pressure of lean solvent	1.5 [<i>bar g</i>]
Density of lean solvent	1039 $\left[\frac{Kg}{m^3}\right]$
Temperature of rich solvent	113 °C
Temperature of reboiler stream	120 °C

4.4 Discussion

The ANN model fits the data roughly 98.5% in the training and 97.5% in the test set that is due to the fine tuning of the number of neurons, number of epochs, number of hidden layer and the split ratio (train and test). Furthermore, the results of the confusion matrix on the test set (831 time-steps) show that the model predicted foaming incorrectly 14 times (i.e., false positive) when there was no foaming, and it predicted not-foaming incorrectly 12 times (i.e., false negative) when there was foaming. The overall percentage of wrong prediction is approximately 3% which indicates that the model generalizes on the new samples well. False not-foaming prediction (when there was foaming) case is more critical than the false foaming prediction (when there was no foaming) case. This is due to the fact that false not-foaming prediction send wrong signal that there is no probability of foaming, and as a result, there are no preventive reactions to stop the drop in the CO2 capture efficiency. Considering the latter argument together with the fact that the predicted false not-foaming cases are less than predicted false foaming cases (i.e., 12 comparing to 14), one can notices that the inefficiency of the model happens less frequent in the critical conditions (i.e., false not foaming).

The ultimate goal of foaming prediction is to implement preventive measures prior to foaming. Although adding anti-foam is one option, it will be beneficial to avoid foaming by fine-tuning the features that have the greatest impact on foaming occurrence. As a result, it is vital to better understand which features influence more the output of the ANN model. For this purpose, a feature importance method (i.e., SHapley Additive exPlanations (SHAP)) was applied to the final achieved ANN model. SHAP [66] is the method used to calculate the shap value for each feature. The shap value represents how influential that feature is on the model's output. To get the shap value, the impacts of that specific feature on the output for each combination of features (including the one which the shap value is calculated for) is computed, and then the average of all computed values are taken to obtain shap value for that feature. The results of feature importance method are shown in the Figure 4.8.

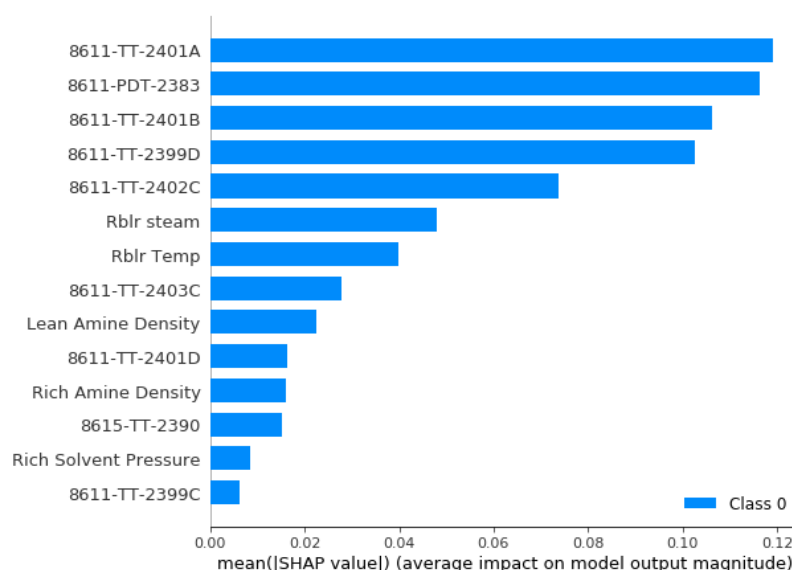


Figure 4.8: Feature importance applied on the fitted ANN model using SHAP technique.

Considering the Figure 4.8 reveals that temperature at the bottom of stripper's packing and packing pressure drop are the features that have the greatest impact on the output of the ANN model. Furthermore, the reboiler stream rate and reboiler temperature have also substantial impact on the ANN model's output (i.e., sixth and seventh shap value among all features, respectively). The reboiler duty, that is calculated based on the equation 4.1, has direct correlation with reboiler stream rate and temperature.

$$\dot{Q} = \dot{m}c\Delta T, \quad 4.1$$

where \dot{Q} is the reboiler duty, \dot{m} is the reboiler mass stream rate, c is the specific heat capacity and ΔT is the temperature difference between inlet and outlet of reboiler streams. As discussed in the section 3.6, CO₂ production rate is highly correlated with reboiler duty and stream. When foaming occurs within the stripper, the CO₂ production rate tends to reduce. In this case, an increase in reboiler duty compensates for the foaming effect and prevents efficiency reduction. Therefore, it seems reasonable to consider increasing in reboiler duty is a sign to react against decreasing in the CO₂ production rate when foaming occurs.

Furthermore, it should be noted that the findings of feature importance (Figure 4.8) show the importance of the packing pressure drop, packing temperature, and reboiler duty in foaming occurrence which is in turn consistent with the mentioned foaming symptoms in the literature (see section 2.1.5). Also, the exploratory analysis which has been done in the section 4.3 suggests that the temperature at the bottom and middle of packing, temperature of lean and rich solvent and temperature of reboiler stream are indicators of foaming occurrence as this has been mentioned in the literature too. As a result, if these features meet their thresholds during the process, there will be a symptom of foaming and preventive actions such as introducing anti-foam or fine-tuning the physical and thermodynamical parameters of the process can be applied to improve the efficiency of CO₂ capture process.

5 Conclusion

Post-combustion CO₂ capture plants (amine-based) face some challenges such as foaming that occurs within the plant's columns (absorber and stripper) and results in a decreased CO₂ capturing efficiency. Methods such as adding anti-foam to the columns can mitigate the effect of foaming. However, the main challenge is the foaming prediction before its occurrence. Despite its importance, there has been no model that can simulate the foaming occurrence based on the physics or thermodynamics of the process. Therefore, this report focused on the developing, for the first time, a data-driven model that can simulate and predict the undesirable foaming occurrence. The data used in this report was provided by technology center mongstad (TCM) that is the external partner of this project. The data includes the time series (10968 hourly time-steps) of 35 features (i.e., physical properties of the process) regarding the CO₂ capture process of TCM post-combustion CO₂ capture plant (amine-based). Comprehensive data preprocessing and feature selection was performed to ensure the high data quality before feeding the data to the models. An artificial neural network (ANN) was used to build a predictive model (the tuned ANN architecture includes 1 hidden layer with 60 neurons and 1 output). The results showed 98.5% and 97.5% accuracy in the training and test sets, respectively. In order to better understand the performance of ANN model, a confusion matrix was created based on the obtained results. Regarding the obtained matrix, it was revealed that the false predictions on the test set include only 3% where the number of false not-foaming prediction (when there was foaming) cases is less than the number of false foaming predictions (when there was no foaming) cases (foaming prediction is more important than non-foaming prediction). To validate the obtained results of ANN model, a decision tree model was used where the input data and features to the decision tree model were identical to the ones to the ANN model. The obtained results in the training and test of the decision tree model were found to be worse than the ANN results (96% and 94.5% accuracy in the training and test sets, respectively), indicating that the ANN model offers promising accuracy. Furthermore, the feature importance of the fitted ANN model was studied to determine the impact of features on the model's output. For this sake, a feature importance method, SHAP method, was applied on the finalized ANN model, and the results showed that temperature at the bottom of stripper's packing and packing pressure drop are the features that have the greatest impact on the output of the ANN model. Furthermore, the reboiler stream rate and temperature, which are directly related to the reboiler duty, have also significant impact on the ANN model's output. In addition, an exploratory analysis based on the foaming condition was performed on the features and the results indicated that the temperature at the bottom and middle of packing, temperature of lean and rich solvent and temperature of reboiler stream are the main indicators of foaming occurrence. It is worth to mention that the findings of the feature importance method and exploratory analysis are consistent with the symptoms of foaming within the columns reported in the literature. Overall, this study reported that the temperature of the column and solvent, packing pressure drop, and reboiler duty have the greatest impact on the occurrence of foaming within the post-combustion CO₂ capture plant (amine-based).

5.1 Future studies

Impurities (particulates) within the solvent, surface tension, lean and rich solvent viscosity have been mentioned in the literature (i.e., section 2.1) that have the potential to contribute the formation and stabilization of the foaming. Among these features, surface tension is an important solvent property that can directly influence foaming occurrence. As mentioned in the section 2.1.3 and 2.1.4, surface tension can contribute in the foaming formation at the first stage, and then it has the potential to prevent foam rupturing (make it stabilized). The other mentioned features (i.e., solvent, stripper and process properties) seems to have an indirect effect on foaming occurrence by affecting surface tension. Therefore, the inclusion of the mentioned features, especially surface tension, may results in more accurate modeling of foaming.

In this report, the presence or absence of foaming within each time-step was determined using some criteria applied to the provided data (i.e., section 3.5). These criteria were defined based on the researches discussed in section 2.1.5 as well as the exploratory analysis on the provided data throughout the time period of foaming labeled figure (i.e., Figure 3.4) given by TCM. The employed labeling technique may classify the time-steps not as accurately as the real foaming occurrence in the plant's actual strippers. For future studies, using an experimental method to determine the occurrence of foaming at different time-steps can represent more realistic characteristics of foaming process in the columns and therefore improve the model's accuracy.

As previously stated, TCM's provided data includes 10968 time-steps and the time interval between time-steps is one hour. Since the nature of the physical reactions that results in foaming within the stripper in each time-step may be influenced by previous time-steps, it can be insightful to simulate the process using a recurrent neural network (RNN). In terms of data preprocessing in this report, 6813 time-steps (62% of entire time-steps) were deleted, and due to the gaps between time-steps, it was not possible to implement the RNN model. Nevertheless, although the performance of the ANN model was satisfactory, employing RNN model by use of consecutive time-steps data can unveil new aspects of the foaming process.

References

- [1] G. Luderer *et al.*, "Residual fossil CO₂ emissions in 1.5-2 °C pathways," *Nature climate change*, vol. 8, no. 7, pp. 626-633, 2018.
- [2] R. Cavicchioli *et al.*, "Scientists' warning to humanity: microorganisms and climate change," *Nat Rev Microbiol*, vol. 17, no. 9, pp. 569-586, 2019.
- [3] M. Rahimi, "Public awareness: What climate change scientists should consider," *Sustainability (Basel, Switzerland)*, vol. 12, no. 20, pp. 1-4, 2020.
- [4] M. Rahimi, S. M. Moosavi, B. Smit, and T. A. Hatton, "Toward smart carbon capture with machine learning," *Cell reports physical science*, vol. 2, no. 4, p. 100396, 2021.
- [5] H. F. Svendsen, E. T. Hessen, and T. Mejdell, "Carbon dioxide capture by absorption, challenges and possibilities : Symposium on Post-Combustion Carbon Dioxide Capture," *Chemical engineering journal (Lausanne, Switzerland : 1996)*, vol. 171, no. 3, pp. 718-724, 2011.
- [6] T. C. Merkel, H. Lin, X. Wei, and R. Baker, "Power plant post-combustion carbon dioxide capture: An opportunity for membranes," *Journal of membrane science*, vol. 359, no. 1, pp. 126-139, 2010.
- [7] Z. Liang, K. Fu, R. Idem, and P. Tontiwachwuthikul, "Review on current advances, future challenges and consideration issues for post-combustion CO₂ capture using amine-based absorbents," *Chinese Journal of Chemical Engineering*, vol. 24, no. 2, pp. 278-288, 2016.
- [8] Z. Zhang *et al.*, "Effectiveness of amino acid salt solutions in capturing CO₂: A review," vol. 98, pp. 179-188, 2018.
- [9] T. N. G. Borhani, A. Azarpour, V. Akbari, S. R. W. Alwi, and Z. A. J. I. J. o. G. G. C. Manan, "CO₂ capture with potassium carbonate solutions: A state-of-the-art review," vol. 41, pp. 142-162, 2015.
- [10] F. Li, J. Zhang, E. Oko, M. J. I. J. o. C. S. Wang, and Technology, "Modelling of a post-combustion CO₂ capture process using extreme learning machine," vol. 4, no. 1, pp. 33-40, 2017.
- [11] R. M. Davidson, "Post-combustion carbon capture from coal fired plants-solvent scrubbing," 2007.
- [12] K. A. Hoff, E. F. da Silva, I. Kim, A. Grimstvedt, and S. J. E. P. Ma'mun, "Solvent development in post combustion CO₂ capture-Selection criteria and optimization of solvent performance, cost and environmental impact," vol. 37, pp. 292-299, 2013.
- [13] C. Saiwan, T. Supap, R. O. Idem, and P. J. C. M. Tontiwachwuthikul, "Part 3: Corrosion and prevention in post-combustion CO₂ capture systems," vol. 2, no. 6, pp. 659-675, 2011.
- [14] D. J. Heldebrant, P. K. Koech, V.-A. Glezakou, R. Rousseau, D. Malhotra, and D. C. J. C. r. Cantu, "Water-lean solvents for post-combustion CO₂ capture: fundamentals, uncertainties, opportunities, and outlook," vol. 117, no. 14, pp. 9594-9624, 2017.
- [15] B. Thitakamol and A. Veawab, "Foaming Behavior in CO₂ Absorption Process Using Aqueous Solutions of Single and Blended Alkanolamines," *Ind. Eng. Chem. Res*, vol. 47, no. 1, pp. 216-225, 2008.

- [16] D. J. H. P. Ballard, "How to operate an amine plant," vol. 45, no. 4, pp. 137-&, 1966.
- [17] D. Ballard, "Techniques to cut energy/corrosion/chemical costs in amine units," in *Proceedings of Laurance Reid Gas Conditioning Conference*, 1986, pp. A1-A38: University of Oklahoma.
- [18] R. Smith and S. RF, "Curing foam problems in gas processing," 1979.
- [19] N. Liebermann, "Amine appearance signals condition of system," 1980.
- [20] M. M. Keaton and M. J. Bourke, "Activated carbon system cuts foaming and amine losses," *Hydrocarbon processing (International ed.)*, vol. 62, no. 8, pp. 71-73, 1983.
- [21] J. Thomason, "RECLAIM GAS TREATING SOLVENT," *Hydrocarbon processing (International ed.)*, vol. 64, no. 4, pp. 75-78, 1985.
- [22] C. R. J. C. e. p. Pauley, "Face the facts about amine foaming," vol. 87, no. 7, pp. 33-38, 1991.
- [23] E. Stewart and R. J. H. P. Lanning, "Reduce amine plant solvent losses; Part 1," vol. 73, no. 5, 1994.
- [24] C. R. Pauley, R. Hashemi, and S. Caothien, "Ways to control amine unit foaming offered," *The Oil & gas journal*, vol. 87, no. 50, pp. 67-75, 1989.
- [25] J. McCarthy and M. J. C. E. C. Trebble, "An experimental investigation into the foaming tendency of diethanolamine gas sweetening solutions," vol. 144, no. 1, pp. 159-171, 1996.
- [26] J. J. Bikerman, "The unit of foaminess," *Transactions of the Faraday Society*, vol. 34, pp. 634-638, 1938.
- [27] R. Thiele, O. Brettschneider, J.-U. Repke, H. Thielert, and G. Wozny, "Experimental Investigations of Foaming in a Packed Tower for Sour Water Stripping," *Ind. Eng. Chem. Res.*, vol. 42, no. 7, pp. 1426-1432, 2003.
- [28] J. J. N. Y. Bikerman, "Foams Springer," 1973.
- [29] M. Pilling, "Foaming in Fractionation Columns," ed, 2015.
- [30] G. Senger and G. J. C. T. M. Wozny, "Impact of foam to column operation," vol. 109, no. 1-M, pp. 209--222, 2012.
- [31] H. Z. J. C. E. R. Kister and Design, "What caused tower malfunctions in the last 50 years?," vol. 81, no. 1, pp. 5-26, 2003.
- [32] N. Sipöcz, F. A. Tobiesen, and M. Assadi, "The use of Artificial Neural Network models for CO₂ capture plants," *Applied energy*, vol. 88, no. 7, pp. 2368-2376, 2011.
- [33] V. Chan and C. Chan, "Learning from a carbon dioxide capture system dataset: Application of the piecewise neural network algorithm," *Petroleum*, vol. 3, no. 1, pp. 56-67, 2017.
- [34] A. Shalaby, A. Elkamel, P. L. Douglas, Q. Zhu, and Q. P. Zheng, "A machine learning approach for modeling and optimization of a CO₂ post-combustion capture unit," *Energy (Oxford)*, vol. 215, 2021.
- [35] M. Afkhamipour and M. Mofarahi, "Modeling and optimization of CO₂ capture using 4-diethylamino-2-butanol (DEAB) solution," *International journal of greenhouse gas control*, vol. 49, pp. 24-33, 2016.

- [36] C. Nwaoha, K. Odoh, E. Ikpatt, R. Orji, and R. Idem, "Process simulation, parametric sensitivity analysis and ANFIS modeling of CO₂ capture from natural gas using aqueous MDEA–PZ blend solution," *Journal of environmental chemical engineering*, vol. 5, no. 6, pp. 5588-5598, 2017.
- [37] A. Nuchitprasittichai and S. Cremaschi, "Optimization of CO₂ Capture Process with Aqueous Amines A Comparison of Two Simulation–Optimization Approaches," *Ind. Eng. Chem. Res*, vol. 52, no. 30, pp. 10236-10243, 2013.
- [38] J. A. Bullin and W. G. Brown, "Hydrocarbons and BTEX pickup and control from amine systems," in *83 rd Annual Convention of the Gas Processors Association, Texas*, 2004.
- [39] X. Chen, S. A. Freeman, and G. T. Rochelle, "Foaming of aqueous piperazine and monoethanolamine for CO₂ capture," *International journal of greenhouse gas control*, vol. 5, no. 2, pp. 381-386, 2011.
- [40] H. Z. Kister, *Distillation design*. New York: McGraw-Hill, 1992.
- [41] M. H. Pahl and D. Franke, "Foam and foam breaking - A review," *Chemie ingenieur technik*, vol. 67, no. 3, pp. 300-312, 1995.
- [42] *Perry's Chemical engineers' handbook*, 6th ed. / prepared by a staff of specialists under the editorial direction of late editor Robert H. Perry ; editor, Don W. Green ; assistant editor, James O. Maloney ed. New York: New York: McGraw-Hill, 1984.
- [43] D. H. Smith, "Foams: Fundamentals and Applications in the Petroleum Industry Edited by Laurier L. Schramm (Petroleum Research Institute). American Chemical Society: Washington, DC, 1994. 555 pp. ISBN 0-8412-2719-5. \$109.95," *Energy Fuels*, vol. 10, no. 1, pp. 266-266, 1996.
- [44] S. Ross and G. Nishioka, "Foaminess of binary and ternary solutions," *J. Phys. Chem*, vol. 79, no. 15, pp. 1561-1565, 1975.
- [45] W. L. J. C. E. P. Bolles, "Solution of a foam problem," vol. 63, no. 9, pp. 48-&, 1967.
- [46] E. Heinerth, "herausfinden wollen."
- [47] J. S. Charlton, *Radioisotope techniques for problem-solving in industrial process plants*. Springer Science & Business Media, 2012.
- [48] H. Z. Kister and T. C. J. P. O. P. Hower Jr, "Unusual operating histories of gas processing and olefins plant columns," vol. 6, no. 3, pp. 151-161, 1987.
- [49] C. Pratt and S. J. C. E. Hobbs, "QUICK KILL OF FOAMS ON FRACTIONATOR TRAYS," vol. 82, ed: MCGRAW HILL INC 1221 AVENUE OF THE AMERICAS, NEW YORK, NY 10020, 1975, pp. 112-112.
- [50] A. Afram and F. Janabi-Sharifi, "Review of modeling methods for HVAC systems," *Applied thermal engineering*, vol. 67, no. 1-2, pp. 507-519, 2014.
- [51] Great Learning Team, "What is Machine Learning? How Machine Learning Works and future of it?," 2022, Jan 19.
- [52] Sakshi Gupta, "Regression vs. Classification in Machine Learning: What's the Difference?," 2021, October 6.
- [53] G. Zhang, B. E. Patuwo, and M. Y. J. I. j. o. f. Hu, "Forecasting with artificial neural networks:: The state of the art," vol. 14, no. 1, pp. 35-62, 1998.

- [54] Jack Lodge. (2021, Oct 19). *How Decentralised Autonomous Organisations Leverage Nature's Complexity Engine*. Available: <https://medium.com/deeplink-labs/how-decentralised-autonomous-organisations-leverage-natures-complexity-engine-a61a4c26abe8>
- [55] R. Yamashita, M. Nishio, R. K. G. Do, and K. J. I. i. Togashi, "Convolutional neural networks: an overview and application in radiology," vol. 9, no. 4, pp. 611-629, 2018.
- [56] M. Shariati *et al.*, "Application of a hybrid artificial neural network-particle swarm optimization (ANN-PSO) model in behavior prediction of channel shear connectors embedded in normal and high-strength concrete," *Applied sciences*, vol. 9, no. 24, p. 5534, 2019.
- [57] Brendan Coady, "Gradient Ascent," 2017, Nov 23.
- [58] P.-H. C. Chen, Y. Liu, and L. Peng, "How to develop machine learning models for healthcare," *Nat Mater*, vol. 18, no. 5, pp. 410-414, 2019.
- [59] H. Jabbar, R. Z. J. C. S. Khan, Communication, and I. Devices, "Methods to avoid over-fitting and under-fitting in supervised machine learning (comparative study)," vol. 70, 2015.
- [60] GeeksforGeeks, "ML | Underfitting and Overfitting," 2021, Oct 20.
- [61] C. Z. J. I. T. o. S. Janikow, Man, and P. B. Cybernetics, "Fuzzy decision trees: issues and methods," vol. 28, no. 1, pp. 1-14, 1998.
- [62] Avinash Navlani, "Decision Tree Classification in Python Tutorial," 2018, December 28.
- [63] European Commission, "Final Report Summary - CESAR (CO2 Enhanced Separation and Recovery)," 2012, October 31.
- [64] K. Johnsen *et al.*, "CO2 Product Quality: Assessment of the Range and Level of Impurities in the CO2 product Stream from MEA Testing at Technology Centre Mongstad (TCM)," in *14th Greenhouse Gas Control Technologies Conference Melbourne*, 2018, pp. 21-26.
- [65] J. Li *et al.*, "Feature selection: A data perspective," vol. 50, no. 6, pp. 1-45, 2017.
- [66] GIANLUCA MALATO, "How to explain neural networks using SHAP," 2021, MAY 17

Appendix A

FMH606 Master's Thesis

Title: Foaming prediction in the post-combustion CO₂ capture plants (amine-based) by utilizing machine learning techniques

USN supervisors: Leila Ben Saad and Ru Yan

External partner: Technology Centre Mongstad (TCM) v/ Rune Teigland

Task background:

Technology Centre Mongstad (TCM) is the world's largest and most flexible test center for developing CO₂ capture technologies and a leading competence center for carbon capture. TCM has been operating since autumn 2012, providing an arena for qualification of CO₂ capture technologies on an industrial scale. A vast amount of data is collected from more than 1000 online instruments in the amine plant and more than 1100 in the utility plant. In addition, there are multiple sampling points for liquid sampling throughout the amine plant. Hence a vast amount of data is readily available to be exploited.

Machine Learning (ML) can be applied when you have a complex task or problem involving big data and many variables, but you do not know the formula/equation or classic regression methods do not fit well.

Machine Learning methods has the potential to design, test and improve various aspects of the CO₂ process that are computationally time consuming or experimentally time consuming and expensive. The use of ML techniques for carbon capture processes [1-4] is still emerging and most investigations have been on simplified models.

The objective of this project is to build data driven models that can enhance the understanding of relationships among key process parameters. The data available from TCM open campaigns will be used as training data sets for ANN model to be built. The aim of ANN model is to provide an analysis of the extracted rules and reveal most significant relationships between them.

In this project, the focus will be mainly on the problem of foaming, which is a common issue in gas-liquid scrubbing processes and results in poor CO₂ capture rate, increased energy penalty for CO₂ capture, emissions and CO₂ stream purity challenges and solvent losses. The use of ANN model in this context will allow to predict the foaming risk.

References:

[1] Rahimi, M., Moosavi, S. M., Smit, B., and Hatton, T. A., "Toward smart carbon capture with machine learning", Cell Reports Physical Science, vol. 2, 2021.

[2] Chan, V. and Chan, C., Learning from a carbon dioxide capture system dataset: Application of the piecewise neural network algorithm, *Petroleum*, Volume 3, Issue 1, pages 56-67, 2017.

[3] Li, F., Zhang, J., Oko, E. et al. Modelling of a post-combustion CO₂ capture process using extreme learning machine. *Int J Coal Sci Technol* 4, 33–40, 2017.

[4] Nikolett Sipöcz, Finn Andrew Tobiesen, Mohsen Assadi, The use of Artificial Neural Network models for CO₂ capture plants, *Applied Energy*, Volume 88, Issue 7, Pages 2368-2376, 2011.

Task description:

- Give an overview of CO₂ capture technologies with focus on data monitoring and control parameters.
- Understand the problem of foaming in CO₂ capture process and select its main key parameters.
- Review and inspect the state-of-the-art of adopting machine learning in the context of CO₂ capture process.
- Describe the steps of developing data driven models using a machine learning approach.
- Analyze and investigate various approaches utilizing Artificial Neural Networks to find the most suitable models for this application.
- Develop some of these models based on data from TCM and discuss the accuracy of these models

Development languages & tools: Python offers a good platform for software development as it is a well-known programming language and has access to great (open source) libraries and frameworks for AI and machine learning (ML), flexibility, platform independence, some of the libraries are:

- [Keras](#), [TensorFlow](#), [PyTorch](#) and [Scikit-learn](#) for machine learning
- [NumPy](#) for high-performance scientific computing and data analysis

Student category: IIA (EET, EPE, IIA or PT students)

Requirements: advanced coding skills in Python and some knowledge in machine learning

The task is suitable for online students (not present at the campus): Yes

Practical arrangements:

Data will be provided by TCM

Supervision:

As a general rule, the student is entitled to 15-20 hours of supervision. This includes necessary time for the supervisor to prepare for supervision meetings (reading material to be discussed, etc).

Signatures:

Supervisor (date and signature): 01/02/2022

Leila Ben Saad



Student (write clearly in all capitalized letters):

Nasir Niayifar 238727@student.usn.no

Student (date and signature): 01/02/2022



Appendix B

Finding offline time-steps (The same script is encoded for RFCC) script

```
import pandas as pd
import numpy as np

# %% Read the data and convert it to numpy
data_sheet1 = pd.read_csv(r'C:\Users\amin\Desktop\USN - Foaming Data Extract.csv')
data_sheet1.head()
data_sheet1 = data_sheet1.to_numpy(dtype='float32')

# %% Finding CHP offline
rich_solventflow = []
rich_solventflow = data_sheet1[:, 1]
lean_solventflow = []
lean_solventflow = data_sheet1[:, 0]
rblr_temp_chp = []
rblr_temp_chp = data_sheet1[:, 6]
rblr_temp_rfcc = []
rblr_temp_rfcc = data_sheet1[:, 7]
co2_chp = []
co2_chp = data_sheet1[:, 3]
co2_rfcc = []
co2_rfcc = data_sheet1[:, 5]
plant_offline_flow = np.asarray(np.where((rich_solventflow < 10000) & (lean_solventflow < 10000))).T
co2_prud_offline = np.asarray(np.where((co2_chp < 1000) & (co2_rfcc < 1000))).T
reboilers_offline = np.asarray(np.where((rblr_temp_chp < 33) & (rblr_temp_rfcc < 33))).T
chp_stripper_offline = np.asarray(np.where(rblr_temp_chp < 33)).T
chp_offline = np.concatenate((plant_offline_flow, co2_prud_offline, reboilers_offline,
chp_stripper_offline), axis=None)
chp_offline = set(chp_offline)
chp_offline = np.asarray(sorted(chp_offline))
chp_offline = pd.DataFrame(chp_offline, columns = ['column1'])
chp_offline.to_csv(r'chp_offline.csv', index = False)
```

Data preprocessing script

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# %% Flow Pressure & Temp
pres_temp_chp = pd.read_excel('USN - Foaming Data Extract.xlsx', sheet_name= 'Flow Pressure &
Temp', header=5)
pres_temp_chp = pres_temp_chp.iloc[2:,2:]
pres_temp_chp = pres_temp_chp.reset_index(drop=True)
pres_temp_chp.rename(columns={'Unnamed: 2': 'date_time'}, inplace=True)
#set the date-time format
pres_temp_chp['date_time'] = pd.to_datetime(pres_temp_chp['date_time'], format = '%d.%m.%Y
%H:%M')
#drop the columns belong to rfcc
pres_temp_chp.drop(['Rblr steam.1', 'CO2 Prod.1', 'RFCC rblr temp'], axis=1, inplace=True)

# %% CHP Stripper Profiles
chp_stripper = pd.read_excel('USN - Foaming Data Extract.xlsx', sheet_name= 'CHP Stripper
Profiles', header=6)
chp_stripper = chp_stripper.iloc[2:,3:]
chp_stripper = chp_stripper.reset_index(drop=True)
#drop demister and waterwash pressure drops
chp_stripper.drop(['8611-PDT-2442', '8611-PDT-2441'], axis=1, inplace=True)

# %% Density
density = pd.read_csv('Density_include.csv')

# %% concatenate
chp_tot = pd.concat([pres_temp_chp, chp_stripper, density], axis=1)

# %% removing chp offline time-steps and columns with high number of nan values
offline_chp = pd.read_csv('chp_offline.csv')
chp_tot = chp_tot.loc[~chp_tot.index.isin(offline_chp['column1'])]
chp_tot = chp_tot.reset_index(drop=True)
xx = chp_tot.isnull().sum(axis = 0)
chp_tot.drop(['8611-TT-2399A', '8611-TT-2399B', '8611-TT-2403D', '8611-TT-2404A', '8611-TT-
2404B', '8611-TT-2404D', '8611-TT-2405D', '8611-TT-2446C'], axis=1, inplace=True)

# %% nan & zeros
chp_tot = chp_tot.dropna()
```

```

chp_tot = chp_tot[(chp_tot != 0).all(1)]
chp_tot = chp_tot.reset_index(drop=True)

# %% Flow Pressure & Temp rfcc
pres_temp_rfcc = pd.read_excel('USN - Foaming Data Extract.xlsx', sheet_name= 'Flow Pressure &
Temp', header=5)
pres_temp_rfcc = pres_temp_rfcc.iloc[2:,2:]
pres_temp_rfcc = pres_temp_rfcc.reset_index(drop=True)
pres_temp_rfcc.rename(columns={'Unnamed: 2': 'date_time'}, inplace=True)
#set the date-time format
pres_temp_rfcc['date_time'] = pd.to_datetime(pres_temp_rfcc['date_time'], format = '%d.%m.%Y
%H:%M')
#drop the columns belong to chp
pres_temp_rfcc.drop(['Rblr steam', 'CO2 Prod', 'CHP Rblr Temp'], axis=1, inplace=True)

# %% RFCC Stripper Profiles
rfcc_stripper = pd.read_excel('USN - Foaming Data Extract.xlsx', sheet_name= 'RFCC Stripper
Profiles', header=5)
rfcc_stripper = rfcc_stripper.iloc[:,3:]
rfcc_stripper.drop([0], axis=0, inplace=True)
rfcc_stripper = rfcc_stripper.reset_index(drop=True)
#drop demister and waterwash pressure drops
rfcc_stripper.drop(['8611-PDT-2435', '8611-PDT-2436'], axis=1, inplace=True)

# %% concatenate
rfcc_tot = pd.concat([pres_temp_rfcc, rfcc_stripper, density], axis=1)

# %% removing rfcc offline time-steps and columns with high number of nan values
offline_rfcc = pd.read_csv('rfcc_offline.csv')
rfcc_tot = rfcc_tot.loc[~rfcc_tot.index.isin(offline_rfcc['column1'])]
rfcc_tot = rfcc_tot.reset_index(drop=True)
yy = rfcc_tot.isnull().sum(axis = 0)
rfcc_tot.drop(['8611-TT-2175A', '8611-TT-2175B', '8611-TT-2178D', '8611-TT-2179A', '8611-TT-
2179B', '8611-TT-2179D', '8611-TT-2180D', '8611-TT-2181C'], axis=1, inplace=True)

# %% nan & zeros
rfcc_tot = rfcc_tot.dropna()
rfcc_tot = rfcc_tot[(rfcc_tot != 0).all(1)]
rfcc_tot = rfcc_tot.reset_index(drop=True)
# %% total plant
plant_tot = pd.DataFrame( np.concatenate( (chp_tot.values, rfcc_tot.values), axis=0 ) )

```

```
plant_tot.columns = chp_tot.columns
plant_tot.to_excel('plant_tot.xlsx', startcol=-1)
```

```
# %% chp excel
chp_tot.to_excel('chp_tot.xlsx', startcol=-1)
```

```
# %% rfcc excel
rfcc_tot.to_excel('rfcc_tot.xlsx', startcol=-1)
```

Foaminess labeling script

```
import pandas as pd
import numpy as np
```

```
# %% read the chp and rfcc preprocessed tables
```

```
chp_tot = pd.read_excel(r'C:\USN\fourth semester\thesis\programing\pandas\chp_tot.xlsx')
rfcc_tot = pd.read_excel(r'C:\USN\fourth semester\thesis\programing\pandas\rfcc_tot.xlsx')
```

```
# %% foaming detection chp_tot
```

```
chp_tot['foaming'] = 0
```

```
chp_tot.loc[(chp_tot['8611-TT-2401A'] > 107) & (chp_tot['8611-TT-2403A'] > 109) & (chp_tot['Lean Amine Density'] < 1039) & (chp_tot['8611-PDT-2383'] > 1), 'foaming'] = 1
```

```
chp_tot['foaming'].value_counts()[1]
```

```
# %% foaming detection rfcc_tot
```

```
rfcc_tot['foaming'] = 0
```

```
rfcc_tot.loc[(rfcc_tot['8611-TT-2176A'] > 107) & (rfcc_tot['8611-TT-2178A'] > 109) & (rfcc_tot['Lean Amine Density'] < 1039) & (rfcc_tot['8611-PDT-2155'] > 0.1), 'foaming'] = 1
```

```
rfcc_tot['foaming'].value_counts()[1]
```

```
# %% foaming detection plant_tot
```

```
plant_tot = pd.DataFrame( np.concatenate( (chp_tot.values, rfcc_tot.values), axis=0 ) )
```

```
plant_tot.columns = chp_tot.columns
```

```
plant_tot['foaming'].value_counts()[1]
```

```
plant_tot.to_excel('plant_tot_plusfoam.xlsx', startcol=-1)
```


ANN model script

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
import tensorflow as tf
from keras.models import Sequential
from keras.layers import Dense
import matplotlib.pyplot as plt
import seaborn as sns

# %% Importing the dataset
plant_tot = pd.read_excel(r'C:\USN\fourth semester\thesis\programing\foaming
detection\plant_tot_plusfoam.xlsx')

# %% Feature selection based on correlation
x = plant_tot.iloc[:, 1:36]
corr = x.corr()
plt.subplots(figsize=(10,8))
sns.heatmap(corr)
columns = np.full((corr.shape[0],), True, dtype=bool)
for i in range(corr.shape[0]):
    for j in range(i+1, corr.shape[0]):
        if corr.iloc[i,j] >= 0.9:
            if columns[j]:
                columns[j] = False
selected_columns = x.columns[columns]
x = x[selected_columns]
y = plant_tot.iloc[:, -1]
plt.savefig('Correlation.pdf', dpi=120, format='pdf', bbox_inches='tight')
plt.show()

# %% Concatinate the data
plant_tot = pd.concat([x, y], axis=1)
# %% Shuffle the data
plant_tot_shuffled = plant_tot.sample(frac=1).reset_index(drop=True)

# %% Input output separation
```

```

x = plant_tot_shuffled.iloc[:, 1:15]
y = plant_tot_shuffled.iloc[:, -1]

# %% Splitting the dataset into the Training set and Test set
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2)

# %% Input Scaling
sc = StandardScaler()

x_train = sc.fit_transform(x_train)
x_test = sc.transform(x_test)
x_train = pd.DataFrame(x_train)
x_train.columns = x.columns
x_test = pd.DataFrame(x_test)
x_test.columns = x.columns

# %% ANN model
foam_predictor = Sequential()
foam_predictor.add(tf.keras.Input(shape=(len(x_train.columns),)))
foam_predictor.add(Dense(
    units = 60,
    activation='ReLU',
    use_bias=True,
    kernel_initializer="glorot_uniform",
    bias_initializer="zeros",
))
foam_predictor.add(Dense(
    units = 1,
    activation='sigmoid',
    use_bias=True,
    kernel_initializer="glorot_uniform",
    bias_initializer="zeros",
))
foam_predictor.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=1e-3),
    loss=tf.keras.losses.BinaryCrossentropy(),
    metrics=[tf.keras.metrics.BinaryAccuracy(),
    tf.keras.metrics.FalseNegatives()])
history = foam_predictor.fit(

```

```

x=x_train,
y=y_train,
batch_size=64,
epochs=400,
validation_split=0.1,
initial_epoch=0,
)

# %% Model Accuracy
print("Evaluate on test data")
results = foam_predictor.evaluate(x_test, y_test, batch_size=128)
print("test loss, test accuracy:", results)

```

Decision tree model script

```

import pandas as pd
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn import metrics
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

# %% Importing the dataset
plant_tot = pd.read_excel(r'C:\USN\fourth semester\thesis\programing\foaming
detection\plant_tot_plusfoam.xlsx')

# %% Feature selection based on correlation
x = plant_tot.iloc[:, 1:36]
corr = x.corr()
plt.subplots(figsize=(10,8))
sns.heatmap(corr)
columns = np.full((corr.shape[0],), True, dtype=bool)
for i in range(corr.shape[0]):
    for j in range(i+1, corr.shape[0]):
        if corr.iloc[i,j] >= 0.9:
            if columns[j]:
                columns[j] = False
selected_columns = x.columns[columns]

```

```

x = x[selected_columns]
y = plant_tot.iloc[:, -1]
plt.savefig('Correlation.pdf', dpi=120, format='pdf', bbox_inches='tight')
plt.show()

# %% Concatenate the data
plant_tot = pd.concat([x, y], axis=1)

# %% Shuffle the data
plant_tot_shuffled = plant_tot.sample(frac=1).reset_index(drop=True)

# %% Input output separation
x = plant_tot_shuffled.iloc[:, 1:15]
y = plant_tot_shuffled.iloc[:, -1]

# %% Splitting the dataset into the Training set and Test set
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2)

# %% Decision tree model
# Create Decision Tree classifier object
clf = DecisionTreeClassifier()
# Train Decision Tree Classifier
clf = clf.fit(x_train,y_train)
#Predict the response for test dataset
y_pred = clf.predict(x_test)

# %% Model Accuracy
print("Accuracy:",metrics.accuracy_score(y_test, y_pred))


```

Appendix C

Clarification regarding unclear feature's names:

Regarding the features that show the temperatures within the packing, the table below clarifies where each temperature is placed.

Temperatures inside packing			
A = Closest to packing wall	B	C	D = Closest to packing center
8611-TT-2446A	8611-TT-2446B	-----	8611-TT-2446D
8611-TT-2405A	8611-TT-2405B	8611-TT-2405C	-----
-----	-----	8611-TT-2404C	-----
8611-TT-2403A	8611-TT-2403B	8611-TT-2403C	-----
8611-TT-2402A	8611-TT-2402B	8611-TT-2402C	8611-TT-2402D
8611-TT-2401A	8611-TT-2401B	8611-TT-2401C	8611-TT-2401D
-----	-----	8611-TT-2399C	8611-TT-2399D



In addition, the location of temperatures outside of the packing is listed in the following table.

Temperature inside the stripper (Outside of packing)	
8615-TT-2390	Temperature at the stripper's outlet
8611-TT-2382	Temperature between packing's top and stripper's outlet
8611-TIC-2379	Temperature at the stripper's bottom

Furthermore, the feature 8611-PDT-2383 represents pressure drop within the packing.

Appendix D

Technology Centre Mongstad (TCM):

TCM as the world's largest carbon capture test center was founded in 2012. Gassnova, Statoil, Total, and Shell own TCM and the major attention of the TCM is the evaluation and development of new carbon capture technologies prior to their application on the full-scale commercial plants. Companies who wish to test their carbon capture system before implementing it in a full-scale plant (i.e., post-combustion in TCM) can decrease their risks and expenses by using two units constructed in the TCM, i.e., Mongstad, Bergen in Norway; that operate with various solvent-based techniques. Each of these units has a 12 megawatt electrical capability and can capture 100,000 tons of CO₂ (together) per year.