

MULTIVARIATE TIME SERIES PREDICTION OF DRILLING RATE AND HAZARDS

BY
**BASIRUDIN RACHMAN BACHTIAR
DJAMALUDDIN**

A Thesis Presented to the
COLLEGE OF GRADUATE AND INTERDISCIPLINARY STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER SCIENCE

JANUARY 2024

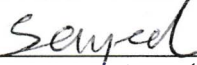
KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **BASIRUDIN RACHMAN BACHTIAR DJAMALUD-DIN** under the direction of his thesis adviser and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN COMPUTER SCIENCE**.

Thesis Committee


Dr. Salahadin Adem Mohammed (Adviser)


Prof. El-Sayed M. El-Alfy (Member)


Dr. Emad Yehia Ahmed Ramadan (Member)



Dr. Motaz Abdulrahman
Alfarraj
Department Chairman


Prof. Suliman Saleh Al-
Homidan
Dean of Graduate Studies

Date



©Basirudin Rachman Bachtiar Djamaluddin
2024

To my wife, Maulida!

ACKNOWLEDGMENTS

I would like to express my gratitude for the unwavering support provided by my advisor, Dr. Salahadin Adem Mohammed, who has consistently shown trust in me. I greatly appreciate the valuable input from Prof. El-Sayed M. El-Alfy, which has greatly enhanced the quality of my work. I would also like to express appreciation to Dr. Emad Yehia Ahmed Ramadan for his continuous support.

TABLE OF CONTENTS

ACKNOWLEDGMENTS	vi
LIST OF TABLES	x
LIST OF FIGURES	xi
LIST OF ABBREVIATIONS	xiv
ABSTRACT (ENGLISH)	xvii
ABSTRACT (ARABIC)	xix
CHAPTER 1 INTRODUCTION	1
1.1 Introduction	1
1.2 Research Statement and Motivation	3
1.3 Thesis Research Objectives	3
1.4 Thesis Organization	4
CHAPTER 2 BACKGROUND	5
2.1 Drilling Operation	5
2.1.1 Drilling Operation Systems	6
2.1.2 Drilling Rate of Penetration (DROP)	12
2.1.3 Drilling Hazards	14
2.2 Time Series	18
2.2.1 Time-Series Data	19
2.2.2 Time-Series Analysis	19
2.2.3 Time-Series Prediction	20

2.2.4	Time-Series Characteristics	21
2.2.5	Time-series Analysis Problems	22
2.2.6	Univariate vs Multivariate Time Series	24
2.2.7	Single-Step vs Multi-Step (Horizon-Style)	25
2.3	Machine Learning (ML) Approach	26
2.3.1	Workflow of Machine Learning Approach	27
2.3.2	Supervised Learning	30
2.3.3	Deep Learning	33
CHAPTER 3 RELATED WORK		42
3.1	DROP Prediction Techniques	42
3.1.1	Mathematical Models	43
3.1.2	Statistical Models	46
3.1.3	Data Driven Model or Machine Learning Models	47
3.2	Hazard Prediction Techniques	52
3.2.1	Statistical Models	53
3.2.2	Data-Driven or Machine Learning Models	53
CHAPTER 4 METHODOLOGY		58
4.1	Data Analysis and Preprocessing	59
4.1.1	Data Analysis	59
4.1.2	Data Preprocessing	60
4.2	Model Construction	61
4.2.1	Lost Function and Model Optimization	62
4.2.2	RNN Based Methods	63
4.2.3	Transformer	66
4.3	Model Evaluation	67
CHAPTER 5 RESULTS AND DISCUSSIONS		70
5.1	Experimental Machines and Software	71
5.2	Experimental Dataset Analysis and Preprocessing	71

5.2.1	Data Analysis	71
5.2.2	Data Preprocessing	73
5.2.3	Dataset Result for Experiment	77
5.3	Performance of the Proposed DROP Prediction Solutions	77
5.3.1	Architecture and Performance of the Proposed Class 1 Models .	78
5.3.2	Architecture and Performance of the Proposed Class 2 Models .	80
5.3.3	Comparison of Class 1 and Class 2 models	82
5.3.4	Comparison of the Time-Based DROP Model with The Best Depth-Based Model	82
5.3.5	Comparison of the Proposed DROP Models with Existing Models	86
5.4	Performance of the Proposed DH Prediction Solution	89
5.4.1	Performance of the proposed DH prediction model	90
5.4.2	Comparison of the Proposed DH Prediction Model with Exist- ing Models	93
CHAPTER 6 CONCLUSION AND FUTURE WORK		101
6.1	Conclusion	101
6.2	Future Work	103
REFERENCES		104
VITAE		121

LIST OF TABLES

Table 2.1	Variable for DROP Prediction	14
Table 3.1	ML Comparison for DROP Prediction	57
Table 5.1	The experimental machines and software	71
Table 5.2	The experimental depth-based dataset and their corresponding wells	94
Table 5.3	The experimental time-based for DROP prediction dataset and their corresponding wells	95
Table 5.4	The experimental time-based for DH prediction dataset and their corresponding wells	95
Table 5.5	The proposed LSTM1 model	96
Table 5.6	Architecture of the proposed GRU1	96
Table 5.7	Hyper-parameters of the proposed TNN1 model	97
Table 5.8	Hyper-parameters of the proposed LSTM2 model	97
Table 5.9	Hyper-parameters of the proposed GRU2 model	98
Table 5.10	Hyper-parameters of the proposed TNN2 model	98
Table 5.11	All Metrics for Class 1 and Class 2 Models	99
Table 5.12	The Best Depth Models vs Time Models	99
Table 5.13	Hyper-parameters of the proposed TNN DH model	99
Table 5.14	DH Prediction Accuracy	100

LIST OF FIGURES

Figure 2.1	Hoisting system [1]	8
Figure 2.2	Rotary system [1]	9
Figure 2.3	Circulating system [1]	11
Figure 2.4	Relationship between DROP and WOB [1]	13
Figure 2.5	Relationship between DROP and bit rotary speed [1]	14
Figure 2.6	Distribution of DH types [2]	15
Figure 2.7	Non linear trend in time series	23
Figure 2.8	Multivariate many input to many output	25
Figure 2.9	Multivariate input to multivariate output with different model for each output	25
Figure 2.10	Single-step technique for hourly periods	26
Figure 2.11	Multi-step technique for hourly periods	26
Figure 2.12	ML Workflow	27
Figure 2.13	Supervised Learning Workflow	31
Figure 2.14	DH as Binary Class Classification Illustration	32
Figure 2.15	DH as Multi Class Classification Illustration	32
Figure 2.16	DROP Regression Illustration	33
Figure 2.17	An unfolded RNN architecture	35
Figure 2.18	LSTM architecture	37
Figure 2.19	GRU architecture	38
Figure 2.20	Transformer Model Architecture	41
Figure 3.1	DROP Prediction Techniques	44
Figure 4.1	Methodology Phases	58

Figure 4.2	Data Analysis Phases	60
Figure 4.3	Model construction workflow [3]	62
Figure 4.4	LSTM Model Architecture	65
Figure 4.5	GRU Model Architecture	65
Figure 4.6	Transformer Model Architecture for Regression on ROP Prediction	67
Figure 5.1	Feature rank: The number of Datasets a feature was se- lected from	73
Figure 5.2	Class 1 Models Comparison	80
Figure 5.3	Class 2 Models Comparison	82
Figure 5.4	Time Based Data Depth vs DROP Chart	83
Figure 5.5	Prediction (Red) vs Test DROP for Time Based Data Above Depth 3800m	84
Figure 5.6	Prediction vs Test DROP Chart for Time Based Data Above Depth 3800m	84
Figure 5.7	Training Data of DROP for Time Based Data Depth Be- tween 1200m and 3700m	85
Figure 5.8	Prediction (Red) vs Test DROP for Time Based Data Above Depth 3700m	86
Figure 5.9	Prediction vs Test DROP Chart for Time Based Data Above Depth 3800m	86
Figure 5.10	Comparison of LSTM using MAE with [4], [5]	87
Figure 5.11	LSTM MAPE based comparison with [6], [7]	88
Figure 5.12	GRU MAPE and MAE Comparison With Other Studies: [5], [8]	89
Figure 5.13	TNN2 performance Comparison With [9]	89
Figure 5.14	Dataset with Multi Class Hazard Information - Shown as horizontal line from the top	91
Figure 5.15	Test Data For Model	91

Figure 5.16	Test Data For Model Zoomed to February the 24 2008 and April the 13th 2008	91
Figure 5.17	Training Accuracy for DH Model Using TNN	92
Figure 5.18	Predicted Data Using TNN	92

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
ARIMA	Autoregressive Integrated Moving Average
BP	Back Propagation
CSA	Coupled with Simulated Annealing
DDR	Daily Drilling Report
DH	Drilling Hazard
DROP	Drilling Rate of Penetration
GRU	Gated Recurrent Unit
LC	Lost Circulation
LSTM	Long Short Term Memory
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
ML	Machine Learning
MLP	Multi-Layer Perceptron
MNR	Multiple Nonlinear Regression
MON-MLR	Monotone Multi-Layer Perceptron

MSE	Mean Square Error
NN	Neural Network
NPT	Non Productive Time
R^2	Coefficient of Determination
RF	Random Forest
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
RPM	Rotation Per Minute
SARIMA	Seasonal Autoregressive Integrated Moving Average
SVR	Support Vector Regression
TNN	Transformers Neural Network
WOB	Weight On Bit

THESIS ABSTRACT

NAME: Basirudin Rachman Bachtiar Djamaluddin

TITLE OF STUDY: Multivariate Time Series Prediction of Drilling Rate and Hazards

MAJOR FIELD: Computer Science

DATE OF DEGREE: January 2024

Optimizing drilling operations in the oil and gas field is an important objective that can result in significant cost savings and decreased risks. Predicting the rate of penetration is a crucial factor in optimizing drilling operations. The penetration rate is determined by the physical process, which involves a combination of factors that exert influence on its outcomes. Leveraging these influential factors holds promise for refining the accuracy of penetration rate predictions during drilling operations, possibly by considering the problem as a multivariate time-series process, enabling better target predictions. Simultaneously, drilling operations are fraught with numerous potential hazards and failures, necessitating proactive identification to prevent undesired outcomes. Utilizing these features, enables the identification of potential drilling hazards or instances of failure. This study aims to compare deep learning models, specifically

those based on deep recurrent neural (RNN) networks and transformers to predict the penetration rate. Consequently, it uses the best model of penetration rate prediction to identify rare events or failures to mitigate hazards during drilling operations. The proposed methodology involves regression for the drilling rate penetration prediction and classification for the hazard detection on time series data. The efficacy of the proposed methodology will be assessed and evaluated, then compared with other existing methods in the literature to ascertain its superiority in drilling rate penetration prediction and hazard detection.

ملخص الرسالة

الاسم:

بصير الدين راشمان بختيار جمال الدين

عنوان الدراسة:

التنبؤ بالسلاسل الزمنية متعددة المتغيرات لمعدل الحفر و المخاطر

التخصص:

علوم الحاسب

تاريخ الدرجة العلمية: جمادى الثاني 1445

د تحسين عمليات الحفر في حقل النفط والغاز هدفًا مهمًا يمكن الى تقليل التكاليف وتخفيف المخاطر. يعد التنبؤ بمعدل الاختراق بدقة عاملاً حاسماً في تحسين عمليات الحفر، متأثراً بعوامل مختلفة. يتم تحديد معدل الاختراق من خلال العملية الفيزيائية، والتي تتضمن مجموعة من العوامل التي تؤثر على نتائجها. إن الاستفادة من هذه العوامل المؤثرة تبشر بتحسين دقة تنبؤات معدل الاختراق أثناء عمليات الحفر، قد يكون ذلك من خلال النظر في المشكلة باعتبارها عملية سلسلة زمنية متعددة المتغيرات، مما يتيح تنبؤات أفضل للهدف. وفي الوقت نفسه، فإن عمليات الحفر محفوفة بالعديد من المخاطر والإخفاقات المحتملة، مما يستلزم تحديداً استباقياً لمنع النتائج غير المرغوب فيها. يتيح استخدام هذه الميزات تحديد مخاطر الحفر المحتملة أو حالات الفشل. تهدف هذه الدراسة إلى مقارنة نماذج التعلم العميق، وتحديداً تلك القائمة على الشبكات والمحولات العصبية المتكررة العميقة (RNN)، للتنبؤ بمعدل الاختراق. وبالتالي فإنه يستخدم أفضل نموذج للتنبؤ بمعدل الاختراق لتحديد الأحداث النادرة أو الإخفاقات بهدف تخفيف المخاطر أثناء عمليات الحفر. تتضمن المنهجية الانحدار للتنبؤ باختراق معدل الحفر والتصنيف للكشف عن المخاطر على بيانات السلاسل الزمنية. سيتم تحديد وتقييم فعالية المنهجية المقترحة، ثم مقارنتها مع الدراسات السابقة للتأكد من تفوقها في تحسين عمليات الحفر والكشف عن المخاطر.

CHAPTER 1

INTRODUCTION

1.1 Introduction

In the oil and gas industry, the time to complete one cycle of drilling operation, from spud to target depth, is critical because it affects cost. Faster drilling means shorter time and thus lower cost. To achieve faster drilling, it is important to have drilling rate of penetration (DROP) prediction [10], [11]. There are many techniques to predict DROP, which can be classified into the following four classes [12]:

1. Mathematical models [13]–[17]
2. Statistical models [18]
3. Machine learning (data-driven) models, [19]–[25]
4. Hybrid models which combine mathematical and machine learning models [26]

The machine learning models are the latest and more advanced whereas the mathematical and the statistical solutions are basic.

Drilling operation is a continuous process with time bounds and can be treated as a time-series problem. During drilling, measurements, such as weight on bit (WOB), revolutions per minute for the cutting rate of bit, mud properties, must be collected. We can call these measurements as features or variables that can affect DROP. This implies that the prediction of future DROP is a multivariate time-series forecasting problem. The studies on DROP prediction using time series have gained popularity recently [6]–[8], [27]–[34].

Moreover, during drilling various hazards (DHs) can occur, where a drilling hazard (DH) is any event that deviates from the drilling plan. The causes of deviation can vary based on the condition of the drilling site, the formation of the underneath earth, or the drilling equipment. Similar to DROP, the various DHs can also be predicted using measurements taken during drilling operations.

Existing studies usually focus either on predicting a particular DH, such as stuck pipe, or on DROP; even though there is a close relationship between predicting DROP and predicting DH. For example, from the measurements collected during cutting removal process, both a value of DROP and a particular DH, such as stuck pipe, can be predicted. In the past, mathematical or statistical models used mathematical equations based on selected parameters [12]. However, these approaches use complex relationship between the parameters and require huge amount of data from the drilling site [35], [36]. The other limitations of the these approaches are their requirement of complex coefficients and their low prediction accuracy [37], [38]. On the other hand, data-driven techniques are the latest approach and do not require pre-selected models [12], [37]. They are based on machine learning techniques to predict DROP and

potential DHs. In this research, we are going to leverage the advances in intelligent solutions, such of deep learning, to propose a novel model that predicts an optimal DROP and identifies a particular DH.

1.2 Research Statement and Motivation

Optimizing drilling process by predicting DROP has attracted many researchers for many years. Most of the existing DROP predictors are based on the mathematical and statistical models. These models have many limitations because they are not dynamic and often depend on experimental results and not on empirical real conditions. These limitations led the research towards data-driven models which are more dynamic.

Like DROP, recent DH predictors are also based on data-driven models. However, there is no work done that suggests one type of model as the best model for both predictions. Furthermore, drilling process should be treated as a time series problem where the modelling should take time series into consideration for predicting both DROP and DH. There is no research done to use one type of model for the prediction of DROP and DH as time series problem, it is either for the prediction of DROP or one particular DH.

1.3 Thesis Research Objectives

This thesis aims at answering the following research questions (RQs):

1. RQ 1: What are the main features that can be used to predict DROP and a

particular DH?

2. RQ 2: If the drilling process is treated as a multivariate time-series problem, can deep-learning models give accurate predictions?
3. RQ 3: What type of deep recurrent neural network model that will give the best result as multivariate time series prediction solution for predicting DROP and DH?

Therefore, the research objectives are as follow:

- Conduct literature survey of existing solutions of DROP and DH predictors.
- Identify common features that can be used to predict DROP and DH.
- Design an efficient deep recurrent neural model that can predict DROP and DH.
- Evaluate the proposed model using benchmark datasets.

1.4 Thesis Organization

This thesis is organized into six chapters. Chapter 1 is an introduction to DROP and DH prediction. Chapter 2 gives background information. Related work is discussed in Chapter 3. Chapter 4 explains the details of the adopted research methodology. Results and analysis of the experiments are presented in Chapter 5. Conclusion and future work are discussed in Chapter 6

CHAPTER 2

BACKGROUND

This chapter presents background information related to the proposed work and consists of two main sections. Section 2.1 discusses the drilling operation and explains the drilling operation phases, processes, systems, and equipment. It also presents drilling cost calculation, features that affect DROP, and the relationship between DROP and drilling cost. Drilling hazards (DH), their causes, types, and the main features that affect DH are also discussed in section. Section 2.2 gives a brief introduction to time series, and Section 2.3 to machine learning, and the state-of-the-art deep learning models.

2.1 Drilling Operation

A drilling operation is a process of digging a hole from the surface of the earth to the reservoir of hydrocarbons underneath. This operation is complex and consists of several phases, systems, and equipment. It must be done efficiently and with utmost care to minimize cost and avoid issues that can cause disaster [39]. This section briefly

discusses the required systems, equipment, and cost parameters of drilling.

2.1.1 Drilling Operation Systems

In order to achieve its objectives, a drilling operation requires several systems, such as power and transmission system, hoisting system, rotary system, mud circulating system, well-control system and well monitoring system [1], [39]–[41]. Let us first discuss these systems briefly.

Power and Transmission System

In most of the modern rigs, power is generated by diesel engines [39]. The majority of the power is consumed by the hoisting, circulating, and the rotary systems [40]. The performance of a power system is measured in horsepower, torque, and fuel consumption [40]. The power required by shallow or moderate rigs is between 500 and 1000 horsepower (hp) whereas the one required by some modern rigs can reach up to 3000 hp [39], [40]. The generated power is used by the electric motors for drawworks, rotary table, and mud pumps [40]. It can be transmitted through [41]:

- Mechanical drive
- Direct-current (DC) generator and motor
- Alternating-current (AC) generator, silicon control rectifier (SCR) [39], and DC motor.

The mechanical drive is used in the early drilling rigs to give power to the drawworks and pumps. The DC generator has speed-torque suitable for hoisting and

circulating [41]. Modern rigs use internal combustion diesel engines and modern electric transmission which provide the driller with smoother power to avoid shock and vibration at the rigs [40].

Hoisting System

The objective of the hoisting system is to move the drill pipe up and down in the hole and to maintain the weight of the drill string at the bottom [39]. The hoisting system is also responsible to move the casing strings or other subsurface tools up or down the hole [1], [40], [41]. There are two main tasks that are the responsibilities of the hoisting system in drilling operations [1], [40], [41]:

1. Making connections, which is the process of adding new joint for the drill pipe when the hole deepens.
2. Pulling out the drill string from the hole to change the bottom hole assembly (BHA) and then put the string back to the bottom of the hole.

Figure 2.1 shows the hoisting system components: drawworks, block and tackle, and derrick [1], [41].

The drawwork consists of drum, brakes, transmission and catheads [1]. It provides the torque for moving the drill string (drill pipe) up or down. Block and tackle makes the handling of heavy loads easier and consists of crown block, traveling block, and drilling line, Figure 2.1 [1], [41]. The derrick is used to insert in or lift out a pipe in a hole. The higher the derrick, the longer the pipe that can be handled. The derrick can also handle sections known as stands. To support *blowout preventers*, a substructure

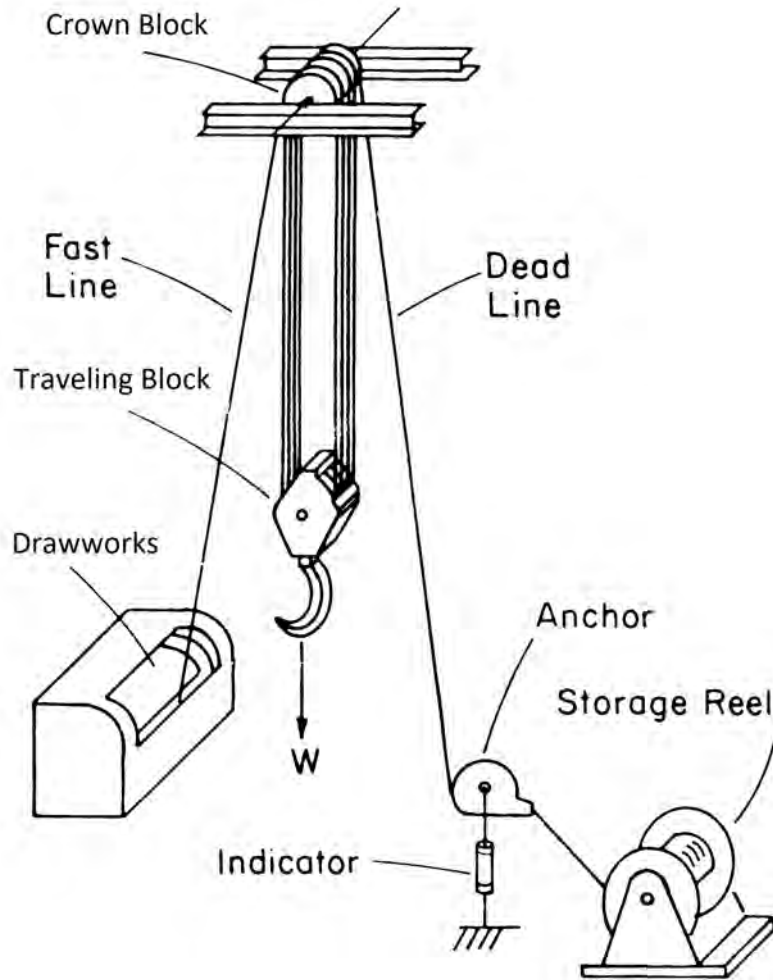


Figure 2.1: Hoisting system [1]

under rig floor supports the elevated derrick which is enough to support loads hooked to the pipe set in the derrick and the wind loads. Hoisting also requires additional equipment like elevators, hooks, and bails [41].

Rotary System

The job of the rotary system is to rotate the drill bit at the bottom hole in a continuous manner through the drill string [1], [39]–[41]. Power to the rotary system is transmitted using gear-driver to rotary table or is taken from drawworks [1], [39]. This power makes

the drill pipe and bit rotate to cut through the rocks formation at the bottom of the hole [40]. The rotary system consists of several equipment starting from the rotary swivel at the top to the drill bit at the bottom of the hole [39]. Figure 2.2 shows the rotary system with its equipment rotary swivel, kelly, rotary table, and drill pipe collars [1], [39].

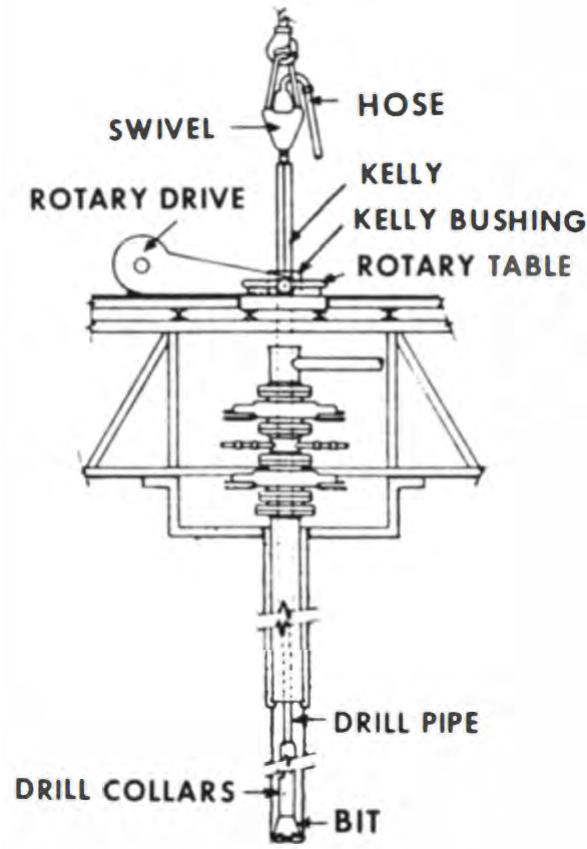


Figure 2.2: Rotary system [1]

The rotary swivel is the point where the circulation system is connected to the rotary system and acts as a seal of fluid. Swivel also supports the drill string and allows it to rotate. It attaches the hook of the traveling block and *gooseneck*, securing the rotary hose downward-pointing [1], [39], [41].

The kelly is where the drill string is attached and acts as the first section right below

the swivel. The torque required by the kelly is transmitted through *kelly bushings* which is inside the *rotary table*. The drill pipe is attached at the base of the kelly and under it there are drill collars and then drill bit. The drill collars can also be used to add weight on the bit or to assist the drilling process to drill a straight hole [1], [39], [41].

Circulating System

The circulating system circulates the drilling fluid or mud starting from the mud pit to the drill bit at the bottom of the drilled hole where it picks the cuttings up back to the shale shaker, sump pit and suction pit [39]. Its objective is to remove the rock cuttings from the drilled hole during the drilling process [1]. This system works as a hydraulic system for the circulation [41]. Figure 2.3 shows the main equipment of the circulating system: mud pumps, mud pits, mud mixer, and contaminant-remover [1], [40], [41].

Mud pump require high pressure, flow rate and horsepower [41]. The requirement for high pressure is to circulate heavy mud and to clean drilled holes. Mud pits are used at the surface to handle the excess volume of drilling fluid [41]. Mud-mixing equipment is used to mix the mud at the mixing hopper using the base fluid (water) and the bulk materials [39]. Contaminant-removal equipment servers to eliminate solids and gases from the mud [1].

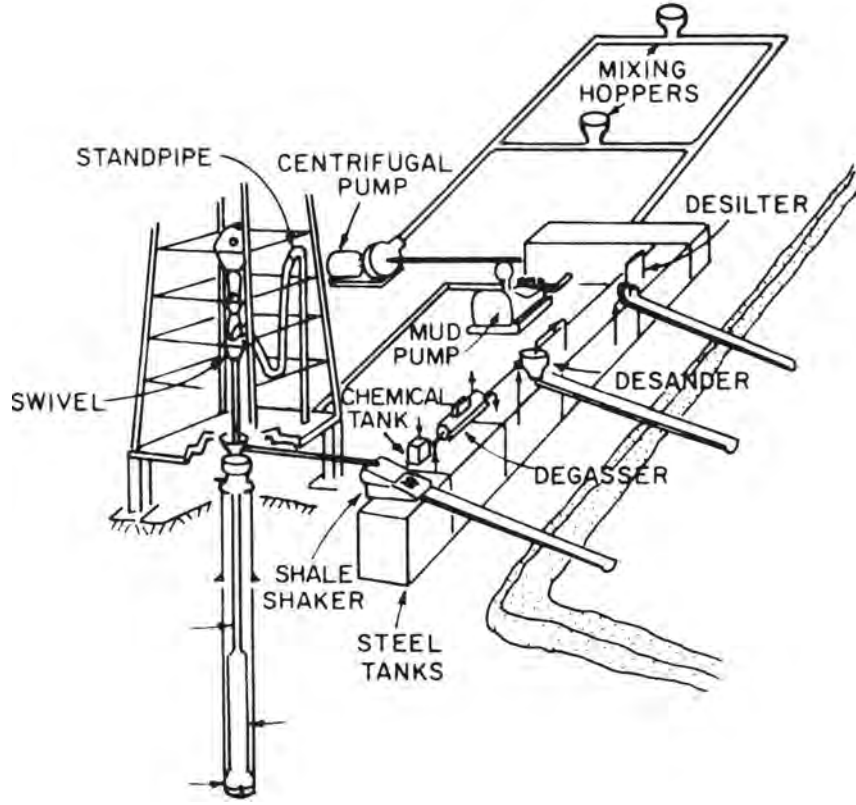


Figure 2.3: Circulating system [1]

Drilling Cost Parameters

The cost on drilling can be approximated from its fixed and variable expenses as follows [1]:

$$C_f = \frac{C_b + C_r(t_b + t_c + t_t)}{\Delta D} \quad (2.1)$$

where: C_f = drilled cost per unit depth

C_b = cost of bit

C_r = fixed operating cost of the rig per unit time

t_b = total rotating time during the bit run

t_c = non-rotating time during the bit run

t_t = trip time

ΔD = footage (depth) drilled with bit ($t_b \times DROP$)

2.1.2 Drilling Rate of Penetration (DROP)

Ahmed et al. [42] defines DROP as the speed of a drill bit to break rocks below it in order to deepen the hole during drilling. DROP affects drilling cost as shown in the aforementioned drilling cost formula. Predicting DROP minimizes drilling cost. Researchers have identified more than fifty parameters that affect DROP, and these parameters have complex relationships with each other. The selection of these parameters correctly is essential in order to accurately predict DROP and to reduce drilling cost.

Two of the main and controllable parameters that are frequently used by researchers to predict DROP are weight on bit (WOB) and bit rotary speed [6], [13]–[17], [19]–[21], [21], [25], [37], [43]–[51]. WOB is the weight that is applied by drill collars, which are thick and heavy part of the drill string directly connected to the drill bit. The value of WOB can be controlled and depends on the hookload from the surface which is important to achieve the DROP [1], [52]. Figure 2.4 shows the effect of WOB

on DROP. As can be seen from the figure, the increase in WOB until certain level increases DROP, but the increase of WOB beyond that level decreases DROP [1]. The rotational speed of drill bit, known as bit rotary speed, is measured as rotation per minute (RPM) [25]. Different types of cuttings require different speed, thus controlling the rotary speed is very important [53]. Low rotary speed values increase DROP linearly until certain point after which it will start declining as shown in Figure 2.5 [1].

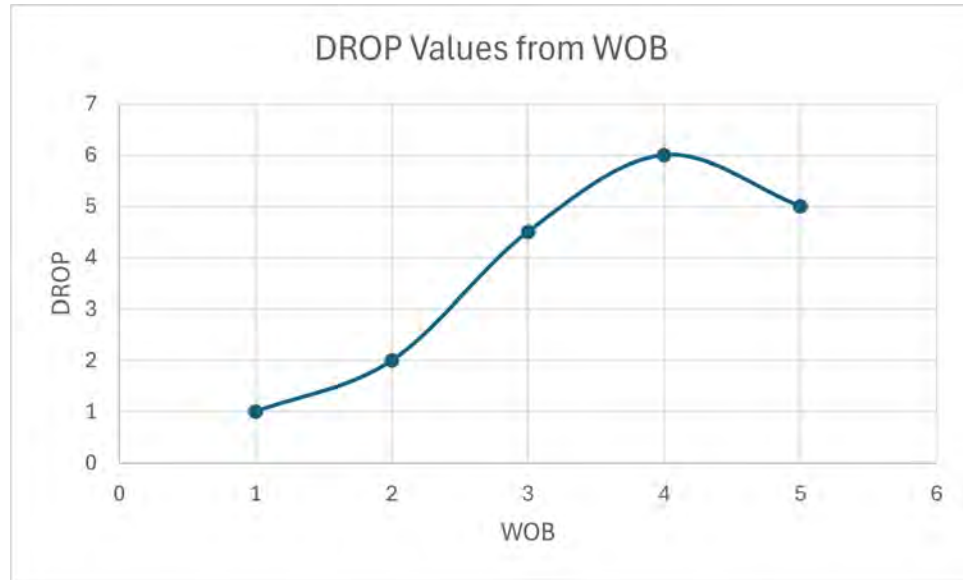


Figure 2.4: Relationship between DROP and WOB [1]

Beside the two major parameters, WOB and bit rotary speed, other three parameters that affect DROP and that are frequently used by researchers are the flow-rate of the mud pumps, the depth of the well, and the weight of mud [37], [41], [44]–[50].

Table 2.1 lists the seven main parameters that affect DROP, namely flow rate, measured depth, mud weight, plastic viscosity, standpipe pressure, and the two main parameters WOB and bit rotary speed. All these parameters except standpipe pres-

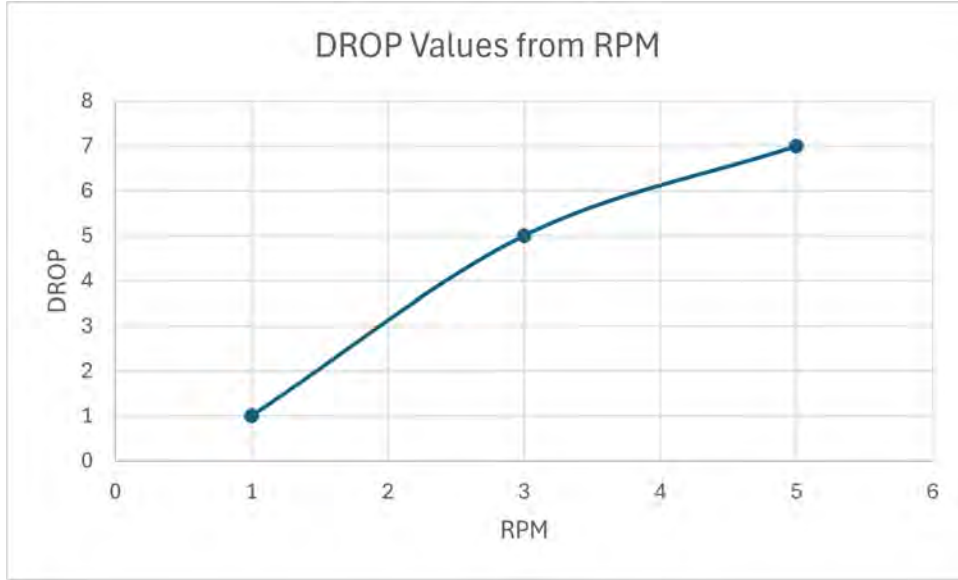


Figure 2.5: Relationship between DROP and bit rotary speed [1]

Table 2.1: Variable for DROP Prediction

Variable	References	Count
WOB	[6], [17], [19]–[21], [25], [37], [38], [43]–[50]	16
Bit Rotary Speed	[6], [17], [19]–[21], [25], [37], [43]–[50]	15
Flow rate	[20], [25], [37], [44]–[47], [49], [50]	9
Measured depth	[6], [20], [44], [45], [47]–[50]	8
Mud weight	[17], [19], [20], [38], [44], [50]	6
Plastic viscosity	[20], [21], [25], [38]	4
Standpipe pressure	[6], [25], [44], [49]	4

sure also affect the lost circulation, one of the most frequently occurring DHs [54]–[57].

2.1.3 Drilling Hazards

A drilling hazard (DH) is an incident that occurs during drilling operation and results in delaying or halting the operation. It can occur at any time while drilling a well and thus the drilling crew must be prepared [39]. A DH is an unforeseen event that causes the drilling operation to deviate from its planning course [58] or from the critical path

[59]. It is also referred to as "trouble" when it causes a nonproductive time (NPT) during drilling, which is a minor interruption, or more devastating occurrences like wellbore failures or loss of control during drilling operations. [58].

Natural conditions, such as geological complexity at the drilling site, are not the only potential causes of DH; unforeseen circumstances during the drilling operation, such as mechanical failure or human error, may also contribute [58]. Such occurrences can have significant repercussions on the drilling schedule, budget, completion time, and reputation of the companies involved [58]. For a recurring DH, a drilling crew can benefit from the historical data to predict and avoid such hazards [60].

According to Skalle [2], there are four major types of DHs, namely stuck pipe, lost circulation, gas kick, and bit balling. The two frequently occurring DHs are stuck pipe and lost circulation, 16% and 12.4% respectively. Figure 2.6 shows DH distribution with additional types beside the above mentioned.

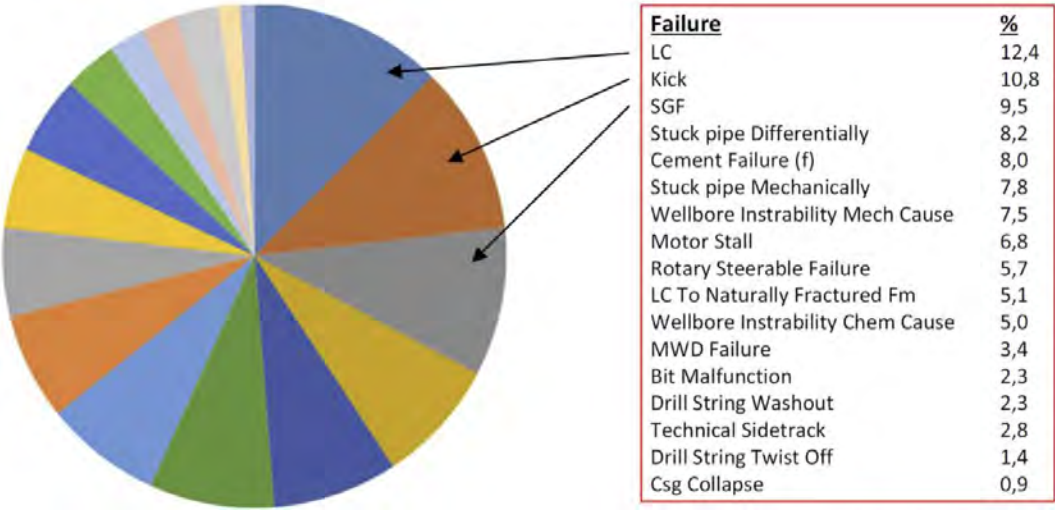


Figure 2.6: Distribution of DH types [2]

A more comprehensive approach to classifying DH involves associating unantici-

pated occurrences with the subsequent categories [58]:

- Heterogeneity in subsurface formations is the most prevalent DH encountered during drilling operations. It encompasses a range of issues such as stuck pipes, lost circulation, geological faults, contamination of formations, hydrogen sulfide (H₂S) or other gases, and buried valleys.
- The complexity of the drilling mechanism and equipment poses challenges in its utilization, requiring correct operation and maintenance to prevent any failures.
- Risks associated with the well site must be accounted for in the drilling plan and assessment.

Stuck Pipe

Stuck pipe is defined as the condition where a drill string is stuck and unable to move vertically or rotate during drilling operations [61]. In Oil and Gas industry, stuck pipe is categorized as one of non-productive time (NPT) due to the downtime it causes. According to Yarim [61], stuck pipe is one of the major DH problems which causes drilling down time and high cost. It also incurs costs for restoring the condition back to normal so that drilling can resume. There are many events that can result in stuck pipe. Tripping and backreaming have the highest possibility of a stuck pipe events, from 54% to 90%. Other reasons include differential sticking 21% to 60%, poor hole cleaning 65%, and mechanical and wellbore geometry related 14% [61]. Stuck pipe can also happen during drilling into rhyolite magma, due to the challenges of geothermal well (high temperature well). For example, drilling close to the magma increases

viscosity which makes releasing a stuck pipe difficult [62].

Lost Circulation

Normal flow of mud in circulation is to move up to the surface from the bottom of the drilling hole. A situation where the mud that is supposed to move towards the surface but instead remains in the hole is known as lost circulation (LC) [60]. The loss of mud can be significant and continuous [41]. The condition where no mud flows to the surface is known as total LC, otherwise it is a partial LC [60].

LC is one of the critical DH which can cause huge financial loss. The need to predict whether LC will occur is important for the sustainability of the drilling operation. As a result, LC prediction has attracted a lot of research for decades, from the detection of return flow rate , pit volume and stand pipe pressure [63] until the modern way using machine learning [64].

Kick

A kick is defined as an unexpected and unwanted flow of reservoir fluid or gas into the wellbore. Gas kicks are riskier than fluid kicks because of their high mobility in the wellbore. During drilling operation, a gas kick can happen if the rate of gas flow forced to the wellbore suddenly increases causing a blowout [65]. Gas kick can also be triggered because of mud loss – the condition where the formation porosity increases. These conditions, gas kick and mud loss, must be detected at the earliest in order to ensure the continuity of drilling operation. There are a number of proposed techniques to detect gas kick, but they have limitations, such as the generation of false alarms or

the requirement of expensive resources [65].

Bit Balling

During drilling, the soft and sticky cuttings absorb water from the drilling fluid, swell, and stick to the drilling bit. Bit balling occurs when the sticky solids cling to the metal surfaces of the bit and pipe. Bit balling happens frequently when drilling in sticky shales or loose sandstone. It can reduce DROP and increase standpipe pressure.

The most frequently occurring DH is stuck pipe, hence on this work, we will study the features that are related to this hazard, such as equivalent circulating density, torque, hook load, pump pressure, standpipe pressure, DROP, WOB, drilling fluid flow-in rate and RPM [66], [67].

2.2 Time Series

A time series refers to a collection of data points that are obtained by measuring a process or set of processes over a specific period of time; these data points are recorded and preserved, providing observations on how the processes change over time [68]–[70].

Examining the distinct attributes of a specific subject inside a time series dataset is crucial for comprehending patterns and making accurate predictions about future trends [68], [69]. The measurement used for analyzing and understanding patterns is determined by a specific time interval, such as seconds, minutes, hours, days, months, years, or any other unit of time [69], [70].

2.2.1 Time-Series Data

There are two primary categories of time-series data: *classical time series* and *point processes* [68]. When gathering data, if only one variable is being measured, it is referred to as a *univariate* time series; however, if there are multiple variables being measured, it is called a *multivariate* time series [69].

The definition and example of the first two types of time-series data are as follows:

- *Classical time series*: this sort of time-series data captures one or several numerical measurements at a specific moment in time and is recorded at regular intervals, such as every second, minute, hour, day, or longer intervals. The classical time-series data examples include stock market data, energy consumption data, and various measurements during drilling operations such as weight on bit (WOB), rotation per minute (RPM), mud flow, and other measurement while drilling (MWD) data.
- *Point processes*: this type of time-series data does not represent a measurement of specific data, but rather records an occurrence that takes place at a specific moment. The timestamp of the event occurrence may not be evenly distributed, and the value or category will also be captured during the event recording. Examples of this type of data include: the activity of a specific user on a website, the occurrence of natural disasters, or incidents in drilling operations (DH).

2.2.2 Time-Series Analysis

The process of analyzing time-series data to extract unseen patterns and to get valu-

able insight through mathematical and statistical analysis is known as *time-series analysis* [69]. This process will produce essential information of the time-series data such as trend, seasonality and cyclic variations [70]. From the time-series data, using time-series analysis, the following tasks can be resolved [70]:

- Find relevant part of the time-series data in the pre-process task by extracting features from the dataset.
- Once the relevant parts are found then they will be used to draw insight from the historical dataset.
- Part of the time-series analysis is to represent the time-series data in the appropriate format, i.e. visualize in graphical format or tabular format.

2.2.3 Time-Series Prediction

There are two types of time-series prediction [68]:

- *Time-series forecasting* is used to predict future time-series value. The prediction is done based on the historical value measurements. This prediction is applied on the classical time series like predicting next hour temperature in the current city. It can also predict the time of the event in the future, which is categorized as point-process prediction, like predicting the date of next flooding.
- *Time-series classification or regression* will classify or predict numerical value (regression) not only for one data point in the future but a set of data points.

So far this has been dominated by ML approach.

The tasks for time-series prediction consist of the following phases [70]:

- Model development: in the case of time-series classification or regression, it will be ML models.
- Model execution: using the developed model then it can be used to forecast the future prediction.

Prior to the aforementioned tasks, the initial step to be done is the *time-series analysis* for the time-series dataset preparation and analysis.

2.2.4 Time-Series Characteristics

A set of data can be recognized as a time-series data if it has *time-series characteristics*.

There are several characteristics for a time-series dataset [69], [70]:

- *Random walk*: the basic and simplest characteristic for a time-series data. It is formed by the summation of set of random variables with normal distribution E_i where i : 1, 2, ... N. The formula for random walk is:

$$R_{t+1} = R_t + E_t$$

where t is the index of when measurement was taken. Fundamentally, a random walk is stochastic and by default is a part of time-series dataset.

- *Trend*: the pattern or direction of a time-series data for some duration of time. It is either increasing, known as uptrend, or decreasing, known as downtrend.

The formula for a linear trend is:

$$T_t = A + B \cdot t + E_t$$

where A and B are constant, B is multiplied with the time index and E_t is error for each index. For non-linear trend the formula is:

$$T_t = A + F_t + E_t$$

where F_t is a logarithmic function. Figure 2.7 shows the time series with non linear trend.

- *Seasonality*: a periodical variations that happen repetitively with identical pattern within a certain time in the time-series data. For example, weather information in specific season or sales during holidays season.
- *Stationary*: it is not a time series with trend or seasonality. It doesn't have clear pattern that can be categorized into the aforementioned two characteristics (trend or seasonality). The stationary condition is when there is no variation of statistical properties in the time-series data with respect to the time [69]. A *stationary* time-series data has constant statistic properties such as: mean, variance and auto-correlation.

2.2.5 Time-series Analysis Problems

There are three time-series analysis problems [70]:

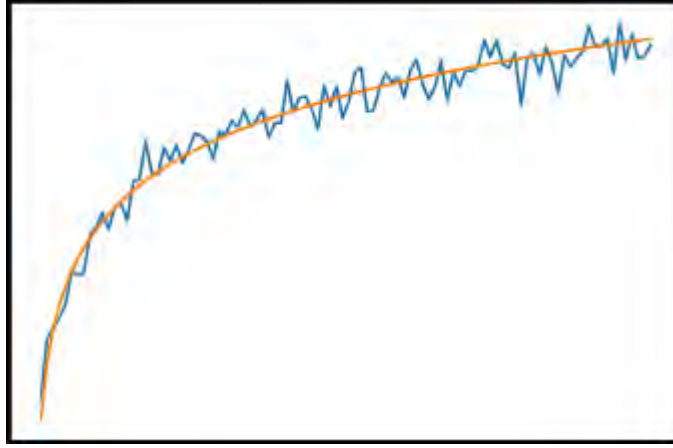


Figure 2.7: Non linear trend in time series

- *Forecasting*: a task to make an approximation of future values of a time-series data. It will give approximation instead of exact value of the forecasted variable.
- *Modelling*: a task to find logical understanding of the model that defines measured time series. Part of the modelling is model training which requires the time-series data in *stationary* format [69].
- *Anomaly detection*: a task to find abnormal pattern in time series which is not expected to be part of the time-series data.

There are several classical methods for the time-series forecasting such as Autoregressive Integrated Moving Average (ARIMA) and Seasonal ARIMA (SARIMA) known as a linear approaches [69], [70], however they have limitations such as [70]:

- Limited on linear problem and not for the complex one.
- Unable to do pre-processing for the features and static observations for lag.
- Unable to support missing data.
- Only support one step prediction.

2.2.6 Univariate vs Multivariate Time Series

The linear or classical time series such as ARIMA or SARIMA are limited to a univariate type of time series data. Real world problem normally consists of multiple data series, known as *multivariate* time-series data [70].

The relationship between the number of input features and output can be divided into the following [70]:

- Univariate input to univariate output: The classical and linear are classified in this type such as ARIMA and SARIMA. This is a straightforward mapping from a set of inputs to a set of outputs.
- Multivariate input to univariate output: This mapping fits with the ML model of deep learning which supports multiple feature inputs for the model.
- Multivariate input to multivariate output: While this is the most complex relationship, it consists of two ways of this relationship can be done. *Many to many* approach uses one model that uses all the features and produce the prediction for all features too as shown in Figure 2.8. Another approach is *many to one* on which there will be n number of models as the number of features and each of them will forecast one feature only. Figure 2.9 shows the multivariate input and output with many to one process.

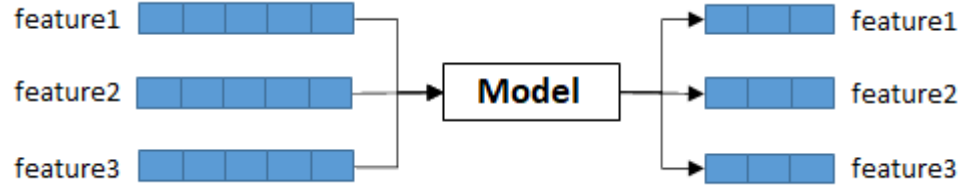


Figure 2.8: Multivariate many input to many output

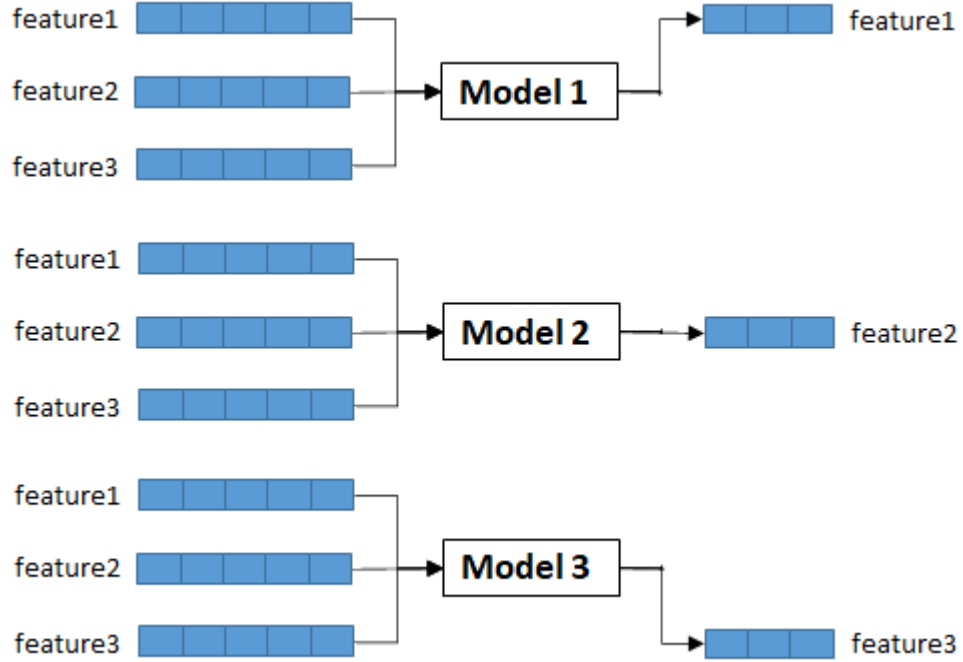


Figure 2.9: Multivariate input to multivariate output with different model for each output

2.2.7 Single-Step vs Multi-Step (Horizon-Style)

From the perspective of forecast data preparation, there are two types of number of steps to be predicted [69], [70]:

- Single-step technique uses a set of time periods (i.e. hours) to generate prediction of the subsequent value (the value of next hour in the case of hour period window). Figure 2.10 shows the single-step technique to predict the next value of s_{t+1} by using previous six hours window.

- Multi-step technique utilizes the same inputs as single-step to predict multiple steps of output, i.e. for the hourly steps case, it is to predict outputs for the next 3 hours. Figure 2.11 shows the technique of multi-step.

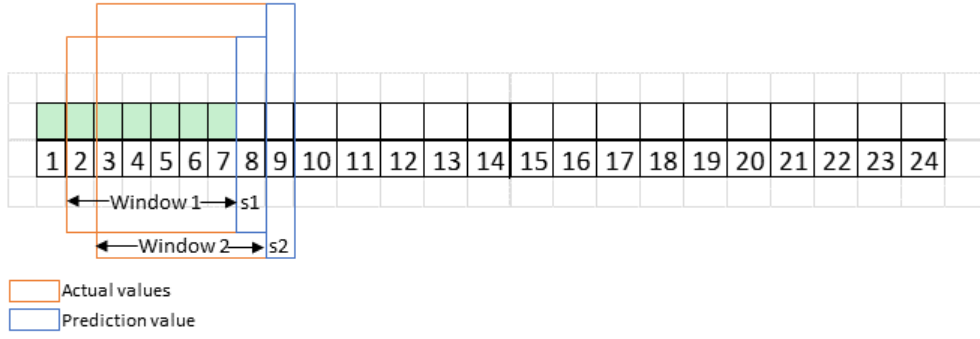


Figure 2.10: Single-step technique for hourly periods

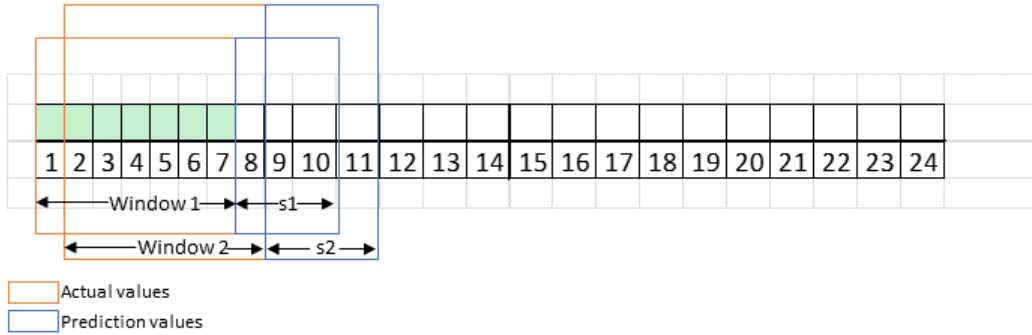


Figure 2.11: Multi-step technique for hourly periods

2.3 Machine Learning (ML) Approach

Machine learning is a field of artificial intelligence that enables computers to extract models from datasets and then use these models for prediction, without being explicitly programmed for each task. A ML model is not a predefined function but built and refined through many cycles of error minimization known as training which is done

using historical data [71]. Brink [68] defines the basic of ML workflow as in Figure 2.12.

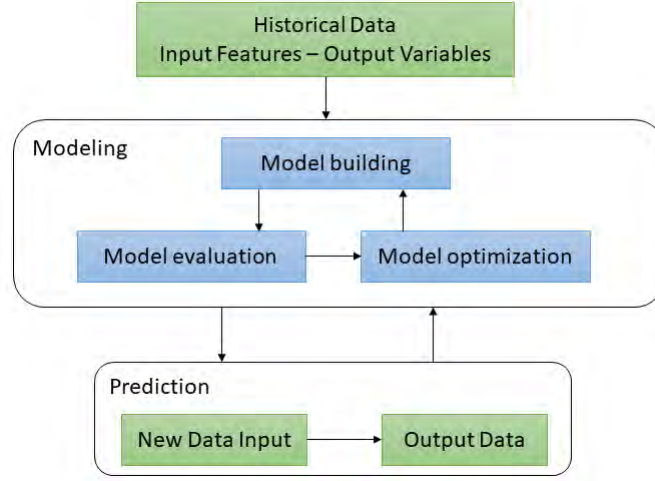


Figure 2.12: ML Workflow

2.3.1 Workflow of Machine Learning Approach

The general workflow of the ML approach contains following steps:

1. Data acquisition and preprocessing by collecting data from the input source, like daily drilling reports, with additional from reservoir and geology [20], [21].
2. Part of preprocessing is to normalize input data [9], [21], [50] and do quality check [44] which can use specific method like Savitzky-Golay (SG) method [46] or several steps like what have been done by [72]: data clustering using K-means algorithm and outlier detection for improving data quality. Mehrad et al. [49] uses Turkey method to remove outliers.
3. Important variables (features) selection, the process to select important features

through feature ranking process [38]. It is also called as feature engineering [37].

Several known methods are:

- Mutual information analysis method [47].
 - Non-dominated sorting genetic algorithm II (NSGA-II) [49]
4. Training, validation and testing use different portions of a dataset. This process is normally done to find the best number of layers and neurons for input, hidden and output layers.
 5. Modeling, various modelling implementations such as:
 - Single model [73], [74]
 - Comparison of several ML models, such as RF, cubist, monotone multi-layer perceptron (MON-MLP) NN [38]; support vector regression (SVR), gradient boosting machine (GBM), NN, K-Nearest Neighbor (KNN), RF, Boosting [75]; ANN, extreme learning machine, SVR and least-square SVR (LS-SVR) [44]. While [11] compared eight ML approaches: RF/Decision Tree (DT), ANN/DL, Recurrent Neural Network (RNN), Multivariate Linear Regression (MLR), LASSO Regression, Ridge Regression, Principle Component Analysis (PCA) and Spline Regression.
 - Comparison of ML models in different approaches: [17], [23], [25], [37], [50].

And more complex comparison as follows:

- Several ML models: linear regression classification and regression tree (CART), RF, GBM has also been compared with several deterministic

models where a hybrid of mathematical Bingham model shown the lowest relative error compared with all [72]. Although this is used when a trust-region mathematical tool method is applied.

- Statistical approach of multiple linear regression (MLR) compared with various forms of SVR to proof that SVR is has better accuracy in the condition of insufficient data [48]
- ANN was compared with other ML approach and showed better performance [44], [76], [77].
- Hybrid methods of ML models with other models can be applied as follows:
 - Combination of three deterministic models: Bingham, BY and Motahari which ensembles using ML model shows that Stacking alogrithm outperformed the other five ML models: mean, regression, bagging, RF, KNN [10]
 - ANN of MLP or radial basis function (RBF-NN) combined with evolutionary algorithms: GA, particle swarm optimization (PSO), biogeography-based optimizer (BBO) or imperialist competitive algorithm (ICA) [46]
 - An improved bat algorithm called iterative local search and stochastic inertia weight (ILSSIWBA) is combined with SVR (SVR-ILSSIWBA) [47].
 - ML estimation tool of SVR, least-squares SVM (LSSVM), and MLP combined with optimization tool for each with: Cuckoo optimization

algorithm (COA), PSO and GA [49]

– Adaptive ANN which was using self-adaptive differential evolution (SaDE) algorithm outperformed four other mathematical models [76]

- Combination of several ML models which is called as stack ensemble has done by [9] which divided the training into two stacks each stack is a single transformer network (STN) which then this whole stack is called as ensemble transformer network (ETN).

6. Evaluation, to find the best result, which normally use the coefficient of determination R^2 which shows that the ML model is more efficient than mathematical models [17]

Based on the relationship between the input features and output result, ML algorithms can be divided into two categories, namely *supervised learning* and *unsupervised learning* [68], [78]–[80]. Awad and Khanna [71] classify more types of ML approaches such as *semi-supervised learning*, *reinforcement learning* (RL), *transductive learning* and *inductive learning*.

2.3.2 Supervised Learning

Supervised learning is a type of ML where the target value of each training data is known. So during training the ML algorithm tries to find the best possible model that maps the input data to the target data. The learned model can then be used to predict target values of similar datasets [78].

There are two types of supervised learning: classification and regression. In clas-

sification the target output is categorical whereas in regression it is numeric [71], [79].

Figure 2.13 shows the workflow of supervised learning which is modified from the diagram proposed by [71] with additional testing data information.

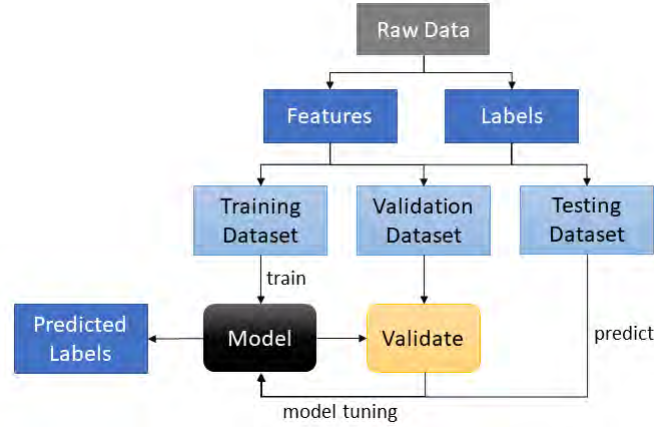


Figure 2.13: Supervised Learning Workflow

An example of regression is to predict the stock price on day t based on the stock prices on days $t - 1, t - 2$ and $t - 3$ [78]; and an example of classification is to predict species of a bird based on the weight, wingspan, and back color of the bird [79].

Classification

As mentioned above, classification models are used to predict categorical values. Classification can be *multiple* or *binary class* classification [68]. For example, predicting type of hazards which can be stuck pipe, lost circulation or other hazards is a *multiple class* classification, whereas classifying whether there will be a DH happening or not is a *binary class* classification. Figures 2.15 and 2.14 depict the classification procedures for multiple and binary classification, respectively. Both workflows utilize the same input data features but yield different types of classification results.

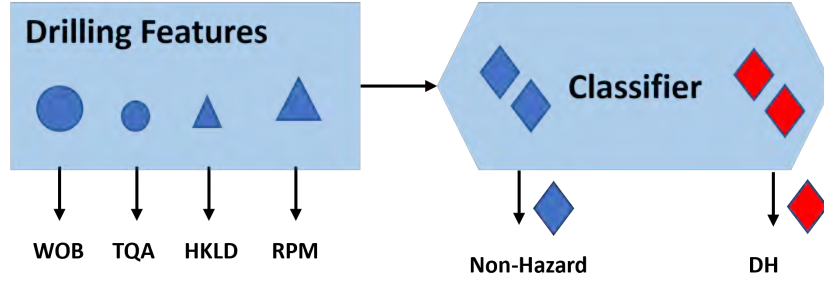


Figure 2.14: DH as Binary Class Classification Illustration

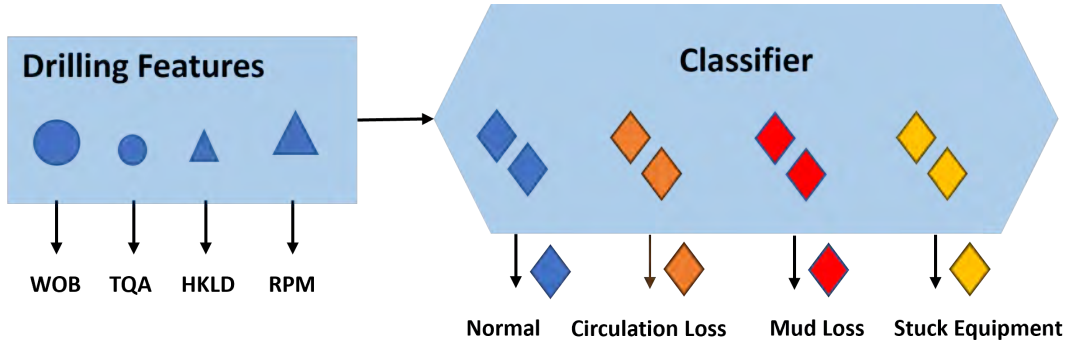


Figure 2.15: DH as Multi Class Classification Illustration

Regression

Unlike classification which is used to predicting limited possible outcomes, regression is used for the prediction of unlimited number of numeric values [68], [80]. Regression uses *regressor* as a model for to predict a target numeric value. According to Brandewinder [80], there are two aspects that differentiates a regression model from a classification model:

- The prediction output is actual numerical values, the range of the values is wide and comparable.
- Point of interest is the relationship between the features and the prediction value, where the changes of features can affect the prediction values.

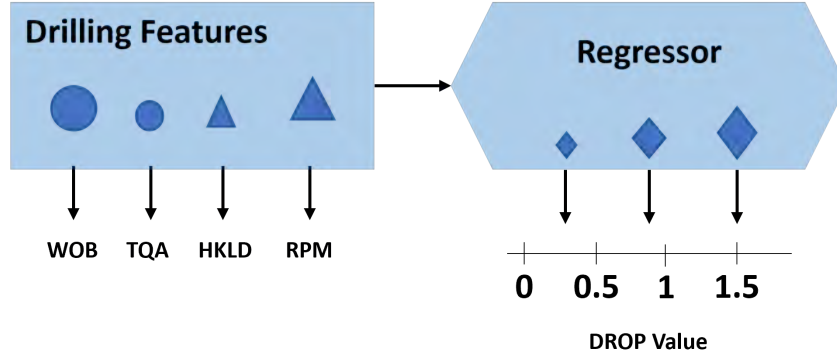


Figure 2.16: DROP Regression Illustration

The prediction of DROP in drilling operations based on the feature inputs of WOB, bit rotary speed, flow rate, measured depth, mud weight and standpipe pressure is an example of a regression model. In this case, the prediction output is a numerical value of DROP where the changes of the feature values can affect the prediction DROP value. Figure 2.16 shows how different features in drilling operations (symbolised with different shapes) through a *regression* model will produce the prediction of target numerical value as DROP value.

2.3.3 Deep Learning

Deep learning (DL) is a type of ML that is able to detect complex pattern between the input, and outputs on which the classical ML algorithms are unable to do [70]. This includes:

- Capability to learn from any complex mappings or patterns between the inputs and outputs.
- Support for multivariate time-series data (inputs or outputs)

- Ability to detect patterns that are scattered through the long series in dataset.

Nowadays, deep learning algorithms (DL) are mainly used for time-series analysis. The notable techniques employed are Convolutional Neural Network (CNN), Recurrent Neural Network (RNN) with its enhancements like Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), and the cutting-edge technique called Transformer [9], [12], [71], [81]–[84].

RNN has two significant advantages over other neural network techniques, such as fully connected neural networks (FNN) as follows [70]:

- RNN allows for a variable number of input sequences, unlike FNN which has a set or limited input sequence length. This makes RNN well-suited for analyzing time-series data.
- The order of sequence is crucial: FNN does not discern the pattern of an input sequence; it only represents a collection of numbers. Nevertheless, Recurrent Neural Networks (RNNs) retain a concealed value following each input processing, which serves as an internal memory to give an accurate prediction for the future values.

A RNN architecture can be divided into three layers: input layer, hidden layer, and output layers. Figure 2.17 illustrates an unfolded RNN architecture where each specific part, such as the input's weight w_i , the hidden's weight w_h , and the output's weights w_o , each different values for each input, hidden and output will be same across the whole sequence. Following equations are showing the update of the hidden (internal) state h and the output y at the time step t [84]:

$$h_t = f(W_i(x_t + b_i) + W_h(h_{t-1} + b_h))$$

$$y_t = g(W_o(h_t + b_o))$$

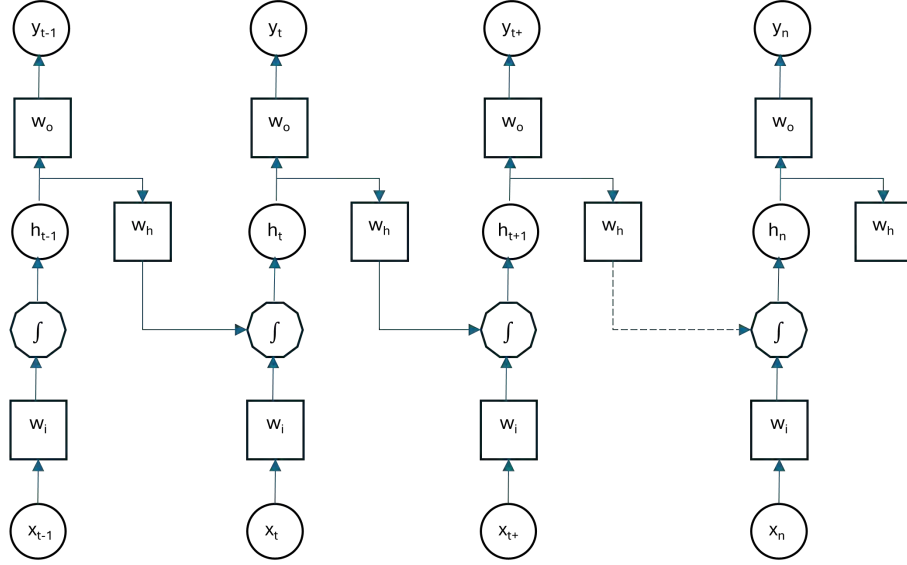


Figure 2.17: An unfolded RNN architecture

The issues of vanishing and exploding gradients were identified in recurrent neural networks (RNN) [81], [82] despite the previously noted benefits [70]. When the length of the unfolded RNN increases and the value of the weight w_h is less than 1, the input value is multiplied by the weight's value raised to the power of the network's length. This multiplication results in a value near to 0, which is known as the vanishing gradients problem. On the other hand, if the value of w_h is greater than 1, it will result in the issue of exploding gradients. These two conditions complicate the process of determining the optimum loss function, as it can result in either too small or excessively big steps. Long short-term memory (LSTM) and Gated Recurrent Unit (GRU) are types of RNN that were specifically designed to address the issue of vanishing and

exploding gradients [84].

Long Short-Term Memory (LSTM)

LSTM works by using an internal memory cell on which there are gates that will control the effect from the previous cell state, how the current cell will be updated and how the output will be generated [82], [84]. The gates are *forget*, *update* and *output*, respectively [84]. Figure 2.18 shows the LSTM architecture [84] where the external input x_t , the prior output y_{t-1} and the prior cell state h_{t-1} influence each gate (sigmoid function) and operator (hyperbolic tangents function) as follows:

- *forget* gate g_f is the result of activation on the W_fx_t, R_fy_{t-1} with bias b_f
- *operator* o_1 will generate candidate of the next cell state $\tilde{h}_t$ based on the W_hx_t, R_hy_{t-1} with bias b_h
- *update* gate g_u is the result of activation on the $W_ux_t, R_u y_{t-1}$ with bias b_u
- the next cell state h_t is the result of linear operation between $\tilde{h}_t$ modified by g_u and h_{t-1} modified by g_f . In this case if the g_f is 1 and g_u is 0 then the result of the current state h_t will be the same as h_{t-1} therefore the issue of *vanishing gradient* does not happen (as the objective of LSTM)
- *output* gate g_o is the result of activation on the $W_o x_t, R_o y_{t-1}$ with bias b_o
- then as the final output is the result of linear operation between g_o and the operator o_2 as y_t

The equations for each previously described processes of LSTM are as follows:

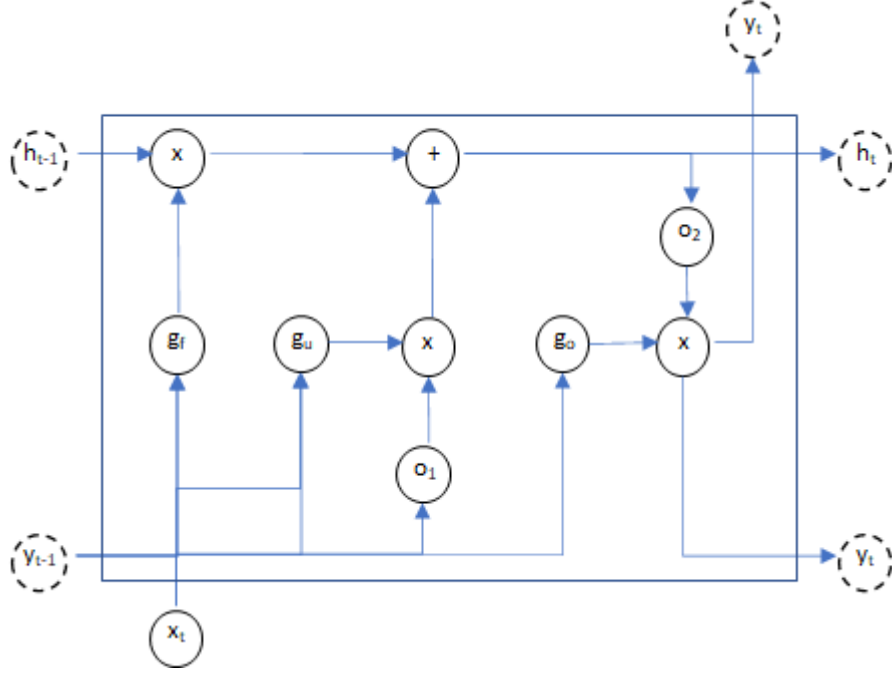


Figure 2.18: LSTM architecture

$$g_f = \sigma(W_f x_t + R_f y_{t-1} + b_f),$$

$$\tilde{h}_t = o_1(W_h x_t + R_h y_{t-1} + b_h),$$

$$g_u = \sigma(W_u x_t + R_u y_{t-1} + b_u),$$

$$h_t = g_u \odot \tilde{h}_t + g_f \odot h_{t-1},$$

$$g_o = \sigma(W_o x_t + R_o y_{t-1} + b_o),$$

$$y_t = g_o \odot o_2(h_t).$$

Gated Recurrent Units (GRUs)

A simplified LSTM called GRU works without using cell states explicitly, and unlike LSTM that has two gates to control hidden state, GRU has only one gate which is reset

gate [81]. Figure 2.19 shows the GRU architecture [84] with following explanation:

- *reset* gate g_r is the result of activation on the $W_r h_{t-1}$, $R_r x_t$ with bias b_r
- *current* state h'_{t-1} is the result of linear operation between h_{t-1} and g_r
- *operator* o will generate candidate of the next hidden state z_t as the result of activation on the $W_z h'_{t-1}$, $R_z x_t$ with bias b_z
- *update* gate g_u is the result of activation on the $W_u h_{t-1}$, $R_u x_t$ with bias b_u
- then as the final output is the result of linear operation between (1 minus g_u linear operation with h_{t-1}) and the result of linear operation between operator o and g_u as new state h_t

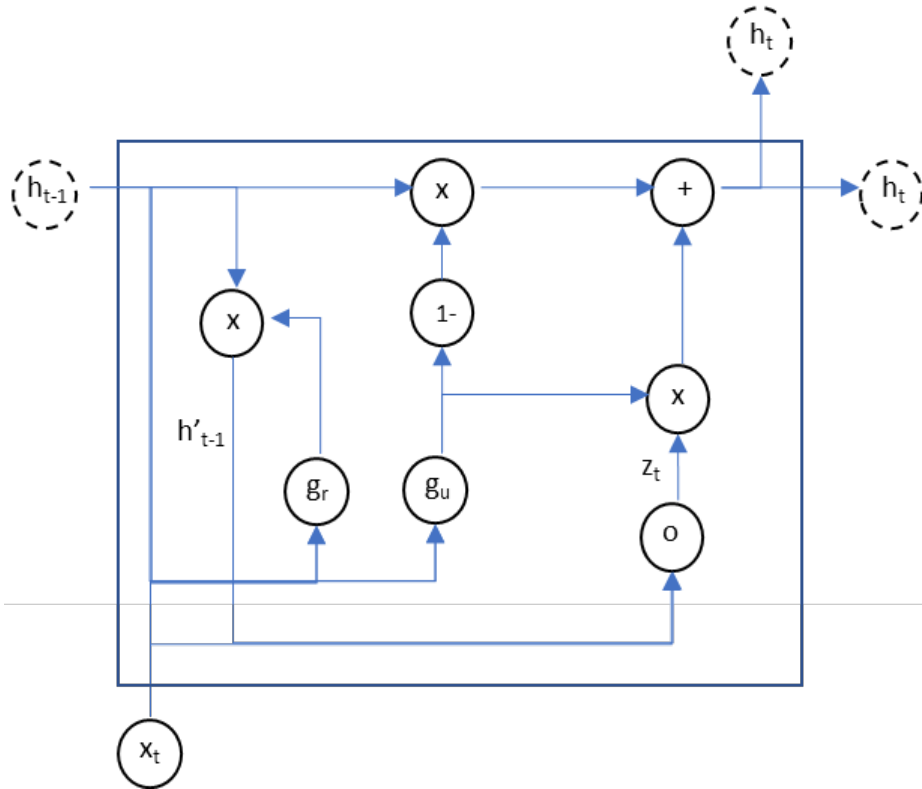


Figure 2.19: GRU architecture

The equations for each previously described processes of GRU are as follows:

$$g_r = \sigma(W_r h_{t-1} + R_r x_t + b_r),$$

$$h'_t = h_{t-1} \odot g_r,$$

$$z_t = o(W_z h'_{t-1} + R_z x_t + b_z),$$

$$g_u = \sigma(W_u h_{t-1} + R_u x_t + b_u),$$

$$h_t = (1 - g_u) \odot h_{t-1} + g_u \odot z_t.$$

Transformer Neural Networks

Transformers uses attention encoder-decoder approach but without the recurrence process [85] with the following architecture:

- Encoder and Decoder, both have similar number of stack, $N = 6$, with each layer has multi-head self-attention part and a position-wised fully connected FFN. In decoder, there is one layer inserted between the two layers which will do multi-head self-attention on the output from the encoder. While the self-attention in the decoder is modified to make sure that the prediction on the i^{th} will only depend on the output from the position prior to it.
- Attention Model, in Transformer, is a mapping of query (Q) to set of key (K) value (V) pairs. This attention model is called Scaled Dot-Product Attention

(SDPA), and is computed as:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

The multi-head layer in the encoder or decoder is the concatenation of the SDPAs which is projected h times in parallel that will produce d_v -dimensional output values.

$$Multihead(Q, K, V) = Concat(head_1, \dots, head_h)W^O$$

$$\text{where } head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

- Additional Components, the other components are as follows:

1. Position-wide FFN, in both encoder and decoder:

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2$$

2. Embeddings and Softmax, are use to convert the input and output tokens to vectors that have dimension d_{model} and to convert the output from decoded for the next token probabilities prediction respectively.
3. Positional encoding, the information about tokens position in the sequence that being added before the inputs or outputs enter the encoder and decoder respectively.

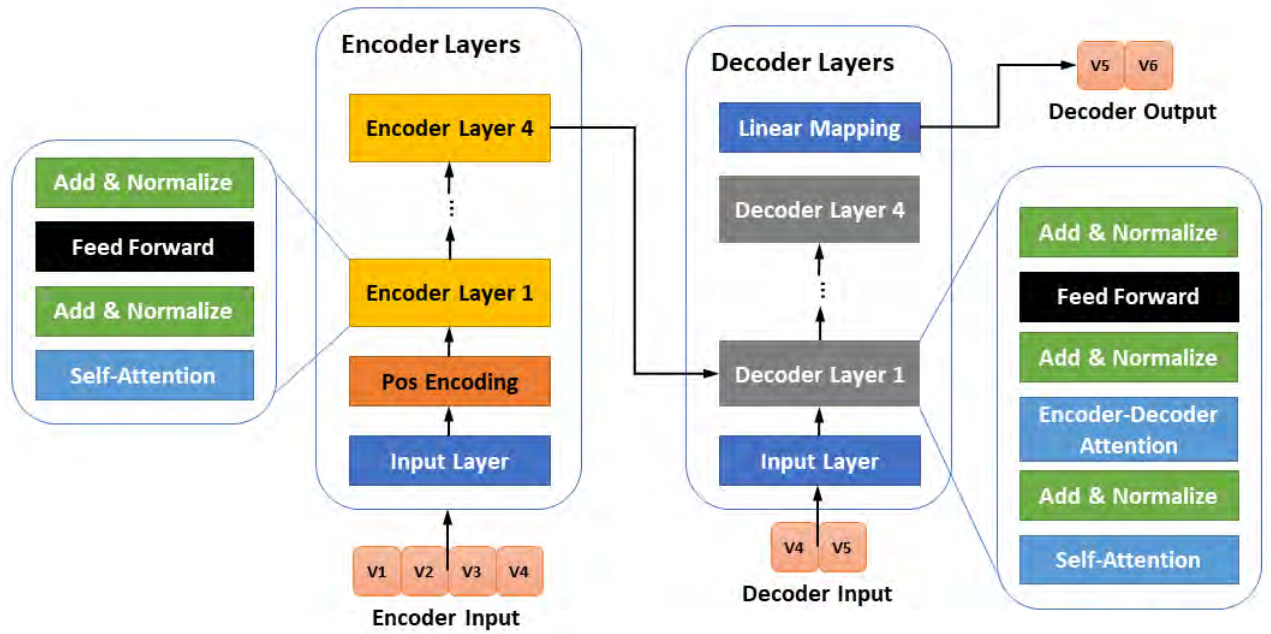


Figure 2.20: Transformer Model Architecture

Figure 2.20 shows the architecture of the Transformer model.

CHAPTER 3

RELATED WORK

Research on DROP and DH predictions have been ongoing for over a decade. Prior to the data-driven approach, mathematical/analytical and statistical methods were employed to predict DROP and DH. Recently, data-driven techniques, such as machine learning, have gained popularity, and they perform better than other methods. This chapter consists of two sections. The first section discusses the various techniques to predict DROP and the second section discusses the techniques to predict DH.

3.1 DROP Prediction Techniques

A drilling operation carried out without careful quantitative calculation of the interrelated features obtained from several different individual operations can result in a significant increase in cost [14]. The relationship between these features must be studied because they affect the speed of the drill bit to penetrate into earth formation [13].

Since the DROP is directly related to the drill bit, it is important to be modeled and

predicted. Once a model is built, it will be used as a reference for DROP prediction. If the observed values depart from the predicted values, it might be considered an anomaly, as it has the potential of extending drilling time and increase drilling costs. The reduction of DROP during drilling is a possibility of drilling problems such as, the vibrations of the bit, drill string problems or bit bailing problems [15].

Increasing the DROP beyond certain rate is also counter productive. It can cause serious drilling problems like stuck pipe. The DROP given by a certain model depends on the values of the various features that affect the speed of the drill bit [21], [25].

There are several techniques of DROP prediction that researches have used and they can be classified into four categories, namely, mathematical models, statistical models, data-driven models, and hybrid models [12]. Figure 3.1 shows the summary of existing technique for DROP predictions.

3.1.1 Mathematical Models

In mathematical models, researches try to model the correlation between features related to DROP using some mathematical formula. An example of using physical processes to form a model that can predict DROP is proposed by Warren [13]. This model predicts DROP on cone bits when the borehole-pressure is low. It also calculates both cuttings generation and removal. The cause of DROP reduction is also investigated and found that pieces of rocks that piled up during the drilling or local catering affect both are react differently when level of hydraulics increases. The former will improve DROP but the latter will not. It was also found that increasing WOB with oil mud will increase DROP than using water mud.

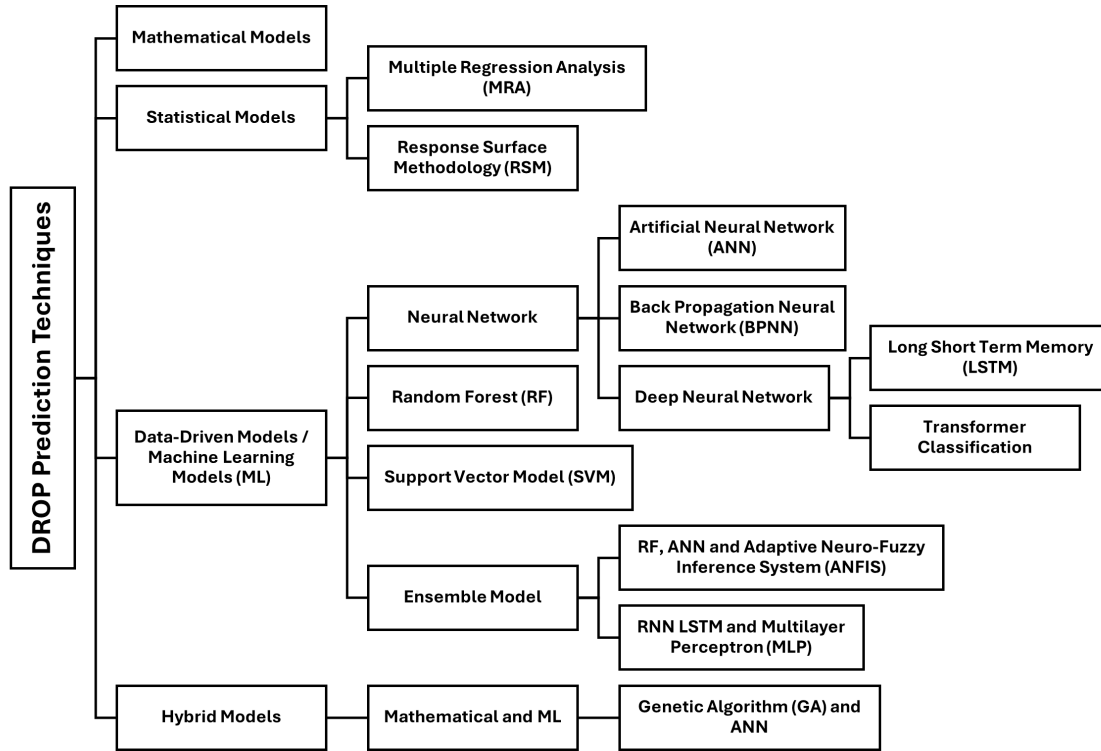


Figure 3.1: DROP Prediction Techniques

Bizanti [86] focused on building mathematical model using field data in the state of Louisiana taken from the drilling operation up to 20,000 ft. He investigated the correlation between depth, WOB, rotational speed, bit characteristics and DROP. The objective was to develop a model based on the aforementioned features on soft and hard formations in Louisiana. It was concluded that if bit weight increases, the coefficient of rotary speed exponent decreases and vice versa. DROP was impacted by the rotary speed on formations significantly at shallow and less at the deeper formations. In soft formations, DROP was maximum with low WOB and high RPM, and highly dependent on depth, WOB and RPM. While in the hard formations, the highest DROP was when WOB and RPM were high, and weakly dependent on those features.

Xu et al. [14] presented a more thorough mathematical model which takes in to consideration rock or formation strength as well as workflows and data pre-processing like removing null data and outliers. They also proposed process steps like in the modern time series machine learning approach by calculating values of a parameter using its previous values. At the end, a mathematical formula was constructed and a DROP model for PDC bits was built based on mud logging data with wireline logging data. Not only did they focus on DROP prediction, but they also demonstrated the utility of acoustic transit time (δt) in assessing rock drillability.

By employing mechanical specific energy (MSE), which represents the energy required to cut out the rock, Bjornsson et al. [15] developed a formula to predict DROP via three distinct processes. Initially, the minimum MSE was estimated by employing the wireline log data, downhole pressure, and lithology. The power required to cut the rock is then determined using the WOB, rotary speed, bit diameter, and friction factor. Ultimately, the DROP is estimated by combining the aforementioned data with additional information regarding the diameter of the holes. As a result, the DROP for different hole sections increased by 15%, 52%, and 60%, respectively, for hole sections measuring 311.2mm, 215.9mm, and 142.9mm. Additionally, it prolonged the life of the TCI bits by 33% and decreased the total duration by 15 days in comparison to the budgeted authorization for expenditure (AFE).

Caicedo et al. [16] developed specific energy ROP (SEROP) to predict DROP. It considers the method as rock strength function which can be applied to all various bit types. The generated model computes minimum, maximum and average DROP. Since the developed method is based on rock cutting principles, it is flexible and easy to

update. Also the method doesn't require a specialist to operate it. It can be operated by any skilled drilling engineer.

3.1.2 Statistical Models

There are several techniques being used that can be categorized as statistical models. Multiple regression analysis (MRA) uses historical dataset to calculate features that will be used in an equation to produce best result of the equation. Some variables of the mathematical equation is determined by measurement and others are computed by method known as least squares. The process iterates through all the datasets and each value of the dataset is used to compute the coefficients of the equation.

Eren and Ozbayoglu [18] used statistical method by doing multiple regression analysis against historical data to find coefficients for pre-defined DROP model. The predicted DROP can represent the actual measurements realistically and the accuracy of the predicted value increased as the data volume increased.

Seifabad and Ehteshami [87] developed generic model for DROP prediction which can be used in different formations, each with unique characteristics. The coefficient of the models were determined using the following features: bit diameter, bit rotation, WOB, depth, mud density, and mud viscosity. MRA is used to determine the correct model. Experimental results show that the model accuracy degrades with lower DROP values.

Morajevi and Naderi [88] investigated technique which uses historical dataset to calculate features called response surface methodology (RSM). It uses quadratic form with the DROP as dependent variable and a set of constant coefficients as independent

variables. The coefficients are derived from the historical data and are estimated by minimizing their sum of squared errors. The DROP prediction has 95% confidence interval and it can be affected by many factors, such as well depth, plastic viscosity, WOB, rotation speed, bit jet impact force, and 10 min to 10 s gel strength.

3.1.3 Data Driven Model or Machine Learning Models

Data driven models are the most recent techniques and use machine learning [17], [19]–[21], [25], [43]. Barbosa, et al. [12] named this as machine learning (ML) models. Unlike the mathematical approach, the data driven model doesn't try to model the physical process into mathematical model but it uses data as input and output and identify the relationships between them and then it uses that for prediction.

Some proposed hybrid approaches which combined the data driven model with other aforementioned models. The combination can be of two or more techniques. For example, the combination of mathematical model based on Genetic Algorithm (GA) with ANN which was proven to be more accurate than regression or ANN only models [26], [89].

Artificial Neural Network (ANN)

In recent years ML approach has been given more attention by researchers. Most of the early ML solutions were based on ANN. For example, Rahimzadeh et al. [17] proposed a DROP prediction model based on ANN. The model takes as input eight features, namely true vertical depth (TVD), WOB, rotary speed (N), mud weight, bit tooth wear, pore pressure, rock strength (S), and hydraulic jet impact force across the

bit (F_{jm}). The model was built using MATLAB Neural Network Fitting Tool (NFT). The dataset was split into three sets of training, validation and testing, 60%, 20% and 20% respectively. After many experiments they found the right number of layers which consists of 20 neurons. This approach improved the correlation factor, R^2 , from 0.812 to 0.825.

Moran et al. [19] used ANN to improve a simple DROP prediction algorithm which used six input features, namely rock strength (S), rock type, abrasion, WOB, rotary speed (N) in RPM, and mud weight. The results show significant improvement with excellent DROP prediction derived from historical data from offset wells. First, a model for each offset well is built to identify lithology, rock strengthness, and well correlation and then ANN model is built with 8 hidden layers and two outputs, DROP and bit wear. The result show a correlation, R^2 , of 0.8. They showed three scenarios that can predict DROP for each hole size.

Monazami et al. [20] used 13 features, namely drill collar outside diameter (in), drill collar length (m), kick of point (m), azimuth (degree), inclination angle (degree), WOB (lb), flow rate (gallon per minute), bit rotation speed (N) in RPM, mud weight, solid percent, plastic viscosity (cp), measured depth (m), and yield point ($lb/100ft^2$). They show that the prediction accuracy is improved significantly by using ANN, regardless of the changes in the features. The given correlation factor is 0.8.

Jahanbakhshi et al. [21] also proposed ANN with 22 features: DROP, WOB, rotary speed, differential pressure, density of the overlying rocks, hole depth, pump pressure (average), equivalent circulating density (ECD), hole size, formation drillability, permeability, porosity, drilling fluid type, plastic viscosity, yield point of mud, initial gel

strength of mud, 10 min. gel strength of mud, solid percent of mud, mud PH, bit type, bit wear and bit hydraulic power (HHP). They studied various combinations of hidden neurons in a maximum of two ANN layers. They found that with two layers, with 9 and 3 neurons respectively, it gave the best result with a correlation factor of 0.916.

Duan et al. [43] proposed a back propagation neural network. BPNN, for predicting DROP. They trained it using six features, namely pressure difference, formation drillability, mud density, WOB, rotation rate, and delivery capacity. The method introduced the usage of momentum factor and dynamic learning rate to enhance the traditional BPNN. First, fine-tune weights to make the network quickly converge and to avoid a local minimum; and then, using random learning rate to help the network converge not in an irregular way. The architecture with 8 to 10 hidden layers produced better results with an accuracy of more than 85%.

Ahmed et al. [25] proposed as input ten features for the ANN model consisting of five drilling features and five fluid properties, namely, rotary speed, WOB, torque, pump flow rate, standpipe pressure, mud density, funnel viscosity, plastic viscosity, yield point and solid percentage. It used 18 hidden layers and one output as DROP prediction. Beside the ANN model, this study also proposed an equation for DROP prediction which gave a correlation coefficient of 0.996 and average absolute percentage error (AAPE) of 5.776%.

Han et al. [6] proposed a model for DROP prediction that takes time as an important parameter as mentioned by Xu et al. [14]. The features that are used in the study are considered as features of a time series dataset. The prediction was first done through a BPNN model and then through LSTM. The BPNN model gave accuracy

of 86% and the LSTM model gave accuracy of 93%.

Diaz et al. [22] divided the training stages into 10 equally divided sections. Each stage is used for the prediction of several DROPs in a cumulative way: stages 1 is used to predict stage 2; 1 and 2 to predict stage 3; 1, 2 and 3 to predict stage 4 and so on until stage 10. The predicted values differed from the measured ones with mean error ranging from 3.7% (on the training stage 1) to the 43.5% on the last training stage.

Unlike the mathematical modeling approaches which normally depend on specific features like WOB and rotary speed like the model developed by Bourgoyne, et al. [1] or rock strength information like the model developed by Warren [13], a ML model has advantage to use any number of features for the development of a model [17]. DROP prediction is affected by a combination of many features. DROP prediction will be inaccurate if data of some features is not available. In contrast to the mathematical model, machine learning models can extrapolate the unavailable data [19]. Mathematical models have many limitations. They have low accuracy, they require huge resources, their coefficients are very difficult to accurately compute, and they are less able to adapt if applied in different environments [43]. Table 3.1 shows the comparison of some of the ML and its hybrid approaches covered in this study. The study included many comparisons but the table only shows the one with significant result.

Comparison of Multiple Or Ensemble ML Models

Soares and Gray [45] compared some mathematical and ML models using drilling

features, such as WOB, drillstring rotary speed, and mud flow rate. The three ML models outperformed the mathematical models. The mean error of the best ML model, random forests (RF), was lower by more than 20%. RF itself outperformed the other two ML models, namely support vector model (SVM) and NN. RF also outperformed the other models when tests were done using real-time data.

Alsaihati et al. [99] introduced ensemble model built from RF, ANN and adaptive neuro-fuzzy inference system (ANFIS). The model was trained using 70% of the dataset and tested using the 30%. First, the ANN and ANFIS, known as base learners, were trained and tested and resulted in meta features. Then both of the original features and meta features were used to train the RF based meta model to predict DROP. Test results showed that RF based meta model outperformed the original RF model with R^2 of 0.97 and 0.96 respectively.

Deep Learning (DL)

Han et al. [6] considered predicting DROP as a time-series forecasting problem. They used LSTM to predict DROP. They compared the performance of their proposed model with ANN BP. They showed that the accuracy of their LSTM was 93% whereas that of ANN BP was 86%.

Hong et al. [9] used Transformer network (TN) with a classification algorithm on time series data to predict DROP. Unlike RNN that uses recurrent connections, TN uses a technique called self-attention that relates the input sequence to a calculated score of attention. The proposed approach outperformed mathematical models with MAE of 3.46 and MRE of 23.68%. They also combined multiple DROP prediction

models using a technique known as stacked ensemble which consisted of 17 models. They trained it using a dataset of 16 wells. The later model outperformed the first one. Its MAE was 2.96 and its MRE was 20.71%.

Encinas et al. [100] proposed a model with two branches RNN with LSTM cells and multilayer perceptron (MLP). The RNN branch input is the past DROP data on which a Gaussian noise layer is applied before going to the RNN and the result will go through a dropout layer. The other branch inputs are the drilling features such as depth, calculated DWOB, hookload, surface torque, RPM and standpipe pressure which will go through set of layers: MLP, dropout, flatten and another MLP layer. Both of the outputs from the RNN branch and the latter MLP branch are concatenated before going to the final MLP layer to generate DROP prediction. The proposed model was compared with other ML models: RF, GBM and extreme GBM, and it outperformed all of them.

3.2 Hazard Prediction Techniques

One significant piece of research that helps drilling operations minimize unproductive time and become more cost-effective is drilling hazard prediction. The same group of strategies, similar to those for DROP prediction, have also been put out by several studies for DH prediction. The methods use statistics and the most modern machine learning data-driven techniques. The main DH topics studied by researchers include bit balling detection, kicked detection, lost circulation, and stuck pipe detection. Various methods proposed by researchers to predict DH are explained in this section.

3.2.1 Statistical Models

Shoraka et al. [101] used multivariate statistical regression and discriminated analysis method to predict stuck pipe. The proposed method uses 26 features to develop equation that represents stuck pipe event. Dataset were collected on daily basis and labelled as stuck or non-stuck and was used in the equation. In general, regression analysis is proven to give more accurate predictions than discriminant analysis. Furthermore, the latter analysis can predict stuck pipe event three days before it happens, while the former only one day prior to the event.

Chamkalani et al. [102] incorporated support vector regression (SVR) with hybrid least square support vector regression Coupled with Simulated Annealing (CSA) optimization technique (LSSVM-CSA) to tune the SVR hyper parameters. This approach is applied to distinguish an event whether it is differential or mechanical stuck. The accuracy of the differential stuck, mechanical stuck, and non-stuck was 96.6%, 96.1%, and 97.2% respectively.

Hess [53] used statistical linear regression model to calculate coefficients, subsequently multiplied with collected data for each selected feature taken from the logging-while-drilling (LWD) measurement. Some of the features are characteristics of the wellbore and BHA, downhole pressure and temperature, mud properties and drilling practices. The model had R^2 of 0.902 with confidence level of 95%.

3.2.2 Data-Driven or Machine Learning Models

Salminen et al. [66] combined two analysis methods to predict stuck pipe. The first one

is to predict real-time data deviation based on real-time model using specific software; and the second one is to analyse the trend of the real time data. These methods use features such as pump pressure, flow rate, torque, rotary speed, hookload and drag, and WOB. An optional equivalent circulating density (ECD) information will improve the result if available. The result will be an alert issued approximately 38 minutes prior to the event.

Alshaikh et al. [103] built a model to predict stuck pipe using real-time drilling data. The model was tested on some wells and gave good result and was able to predict stuck pipe event. There were two early detection cases reported when the model was applied. In the first case, it predicted a stuck pipe event 2.5 hours prior the incident mainly due to decrease of standpipe pressure and subsequently an increase of mud flow. In the second case, it was when a rotary drilling experienced dynamic loss and restrictions during running in a hole, the model was able to predict stuck pipe event five hours before the actual incident.

Murillo et al. [104] used fuzzy logic and NN for stuck pipe prediction with discriminant analysis first for grouping the predictive model as differentially stuck, mechanically stuck and non-stuck. Both of the models divided the dataset into 75% for training and the rest for testing. NN outperformed fuzzy logic with lower number of misclassification.

Al-Baiyat and Heinze [105] implemented ANN and SVM models to predict stuck pipe. It also gave more detail of the stuck event such as differentially or mechanically stuck and non-stuck. The accuracy for both the proposed models are 85%. However, the SVM model produced more reliable predictions and was more practical because it

needed fewer features to be optimized.

Zhu et al. [106] also used ANN model to predict stuck pipe. First, features for the model were taken from the well's formation pressure. Subsequently, the weight of each feature of the model is calculated using drilling fluid performance data. Finally, the model calculates the probability of stuck pipe.

Alshaikh et al. [67] compared three classification models to detect stuck pipe event, namely ANN, decision tree, and SVM. These models utilized following features: hook load, standpipe pressure, torque, DROP, WOB, drilling fluid flow-in rate and rotary speed in RPM. The performance was measured using 3-fold cross-validation technique. The SVM model gave: 98.62% accuracy, 96.88% precision, 94.88% sensitivity, and 99.39% specificity.

Magana-Mora et al. [107] implemented a framework that covers end-to-end processes from automation on data extraction, analysis on the feature selection, until the model prediction. Accordingly, 11 features were selected from drilling parameters, drilling performance, drilling fluid, and survey data. Finally, four models of RF, DT, ANN and SVM were compared. RF outperformed the others with 82.75% accuracy.

Abbas et al. [56] investigated lost circulation prediction using ANN and SVM. The proposed models were built a dataset collected from 385 wells that covers features such as drilling operation parameters, formation type, lithology information from the rock, and characteristics of drilling fluid. Following this, both models, ANN and SVM, were optimized using hyper parameters specific to each model. At the end, the two models were tested and the accuracy of SVM was 0.91 and ANN was 0.83.

Yang et al. [108], proposed machine learning approach to detect gas kick hazard

in real time. An ANN model was selected and utilized to identify gas kick after gas cased compiled from logging data. The model uses seven features which were selected through PCA method. The selected features were weight of hook (WOH), ROP, SPP, RPP, flow pumps, PUMP1, and PUMP2. The proposed model was tested and it showed its ability to predict gas kick earlier to the incident.

Yi et al. [109] investigated five drilling dysfunction events including stuck pipe and bit balling with Bayesian networks classification models. The models utilize only surface data features to detect the dysfunction and was proven that the detection models give robust result. Further detail of bit balling detection was concluded after aggregating information from trends of the surface data. The trends started with the decrease of surface torque through the Bayesian network, and finally the condition of bit assertiveness that became decreased. Eventually, the classifier model was able to predict the bit balling event.

These studies highlight the successful outcome of deep learning recurrent neural network (RNN) models, specifically long short-term memory (LSTM) and transformers neural network (TNN), in predicting DROP. The LSTM model had superior performance compared to other machine learning methods, including BP ANN [6], SVR, RF, and GRU [34]. Additionally, an ensemble TNN with classification beat mathematical approaches Bingham and Warren [9]. However, there is a lack of literature comparing the TNN and LSTM models. Both models are deep learning algorithms that handle time-series data. Furthermore, each publication faces the difficulty of using diverse datasets, with only a limited number of them utilizing publicly available datasets such as Volve [4], [9], [90] or FORGE [93], [98] datasets.

Table 3.1: ML Comparison for DROP Prediction

References	Models	Inputs	Performance	Dataset and Training Splits [(%)]
Rahimzadeh et al., 2010 [17]	(ML): NN , (Math): BY, Warren	8	R^2 : 0.825	2 wells in SK gas field, Qatar-Fars Arch Persian Gulf, Iran [60, 20, 20]
Moran et al., 2010 [19]	NN	6	R^2 : 0.8	
Monazami et al., 2012 [20]	NN	13	R: 0.98913, MSE: 0.0591	336 cases from DDR of 1 oil field, Southern Iran [79.76, 10.12, 10.12]
Jahanbakhshi et al., 2012 [21]	NN	21	R: 0.9160, MSE: 0.015	[75, 15, 10]
Bezminabadi et al., 2017 [50]	(ML): NN , (Stat): MNR	9	R: 0.87, RMSE: 0.009	604 data in 1 well of Azadegan, Southwest Iran [80, , 20]
Eskandarian et al., 2017 [38]	MON-MLP , RF, Cubist	6	R^2 : 0.8009, RMSE: 1.33	5 wells in Southwest Iran [10 CV]
Hegde et al., 2017 [37]	(ML): RF , Ensemble; (Mat): Bingham, Motahhari, Hareland	3	R^2 : 0.84, RMSE: 8.3	Bakken shale [10 CV]
Ashrafi et al., 2018 [46]	(ML): MLP-NN , RBF-NN	8	R: 0.933, RMSE: 1.12	Marun oil field, Southwest Iran [10 CV]
Han et al., 2019 [6]	LSTM , BP ANN	32	MAPE: 7%	1 well in Southern China [55.56,,44.44]
Noshi, 2019 [75]	RF , ANN, SVR, RR, GBM	18	R^2 : 0.88, MAPE: 10%	60 wells
Gan et al., 2019 [47]	SVR-ILSSIWBA , BP, ELM, RS, RF, SVR, LSSVR, SVR-PSO, SVR-Simplex	4	RMSE: 0.398, NRMSE: 0.343, MAE: 10.586%, MAPE: 10.114%	95 drilling data from 1 well in Shennongjia, Central China [84.21,,15.79] and 10 CV
Singh et al., 2019 [11]	Spline Regression , RF, NN, RNN, MLR, RR, PCA	15	MAPE: 13%	50 wells in Permian Basin, Texas, USA. 45 wells for training and 5 wells for testing with leave one out cross validation
Hadi et al., 2019 [23]	NN , MRA	8	R^2 : 0.96, MSE: 1.89	1 well in Southeast Iraq [70,,30]
Ahmed et al., 2019 [44]	(ML): NN , (Math): Bingham, Maurer, BY	10	R^2 : 0.9912, R: 0.996, MAPE: 5.776%	347 data points of deep shale formation [70,,30]
Diaz et al., 2019 [22]	NN	9	MAPE: 23.7%	2 wells PX-1 and PX-2 from South Korea, divided data into 5 cases.
Al-AbdulJabbar et al., 2020 [76]	NN	6	R: 0.956, MAPE: 5.29%	2 wells Well-A and Well-B [85,,15]
Najjarpour et al., 2020 [72]	Hybrid Bingham	5	MAPE: 13.59%	Well in Southwest Iran, using DDR, BHA report, bit report and geology report
Quisbert, 2020 [4]	LSTM	6	R^2 : -3.17, MAE: 13.83	1 well: 15/9-F-5 from Volve dataset, Norway [60,,40]
Hong et al., 2020 [9]	(ML): Ensemble TNN , TNN; (Math): Bingham, Warren	14	MAE: 2.96, MAPE: 20.71%	23 wells from Volve dataset, Norway [73.91,,26.09] where 17 wells for training and 6 for testing
Tunkiel et al., 2020 [90]	GB	11	R^2 : 0.82, MAE: 4.75	7 wells, Volve dataset, Norway [90,,10]
Alali et al., 2020 [77]	NN	3	R: 0.94, RMSE: 9.224, MAPE: 5.95	Onshore Field-A 12" section, 2500 wells in 10 rigs [70,,30]
Hazbeh et al., 2021 [91]	MLP-ABC	7	R^2 : 1 , RMSE: 0.009312, MSE: 0.000087	Kaboud oil field well 7 with 1878 dataset [80,,20]
Brenjkar and Delijani, 2021 [92]	(ML): PSO-MLPNN , RBFNN, ANFIS, SVR, (Math): BYM, Bingham	13	R: 0.698, RMSE: 4.916	1912 samples from southwest Iran, divided by 6 cases
Gan et al., 2021 [30]	ELM	3	RMSE: 0.85	Well ZK3 in Xiangyang area, Central China [10 CV]
Ben Aoun and Madarász, 2022 [93]	RF , NN	6	R^2 : 0.84, MAE: 2.46	FORGE site, Pason log files for well 58-32, 7311 rows of data [70,,30]
Fan et al., 2022 [94]	RF	7	R^2 : 0.82, MAPE: 14.52%	4 wells in X block of T oil field [75,,25]
Zhou et al., 2023 [95]	(Hybrid): Mota-RF , Mota-ANN, Mota-SVM; (Math): Hareland, Motahhari; (ML): ANN, SVM, RF	13	RMSE: 2.65, MAPE: 9%	70 wells [90,,10]
Chen et al., 2023 [96]	LSTM , SVR, BPNN, RF, RNN, GRU	27	MSE: 80.95, RMSE: 9, MAE: 6.12, MAPE: 12%	Block N of Bohai oilfield, China with 9.193.408 data points [75,,25]
Jiao et al., 2024 [97]	(Hybrid): Eckel-SVM , Eckel-RF, Soares-SVM, Soares-RF; (ML): RF; (Math): Soares	14	R: 0.9936 , RMSE: 1.5861, MAPE: 5.09%	5655 data points from Halahatang oil field, Tarim Basin, China
Yehia et al., 2024 [98]	ET , LGBBoost, RF, GBR, KNN, Cat-Boost, SVM, XGBoost, ANN	11	R^2 : 0.933, RMSE: 8.13, MAE: 4.23, MAPE: 15.24%	FORGE dataset well 58-32, 7310 rows [80,,20]

CHAPTER 4

METHODOLOGY

To meet all the objectives of this thesis, a methodology of four stages was adopted. In the first stage, benchmark dataset relevant to this study were analyzed. In the second stage, the analyzed data were preprocessed and made ready to be consumed by deep learning algorithms. In the third stage, a number of deep learning models for the prediction of DROP and DH were selected and optimized. In the last stage, the performance of the proposed deep learning models was analyzed and compared. Figure 4.1 illustrates the adopted methodology which consists of the above mentioned four stages: data analysis, data preprocessing, model selection, and model assessment. The subsequent sections will explain each stage.



Figure 4.1: Methodology Phases

4.1 Data Analysis and Preprocessing

4.1.1 Data Analysis

The dataset used in this study is called the Volve dataset. It was collected from 24 wells by Equinor, a Norwegian state-owned international energy firm. We chose it for this study because it is widely used as a benchmark dataset, and it is easy accessible by any researcher. It has been used in a variety of research related to drilling operation in the oil and gas industry [4], [9], [90].

Similar to the study done by Hong et al. [9], this work uses the complete Volve dataset. This is in contrast to existing studies, where a subset of the Volve dataset is used. They used subset of the dataset that belong to few wells [100], [110], [111] or those with specific format [90], [112]. More details of this dataset is given in Section 5.2.

The Volve dataset provides two types of well data: time-based log data and depth-based log data. We utilized the depth-based log data to construct and assess the deep learning models. Subsequently, the time-based data will be chosen and combined with the daily drilling report (DDR) information. This will generate a dataset that includes both time and status information for DH prediction. The analysis process will be segmented into three sub-processes: data import, data selection, and data combining.

In the data import stage, relevant data was transformed into a comma-separated values (CSV) format, and then in the data selection stage, irrelevant data was removed and separate depth and time based data. The last stage of data analysis is to do data

merging with activity information that contains time and status of the operation (i.e. normal or hazard/failure happened). Figure 4.2 shows the data analysis sub-processes.

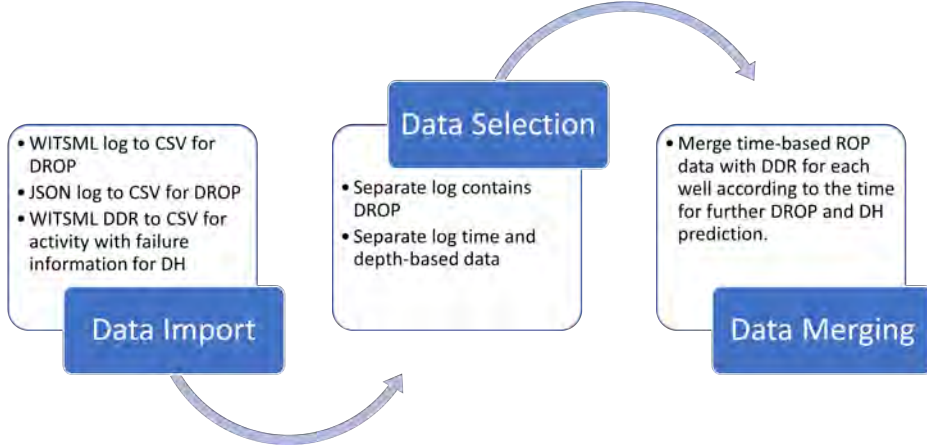


Figure 4.2: Data Analysis Phases

4.1.2 Data Preprocessing

Data preparation is an essential step that must be carried out before the deep learning models can use the data. This procedure consists of four consecutive steps. The first two phases were exclusively implemented for the data meant for DROP prediction, as the absence of data and outliers are crucial for DH prediction. In the initial stage, data quality for DROP were validated and all missing values were either substituted with suitable alternatives or eliminated from the dataset. Following that, in the second stage, extreme outlier removal was performed. In Step 3, the data was labeled, specifically for hazards detection. The process of data normalization was executed as the final stage in the operation of the deep learning models. The details of the above mentioned steps is explained in 5.2.2

4.2 Model Construction

DNN models, such as LSTM, outperform other ML models when forecasting using sequentially sampled time series data [4]. Hence, in this study, time series forecasting of DROP and DH is done using DL models.

In recent studies, LSTM has been shown to outperform other machine learning (ML) models in forecast DROP using time series data [6], [100]. However, recently introduced Transformer networks (TN) has show better time series forecasting performance than LSTM. So far, and to the best of our knowledge, the use of TN in the prediction of DROP has only been published in [9] and as a classification problem .

In this study, TN is used a regression model to predict DROP and is compared with RNN-based models, LSTM and GRU. After predicting DROP, the next objective of this thesis is to predict DH using DL. We chose classification model based on TN for the prediction of DH using the same dataset as for DROP prediction, but with the addition of a target column labeled with values representing the absence or presence of hazards.

A typical workflow in model construction consists of: training, validation and testing. Optimizing the model is done during the training by measuring the difference between data that being observed (from the dataset) and the calculated value by the model, called as residual through a loss function mechanism [45]. The model will be optimized to reduce the residual. Figure 4.3 shows the model construction workflow.

A DL model consists of input, hidden and output [113] layers. The shape of the input layer is determined by the shape of the pre-processed data. The number of

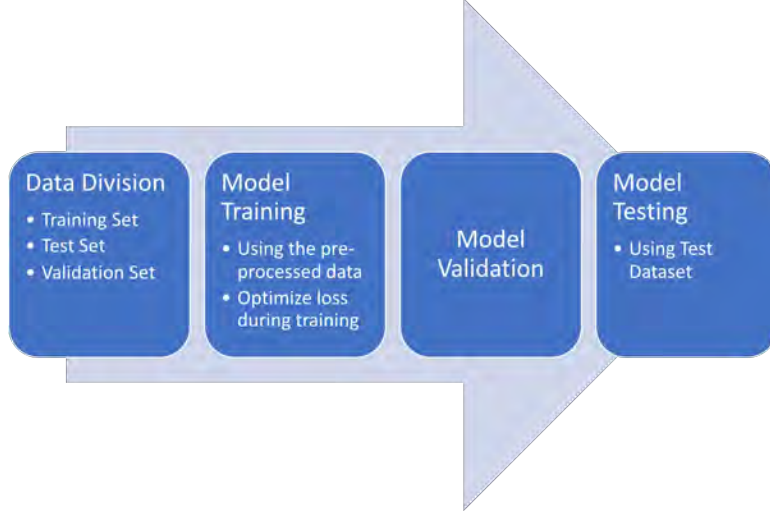


Figure 4.3: Model construction workflow [3]

hidden layers is determined by trial and error and by optimizing the applicable hyper parameters, such as the number of epochs, learning rate, etc. The shape and form of the output layer depends on the required output format.

4.2.1 Lost Function and Model Optimization

When the model is learning, the objective is to adjust the weight of the neural network to make the difference between value from the observed data with the predicted value calculated by the model down to 0 which is called as a loss function [78]. In this study we utilized mean square error (MSE) for the loss function.

Mean Square Error (MSE): Given the model parameter (weight) w , training data set X , the size of training data set n , the number of neurons in the output layer D , the target output t , and the actual output o , MSE is calculated as following:

$$MSE = J(w) = \frac{1}{2n} \sum_{x \in X, d \in D} (t_d - o_d)^2$$

. The MSE is consider better is closer to zero [3].

Based on the MSE result to reduce or optimize the residual, the optimized gradient descent is required, it is called as optimizer. This study utilizes Adam optimizer [114] as the model optimizer.

Adaptive Moment Estimation (Adam) Optimizer: Given the estimates of the first moment (mean) and second moment (uncentered variance) $\mathbf{m}$ and v of the past gradients respectively, exponential decay η , Adam optimizer is shown as below:

$$\Delta w = -\frac{\eta}{e + \sqrt{v(n)}}m(n)$$

Additionally, $g(n)$ represents the gradient dJ/dw and e is a small constant. Thus the formula for $v(n)$ and $m(n)$ are shown as below:

$$m(n) = am(n-1) + (1-a)g(n)$$

$$v(n) = bv(n-1) + (1-b)g^2(n)$$

The experiments have demonstrated that the initial values of variables a and b are 0.9 and 0.999, respectively. Additionally, it has been shown that the learning rate is set at 0.0001 [3].

4.2.2 RNN Based Methods

Typically, at time t , an RNN based model consists of the input parameter x_t , a memory cell s_t , set of model weights w_t and output o_t which may be in hidden layer [3].

The following function shows the structure of RNN based model [3]:

$$s_t = f(Ux_t + Ws_{t-1})$$

where f is a function known as activation function, which may be a logistic sigmoid or hyperbolic tangent function [115] and U and W are the model weights [3]. Vanilla RNN suffers from the gradient vanishing problem, and to counter that LSTM and GRU are used which offer an enhancement of the activation function using a combination of sigmoid and hyperbolic tangents along with gating units for element-wise operation [115].

In this study, the input for LSTM and GRU based models are preprocessed and normalized. The hyper parameter, such as epochs are optimized by many iterations; ADAM was used as optimizer; RELU was used for the dense layer, and mean squared error (MSE) was used as the loss function.

The LSTM model uses k input sequences to initiate output generation. The input consists of a set of specified features, such as depth, WOB (weight on bit), torque, RPM (revolutions per minute), SPPA (specific power per area), hook load, and mud flow, represented by their respective values. The goal is to predict the first DROP (drilling rate of penetration) after k sequences. The number of cells for both cell states and hidden units is n . This will generate the next n cell states and hidden units. After every k sequence, the hidden units from the k^{th} LSTM cell will produce a predicted DROP after passing through two hidden layers. Figure 4.4 displays the architecture of the proposed LSTM model.

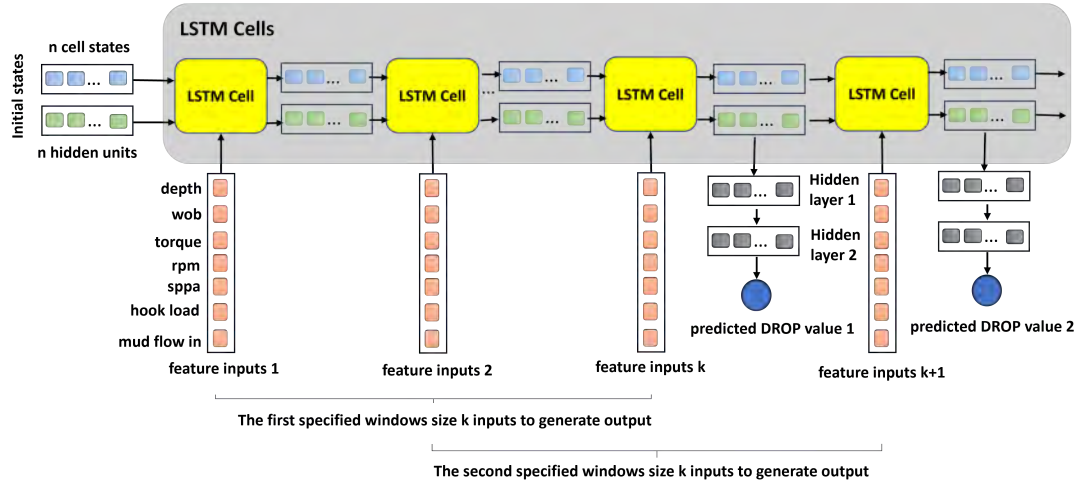


Figure 4.4: LSTM Model Architecture

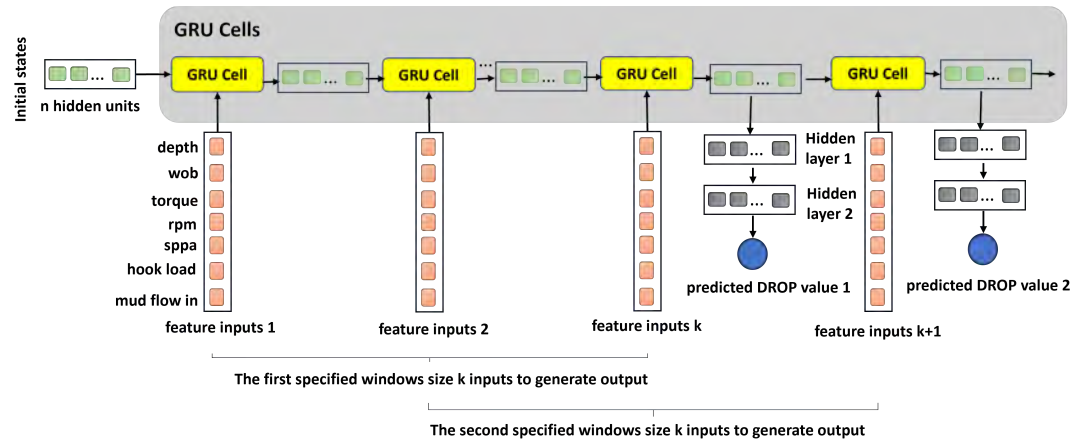


Figure 4.5: GRU Model Architecture

The GRU model, like LSTM, utilizes k input sequences with similar features that initiate the generation of output. The GRU cell consists solely of hidden units of size n , which are essential to generating the subsequent n hidden units. Following each k sequence, the hidden units originating from the k^{th} GRU cell will generate a predicted DROP after going through two hidden layers. The architecture of the suggested GRU model is shown in Figure 4.5.

4.2.3 Transformer

The typical architecture of Transformer model consists of two major components: encoder and decoder. Each component is stacked with two sub-layers: multi-head self-attention and fully connected feed-forward network (FFN) layer and with additional sub-layer of multi-head attention to receive the output from the encoder in to the decoder stack [85], [116]. The architecture of the transformer used in this study is as follow:

- **Input Part:** The set of selected features of k sequence will be used as the input to the encoder.
- **Encoder Stack:** An encoder with four layered stack was selected. The sub-layer of multi-head consists of four parallel attention layers (*heads*) [85]. For the next sub-layer, FFN, of 128 dimensions with ReLU activation was used. The output from here is sent to Decoder.
- **Decoder Stack:** The number of stack layers in the decoder is equal to the number of stack layers in the encoder. Furthermore, the identical number of *heads*, FFN dimensions, and activation functions were employed. The decoder will generate one output, and in the subsequent step, the output of the encoder for sequences 2^{nd} to $k+1$ will serve as the additional input for the decoder, with a maximum of k sequences.

Figure 4.6 shows the architecture of the proposed Transformer model for ROP prediction.

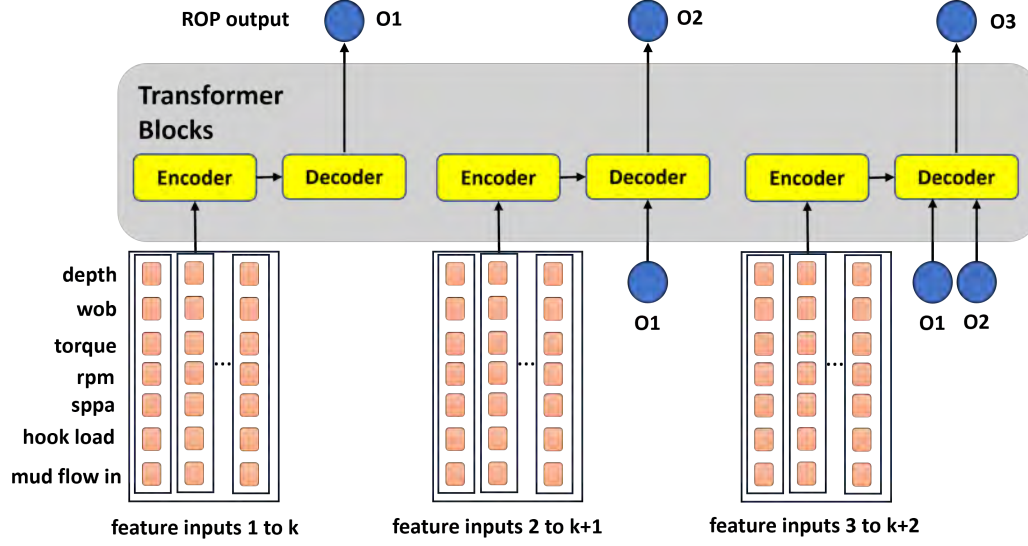


Figure 4.6: Transformer Model Architecture for Regression on ROP Prediction

4.3 Model Evaluation

To study the performance of the proposed model, it is important to evaluate its prediction accuracy, also known as generalization ability [3]. The evaluation is done by comparing the predicted values with the true or actual values. The average difference between the predicted values and the actual values is called a residual or error [3]. The lower the residual error is, the better the performance of the proposed model [117].

In this study, four known evaluation metrics, namely mean absolute error, mean square error [3], mean absolute percentage error [118] and root mean squared error [117], were used to evaluate the performance of the proposed regression models to predict DROP. While for the classification model to predict DH, F_1 score is being used.

Mean Absolute Error (MAE): Given a sample output of $y_1, y_2, \dots, y_n$ and a predicted output as $\hat{y}_1, \hat{y}_2, \dots, \hat{y}_n$, MAE is calculated by taking the average of all the

absolute values of residuals as:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

. The performance of a model is considered to be better if its MAE is closer to zero [3].

Mean Square Error (MSE) MSE is similar to MAE but it uses square of each residual instead of its absolute value. This means that the greater the error is, the higher is the MSE. The following equation shows how MSE is calculated:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

Mean Absolute Percentage Error (MAPE) While the MAE is the average of the residual absolute values, MAPE is calculated as a ratio of the actual and residual values as shown by equation:

$$MAPE = \frac{100\%}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right|$$

Root Mean Squared Error (RMSE) RMSE has the same unit as the predicted variable which simplifies the interpretation of the evaluation. The following equation shows how RMSE is calculated.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

Coefficient of Determination (R^2)

The coefficient of determination compares the sum of squared prediction error to the sum of squared deviations of observed value Y with its mean [100]. The following equation shows how R^2 is calculated.

$$R^2 = \frac{\sum_{i=1}^{n_{samples}} (y_i - \hat{y})^2}{\sum_{i=1}^{n_{samples}} (y_i - \bar{y})^2}$$

F_1 Score Using Sensitivity and Specificity F_1 score to calculate the accuracy due to imbalance data is defined using *Sensitivity* and *Specificity*. Sensitivity is defined as

$$Sensitivity = \frac{TP}{TP + FN}$$

where TP and FN represent true positives and false negatives respectively. FN is defined as *false negative* because a hazard is detected as non hazard.

When a non hazard is detected as hazard it is called *false positive* (FP) and accordingly *Specificity* is defined as:

$$Specificity = \frac{TN}{TN + FP}$$

where TN is defined as *true negative*, where a non hazard is detected as non hazard.

F_1 score is defined from *Sensitivity* and *Specificity* as [71]:

$$F_1 = 2 \times \frac{Sensitivity \times Specificity}{Sensitivity + Specificity}$$

CHAPTER 5

RESULTS AND DISCUSSIONS

This chapter presents the results and analysis done to study the performance of the proposed solutions for the prediction of DROP and DH. The conducted experiments can be classified in to three sets.

The first set of experiments was done to find the best possible deep learning architectures for the prediction of DROP and use the best of DROP DL for DH.

The second set of experiments was done to study the performance of the proposed DROP prediction solution and the third set was done to study the proposed DH prediction solution.

But before the results and analysis of the experiments are discussed, let us present the machines, the software, and the datasets used in the experiments, and the data preprocessing done to transform the data into quality data and into a format that can efficiently be processed by machine learning algorithms.

5.1 Experimental Machines and Software

The machines and software specification for the experiment is shown in the following Table 5.1.

Table 5.1: The experimental machines and software

Component	Specification
OS	Windows 11 and Linux (under WSL, Windows Sub-system for Linux)
Processor	Intel i7-10875, 8 Cores
RAM	32GB
Development Tools	Visual Studio Code
Programming Language	Python
Main Libraries	tensorflow, pandas, darts

5.2 Experimental Dataset Analysis and Preprocessing

5.2.1 Data Analysis

As mentioned in Section 5.2, the dataset used in this study is called the Volve dataset. It was collected from 24 wells by Equinor, a Norwegian state-owned international energy firm. The dataset is stored in two file formats, namely, JavaScript Object Notation (JSON) and the Wellsite Information Transfer Standard Markup Language (WITSML). The JSON formatted data consists two types namely, mud log data and logging while drilling (LWD) data; whereas the WITSML formatted data consists of many types of data.

The preprocessed data from the Volve dataset is stored in three primary sources as JSON and WITSML files. Data from two sources in JSON format were divided into two categories: mud log data and logging while drilling (LWD) log. Despite the fact that the WITSML source contains a variety of various data types, we only include the log type and daily drilling report (DDR) during the preprocessing stage.

Data Import

The data import technique involved segregating the JSON and WITSML log data based on their depth and time attributes. Afterwards, another import process is carried out for the time-based log data items and DDR reports related to the drilling activities, taking into account the time and status of each activity. Following that, the state of the activity will serve as the indicator of the specific failure that occurred during that time frame.

Data Selection

The data selection process will specifically choose the imported dataset that contains DROP information. Subsequently, the depth and time variables will be separated, with the time variable being combined with the activity dataset for hazards detection purposes. Out of the total of 24 wells, 14 wells were chosen from the time-based log that had DROP information.

We selected the candidate of the features for the study similar to those frequently used by similar studies to predict DROP, Table 2.1.

The feature names in Figure 5.1 have been standardized because the name of a

feature may vary from one dataset to another. For example, the names TQA, TQX, TQM, TQ5 and TORQUEA are used for the feature torque.

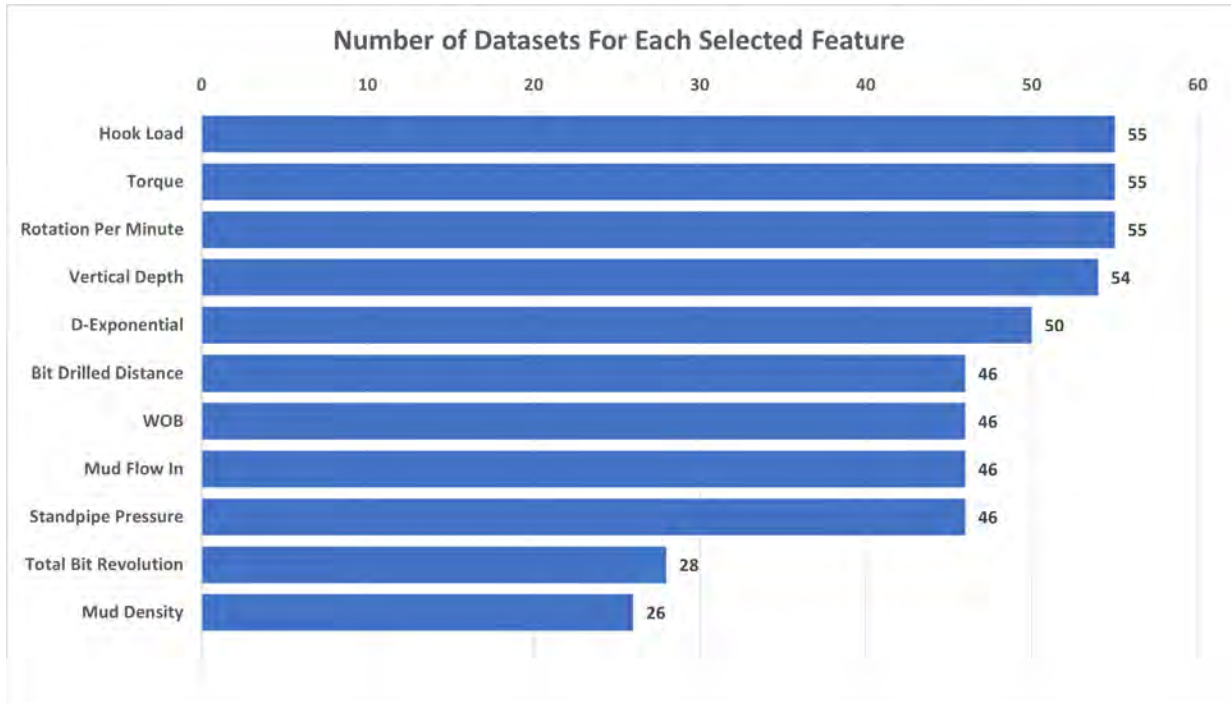


Figure 5.1: Feature rank: The number of Datasets a feature was selected from

Data Merging

In the last stage of data analysis, the method involves combining time-based log data with activity data for each well and time reference. This merging creates a new dataset that includes a new feature and is also labeled with the activity status. The drilling activity is categorized into seven labels: success, circulation loss, equipment failure, injury, mud loss, operation failed, and stuck equipment.

5.2.2 Data Preprocessing

The data preparation utilized in this study encompasses four separate procedures, as

outlined in 4.1.2. The detailed processes are explained in the subsequent sections, and it should be noted that the initial two stages were only executed for the DROP prediction dataset.

Data Quality Validation, Handling Missing Values

The Volve dataset contains many types of data some of which are not relevant to the prediction of DROP and DH, hence deleted.

Also, according to the study in [119], datasets used to predict DROP must have DROP and WOB values greater than or equal to 2 ft/h and 5 Klbh respectively; otherwise, the values are considered too low and thus deleted.

Quality data results in better deep learning models. In order to improve data quality, noise reduction or data smoothing is required [100], [110]. In this work, an enhanced moving average filter (MAF) known as trailing average was employed [99]. This technique first calculates the MAF using a window size of 20 data points, then calculates a Z-score score by dividing the difference of each data point and the MAF with the standard deviation for that window. It uses only the Z-score which are inside the windows between -3 and 3. Following is the formula:

$$Z = (x_i - \mu) / \sigma$$

where x_i is the data point, μ is the average and σ is the standard deviation.

An important step in data engineering is handling missing values and outliers. Missing values are handled by substitution or elimination [6]. A very popular substi-

tution technique is interpolation [4], [100].

In this study, *linear* and *bfill* [120] interpolation techniques were used. Linear was used for the substitution and *bfill* was used to make sure no value is missing by filling with the next existing value.

The linear interpolation uses following equation to estimate the substitution value for the null or empty one (x) between two known values (x_0 and x_1):

$$x = x_0 + \left(\frac{x_1 - x_0}{2}\right)$$

Outliers Removal

Outliers were removed using inter-quartile range (IQR) technique [100], [119]. Points less than LowerOuterFence and points greater than UpperOuterFence are considered outliers. IQR, LowerOuterFence and UpperOuterFence are defined as follows:

$$IQR = P_{75} - P_{25}$$

where,

$$LowerOuterFence = P_{25} - 3 * IQR$$

$$UpperOuterFence = P_{75} + 3 * IQR$$

P_{25} and P_{75} are the 25th and 75th percentiles of data points. The number of data samples in the original Volve datasets ranges from 202 to 54,361 per dataset. After the above mentioned preprocessing steps, the range was 113 to 17,625.

Data Labeling

As the result of data merging for the time-based log data with drilling activity data, a drilling category is already available as the status of the activity in particular time. This information will be used for the DH prediction that uses multi class classification. The label of the activity then encoded with following result:

- "success": 6
- "circulation loss": 0
- "equipment failure": 1
- "injury": 2
- "mud loss": 3
- "operation failed": 4
- "stuck equipment": 5

Data Normalization

The last data preprocessing step that we preformed was data normalization, also known as feature scaling. In this study we used the Min-Max algorithm which is the recommended normalization technique for both ANN and RNN models [100]. Min-Max normalization is defined as:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

The process of data normalization was executed as the final stage in the operation of the deep learning models.

5.2.3 Dataset Result for Experiment

After conducting data analysis and preprocessing, three distinct dataset have been prepared for experimentation: a depth-based log dataset for DROP prediction, a time-based log dataset for DROP prediction, and a merged time-based log and drilling activity dataset for DH prediction. Since the depth-based dataset includes data from practically all the wells, it will be utilized to determine the most suitable model among the proposed models of LSTM, GRU, and TNN.

The Table 5.2 presents the log data categorized by depth, with the well ID and dataset ID encoded for future reference. The tables 5.3 and 5.4 display the wells used for time-based data DROP prediction and DH prediction, respectively.

5.3 Performance of the Proposed DROP Prediction Solutions

A timeseries forecasting solution assumes the input data to be sorted based on time. In this study, we assume each Volve dataset is sorted according to depth which is equivalent to sorting it according to time. Out of the 55 Volve datasets, we used 46 datasets. The excluded nine datasets were removed during data preprocessing phase because they lacked the necessary features to predict DROP. In the proposed solutions, 31 of these datasets were randomly selected for training and the remaining 15 were

used for testing.

The features of the merged datasets are a subset of the features listed in Figure 5.1. The features are depth, WOB, torque, RPM, mud flow in, stand pipe and pressure, and are available in all the 46 datasets.

Based on the way the training datasets are merged, the proposed deep learning models can be classified in to two. In the first class, the 31 training dataset are merged and then sorted depth wise; and in the second class, they are merged without sorting. In each class, three deep learning models are proposed. The first model is based on LSTM, the second one on GRU, and the last one on TNN. Let us call the three Class 1 models as LSTM1, GRU1, and TNN1; and the three Class 2 models as LSTM2, GRU2, and TNN2.

The time-based dataset will utilize the top-performing model from the Class 1 and Class 2 performance outcomes. This is because the depth-based dataset comprehensively covers all well, therefore, the model that exhibits the highest performance is anticipated to yield the greatest outcome for the time-based dataset.

5.3.1 Architecture and Performance of the Proposed Class 1 Models

As mentioned above, Class 1 models take as input 31 randomly selected Volve datasets for training and 15 for testing. The 31 training datasets are merged into one dataset and sorted depth-wise. As mentioned above, there are three class 1 models, namely LSTM1, GRU1, and TNN1. The architecture and performance of these three models

are as discussed below.

Architecture of the Proposed LSTM1 model

To find the best possible LSTM1 architecture, we tested many LSTM architectures and multiple possible hyper parameter values. For example, we tested using different number of neurons, levels, dropout values, look-back periods and so on. Finally we arrived at the architecture listed in Table 5.5.

Architecture of the Proposed GRU1 model

To find the best possible GRU1 architecture for DROP prediction, we tested a number of GRU architectures and tested multiple hyper parameter values, such as different number of neurons, levels, dropout values, look-back periods and so on. Table 5.6 shows the selected GRU1 architecture.

Architecture of the Proposed TNN1 model

The arrive at the proposed TNN1 architecture, we tested a number of TNN architectures and multiple hyper parameter values. To tune the architecture and hyper-parameters of the model, we conducted many experiments. Table 5.7 shows the model hyper-parameters that were selected for the DROP prediction.

Performance of the Proposed Class 1 model

We assessed the performance of each proposed model in Class 1 by utilizing metrics such as RMSE, MSE, MAE, and MAPE. The LSTM1 model exhibited superior per-

formance compared to the GRU1 and TNN1 models in terms of MSE and RMSE, with scores of 102.08 and 10.10, respectively. Nevertheless, TNN1 demonstrates dominance over the other two models, GRU and LSTM1, in terms of the MAE and MAPE scores, which are 6.70 and 26.35 respectively. Figure 5.2 displays a comparison of the Class 1 models for the RMSE, MSE, MAE, and MAPE.



Figure 5.2: Class 1 Models Comparison

5.3.2 Architecture and Performance of the Proposed Class 2 Models

Class 2 models also take as input 31 randomly selected Volve datasets for training and 15 for testing. The 31 training datasets are merged into one dataset without sorting. There are three class 2 models. The first model is based on LSTM, the second one on GRU, and the last one on TNN. Let us refer to these models as LSTM2, GRU2, and TNN2. The architecture and performance of these three models are as discussed below.

Architecture of the Proposed LSTM2 model

To choose the best possible LSTM2 architecture, we tested many LSTM architectures and multiple hyper parameter values. For example, we tested using different number of neurons, levels, dropout values, look-back periods and so on. Finally we arrived at the architecture listed in Table 5.8.

Architecture of the Proposed GRU2 model

To arrive at the proposed GRU2 architecture, we tested a number of GRU architectures and multiple hyper parameter values. For example, we tested using different number of neurons, levels, Dropout values, look-back periods and so on. Table 5.9 shows the model hyper-parameters that were selected for the DROP prediction.

Architecture of the Proposed TNN2 model

To choose the best possible TNN architecture, we conducted many experiments using multiple hyper parameter values. Table 5.10 shows the model hyper-parameters that were selected for the DROP prediction.

Performance of the Proposed Class 2 model

We evaluated the performance of each proposed model in Class 2 by using measures such as RMSE, MSE, MAE, and MAPE. The TNN2 model outperformed the GRU2 and LSTM2 models in all performance metrics, with scores of 22.96, 2.69, 4.79, and 14.88 for MSE, MAE, RMSE, and MAPE correspondingly. Figure 5.3 illustrates a comparison of the Class 2 models based on the RMSE, MSE, MAE, and MAPE.



Figure 5.3: Class 2 Models Comparison

5.3.3 Comparison of Class 1 and Class 2 models

The performance of Class 1 and Class 2 models was evaluated by comparing their RMSE, MSE, MAE, and MAPE. The TNN2 model demonstrated superior performance compared to all other models in both Class 1 and Class 2, as indicated in Table 5.11. This demonstrates that the TNN2, belonging to Class 2, has strong performance on unsorted data without any feature-based grouping. This enables the model to function with any generic depth-based data for DROP prediction, without requiring it to be sorted by a specific feature.

5.3.4 Comparison of the Time-Based DROP Model with The Best Depth-Based Model

We conducted an analysis of time-based data from 10 wells, as indicated in Table 5.3. We excluded the stand pipe pressure feature from the analysis, as most of the values were abnormal during drilling activity. Figure 5.4 displays the DROP data across a

depth range of 0 to over 5000m.

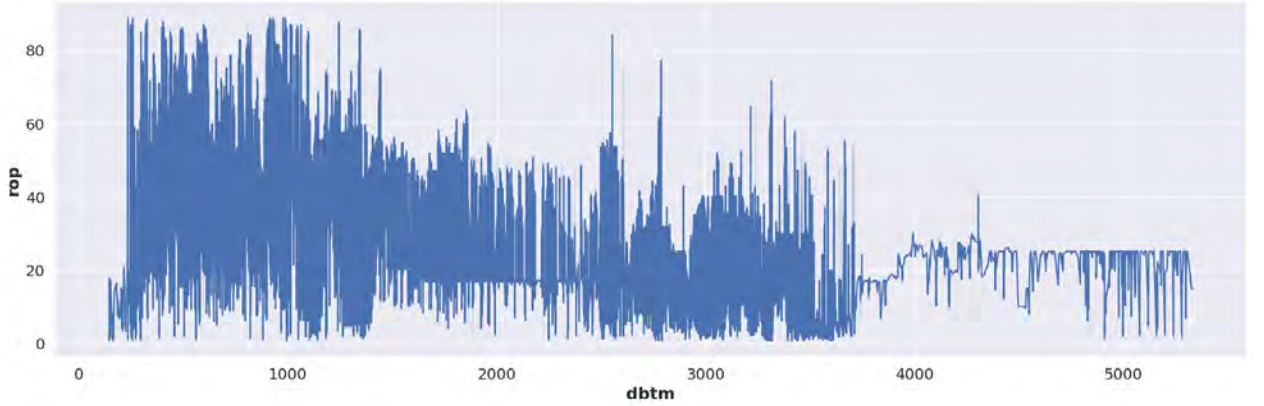


Figure 5.4: Time Based Data Depth vs DROP Chart

In the initial experiment, our focus was on the depth starting from 3800m, as the DROP values exhibited a narrow range of 0 to 40 m/h. The hyper-parameter for the model will be set to the configuration of the best model reported in Table 5.11, which is also shown in Table 5.10. The new model will have a smaller number of epochs, specifically 50, compared to the previous model which had 200 epochs. The training process lasted for a duration of 29 minutes and 8.7 seconds, while the inference process took 6 minutes and 21.8 seconds. The data was split into a 70% training set and a 30% testing set. The experiment yielded the following performance results: $MSE = 1.91$, $MAE = 0.56$, $RMSE = 1.38$, $MAPE = 2.89$, and $R^2 = 0.85$.

Figure 5.5 displays the relationship between the predicted DROP in red color and the test values in blue. Meanwhile, Figure 5.6 illustrates the predicted and test DROP values in relation to the test depth.

The following experiment aims to assess the model's ability to generalize and perform well across different depths. This experiment involves aggregating data from 10 wells into one dataset. The performance results obtained from this experiment

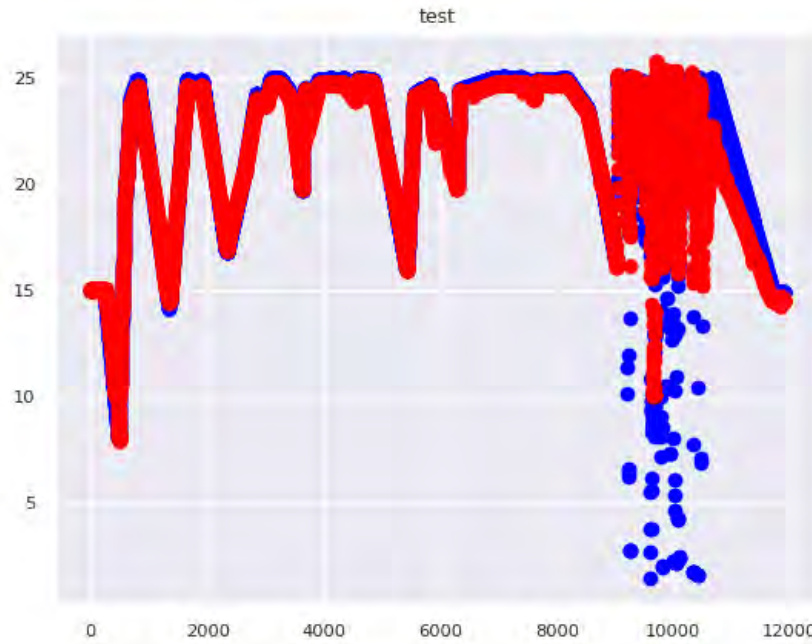


Figure 5.5: Prediction (Red) vs Test DROP for Time Based Data Above Depth 3800m



Figure 5.6: Prediction vs Test DROP Chart for Time Based Data Above Depth 3800m

can serve as a reliable model for predicting outcomes in other wells. The dataset was divided into training and testing data based on the depth of the data. Data with depths greater than or equal to 1200m and less than or equal to 3700m were used for training, while data with depths above 3700m were used for testing. The hyper-parameter for the model remains unchanged from the initial experiment, which

produced the following results: MSE of 1.94, MAE of 1.13, RMSE of 1.39, MAPE of 6.04, and R^2 value of 0.91.

While the performance of the MSE, MAE, RMSE, and MAPE did not match that of the initial experiment, the R^2 showed improvement. This demonstrates the successful training of the model, which accurately predicted the ROP value up to 80m/h, as depicted in Figure 5.7. The model was trained using data from all 10 wells and was able to make accurate predictions for data at a depth of 3700m. The correlation between the predicted and test values increased compared to the previous one, with the coefficient of determination R^2 increasing from 0.85 to 0.91, as depicted in Figure 5.8. The chart illustrating the predicted vs test values can be seen in Figure 5.9.

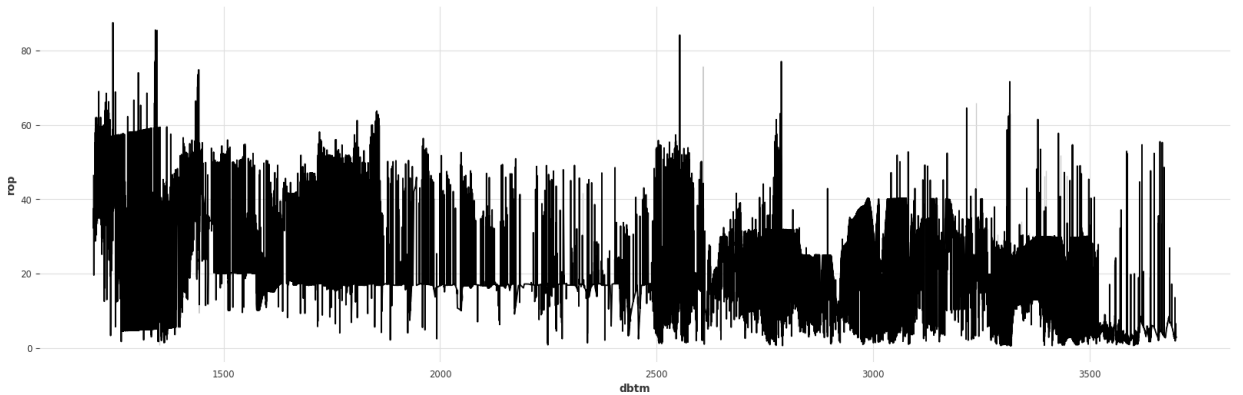


Figure 5.7: Training Data of DROP for Time Based Data Depth Between 1200m and 3700m

We compare the metrics obtained from the time-based trials with the best metrics obtained from the depth experiment (TNN2), as shown in Table 5.12. The time-based performance surpassed the performance of the best depth-based experiment, TNN2.

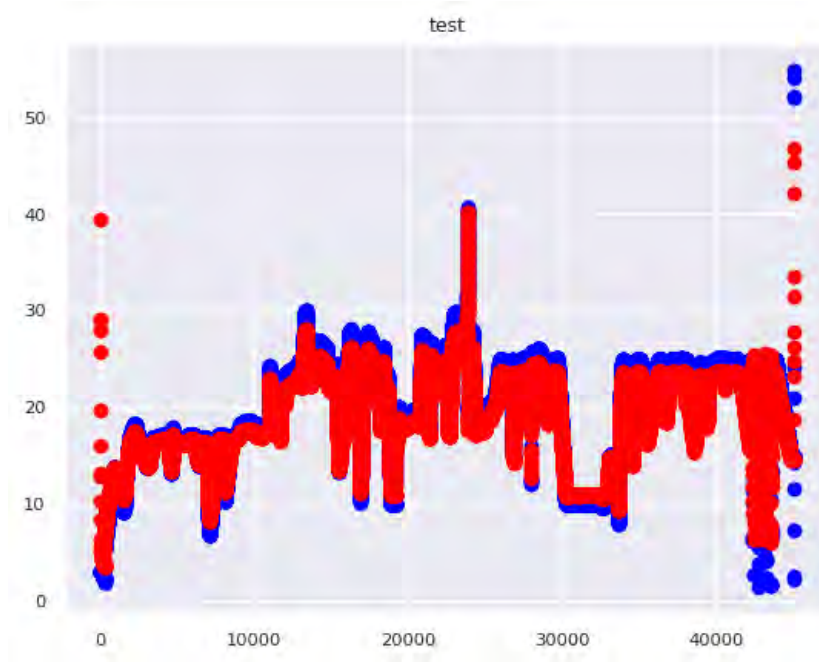


Figure 5.8: Prediction (Red) vs Test DROP for Time Based Data Above Depth 3700m

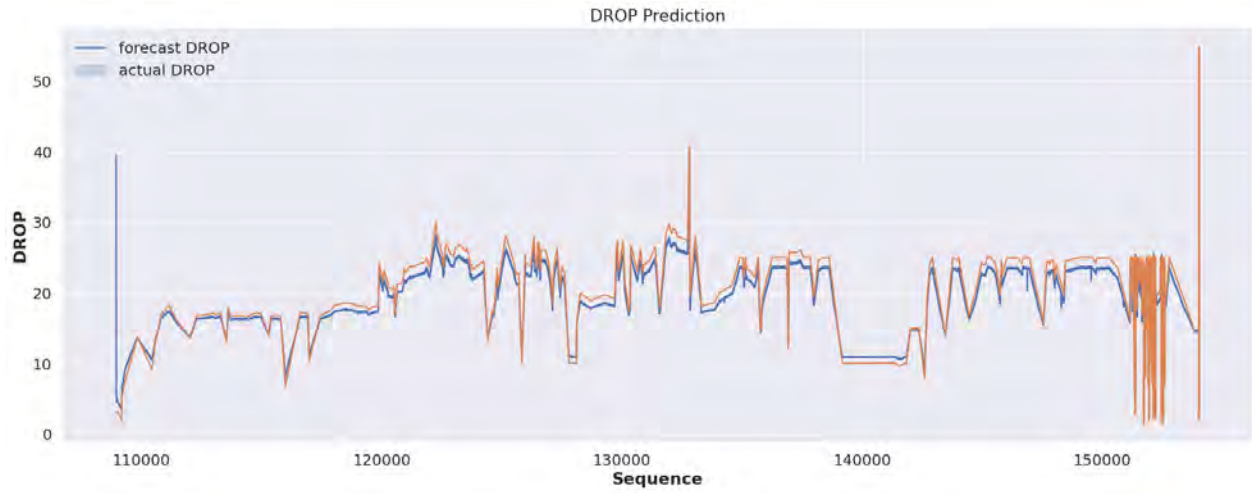


Figure 5.9: Prediction vs Test DROP Chart for Time Based Data Above Depth 3800m

5.3.5 Comparison of the Proposed DROP Models with Existing Models

In this section, the performance of the proposed DROP prediction models with existing DROP prediction solutions. We will show to sets of comparison. The first set of

comparison is done from model perspective, where we will compare the proposed models with existing solutions that are based on deep learning models such as: LSTM, GRU or Transformer. The second set compares the performance of TNN2 with existing solution that were trained using the Volve dataset. We compare them with TNN2 because it outperformed the other proposed models.

Several studies have been conducted for DROP prediction that focus on LSTM model with different datasets and using different evaluation methods. Quisbert [4] proposed a DROP prediction solution based on LSTM and tested it using D2, which is one of the Volve datasets listed in Table 5.2. In his thesis, he reported MAE of 13.83 which is of lower performance than our original model of 10.08 [5]. The proposed LSTM1 and LSTM2 have significantly better performance with MAE of 7.6 and 9.02 respectively. Figure 5.10 shows the MAE performance comparison.

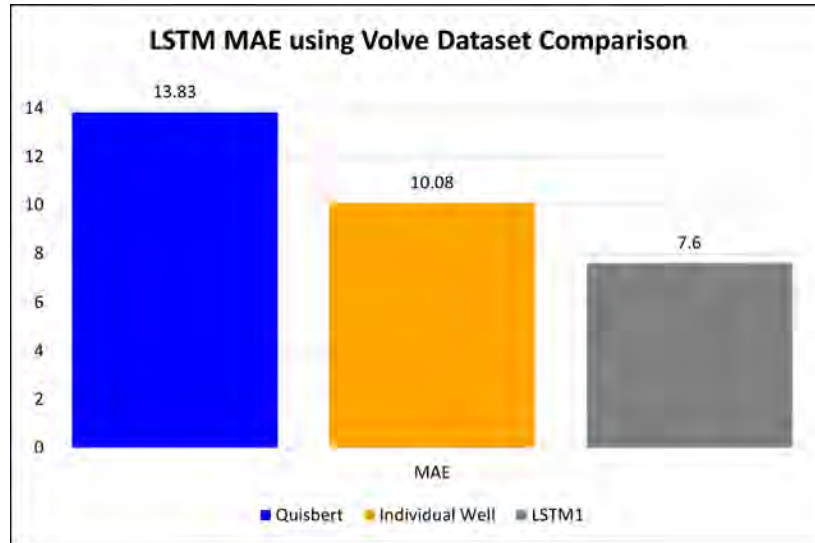


Figure 5.10: Comparison of LSTM using MAE with [4], [5]

Han et al. [6] reported the reduction of MAPE from other method 14% to 7% using LSTM. Liu et al. [7] reported that combination of LSTM and FNN outperformed single

LSTM model with MRE/MAPE from 20.03 to 15.11. Figure 5.11 shows performance comparison based on MAPE.

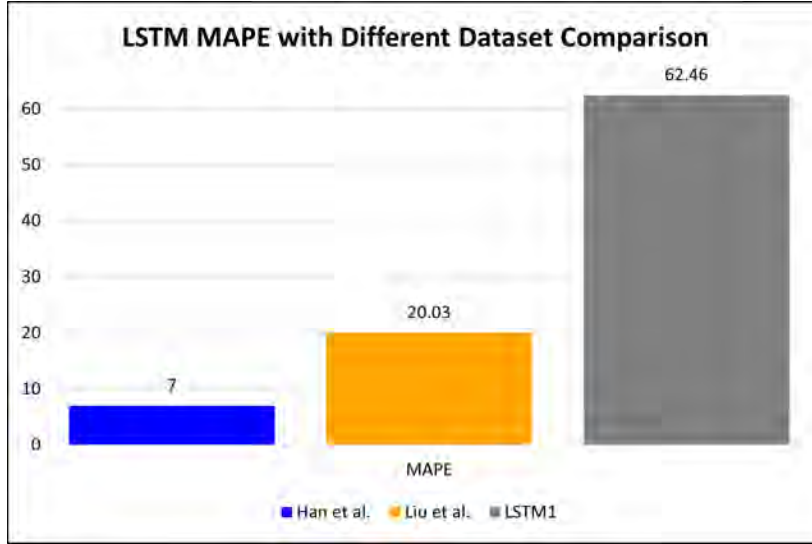


Figure 5.11: LSTM MAPE based comparison with [6], [7]

Zhang et al. [8] divided datasets from four wells into limited data points ranging from 354 to 902 and run the modified GRU model. He reported MAE in the range of 0.22 to 0.55 and MAPE in the range of 4.3 to 7.4. In our previous study where we used data of individual wells we obtained MAE and MAPE of 1.6 and 10.9 respectively [5]. Figure 5.12 shows the comparison for MAPE and MAE between the proposed GRU1 and GRU2 with [8], [5].

Hong, 2020 [9] proposed a solution based on TNN for DROP prediction. They used the Volve dataset for training and testing and obtained MAE of 2.96 and MAPE of 20.71. As can be seen from Figure 5.13, the proposed TNN Time 1 has better performance with MAE of 0.56 and MAPE of 2.89.

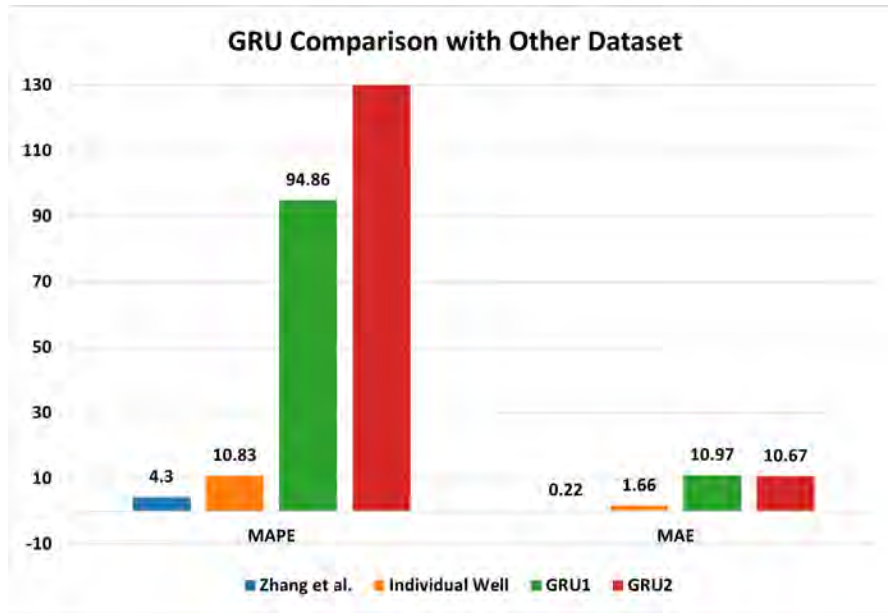


Figure 5.12: GRU MAPE and MAE Comparison With Other Studies: [5], [8]

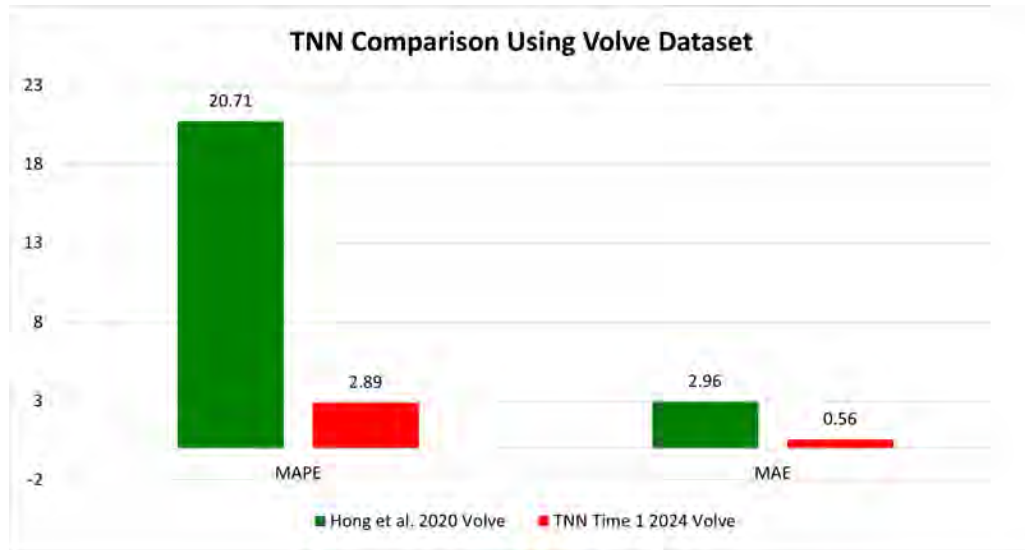


Figure 5.13: TNN2 performance Comparison With [9]

5.4 Performance of the Proposed DH Prediction Solution

The dataset suited for DH prediction is created by integrating time-based log object data with the daily drilling activity report data (DDR). The dataset is derived from 6

wells, as indicated in Table 5.4. It consists of encoded labeled data related to hazards encountered during drilling operations. These hazards are characterized as multi-class hazards, making the proposed DH prediction a multi-class classification task.

Among the six labels given, one of them denotes a typical state, referred to as "success," and is assigned a numerical value of 6. The hazard codes are represented by the remaining codes, specifically 0, 1, 2, 3, 4, and 5. No major adjustments, such as outlier reduction, were performed on the dataset during the data preprocessing stage. Nevertheless, null values were removed. All the features that are same to DROP prediction, including SPPA, were utilized for the DH prediction, such as: WOB, depth, torque, RPM, and mud flow in.

For the DH prediction we used TNN model. We chose TNN because it outperformed LSTM and GRU in the prediction of DROP. To study the performance of the proposed TNN model, the data was preprocessed, reshaped, and split, and then fed to the model. We then conducted many experiments in order to optimize the TNN hyper-parameters. Table 5.13 shows the selected hyper-parameters.

5.4.1 Performance of the proposed DH prediction model

In the experiment, we imported the data from the chosen 6 wells for the DH prediction, which included multi-label condition labels. The Figure 5.14 displays the data collected from July 2007 to July 2009, indicating the occurrence of multiple categories of hazards over that period. The hazards were depicted as a vertical line originating from the top, representing the normal condition at the top.

Once the data has been divided into training and testing sets, the testing data

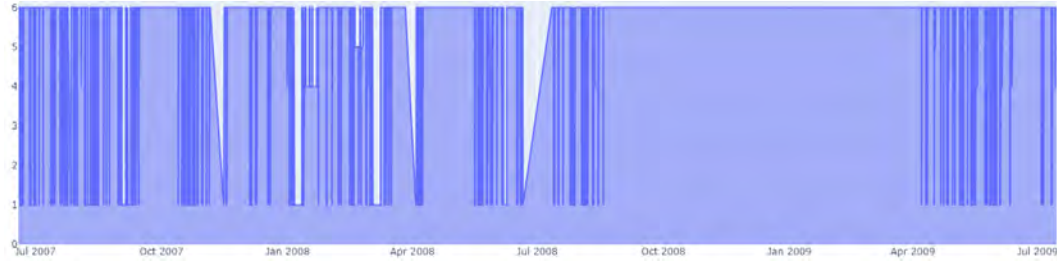


Figure 5.14: Dataset with Multi Class Hazard Information - Shown as horizontal line from the top

will be displayed in the Figure 5.15. By zooming in on the specific date range from February 24th, 2008 to April 13th, 2008, the hazards and input features for that duration can be observed in the Figure 5.16.

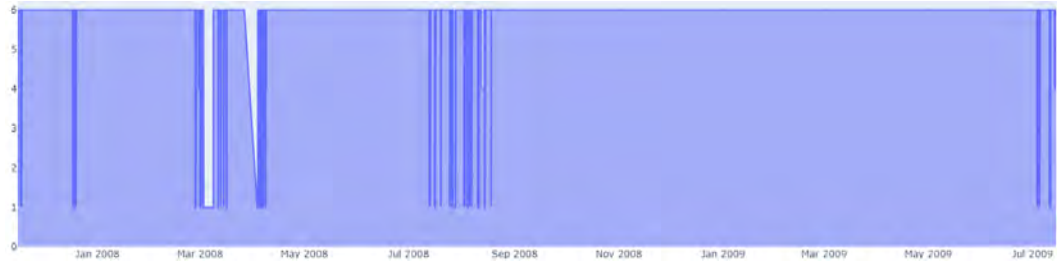


Figure 5.15: Test Data For Model

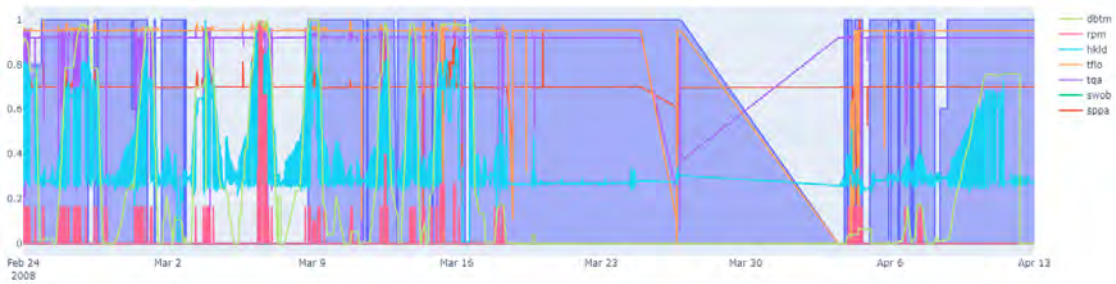


Figure 5.16: Test Data For Model Zoomed to February the 24 2008 and April the 13th 2008

Zooming in on the time period from February 24, 2008 to April 13, 2008 will display the hazard as depicted in the picture 5.16. Additionally, it will reveal the features and their correlation with the hazards. The manual analysis does not easily

allow for the identification of the pattern that correlates the input feature values with hazards. The chosen TNN model will utilize the training data to make predictions, yielding the following results: training accuracy as shown in the Figure 5.17.

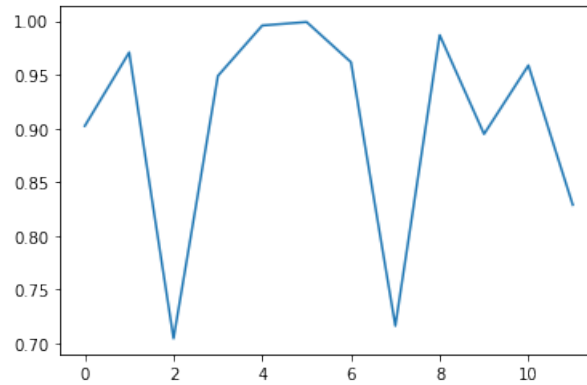


Figure 5.17: Training Accuracy for DH Model Using TNN

The prediction result is displayed in Figure 5.18, which is capable of categorizing hazards at the same time as the test data. Non-hazards are also depicted similarly to the test data. This demonstrates the efficacy of the model in performing multi-class classification, accurately predicting the hazard level at a similar data point as the test data. Table 5.14 displays the model's accuracy in predicting three classes: 1, 4, and 6, which are labeled as equipment failure, operation failed, and success (no hazard).

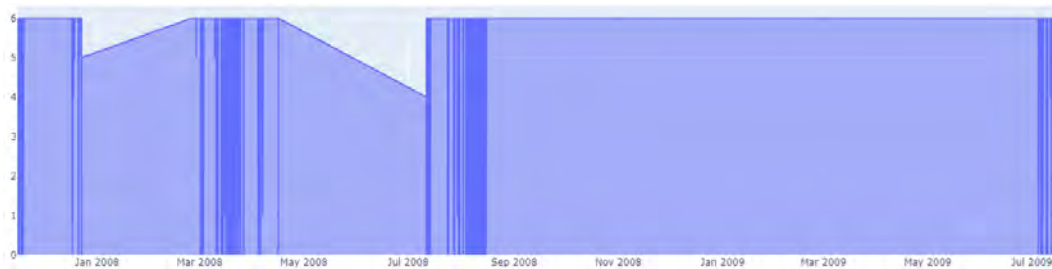


Figure 5.18: Predicted Data Using TNN

5.4.2 Comparison of the Proposed DH Prediction Model with Existing Models

Unlike the proposed model, nearly all the existing models detect one type of hazard. For example, Magana-Mora et al. [107] proposed three models to predict stuck pipe. They reported that the *Sensitivity* of each model was low compared with their *Specificity*. Which means, the prediction of non-hazard events is much better than the prediction of hazard (Stuck Pipe). This is because due to the imbalanced in the datasets used. However, the TNN model that we proposed, worked much better, and if there is any dataset to predict the stuck pipe, it can learn from it.

Abbas et al. [56] proposed a machine learning model to predict Lost Circulation (LC). They achieved good *Sensitivity* and *Specificity* but they can only detect LC. The TNN model that we are proposing can predict any type of hazard and has a good accuracy depends on the given dataset for the hazard.

Table 5.2: The experimental depth-based dataset and their corresponding wells

Well ID	Well Name	Dataset	Dataset IDs
W01	$15_9 - 19A$	2	D01a - D01b
W02	$15_9 - 19BBT2$	3	D02a - D02c
W03	$15_9 - 19SSR$	6	D03a - D03f
W04	$15_9 - F - 1A$	3	D04a - D04c
W05	$15_9 - F - 1B$	3	D05a - D05c
W06	$15_9 - F - 1C$	5	D06a - D06e
W07	$15_9 - F - 1$	1	D07
W08	$15_9 - F - 10$	15	D08a - D08o
W09	$15_9 - F - 11A$	1	D09
W10	$15_9 - F - 11B$	1	D10
W11	$15_9 - F - 11 - T2$	1	D11
W12	$15_9 - F - 11$	2	D12a - D12b
W13	$15_9 - F - 12$	1	D13
W14	$15_9 - F - 14$	1	D14
W15	$15_9 - F - 15A$	1	D15
W16	$15_9 - F - 15B$	1	D16
W17	$15_9 - F - 15C$	1	D17
W18	$15_9 - F - 15 - D$	1	D18
W19	$15_9 - F - 15$	1	D19
W20	$15_9 - F - 4$	1	D20
W21	$15_9 - F - 5$	1	D21
W22	$15_9 - F - 7$	1	D22
W23	$15_9 - F - 9A$	1	D23
W24	$15_9 - F - 9$	1	D24

Table 5.3: The experimental time-based for DROP prediction dataset and their corresponding wells

Well ID	Well Name	Dataset IDs
W07	$15_9 - F - 1$	T07D
W08	$15_9 - F - 10$	T08D
W13	$15_9 - F - 12$	T13D
W14	$15_9 - F - 14$	T14D
W16	$15_9 - F - 15B$	T16D
W20	$15_9 - F - 4$	T20D
W21	$15_9 - F - 5$	T21D
W22	$15_9 - F - 7$	T22D
W23	$15_9 - F - 9A$	T23D
W24	$15_9 - F - 9$	T24D

Table 5.4: The experimental time-based for DH prediction dataset and their corresponding wells

Well ID	Well Name	Dataset IDs
W08	$15_9 - F - 10$	T08DH
W13	$15_9 - F - 12$	T13DH
W14	$15_9 - F - 14$	T14DH
W20	$15_9 - F - 4$	T20DH
W21	$15_9 - F - 5$	T21DH
W23	$15_9 - F - 9A$	T23DH

Table 5.5: The proposed LSTM1 model

Hyper-parameter	Value
Number of epochs	200
Number of steps/epoch	100
Learning rate	0.00055
Hidden neurons	200
Layers	2x20
Output neurons	1
Dropout	0.25
Optimizer	Adam
Activation function	tanh
Loss function	MSE
Look-back period	32

Table 5.6: Architecture of the proposed GRU1

Hyper-parameter	Value
Number of epochs	200
Number of steps/epoch	100
Learning rate	0.00055
Hidden neurons	200
Layers	2x20
Output neurons	1
Dropout	0.25
Optimizer	Adam
Activation function	tanh
Loss function	MSE
Look-back period	32

Table 5.7: Hyper-parameters of the proposed TNN1 model

Hyper-parameter	Value
Number of epochs	200
Learning rate	0.001
Feed forward network	128
Encoder layers	4
Decoder layers	4
Multi head attention	4
Optimizer	Adam
Activation function	ReLU
Loss function	MSE
Look-back period	32

Table 5.8: Hyper-parameters of the proposed LSTM2 model

Hyper-parameter	Value
Number of epochs	200
Number of steps/epoch	100
Learning rate	0.00055
Hidden neurons	200
Layers	2x20
Output neurons	1
Dropout	0.25
Optimizer	Adam
Activation function	tanh
Loss function	MSE
Look-back period	32

Table 5.9: Hyper-parameters of the proposed GRU2 model

Hyper-parameter	Value
Number of epochs	200
Number of steps/epoch	100
Learning rate	0.00055
Hidden neurons	200
Layers	2x20
Output neurons	1
Dropout	0.25
Optimizer	Adam
Activation function	tanh
Loss function	MSE
Look-back period	32

Table 5.10: Hyper-parameters of the proposed TNN2 model

Hyper-parameter	Value
Number of epochs	200
Learning rate	0.001
Feed forward network	128
Encoder layers	4
Decoder layers	4
Multi head attention	4
Optimizer	Adam
Activation function	ReLU
Loss function	MSE
Look-back period	32

Table 5.11: All Metrics for Class 1 and Class 2 Models

Models	MAPE	MAE	MSE	RMSE
GRU1	94.86	10.97	174.4	13.2
LSTM1	62.46	7.6	102.1	10.1
TNN1	26.35	6.7	194.3	13.94
GRU2	130.6	10.67	181	13.45
LSTM2	96.37	9.02	131	11.44
TNN2	14.88	2.69	22.96	4.79

Table 5.12: The Best Depth Models vs Time Models

Models	MAPE	MAE	MSE	RMSE	R^2
TNN2	14.88	2.69	22.96	4.79	
TNN Time 1	2.89	0.56	1.91	1.38	0.85
TNN Time 2	6.04	1.13	1.94	1.39	0.91

Table 5.13: Hyper-parameters of the proposed TNN DH model

Hyper-parameter	Value
Number of epochs	50
Learning rate	0.0001
Feed forward network	4
Encoder layers	4
Decoder layers	No decoder
Multi head attention	4
Optimizer	Adam
Activation function	ReLU
Loss function	Sparse Categorical Entropy
Drop out	0.25
Look-back period	32

Table 5.14: DH Prediction Accuracy

	precision	recall	f1-score	support
1.0	0.99	0.99	0.99	238598
4.0	1.00	1.00	1.00	46858
6.0	1.00	1.00	1.00	1551484
accuracy			1.00	1836940
macro avg	1.00	1.00	1.00	1836940
weighted avg	1.00	1.00	1.00	1836940

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

In this study, we proposed a number of deep learning models for the prediction of DROP and DH. The DROP prediction models were based on LSTM, GRU, and TNN for the depth based data. The DROP prediction for time based data and DH prediction model were based on TNN. The dataset used for training both the DROP prediction models and the DH prediction model was the benchmark dataset known as Volve. It has 55 dataset generated from 24 wells for the depth-based dataset, 10, and 6 for time-based dataset for DROP and DH prediction respectively.

For the prediction of DROP using depth-based dataset, we used 46 of the 55 datasets because the remaining nine datasets don't have data related to the prediction of DROP. We randomly choose 31 of them for training and the remaining 15 for testing. We made two copies of the the 31 datasets. Then we merged the data in the

first copy and sorted them depth-wise. We also merged the data in the second copy but without sorting them. We then created two classes of deep learning models for the prediction of DROP. Class 1 models were trained using the copy sorted depth-wise where as Class 2 models were trained using the unsorted copy. There are three models in each class and they are based on LSTM, GRU, and TNN. The three Class 1 models are known as LSTM1, GRU1, and TNN1; and the three Class 2 models are known as LSTM2, GRU2, and TNN2. These models were optimized by testing them with different hyper-parameters, such as number of neurons, levels, epochs, and look-back periods.

After doing a thorough comparison, we discovered that TNN2 demonstrated superior performance when compared to the other six models. TNN2 was also evaluated against several current models and it demonstrated superior performance compared to all of them. Nevertheless, the outcome of the model's execution on the time-based dataset surpassed TNN2 with a substantial enhancement. The high-quality data in the time-based dataset resulted in a well-trained model, leading to improved outcomes.

For the prediction of DH we used TNN because it gave us better results. To arrive at the final TNN architecture, we experimented with different hyper-parameters, such as number of neurons, levels, epochs, and look-back periods and we selected those who resulted in better performance. We employed a multi-class classification technique to accurately categorize the sort of hazard, wherein one class was designated as "no hazard" (indicating success). We then split the datasets into training and testing. The proposed model is different from all the existing models. This is because all the existing models predict only one type of hazard whereas the proposed solution predicts

all types of hazards. We also compared the proposed algorithm with existing solutions and it outperformed all of them. We believe the proposed DROP prediction and DH prediction solutions are among the best and can be used by drilling companies.

6.2 Future Work

A number of feature research directions may improve the performance results obtained by the proposed models. For example, to employ another type of LSTM and GRU architectures such as stacked LSTM, stacked GRU, bidirectional LSTM, and Bidirectional GRU. Also to investigate ensemble of different deep learning and machine learning architectures.

Validating the suggested model with a different dataset might enhance its maturity. However, it is crucial to subject the dataset to thorough data analysis and preprocessing methods before inputting it into the model.

The *multi-output* model allows for the combination of DROP prediction as a regression output and DH prediction as a multi-class classification output into a single integrated model.

REFERENCES

- [1] A. T. Bourgoyne, *Applied drilling engineering*, Richardson, TX, 1991.
- [2] P. Skalle, A. Aamodt, and I. Swahn, “Detection of failures and interpretation of causes during drilling operations,” 2016.
- [3] *Artificial Intelligence Technology*, 2023.
- [4] M. A. E. Quisbert, “Data driven rop modeling - analysis and feasibility study.”
- [5] B. R. Djamaluddin and S. A. Mohammed, “Comparison of Drilling Rate Penetration Prediction Based on Deep Learning Approaches, A Volve Dataset Use Case,” vol. Day 4 Thu, October, 2023.
- [6] J. Han, Y. Sun, and S. Zhang, “A data driven approach of rop prediction and drilling performance estimation.” International Petroleum Technology Conference, March 2019.
- [7] H. Liu, Y. Jin, X. Song, and Z. Pei, “Rate of penetration prediction method for ultra-deep wells based on lstm-fnn,” *Applied Sciences*, vol. 12, no. 15, 2022.
- [8] C. Zhang, X. Song, Y. Su, and G. Li, “Real-time prediction of rate of penetration by combining attention-based gated recurrent unit network and fully connected

- neural networks,” *Journal of Petroleum Science and Engineering*, vol. 213, p. 110396, 2022.
- [9] Y. Hong, J. Yoo, S. Wang, and S. Yoon, “Augmented machine learning approach of rate of penetration prediction for north sea oilfield,” Kuala Lumpur, Malaysia, p. 9, Oct 2020, oTC.
- [10] C. Hegde, C. Soares, and K. Gray, “Rate of penetration (rop) modeling using hybrid models: deterministic and machine learning.” Unconventional Resources Technology Conference (URTeC), July 2018.
- [11] K. Singh, S. S. Yalamarty, M. Kamyab, and C. Cheatham, “Cloud based rop prediction and optimization in real-time using supervised machine learning.” Unconventional Resources Technology Conference (URTeC), July 2019.
- [12] L. F. F. Barbosa, A. Nascimento, M. H. Mathias, and J. A. de Carvalho, “Machine learning methods applied to drilling rate of penetration prediction and optimization - a review,” *Journal of Petroleum Science and Engineering*, vol. 183, p. 106332, 2019.
- [13] T. Warren, “Penetration-rate performance of roller-cone bits,” March 1987.
- [14] H. Xu, M. Fear, O. Whelehan, J. Ford, and J. Jensen, “How to organise mud logging data for modelling rate of penetration,” March 1995.
- [15] E. Bjornsson, B. Hucik, G. Szutiak, L. Brown Jr., H. Evans, D. Curry, and P. Perry, “Drilling optimization using bit selection expert system and rop predic-

- tion algorithm improves drilling performance and enhances operational decision making by reducing performance uncertainties,” September 2004.
- [16] H. U. Caicedo, W. M. Calhoun, and R. T. Ewy, “Unique rop predictor using bit-specific coefficient of sliding friction and mechanical efficiency as a function of confined compressive strength impacts drilling performance,” February 2005.
- [17] H. Rahimzadeh, M. Mostofi, A. Hashemi, and K. Salahshoor, “Comparison of the penetration rate models using field data for one of the gas fields in persian gulf area,” June 2010.
- [18] T. Eren and M. E. Ozbayoglu, “Real-time drilling rate of penetration performance monitoring.” Offshore Mediterranean Conference, March 2011.
- [19] D. P. Moran, H. F. Ibrahim, A. Purwanto, and J. Osmond, “Sophisticated rop prediction technology based on neural network delivers accurate results sophisticated rop prediction technology based on neural network delivers accurate results,” November 2010.
- [20] M. Monazami, A. Hashemi, and M. Shahbazian, “Drilling rate of penetration prediction using artificial neural network: A case study of one of iranian southern oil fields,” *Oil and Gas Business*, no. 6, 2012.
- [21] R. Jahanbakhshi, R. Keshavarzi, and A. Jafarnejhad, “Real-time prediction of rate of penetration during drilling operation in oil and gas wells.” American Rock Mechanics Association, June 2012.

- [22] M. B. Diaz, K. Y. Kim, and H. S. Shin, “On-line prediction model for rate of penetration (rop) with cumulating field data in real time.” Korean Society for Rock Mechanics, May 2017.
- [23] F. Hadi, H. Altaie, and E. AlKamil, “Modeling rate of penetration using artificial intelligent system and multiple regression analysis,” November 2019.
- [24] P. Batruny, H. Yahya, N. Kadir, A. Omar, Z. Zakaria, S. Batamale, and N. Jayah, “Drilling in the digital age: An aproach to optimizing rop using machine learning,” November 2019.
- [25] A. Ahmed, A. Ali, S. Elkatatny, and A. Abdulraheem, “New artificial neural networks model for predicting rate of penetration in deep shale formation,” *Sustainability*, vol. 11, no. 22, November 2019.
- [26] H. Yavari, M. Sabah, R. Khosravanian, and D. A. Wood, “Application of an adaptive neuro-fuzzy inference system and mathematical rate of penetration models to predicting drilling rate,” *Iranian Journal of Oil and Gas Science and Technology*, vol. 7, no. 3, pp. 73–100, 2018.
- [27] Y. Zhu, Z. Yin, H. Luo, M. Li, J. Wang, and Y. Li, “Prediction and optimization of rop in the absence of geological and lithological information,” in *2020 IEEE Computing, Communications and IoT Applications (ComComAp)*, 2020, pp. 01–06.
- [28] Y. Zhou, X. Chen, H. Zhao, M. Wu, W. Cao, Y. Zhang, and H. Liu, “A novel rate of penetration prediction model with identified condition for the complex

- geological drilling process,” *Journal of Process Control*, vol. 100, pp. 30–40, 2021.
- [29] *Prediction of POR Based on Artificial Neural Network With Long and Short Memory (LSTM)*, ser. U.S. Rock Mechanics/Geomechanics Symposium, vol. All Days, 06 2021.
- [30] C. Gan, W.-H. Cao, K.-Z. Liu, and M. Wu, “A novel dynamic model for the online prediction of rate of penetration and its industrial application to a drilling process,” *Journal of Process Control*, vol. 109, pp. 83–92, 2022.
- [31] M. Nystad, B. S. Aadnøy, and A. Pavlov, “Micro-Testing While Drilling for Rate of Penetration Optimization: Experiments and Simulations,” *Journal of Offshore Mechanics and Arctic Engineering*, vol. 144, no. 3, p. 031801, 12 2021.
- [32] *An Intelligent Prediction Method of Horizontal Well ROP Based on Di-GRU*, ser. U.S. Rock Mechanics/Geomechanics Symposium, vol. All Days, 06 2023.
- [33] *Transformer-based deep learning model for accurate rate of penetration prediction in drilling*, ser. SEG International Exposition and Annual Meeting, vol. All Days, 08 2023.
- [34] Z. Cheng, F. Zhang, L. Zhang, S. Yang, J. Wu, T. Li, and Y. Liu, “A Sequential Feature-Based Rate of Penetration Representation Prediction Method by Attention Long Short-Term Memory Network,” *SPE Journal*, pp. 1–19, 10 2023.

- [35] H. I. Bilgesu, L. T. Tetrick, U. Altmis, S. Mohaghegh, and S. Ameri, “A new approach for the prediction of rate of penetration (rop) values,” Lexington, Kentucky, p. 5, Jan 1997, SPE.
- [36] J. Mendes, T. Fonseca, and A. Serapiao, “Applying a genetic neuro-model reference adaptive controller in drilling optimization,” *World Oil*, vol. 228, pp. 29–36, 10 2007.
- [37] C. Hegde, H. Daigle, H. Millwater, and K. Gray, “Analysis of rate of penetration (rop) prediction in drilling using physics-based and data-driven models,” *Journal of Petroleum Science and Engineering*, vol. 159, pp. 295 – 306, 2017.
- [38] S. Eskandarian, P. Bahrami, and P. Kazemi, “A comprehensive data mining approach to estimate the rate of penetration: Application of neural network, rule based models and feature ranking,” *Journal of Petroleum Science and Engineering*, vol. 156, pp. 605 – 615, 2017.
- [39] E. H. Austin, *Drilling Engineering Handbook*. Springer Netherlands, 1983.
- [40] M. E. Hossain, A. A. Al-Majed, A. Al-Majed, and M. E. Hossain, *Fundamentals of Sustainable Drilling Engineering*. Somerset, UNITED STATES: John Wiley & Sons, Incorporated, 2015.
- [41] R. F. Mitchell and S. Z. Miska, *Fundamentals of Drilling Engineering*, Richardson, UNITED STATES, 2010.

- [42] O. Ahmed, A. Adeniran, and A. Samsuri, “Rate of penetration prediction utilizing hydromechanical specific energy,” in *Drilling*, A. Samsuri, Ed. Rijeka: IntechOpen, 2018, ch. 6.
- [43] J. Duan, J. Zhao, L. Xiao, C. Yang, and H. Chen, “A rop prediction approach based on improved bp neural network.” IEEE, November 2014.
- [44] O. S. Ahmed, A. A. Adeniran, and A. Samsuri, “Computational intelligence based prediction of drilling rate of penetration: A comparative study,” *Journal of Petroleum Science and Engineering*, vol. 172, pp. 1 – 12, 2019.
- [45] C. Soares and K. Gray, “Real-time predictive capabilities of analytical and machine learning rate of penetration (rop) models,” *Journal of Petroleum Science and Engineering*, vol. 172, pp. 934 – 959, 2019.
- [46] S. B. Ashrafi, M. Anemangely, M. Sabah, and M. J. Ameri, “Application of hybrid artificial neural networks for predicting rate of penetration (rop): A case study from marun oil field,” *Journal of Petroleum Science and Engineering*, vol. 175, pp. 604 – 623, 2019.
- [47] C. Gan, W.-H. Cao, M. Wu, X. Chen, Y.-L. Hu, K.-Z. Liu, F.-W. Wang, and S.-B. Zhang, “Prediction of drilling rate of penetration (rop) using hybrid support vector regression: A case study on the shennongjia area, central china,” *Journal of Petroleum Science and Engineering*, vol. 181, p. 106200, 2019.

- [48] K. Kor and G. Altun, “Is support vector regression method suitable for predicting rate of penetration?” *Journal of Petroleum Science and Engineering*, vol. 194, p. 107542, 2020.
- [49] M. Mehrad, M. Bajolvand, A. Ramezanzadeh, and J. G. Neycharan, “Developing a new rigorous drilling rate prediction model using a machine learning technique,” *Journal of Petroleum Science and Engineering*, vol. 192, p. 107338, 2020.
- [50] S. N. Bezminabadi, A. Ramezanzadeh, S.-M. Esmail Jalali, B. Tokhmechi, and A. Roustaei, “Effect of Rock Properties on ROP Modeling Using Statistical and Intelligent Methods: A Case Study of an Oil Well in Southwest of Iran,” *Archives of Mining Sciences*, vol. 62, no. 1, pp. 131–144, Mar. 2017.
- [51] M. J. Fear, “How to improve rate of penetration in field operations,” *SPE Drilling & Completion*, vol. 14, no. 01, pp. 42–49, Mar 1999, sPE.
- [52] M. L. Boucher, A. Schen, C. R. Ivie, and B. P. Jarvis, “Apparatus for weight on bit measurements, and methods of using same,” Patent US 6 957 575, 10 25, 2005.
- [53] J. Hess, “Pipe sticking prediction using lwd real-time measurements,” 2016.
- [54] F. Agin, R. Khosravanian, M. Karimifard, and A. Jahanshahi, “Application of adaptive neuro-fuzzy inference system and data mining approach to predict lost circulation using doe technique (case study: Maroon oilfield),” *Petroleum*,

- vol. 6, no. 4, pp. 423–437, 2020, sI: Artificial Intelligence (AI), Knowledge-based Systems (KBS), and Machine Learning (ML).
- [55] M. Sabah, M. Talebkeikhah, F. Agin, F. Talebkeikhah, and E. Hasheminasab, “Application of decision tree, artificial neural networks, and adaptive neuro-fuzzy inference system on predicting lost circulation: A case study from marun oil field,” *Journal of Petroleum Science and Engineering*, vol. 177, pp. 236–249, 2019.
- [56] A. K. Abbas, N. A. Al-haideri, and A. A. Bashikh, “Implementing artificial neural networks and support vector machines to predict lost circulation,” *Egyptian Journal of Petroleum*, vol. 28, no. 4, pp. 339–347, 2019.
- [57] *Study on Intelligent Prediction for Risk Level of Lost Circulation While Drilling Based on Machine Learning*, ser. U.S. Rock Mechanics/Geomechanics Symposium, vol. All Days, 06 2018.
- [58] A. J. Stumpf, *Drilling Hazards*. Cham: Springer International Publishing, 2018, pp. 1–4.
- [59] D. Pritchard, P. L. York, S. Beattie, and D. Hannegan, “Drilling hazard management: Excellent performance begins with planning,” [Accessed 02-05-2024].
- [60] L. W. Lake and R. F. Mitchell, *Petroleum Engineering Handbook : Drilling Engineering*, Richardson, UNITED STATES, 2006.
- [61] G. Yarim, R. J. Uchytel, R. B. May, A. Trejo, and P. Church, “Stuck pipe prevention—a proactive solution to an old problem,” 2007.

- [62] M. Kruszewski and V. Wittig, “Review of failure modes in supercritical geothermal drilling projects. geothermal energy,” *Geothermal Energy*, vol. 6, no. 28, 2018.
- [63] H. Jiang, G. Liu, J. Li, T. Zhang, C. Wang, and X. Ling, “An innovative diagnosis method for lost circulation with unscented kalman filter,” *Journal of Petroleum Science and Engineering*, vol. 166, pp. 731–738, 2018.
- [64] *Machine Learning Prediction of the Lost Circulation Events at the Well Planning Stage*, ser. Offshore Technology Conference Asia, vol. Day 2 Wed, February 28, 2024, 02 2024.
- [65] A. E. Omrani, M. A. Franchek, K. Grigoriadis, and R. Tafreshi, “Model-based early gas kick and well loss detection.” IEEE, 2016.
- [66] K. Salminen, C. Cheatham, M. Smith, and K. Valiulin, “Stuck pipe prediction using automated real-time modeling and data analysis,” 2016.
- [67] A. Alshaikh, A. Magana-Mora, S. A. Gharbi, and A. Al-Yami, “Machine learning for detecting stuck pipe incidents: Data analytics and models evaluation,” 2019.
- [68] H. Brink, J. W. Richards, and M. Fetherolf, *Real-World Machine Learning*. Manning Publications Co., 2017.
- [69] B. V. Vishwas and A. Patel, “Time-series characteristics,” in *Hands-on Time Series Analysis with Python*. Apress, 2020, pp. 1–21.

- [70] I. Gridin, *Time series forecasting using deep learning combining pytorch, RNN, TCN, and deep neural network models to provide production-ready prediction solutions.* BPB Publications, 2022.
- [71] M. Awad and R. Khanna, *Efficient Learning Machines.* Apress, 2015.
- [72] M. Najjarpour, H. Jalalifar, and S. Norouzi-Apourvari, “The effect of formation thickness on the performance of deterministic and machine learning models for rate of penetration management in inclined and horizontal wells,” *Journal of Petroleum Science and Engineering*, vol. 191, p. 107160, 2020.
- [73] Y. Li and R. Samuel, “Prediction of penetration rate ahead of the bit through real-time updated machine learning models,” The Hague, The Netherlands, p. 8, Mar 2019, SPE.
- [74] M. Diaz, K. Y. Kim, J. Lee, and H.-S. Shin, “Prediction of rate of penetration with data from adjacent well using artificial neural network,” in *Geotechnics for Sustainable Infrastructure Development*, P. Duc Long and N. T. Dung, Eds. Singapore: Springer Singapore, 2020, pp. 517–522.
- [75] C. I. Noshi and J. J. Schubert, “Application of data science and machine learning algorithms for rop prediction: Turning data into knowledge.” Offshore Technology Conference, May 2019.
- [76] A. Al-AbdulJabbar, S. Elkatatny, A. Abdulhamid Mahmoud, T. Moussa, D. Al-Shehri, M. Abughaban, and A. Al-Yami, “Prediction of the rate of penetration

while drilling horizontal carbonate reservoirs using the self-adaptive artificial neural networks technique,” vol. 12, p. 1376, Feb 2020.

- [77] A. M. Alali, M. F. Abughaban, B. M. Aman, and S. Ravela, “Hybrid data driven drilling and rate of penetration optimization,” *Journal of Petroleum Science and Engineering*, p. 108075, 2020.
- [78] A. W. Trask, *Grokking Deep Learning*. Manning Publications Co., 2019.
- [79] P. Harrington, *Machine Learning in Action*. Manning Publications Co., 2012.
- [80] M. Brandewinder, *Machine Learning Projects for .NET Developers*. Apress, 2015.
- [81] C. C. Aggarwal, *Neural Networks and Deep Learning*. Springer International Publishing, 2018.
- [82] G. Rebala, A. Ravi, and S. Churiwala, *An Introduction to Machine Learning*. Springer International Publishing, 2019.
- [83] R. Genuer and J.-M. Poggi, *Random Forests with R*. Springer International Publishing, 2020.
- [84] F. M. Bianchi, E. Maiorino, M. C. Kampffmeyer, A. Rizzi, and R. Jenssen, *Recurrent Neural Networks for Short-Term Load Forecasting*. Springer International Publishing, 2017.
- [85] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” 2017.

- [86] M. Bizanti, “Drilling rate model for louisiana formations,” *Journal of Petroleum Science and Engineering*, vol. 2, no. 4, pp. 261 – 267, 1989.
- [87] M. C. Seifabad and P. Ehteshami, “Estimating the drilling rate in ahvaz oil field,” *Journal of Petroleum Exploration and Production Technology*, vol. 3, no. 3, pp. 169–173, Sep 2013.
- [88] M. Keshavarz Moraveji and M. Naderi, “Drilling rate of penetration prediction and optimization using response surface methodology and bat algorithm,” *Journal of Natural Gas Science and Engineering*, vol. 31, pp. 829 – 841, 2016.
- [89] M. Khandelwal and D. J. Armaghani, “Prediction of drillability of rocks with strength properties using a hybrid ga-ann technique,” *Geotechnical and Geological Engineering*, vol. 34, no. 2, pp. 605–620, Apr 2016.
- [90] A. T. Tunkiel, D. Sui, and T. Wiktorski, “Reference dataset for rate of penetration benchmarking,” *Journal of Petroleum Science and Engineering*, vol. 196, p. 108069, 2021.
- [91] O. Hazbeh, S. K. ye Aghdam, H. Ghorbani, N. Mohamadian, M. Ahmadi Alvar, and J. Moghadasi, “Comparison of accuracy and computational performance between the machine learning algorithms for rate of penetration in directional drilling well,” *Petroleum Research*, vol. 6, no. 3, pp. 271–282, 2021.
- [92] E. Brenjkar and E. Biniiaz Delijani, “Computational prediction of the drilling rate of penetration (rop): A comparison of various machine learning approaches

- and traditional models,” *Journal of Petroleum Science and Engineering*, vol. 210, p. 110033, 2022.
- [93] M. A. Ben Aoun and T. Madarász, “Applying machine learning to predict the rate of penetration for geothermal drilling located in the utah forge site,” *Energies*, vol. 15, no. 12, 2022.
- [94] *Enhanced Rate of Penetration Prediction with Rock Drillability Constraints: A Machine Learning Approach*, ser. U.S. Rock Mechanics/Geomechanics Symposium, vol. All Days, 06 2022.
- [95] F. Zhou, H. Fan, Y. Liu, H. Zhang, and R. Ji, “Hybrid model of machine learning method and empirical method for rate of penetration prediction based on data similarity,” *Applied Sciences*, vol. 13, no. 10, 2023.
- [96] X. Chen, C. Weng, X. Du, J. Yang, D. Gao, and R. Wang, “Prediction of the rate of penetration in offshore large-scale cluster extended reach wells drilling based on machine learning and big-data techniques,” *Ocean Engineering*, vol. 285, p. 115404, 2023.
- [97] S. Jiao, W. Li, Z. Li, J. Gai, L. Zou, and Y. Su, “Hybrid physics-machine learning models for predicting rate of penetration in the halahatang oil field, tarim basin,” *Scientific Reports*, vol. 14, no. 1, p. 5957, Mar 2024.
- [98] T. Yehia, M. Gasser, H. Ebaid, N. Meehan, and E. R. Okoroafor, “Comparative analysis of machine learning techniques for predicting drilling rate of penetration

- (rop) in geothermal wells: A case study of forge site,” *Geothermics*, vol. 121, p. 103028, 2024.
- [99] A. Alsaihati, S. Elkatatny, and H. Gamal, “Rate of penetration prediction while drilling vertical complex lithology using an ensemble learning model,” *Journal of Petroleum Science and Engineering*, vol. 208, p. 109335, 2022.
- [100] M. A. Encinas, A. T. Tunkiel, and D. Sui, “Downhole data correction for data-driven rate of penetration prediction modeling,” *Journal of Petroleum Science and Engineering*, vol. 210, p. 109904, 2022.
- [101] S. A. Shoraka, S. R. Shadizadeh, and M. P. Shahri, “Prediction of stuck pipe in iranian south oil fields using multivariate statistical analysis,” 2011.
- [102] A. Chamkalani, M. P. Shahri, and S. Poordad, “Support vector machine model: A new methodology for stuck pipe prediction,” 2013.
- [103] A. A. Alshaikh, M. K. Albassam, S. H. Al Gharbi, and A. S. Al-Yami, “Detection of stuck pipe early signs and the way toward automation,” 2018.
- [104] A. Murillo, J. Neuman, and R. Samuel, “Pipe sticking prediction and avoidance using adaptive fuzzy logic modeling,” 2009.
- [105] I. A. Al-Baiyat and L. Heinze, “Implementing artificial neural networks and support vector machines in stuck pipe prediction,” 2012.
- [106] Q. Zhu, Z. Wang, and J. Huang, “Stuck pipe incidents prediction based on data analysis,” 2019.

- [107] A. Magana-Mora, S. Gharbi, A. Alshaikh, and A. Al-Yami, “Accupipepred: A framework for the accurate and early detection of stuck pipe for real-time drilling operations,” 2019.
- [108] *Advanced Real-Time Gas Kick Detection Using Machine Learning Technology*, ser. International Ocean and Polar Engineering Conference, vol. All Days, 06 2019.
- [109] *Time-Series Data Augmentation Techniques for Improving Automated Drilling Dysfunction Classifiers*, ser. SPE/IADC Drilling Conference and Exhibition, vol. Day 4 Thu, March 11, 2021, 03 2021, d041S016R003.
- [110] S. C. H. Geekiyanage, A. Tunkiel, and D. Sui, “Drilling data quality improvement and information extraction with case studies,” *Journal of Petroleum Exploration and Production*, vol. 11, no. 2, pp. 819–837, Feb 2021.
- [111] Y. Liu, F. Zhang, S. Yang, and J. Cao, “Self-attention mechanism for dynamic multi-step rop prediction under continuous learning structure,” 2023.
- [112] A. Tunkiel, *Prediction, interpolation and extrapolation of drilling data with Deep Learning*, 2022.
- [113] I. Albaiyat, “Implementing artificial neural networks and support vector machines in stuck pipe prediction,” 2012.
- [114] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2017.

- [115] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” 2014.
- [116] N. Wu, B. Green, X. Ben, and S. O’Banion, “Deep transformer models for time series forecasting: The influenza prevalence case,” 2020.
- [117] J. Soh and P. Singh, *Data Science Solutions on Azure*. Apress, 2020.
- [118] Y. Zhou, X. Chen, M. Wu, and W. Cao, “A comprehensive evaluation method for states adjustment priority of drilling process this work was supported by the national natural science foundation of china under grant 61733016, the national key rd program of china under grant 2018yfc0603405, the hubei provincial technical innovation major project under grant 2018aaa035, the 111 project under grant b17040, and the fundamental research funds for the central universities under grant cugcj1812.” *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 11 956–11 961, 2020, 21st IFAC World Congress.
- [119] A. Al-AbdulJabbar, A. A. Mahmoud, S. Elkatatny, and M. Abughaban, “Artificial neural networks-based correlation for evaluating the rate of penetration in a vertical carbonate formation for an entire oil field,” *Journal of Petroleum Science and Engineering*, vol. 208, p. 109693, 2022.
- [120] G. Rajagopalan, *Prepping Your Data with Pandas*. Berkeley, CA: Apress, 2021, pp. 147–241.

VITAE

- Name: Basirudin Rachman Bachtiar Djamaluddin
- Nationality: Indonesian
- Date of Birth: July the 8th 1978
- Email: *basirudin@gmail.com*
- Permanent Address: Safaniya Drive 525 - Dhahran, Saudi Arabia
- Bachelor Degree in Computer Science from IPB University, Indonesia (September 2003).
- Research Interests: Artificial Intelligence, Database, Cyber security.
- Has been working in the software engineering industry for more than 20 years.
- Published several papers in high reputable conferences related with Oil and Gas Industry.