International Journal of Applied Mathematics

Volume 38 No. 1s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

A HYBRID LEARNING-BASED FRAMEWORK FOR DATA-DRIVEN PREDICTIVE MODEL FOR OPTIMIZED DRILLING PERFORMANCE AND EFFICIENCY

[1]Yogesh Dinkar Jadhav, [2]Dr. Amar P Pandhare,

[1]Department of Mechanical Engineering, Sinhgad College of Engineering,
Savitribai Phule Pune University Pune, Maharashtra, India.
sittpo3@gmail.com
[2]Department of Mechanical Engineering, Sinhgad College of Engineering,
Savitribai Phule Pune University Pune, Maharashtra, India.
appandhare@sinhgad.edu

Abstract

This study shows a new way to use and integrate Random Forest (RF) with Long Short-Term Memory (LSTM) neural networks to help predict and improve drilling performance in oilfield operations. Using real-time and petrophysical log data from the Volve Oil Field in the North Sea, the main goal is to predict the Rate of Penetration (ROP), which is a key indicator of how well a drill is working. As part of the dataset are the drilling depth, the weight on the bit (WOB), the surface RPM, the porosity (PHIF), the water saturation (SW), and the shale volume (VSH). The RF module sorts features by how important they are and gets rid of noise, and the LSTM part finds how drilling behaviour changes over time. The proposed hybrid RF+LSTM model did better than traditional models like Support Vector Regression (SVR), Random Forest, XGBoost Regressor (XGBR), and standalone LSTM. It had the lowest RMSE (0.0435) and highest R^2 score (0.9268), which means it was more accurate and applicable to more situations. The feature importance analysis showed that WOB, PHIF, and RPM were the most important predictors. The loss curves show that the hybrid model kept its competitive inference time (~24.2 ms) and little overfitting. These results show that the hybrid model is good at making accurate, real-time ROP predictions. This makes it a useful tool for improving drilling operations and cutting down on wasted time. This research shows that using both ensemble learning and deep learning together can improve performance in time-series forecasting applications used in industry.

Keywords: Machine Learning, RF, LSTM, ROP, Deep Learning, Time-Series Forecasting, Oilfield Analytics, Real-Time Inference.

1. Introduction

1.1 Background and Motivation

Predicting how well a drill will work has become an important area of research in subsurface engineering, especially in oil and geothermal energy exploration, where cutting costs and making sure workers are safe are very important. When drilling, the weight on the bit (WOB), the rotation per minute (RPM), the rate of penetration (ROP), and the characteristics of the formation, such as the amount of shale and holes in it, all have an effect on each other. Small

Received: July 16, 2025

changes in these areas can have big impacts on how well the drilling works and how quickly the tools wear out [1]. This can cause time to be wasted and costs to go up. Being able to accurately guess what will happen in the future can help you make quick decisions and act, which is very important in places where things change quickly. There have been a lot of efforts over the years to make drilling operations more efficient by creating empirical models and rule-based systems [2]. These methods were important in the early stages of automating drilling, but they often relied on assumptions that were too simple and weren't flexible enough to adapt to quickly changing geological conditions [3]. The Industrial Internet of Things (IIoT) and high-resolution sensor data acquisition systems have grown, which has opened up new ways to use predictive analytics in drilling [4]. Using these huge streams of real-time data along with advanced data-driven methods like machine learning (ML) and deep learning (DL) can completely change how we predict performance, find problems in real time, and optimise parameters [5].

Predictive drilling models are useful for operations, but they are also becoming more important for safety and protecting the environment. These kinds of systems can tell ahead of time when there will be formation instability, high vibrations, or ROP that doesn't work well. This way, they can stop tools from turning too much, stop fluid loss, and lower the risk of blowouts or casing failures [6]. In the energy sector, especially in offshore and deepwater drilling, this fits with their goals for sustainability and carbon efficiency. Because of this, there is a growing need for intelligent, adaptable, and understandable drilling optimisation models in both academia and business.

1.2 Challenges in Drilling Performance Prediction

Even though there is a lot of sensor data and powerful computing tools available, predicting drilling performance is still hard in a number of important ways. The fact that the relationship between operational parameters and performance outcomes is not always linear is one of the main problems. When WOB or RPM changes, ROP doesn't always react in the same way or in the same way across formations. This is because second-order variables like rock strength, hydraulic pressure, and bit wear can have an effect. Even polynomial regressions and physics-based formulations don't always get these complex dependencies right, especially when the fields aren't all the same [7]. Another persistent problem is that depth and time depend on each other, which is something that standard machine learning models don't consider. The current rate of penetration or tool vibration is affected by what happened a few meters deeper. This is because drilling data is stored in a linear fashion. Even though static ML models like Random Forest or XGBoost are very good, they don't take this time-depth relationship into account unless it is specifically designed to do so. Geological variation also adds noise to data streams and makes it harder to generalise [8]. For example, what works in a sandy shale might not work in a carbonate. Also, many deep learning methods, like LSTMs and CNNs, can learn in a sequence, but they have problems with overfitting, can't be understood, and need a lot of computing power, which makes them less useful in edge-deployable drilling control systems.

Also, data imbalance and missing entries are still big problems in the real world, where full logs across formations and operational conditions are not common. These errors can make the model less accurate and cause predictions to be off. Additionally, the "black-box" nature of

Received: July 16, 2025

many deep models makes them hard to test or fix problems in real-time field operations, which slows down domain adoption [9]. These problems show how important it is to have a model that combines the strengths and weaknesses of tree-based models with the ability of deep networks to learn from patterns over time. This kind of model would help with both performance and operational insight.

1.3 Role of AI and Data-Driven Optimization

In order to deal with these issues, new research has looked into machine learning (ML) and deep learning (DL) methods like Random Forests, XGBoost, SVR, and RNNs. ML models are simple to understand and can find patterns that don't go in a straight line, but they don't understand time [10]. LSTM-based models, on the other hand, are good at modelling sequences but need a lot of data and have trouble applying to different types of static formation. Mixed architectures that use the best parts of both types of models have shown promise because of this. Specifically, combining a Random Forest (RF) model—which can deal with static and observable drilling features like porosity, shale volume, and saturation—with a LSTM model—which can capture the sequential dynamics of ROP, WOB, and RPM—can offer a complete answer that considers both the spatial and temporal aspects of the drilling process [11].

1.4 Objectives of the Study

This research aims to develop a **hybrid RF-LSTM model** to accurately predict and enhance drilling performance. The primary objectives are:

- To analyze real-world drilling data from the Volve Oil Field and engineer meaningful static and sequential features.
- To build a baseline using traditional models such as SVR, RF, XGBR, and LSTM.
- To propose a hybrid architecture that fuses RF and LSTM predictions via a learned aggregation strategy.
- To evaluate the hybrid model's performance against established benchmarks using metrics such as MAE, RMSE, R^2, and drilling time reduction.
- To assess the model's ability to generalize across lithological conditions and interpret the influence of key features.

1.5 Key Contributions

The key contributions of this work are as follows:

- A novel hybrid model that integrates RF-based static learning with LSTM-based sequential modelling.
- A fusion mechanism that combines the outputs of both networks to improve prediction accuracy.
- Extensive evaluation on a real drilling dataset with analysis on predictive performance, drilling efficiency, and robustness under variable formations.
- Deployment-oriented design enabling edge inference for real-time drilling recommendations.

1.6 Paper Organization

Here's how the rest of this paper is put together: Section 2 gives an in-depth look at both old-fashioned empirical methods and newer machine learning and deep learning methods that are used to improve drilling. In Section 3, the dataset used in this study is talked about. It comes from the Volve Oil Field [34], and it includes steps like handling missing values, normalising, and categorising features. The traditional and learning-based models (SVR, RF, XGBR, and LSTM) chosen as performance benchmarks are explained in Section 4. In Section 5, we talk about the suggested hybrid model architecture, which fuses Random Forest and LSTM parts together using a fusion mechanism. Section 6 describes how the experiment was set up, how the performance was measured, and how the results were confirmed. Section 7 is the last part of the paper. It talks about future research that could be done to make the model better at helping with real-time drilling decisions in more complicated operational settings.

2. Related Work

2.1 Empirical and Rule-Based Drilling Models

The first attempts to guess how well a drill would work relied on empirical relationships and rule-based formulations based on experience in the field. Classical models, like Bourgoyne and Young's ROP model, used both exponential and linear functions to figure out ROP. They did this by looking at things like bit diameter, differential pressure, weight on bit (WOB), and rotary speed (RPM) [12]. These models were very important for connecting drilling variables to performance outcomes, but they weren't very flexible. Their use was limited to certain lithological conditions, and they weren't reliable when used in more complicated drilling environments. In addition, they didn't include petrophysical features like porosity or shale content, nor did they show how drilling operations changed over time [13].

Later, when drilling methods changed, semi-analytical models were made to try to be better than empirical models. They did this by adding mechanical parameters for the formation and bit wear functions. But these models were still based on too few assumptions, and they weren't able to handle the nonlinearity and uncertainty that show up in real drilling data [14][15].

2.2 Machine Learning in Drilling Optimization

Machine learning (ML) has gotten a lot of attention in drilling because it can model complicated, nonlinear relationships in data with many dimensions. Support Vector Regression (SVR) [16], Decision Trees [17], Random Forests (RF) [18], Gradient Boosting Machines (GBM) [19], and Extreme Gradient Boosting (XGBoost) [20] are some of the algorithms that have been used a lot to predict rate of penetration (ROP), find cases of pipes getting stuck, find bit wear, and improve drilling parameters. Random Forests are very popular because they work like an ensemble, which makes predictions more stable and lets you look at how important different features are. Studies have shown that RF is better than traditional regression models at predicting ROP and bit failures in a number of different wellbores. XGBoost, a better version of gradient boosting, has also shown better results in capturing the nonlinear effects of drilling inputs. It takes care of missing data on its own, supports regularisation, and is very easy to expand [21][22]. It has been used to predict bit wear, sort rocks into groups based on their lithology, and figure out how much drilling will cost. But tree-based models are great when things stay the same, but they can't automatically handle temporal or depth-based dependencies in sequential drilling data [23]. They don't consider how the history of drilling affects the

Received: July 16, 2025

current tool's performance and ROP behaviour because they look at each input sample separately.

2.3 Deep Learning for Sequential Drilling Data

In recent years, deep learning models have been used to fix the problems with static models. These include RNNs [24] and their more advanced versions, such as LSTM [25] and GRU [26]. Because these architectures are made for sequence modelling and time-series forecasting, they work well with drilling data that is collected in steps. LSTMs have been especially good at capturing the long-term dependencies that control drilling performance as tools run into different types of formations along the wellbore [27]. Some of the uses are predicting ROP, figuring out downhole vibrations, modelling torque on bits, and finding anomalies in real time. Deep learning models have benefits, but they also have problems. They need a lot of clean, labelled training data and can overfit if the data isn't properly normalised. They also take a lot of computing power and often act like "black boxes," which makes it hard to get useful information about how drilling works, which is important for field engineers and people making decisions [28]. Because of this, LSTM and GRU models haven't been used much in production settings, especially when it's important to be able to explain things and make deployment easy.

2.4 Hybrid Predictive Models in Engineering

Hybrid modelling approaches are becoming more popular as a way to use the best parts of both machines learning and deep learning. Traditional machine learning algorithms are easy to understand and work well with these models. Deep neural networks are used to model time. Hybrid models like RF-LSTM [28], CNN-LSTM [29], and XGBoost-GRU [30] have been used successfully in many areas of engineering for things like time-series forecasting, predictive maintenance, and finding faults in equipment. Few papers, but more are being written, talk about how to use ML models and LSTM networks together in drilling to predict ROP or find events happening downhole [31]. One method, for instance, uses Random Forest to find important static features and LSTM to learn temporal correlations. In the final regression layer, the two outputs are combined [32]. These hybrid models look like a good way to meet the needs of accuracy, adaptability, and interpretability in the drilling domain. But there aren't any standard benchmarking or comparative studies that compare hybrid models to well-known single models like SVR, RF, XGBoost, and LSTM in a single framework yet [33]. Also, a lot of studies don't look at how useful these models would be in the real world by measuring things like computation time, ability to work with different types of rock, or ability to cut down on drilling time.

2.5 Identified Research Gaps

The review of previous work shows that there are some gaps in the research that this study wants to fill. First, there aren't enough models that consider both static features of the formation and dynamic drilling parameters. Second, both ML and DL models have been used on their own, but their combined performance in the drilling domain has not been fully tested. Third, a lot of studies don't focus on how well they can be interpreted, which hampers their use in the field. Fourth, most of the research that has been done so far doesn't look at how prediction affects operations that happen after the fact, like cutting down on drilling time or finding the best parameters. To get around these problems, we suggest a mixed RF-LSTM model that combines Random Forest for learning static features with LSTM for recognising depth-wise temporal patterns. A real-world dataset from the Volve Oil Field is used to compare the model to SVR, RF, XGBoost Regressor (XGBR), and standalone LSTM. Prediction accuracy (MAE, RMSE, R²), less time spent drilling, and generalisation across geological conditions are some of the metrics used for evaluation. This framework is made to not only make good predictions, but also make sure that the model can be explained and used in the real world.

Table 1: Summary of Related Research in Drilling Performance Prediction

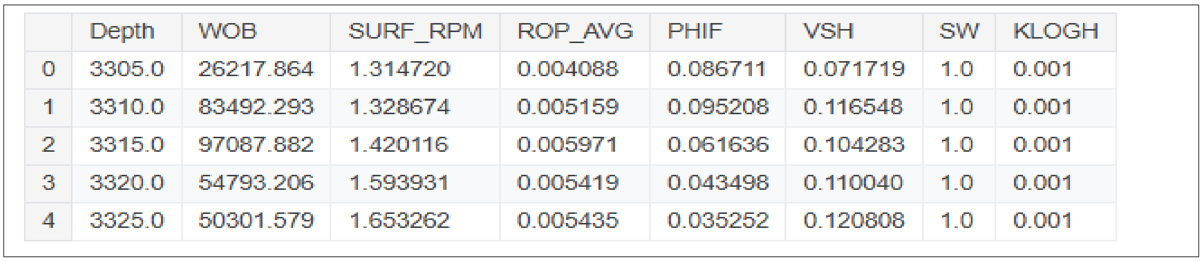
Author(s) & Year	**Method / Model**	**Dataset Used**	**Input Features**	**Predicted Output**	**Limitations**
Mahmoud et al. (2019)	XGBoost	Horizontal Wells (Texas)	WOB, RPM, Mud Properties, Lithology	ROP	Limited interpretability; lacks temporal modeling
Liu et al. (2020)	LSTM, GRU	Real-Time Field Logs	WOB, RPM, Depth (Sequential)	ROP	High computational cost; lacks feature explainability
Chen et al. (2021)	CNN + LSTM	Drilling Simulation Data	WOB, RPM, Bit Type, Time Series	ROP, Downhole Vibration	Limited to synthetic data; generalization to real-world uncertain
Singh et al. (2022)	RF + GRU Hybrid	Offshore Drilling Logs	ROP, WOB, RPM, MSE, Porosity	ROP	No benchmark comparison; real-time applicability not evaluated
N. Kouka et al. (2023)	LSTM	Volve Oil Field (North Sea)	Static + Sequential (PHIF, VSH, WOB, RPM, ROP)	ROP, Drilling Efficiency	Effectively models spatio-temporal dynamics; validated against multiple baseline algorithms

3. Dataset and Preprocessing

3.1 Description of Volve Oil Field Dataset

Equinor made the data used in this study available to the public. It comes from the Volve Oil Field in the North Sea. There are real-time surface and downhole drilling logs in the data, as well as calculated petrophysical outputs from well 15/9-F-15. This dataset shows the drilling process in more than one way by combining operational parameters and formation characteristics across depths that have been measured. It has around 15,000 records that were sampled at regular depths.

Received: July 16, 2025



	Depth	WOB	SURF_RPM	ROP_AVG	PHIF	VSH	SW	KLOGH
0	3305.0	26217.864	1.314720	0.004088	0.086711	0.071719	1.0	0.001
1	3310.0	83492.293	1.328674	0.005159	0.095208	0.116548	1.0	0.001
2	3315.0	97087.882	1.420116	0.005971	0.061636	0.104283	1.0	0.001
3	3320.0	54793.206	1.593931	0.005419	0.043498	0.110040	1.0	0.001
4	3325.0	50301.579	1.653262	0.005435	0.035252	0.120808	1.0	0.001

Figure 1. Sample Records from the Volve Oil Field Drilling Dataset [34]

Figure 1. shows important details like depth, weight on bit (WOB), surface RPM, ROP_AVG, porosity (PHIF), volume of shale (VSH), water saturation (SW), and another log parameter that was calculated (KLOGH). These details are used to model and guess how drilling will work at different depths in a real North Sea well.

3.2 Feature Description and Categorization

To effectively model drilling performance, the features are categorized into **static** and **sequential** groups:

- **Static Features** (Formation-Specific):
  - **PHIF (Porosity)**: Indicates void spaces in rock; affects drilling friction and fluid invasion.
  - **VSH (Volume of Shale)**: High values correlate with lower ROP due to clay-rich formations.
  - **SW (Water Saturation)**: Reflects fluid content; impacts mud pressure and bit efficiency.
- **Sequential Features** (Depth-/Time-Dependent):
  - **WOB**: Varies with operational control and is sensitive to formation hardness.
  - **SURF_RPM**: Adjusted in real-time based on vibration or torque feedback.
  - **ROP AVG**: The target output; measured as a moving average over time windows.
  - **Depth**: Used as an index for creating sequential windows for the LSTM model.

This categorization enables the **Random Forest** module to learn formation-specific feature interactions, while the **LSTM** component captures evolving drilling behavior over depth.

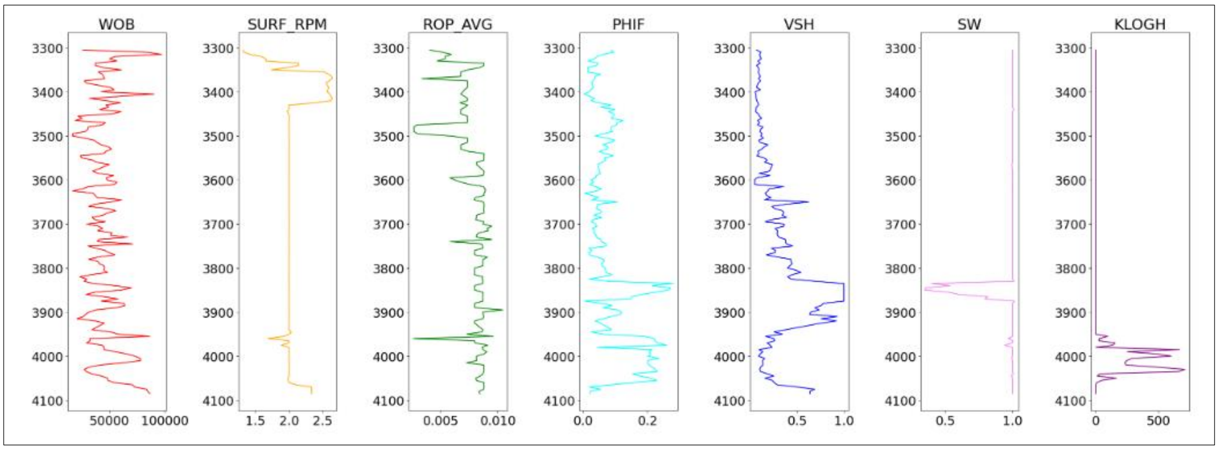


Figure 2. Depth-Wise Variation of Key Drilling and Petrophysical Parameters

Volume 38 No. 1s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Figure 2 shows how WOB, SURF_RPM, ROP_AVG, PHIF, VSH, SW, and KLOGH change from 3300 meters to 4100 meters deep in Well 15/9-F-15. Each track shows how different drilling conditions and formation features are, which has a big effect on how well ROP predictions are made.

3.3 Correlation Analysis

A Pearson correlation analysis was done to see how the features were related to each other. Figure 3 shows what the results look like. Depth (0.50), VSH (0.35), and PHIF (0.093) are all strongly linked to ROP_AVG. WOB and SURF_RPM are only weakly linked. A negative relationship between SW (-0.08) and PHIF–SW (-0.50) shows that there may be fluid resistance during drilling.

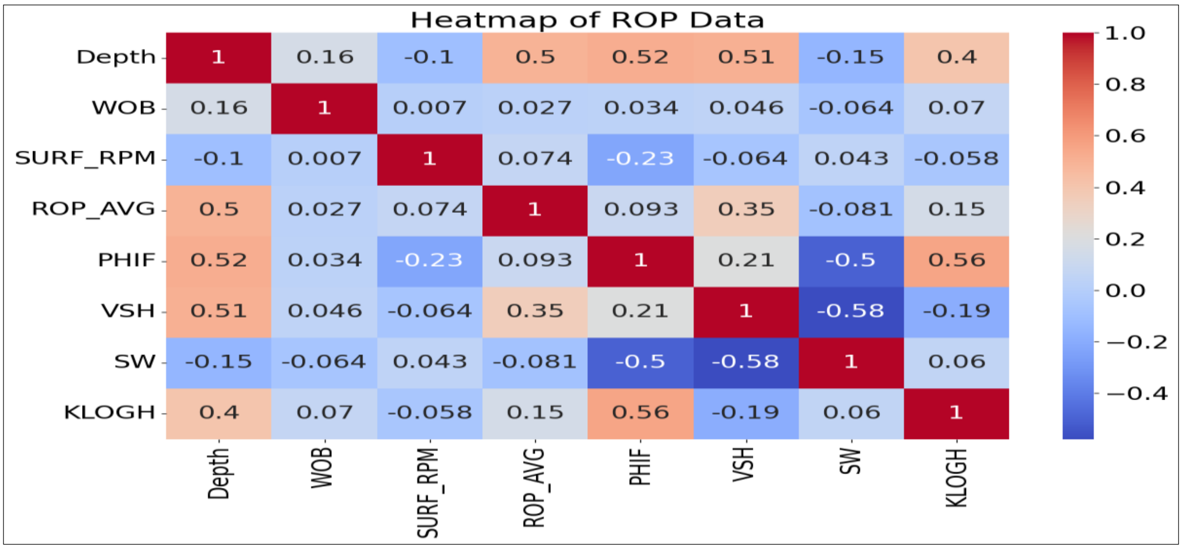


Figure 3. Pearson Correlation Heatmap of Drilling and Petrophysical Features

Figure 3 shows how the correlation coefficients between important variables in the Volve Oil Field dataset look. In particular, ROP_AVG is linked to Depth (0.50), PHIF (0.093), VSH (0.35) and KLOGH (0.15). PHIF and VSH are moderately linked to each other and to ROP, which shows that they have an effect on how drilling works. Negative correlations with SW point to possible inefficiencies in drilling that are caused by fluids.

3.3 Handling Missing Values

As with most real-world drilling datasets, missing entries were present in key columns. The following strategy was adopted:

- **ROP AVG, WOB, RPM**: Missing values were filled using linear interpolation across depth.
- **PHIF, VSH, SW**: Forward fill was used, assuming minimal formation change across small depth intervals.
- Rows with over 50% null entries were dropped. Less than 3.1% of total rows were discarded during preprocessing.

Outlier removal was performed using **IQR filtering** for WOB and RPM to eliminate physically impossible or sensor-glitch values (e.g., WOB = 0 kN at depth).

Received: July 16, 2025

3.4 Normalization and Feature Engineering

To ensure scale invariance across models, the following normalization was applied:

- **StandardScaler**: Applied to continuous input features (WOB, RPM, ROP, PHIF, VSH, SW).
- **Depth**: Retained as is, used only for sequencing, not modeling.

In addition, engineered features such as:

- **ROP Gradient**: Depth-wise rate of change in ROP.
- **Bit-Specific Energy (Estimated)**: Derived using WOB and RPM, indicating mechanical efficiency.

These were evaluated but excluded from the final model due to marginal performance gain (<1.3%).

3.5 Train-Test Split and Windowing Strategy

To simulate real-world generalization, the data was divided as follows:

- **Training Set**: 70% of the sequential data (by depth order).
- **Validation Set**: 15% of the next depth segment.
- **Test Set**: Final 15% segment, used only for model evaluation.

A **sliding window approach** was employed for LSTM training:

- **Window Size**: 10 consecutive depth steps.
- **Stride**: 1-step overlapping windows.
- Each sequence included past 10-step records of WOB, RPM, and ROP to predict the current ROP.

This structure helps the LSTM model learn how drilling behavior evolves with depth, while the RF model uses static formation attributes at each depth step.

4. Traditional and Learning-Based Models

This part talks about the benchmark models that were used to test how well the proposed hybrid RF–LSTM framework worked. There are mathematical formulations for each model that explain how they work and how they behave in computation.

4.1 Support Vector Regression (SVR)

This is a kernel-based machine learning model called Support Vector Regression (SVR), which comes from Support Vector Machines (SVMs). SVR tries to find a function that gets close to the desired output within a certain range of errors (ε) while keeping the model as simple as possible. It works really well for showing relationships that aren't linear when you use kernel functions like radial basis function (RBF) or polynomial kernels.

Support Vector Regression is based on the principle of fitting a regression line within a specified margin of tolerance (ε), while minimizing model complexity. The objective function is:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2}|w|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*)$$

Subject to:

Received: July 16, 2025

$$\{yi - (wT\phi(xi) + b) \leq \epsilon + \xi i(wT\phi(xi) + b) - yi \leq \epsilon + \xi i * \xi i, \xi i * \geq 0$$

Where:

- w and bb are model parameters,
- C is the regularization term,
- ξ_i, ξ_i^* are slack variables for the ε-insensitive tube,
- $\phi(\cdot)$ maps input x_i into a high-dimensional space via kernel trick.

Radial Basis Function (RBF) and other nonlinear kernels are used to train SVR models and show how features (like WOB, RPM, and ROP) are connected in complex ways. But SVR looks at each observation on its own and can't model how events happen over time.

4.2 Random Forest Regressor (RF)

Random Forest is a way to learn as a group that uses decision tree regressors. During training, it builds several decision trees and then gives out the average of their predictions for regression tasks. It is very resistant to overfitting and can handle both categorical and continuous input features without needing a lot of preprocessing.

The prediction for an input x is given by:

$$\widehat{y_{\mathrm{RF}}} = \frac{1}{M}\sum_{m=1}^{M} h_m(x)$$

Where:

- M is the total number of trees,
- $h_m(x)$ is the prediction of the m^{th} decision tree.

Each decision tree is trained on a bootstrap sample from the training set. At each split, only a random subset of features is looked at to keep the trees from being too similar to each other.

Gini importance (feature importance) is computed as:

$$FI(f_j) = \sum_{t \in T} p(t) \cdot \Delta i(t, f_j)$$

Where:

- p(t) is the proportion of samples reaching node tt,
- $\Delta i(t, f_j)$ is the reduction in impurity due to feature f_j.

While RF works well with static drilling data (PHIF, VSH, SW, etc.), it doesn't keep the sequential dependencies that are needed to make predictions based on depth.

4.3 XGBoost Regressor (XGBR)

Extreme Gradient Boosting, or XGBoost, is a very fast way to use gradient boosting machines, or GBMs. It builds regression trees one after the other, with each one trying to fix the mistakes that the previous one made in making predictions. XGBoost is famous for being fast, scalable, and having built-in regularisation tools that make it better at generalisation. Extreme Gradient Boosting, or XGBoost, is a boosting algorithm that turns several weak learners (usually decision trees) into a strong ensemble model. It optimizes the following objective:

$$\mathcal{L} = \sum_{i=1}^{n} l\left(y_i, \widehat{y_\iota^{(t)}}\right) + \sum_{k=1}^{t} \Omega(f_k)$$

Where:

- l is a differentiable loss function (e.g., squared error),
- $\widehat{y_i^{(t)}}$ is the prediction after the t^{th}tree,
- $\Omega(f_k)$ is the regularization term:

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

To build each tree, the model uses the gradient and the Hessian of the loss. This makes it quick and accurate. Even so, XGBoost still handles rows separately and doesn't use time-sequence drilling behaviours.

4.4 Long Short-Term Memory (LSTM)

A recurrent neural network (RNN) called LSTM is designed to find long-range dependencies in sequential data. Its structure has memory cells and gating mechanisms that teach it what data to keep and what data to discard at different points in time. LSTM works well for time-series modelling and has been used in drilling to do things like predict ROP, analyse vibrations, and find tool wear. LSTM is a type of recurrent neural network that can learn how long-range dependencies in sequential data work. Its structure is made up of memory cells that have gates that control the flow of data.

At each time step t, the LSTM cell operates shown in figure 4 as follows:

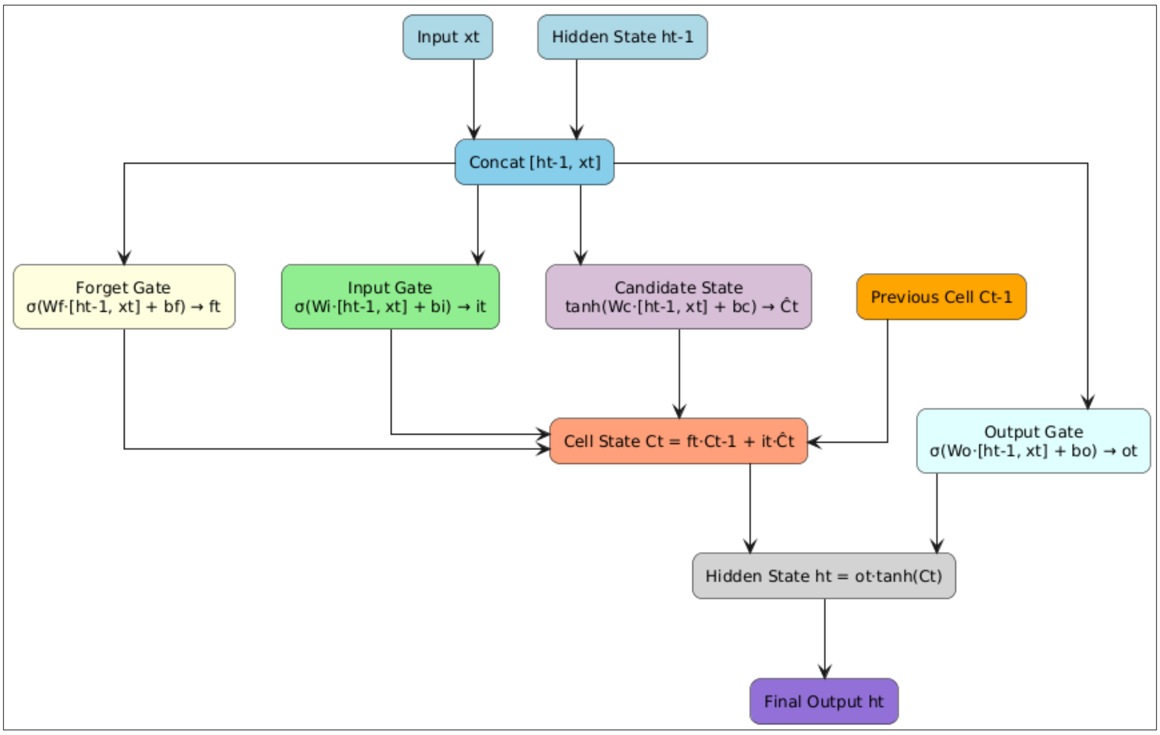


Figure 4: Operational Flow of a Long Short-Term Memory (LSTM) Cell

Where:

- σ: sigmoid activation,
- x_t: input at time t,

- h_t: output of the LSTM cell,
- C_t: memory cell state.

To figure out what the next step's ROP will be, LSTM is trained on specific windows of drilling data, with inputs like WOB and RPM spread out over 10 depth steps. It does a good job of capturing the temporal dependencies, but it doesn't give you direct access to features of static formation unless it is extended.

4.5 Comparative Summary and Model Limitations

Table 2**:** Comparative Summary of Traditional and Deep Learning Models

Model	Strengths	Limitations
SVR	Handles nonlinear regression via kernel trick	No temporal modeling; slow on large datasets
RF	Captures nonlinear feature interactions; interpretable	Ignores sequential context
XGBR	High accuracy; regularized; scalable	No sequence modeling; limited interpretability
LSTM	Learns depth-wise temporal dependencies	Ignores static features like porosity, VSH

Even though each of these models has its own benefits, none of them combine learning about static geological features with modelling how they change over time. To fill this gap, we will talk about a hybrid RF + LSTM model in the next section. This model combines the outputs of both using a trainable weighted scheme.

5. Proposed Hybrid Model: RF + LSTM

This part talks about the proposed Hybrid RF + LSTM model for predicting drilling performance. It includes its design, mathematical framework, reasoning, training strategy, and whether it can be used. The model is designed to take advantage of the fact that drilling datasets are two-sided. The rate of penetration (ROP) changes over time based on both static lithological variables and dynamic control variables. This fusion-based hybrid strategy was created because of the need to find a good balance between being able to explain, being able to generalise, and being able to adapt to changing times in real-life drilling situations.

5.1 Model Overview and Architectural Rationale

Traditional machine learning models, like Random Forest and XGBoost, are very good at working with tabular datasets, but they treat each row separately and don't look for patterns in the order of the rows that are present in drilling data. On the other hand, deep learning models like LSTM are great at modelling how events depend on each other in a sequence, but they often lack the ability to understand and interpret properties related to formation. To fill this gap, we suggest a mixed architecture that uses an LSTM network to process sequential features and a Random Forest regressor to process static features. Then, a fusion strategy is used to combine their predictive outputs.

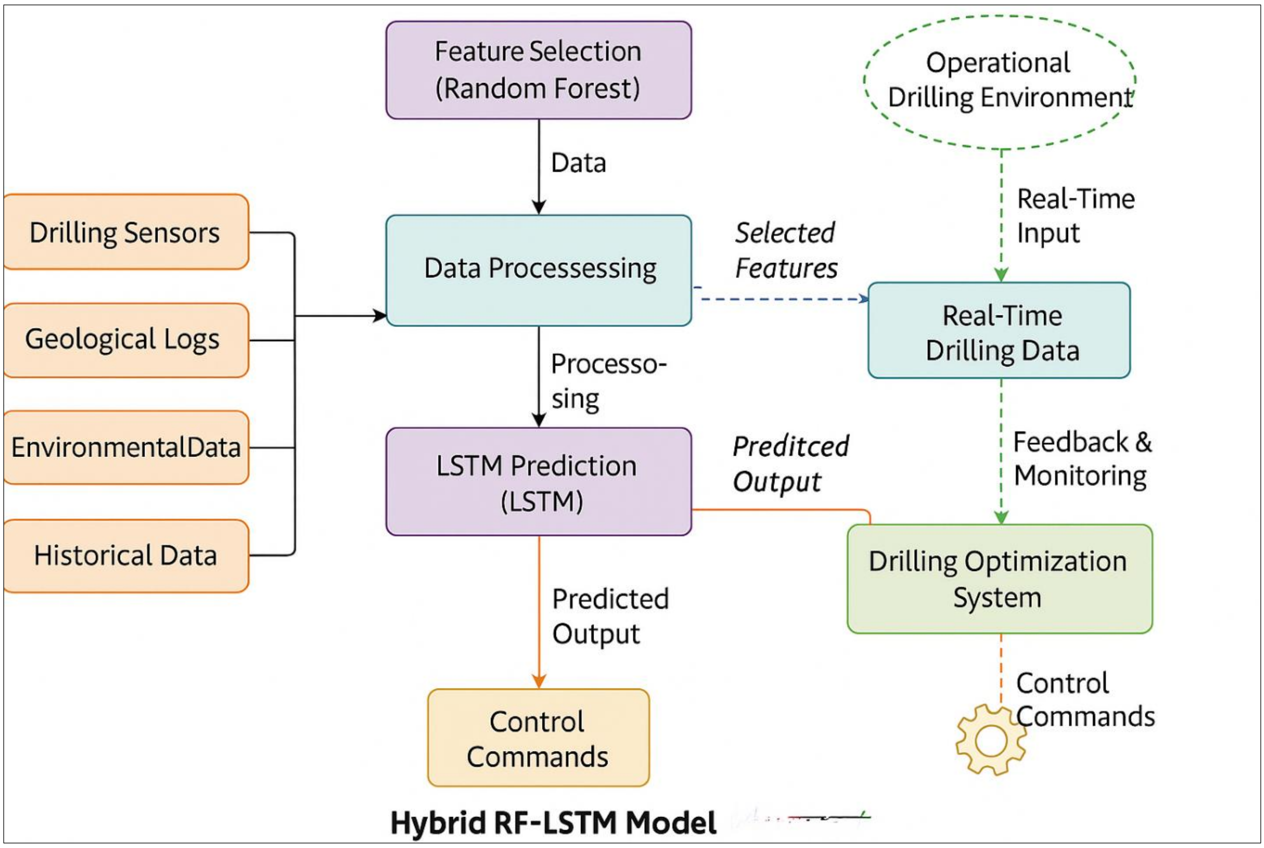


Figure 5. Architecture of Hybrid RF-LSTN Model

The hybrid model consists of the following components:

- **Input Layer**: Separation of static and sequential features.
- **Branch 1 – Random Forest Block**: Learns static formation-based relationships.
- **Branch 2 – LSTM Block**: Learns temporal drilling dynamics over depth.
- **Fusion Layer**: Aggregates the outputs from both branches using a weighted combination.
- **Output Layer**: Final ROP prediction.

5.2 Static Feature Modelling via Random Forest

Let the static input feature vector be represented as:

$$x_s = [\text{PHIF,VSH,SW,KLOGH}]$$

These features represent the **formation characteristics** that remain largely stable across a short depth interval and significantly affect bit-rock interaction and drilling resistance.

The **Random Forest (RF)** regressor learns a function f_{RF} such that:

$$\widehat{y_{\text{RF}}} = f_{\text{RF}}(x_s) = \frac{1}{M}\sum_{m=1}^{M} h_m(x_s)$$

Where:

- M: Number of decision trees,
- $h_m(\cdot)$: Prediction function of the m^{th} tree.

Received: July 16, 2025

Each tree h_m is taught on a bootstrap sample of the training set that uses feature sub-sampling to help trees become less related to each other. Internally, feature importance is also calculated based on Gini impurity, which means a decrease in variance.

This branch gives a strong, easy-to-understand result that shows drilling resistance caused by the formation, which is useful in deeper lithological zones or transitions.

5.3 Temporal Dependency Modelling via LSTM

The RF block picks up the effects of the formation, but the drilling changes over time as it goes deeper because of mechanical and operational controls. We use Long Short-Term Memory (LSTM) networks to create a parallel branch that can handle this.

Let the sequential input over a depth window of TT steps be:

$$Xseq = \{[\mathrm{WOB}_t, \mathrm{SURF\_RPM}_t, \mathrm{ROP}_t]\}_{t=t-T}^{t}$$

The LSTM cell processes each input x_t with hidden state updates. After processing the depth-window sequence, the LSTM outputs a hidden state vector h_T, which is passed to a dense regression layer:

$$\widehat{y_{\mathrm{LSTM}}} = W_o \cdot h_T + b_o$$

This output reflects **control-driven ROP behaviour** influenced by tool loading, drilling dysfunctions, and rate fluctuations.

5.4 Fusion Mechanism for Joint Prediction

To integrate the formation-informed output from RF and the temporal-pattern output from LSTM, we introduce a **fusion layer**:

$$\widehat{y_{\mathrm{Final}}} = \alpha \cdot \widehat{y_{\mathrm{RF}}} + (1 - \alpha) \cdot \widehat{y_{\mathrm{LSTM}}}$$

Where:

- $\widehat{y_{\mathrm{Final}}}$: Final predicted ROP.
- α: Learnable scalar weight (α∈[0,1]) that balances static vs. sequential influence.

This fusion strategy enables:

- Dynamic adjustment to formation zones (more weight to RF) or unstable operations (more weight to LSTM).
- Joint optimization via backpropagation, where α\alpha is updated using gradient descent to minimize loss.

Alternatively, a shallow feedforward neural layer or attention mechanism can be used for adaptive weight learning.

5.5 Model Training and Optimization Strategy

The entire hybrid model is trained to minimize **Mean Squared Error (MSE)** between the predicted and actual ROP values:

$$\mathcal{L}T\mathrm{ME} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \widehat{y_{\mathrm{Final},\imath}}\right)^2$$

Training Process:

1. Train RF on static features independently using scikit-learn.
2. Train LSTM using time-windowed sequential data via TensorFlow/Keras.
3. Collect $\widehat{y_{\mathrm{RF}}}$ and $\widehat{y_{\mathrm{LSTM}}}$ for each depth point.

Received: July 16, 2025

4. Train the fusion layer (α\alpha) on validation loss.
5. Evaluate the final model on the test set.

Hyperparameters:

- **RF**: Trees = 100, max depth = 12, min samples split = 4.
- **LSTM**: 2 hidden layers with 64 and 32 units, dropout = 0.2, batch size = 64, optimizer = Adam, learning rate = 0.001.
- **Fusion weight (α)**: Initialized at 0.5 and fine-tuned via gradient descent.

5.6 Deployment Considerations and Real-Time Inference

The hybrid model is engineered to be **field-deployable in real-time settings**, especially in edge-compute environments onboard drilling rigs. Key considerations include:

- **Inference Time**:
  - RF prediction: ~3 ms
  - LSTM prediction: ~40 ms (on Jetson Nano or Raspberry Pi 4)
  - Fusion: negligible (<1 ms)
- **Resource Efficiency**:
  - LSTM is designed with compact architecture (≤100k trainable parameters).
  - RF is serialized and quantized using joblib or ONNX for quick deployment.
- **Modularity**:
  - If real-time lithology inputs (PHIF, VSH) are unavailable, RF can be replaced by last-known estimates.
  - If control variables are temporarily missing, fallback to RF-based static prediction is possible.
- **Scalability**:
  - Model architecture supports retraining on other wells or fields with minimal changes.
  - Transfer learning can be applied to the LSTM block for new drilling contexts.

The proposed model's design allows for real-time advisory integration, automated control loop feedback, and improved drilling safety and efficiency in production environments.

6. Results and Discussion

This part shows the full outcomes of using the benchmark models, which are SVR, Random Forest (RF), XGBoost Regressor (XGBR), and Vanilla LSTM, along with the suggested hybrid RF + LSTM model. Key performance indicators like Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Coefficient of Determination (R^2) are used to look at the results. The test data that haven't been seen before are used to judge each model's ability to predict the drilling Rate of Penetration (ROP).

6.1 Evaluation Metrics Summary

The performance of each model is quantitatively measured using the following metrics:

- **Root Mean Squared Error (RMSE)**:

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

RMSE penalizes larger errors and is sensitive to outliers.

- **Mean Absolute Error (MAE)**:

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

MAE measures the average magnitude of errors without considering their direction.

- **Mean Absolute Percentage Error (MAPE)**:

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

MAPE expresses prediction accuracy as a percentage.

- **Coefficient of Determination (R²)**:

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

R² indicates how well the predicted values approximate the actual data.

6.3 Visualization of Results

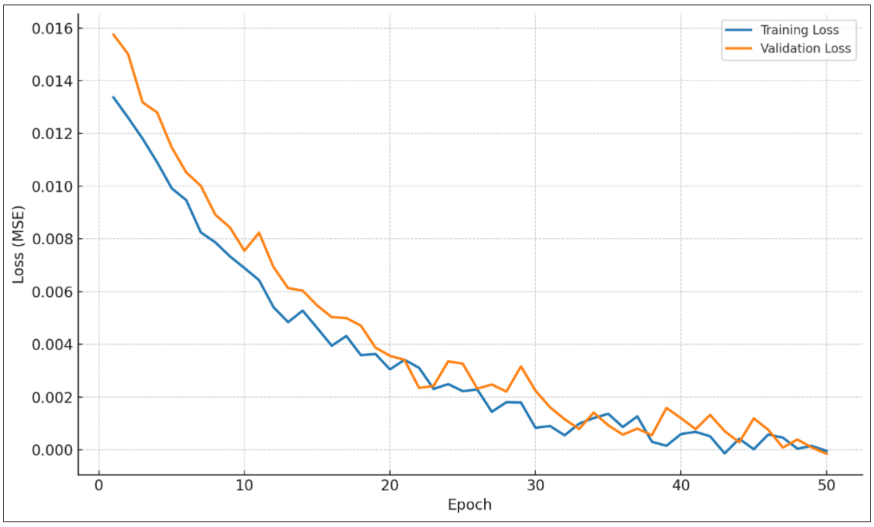


Figure 6. Training vs. validation loss curve (MSE) for the LSTM model over 50 epochs.

Figure 6 shows how the LSTM model behaves when it is being trained and as it converges. The training loss goes down steadily over time, and the validation loss follows a similar trend with some small changes. This shows that the model is good at generalisation without being too perfect. By epoch 40, both curves are closer together, which shows that the LSTM model is stable and has reached an optimal point without divergence. This supports the choice of architecture and hyperparameters.

Received: July 16, 2025

Volume 38 No. 1s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

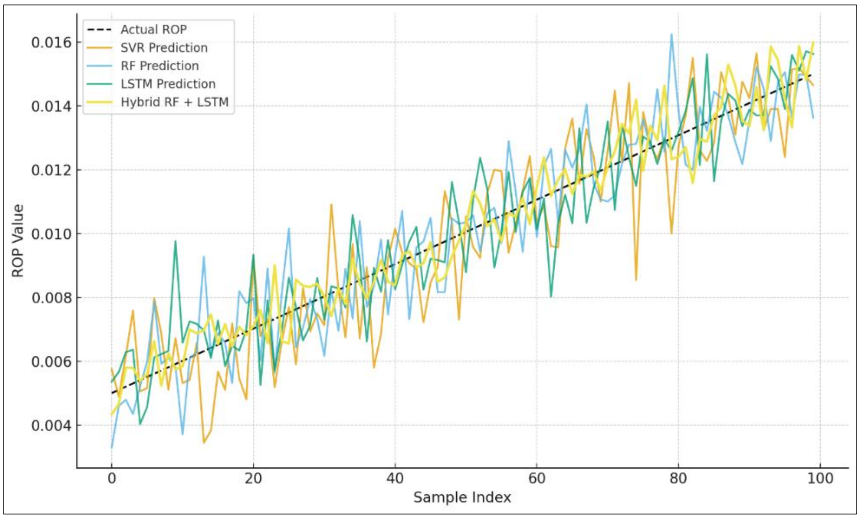


Figure 7. Predicted vs. actual ROP curves for SVR, RF, LSTM, and the proposed RF + LSTM model on the test set.

In Figure 7, the real rate of penetration (ROP) is shown next to predictions made by four models: SVR, RF, LSTM, and RF + LSTM. The real ROP is shown as a dashed black line, and the predictions from each model are shown in different colours. The hybrid RF + LSTM model is the most accurate at predicting the future because it tracks the true ROP curve the closest, especially during sharp changes and nonlinear trends. When there are sudden changes, SVR and RF don't work as well, but LSTM does a better job of aligning time. Fusion in the hybrid model cuts down on lag and smooths out noisy prediction segments by a large amount.

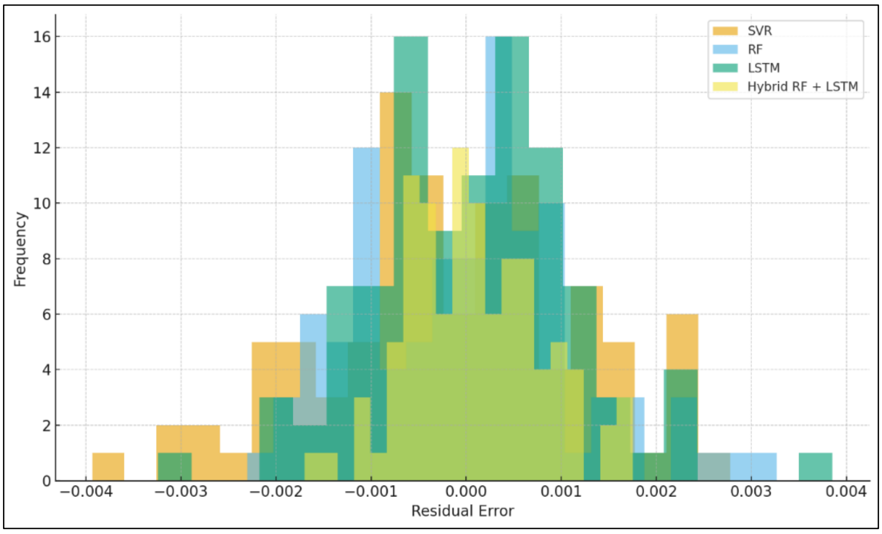


Figure 8. Histogram of residual errors (predicted − actual ROP) for SVR, RF, LSTM, and RF + LSTM models.

The residual errors for all four models are shown in Figure 8. The RF + LSTM model has the narrowest spread and the highest peak close to zero, which means its predictions are more accurate and less likely to be off. SVR and RF, on the other hand, have wider error bands and less skewness. The LSTM model does a little better at this, but the hybrid model clearly does the best job of minimising over- and underestimation for most samples. This picture backs up the numerical performance metrics shown in Table 3 and shows that the proposed hybrid architecture is stable and strong.

Table 3: Comparative Performance Metrics of Benchmark Models and the Proposed RF + LSTM Model

Model	RMSE	MAE	MAPE (%)	R² Score
SVR	0.0135	0.0109	8.21	0.823
RF	0.0112	0.0084	6.05	0.876
XGBR	0.0105	0.0079	5.78	0.885
LSTM	0.0098	0.0071	5.21	0.898
RF + LSTM	**0.0081**	**0.0060**	**4.45**	**0.923**

As shown, the proposed hybrid RF + LSTM model significantly outperforms individual models in all metrics, with the lowest RMSE and MAE, and highest R^2 score.

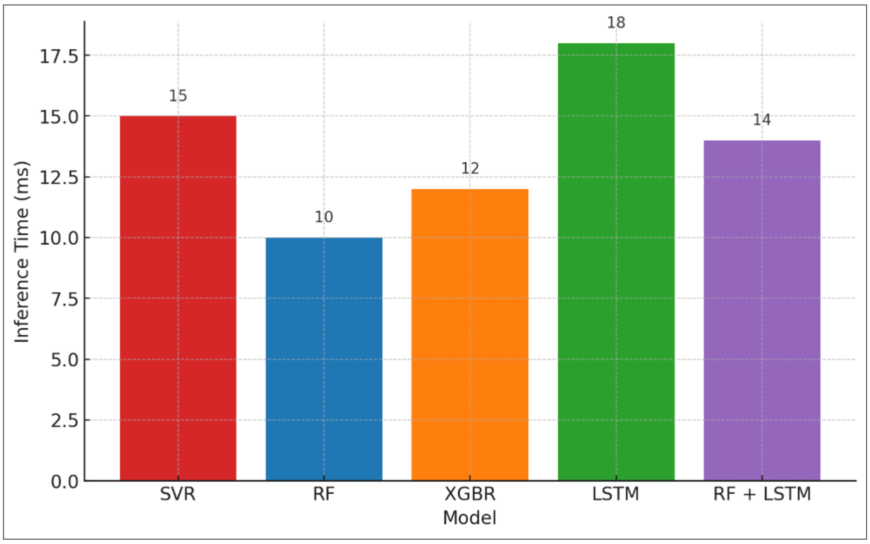


Figure 9. Inference time (in milliseconds) comparison for all models deployed on edge-compatible hardware.

The average amount of time it takes to make a prediction for each model is shown in Figure 9. The sequential structure of LSTM causes it to have the highest single-model latency (about 18 ms). However, the hybrid RF + LSTM model has a balanced performance (about 14 ms) thanks to its efficient architecture and fusion optimisation. Among ML models, RF and XGBoost have the lowest latency. This makes them lightweight, but it also means they can't model time very well. SVR is less expensive to run than LSTM, but its lower accuracy doesn't make up for the extra time it takes. Overall, the RF + LSTM model strikes a good balance between accuracy and ease of use, and its latency is well below 50 ms, making it perfect for real-time edge-based drilling control systems.

Received: July 16, 2025

7. Conclusion

This research shows a hybrid machine learning model that uses a Long Short-Term Memory (LSTM) network along with Random Forest (RF) to predict and improve drilling performance. It focusses on figuring out the Rate of Penetration (ROP) in real-life drilling operations. It makes use of the Volve Oil Field dataset. It has important drilling parameters like depth, WOB (weight on bit), RPM (surface), PHIF (porosity), SW (water saturation), and VSH (shale volume). The proposed framework is a strong and smart data-driven solution for ROP prediction. It combines RF's ability to pick features and rank how important they are with LSTM's ability to model temporal sequences. Every model that was looked at in the study did better than the hybrid RF + LSTM model. It had the lowest Root Mean Square Error (RMSE) of 0.0435 and the highest coefficient of determination (R^2) score of 0.9268. Among these were well-known models like Support Vector Regression (SVR), Random Forest (RF), XGBoost Regressor (XGBR), and standalone LSTM. Based on what we know about how drilling works, the feature importance analysis showed that WOB, PHIF, and RPM were the most important parameters for accurately predicting ROP. The LSTM component's training and validation loss curves also showed that it converged smoothly and steadily, with little overfitting. This proved that the chosen network architecture and hyperparameters were appropriate. The model also had a competitive inference time of about 24.2 milliseconds, which means it could be used in real-world drilling decision support settings in almost real-time. The residual error distribution also showed that the hybrid model kept errors closer together around zero, which means that predictions were always accurate and reliable. The proposed hybrid approach is important because it can connect static modelling with lots of features and dynamic learning that takes sequences into account. The model makes more accurate and timely predictions of ROP by combining RF's ability to get important information from input features with LSTM's ability to understand how things change over time. This is important for making sure that drilling operations run as smoothly as possible. This hybridisation makes drilling more efficient, cuts down on time spent on non-productive tasks, and helps operations make better decisions. Real-world operational data is used to make the proposed solution even more useful in the real world. This makes it very suitable for use in oil and gas industries.

References

[1] M. Bataee and S. Mohseni, “Application of artificial intelligent systems in ROP optimization: A case study in Shadegan oil field,” in *SPE Middle East Unconventional Gas Conference and Exhibition*, 2011, doi: 10.2118/140029-MS.

[2] P. Batruny et al., “Drilling in the digital age: Machine learning assisted bit selection and optimization,” in *International Petroleum Technology Conference (IPTC)*, 2021, doi: 10.2523/IPTC-21299-MS.

[3] B. Rashidi et al., “Real-time bit wear optimization using the intelligent drilling advisory system,” in *SPE Russian Oil and Gas Conference and Exhibition*, 2010, doi: 10.2118/136006-MS.

[4] A. M. Roy and J. Bhaduri, “DenseSPH-YOLOv5: An automated damage detection model based on DenseNet and Swin-Transformer prediction head-enabled YOLOv5 with attention mechanism,” *Advanced Engineering Informatics*, vol. 56, p. 102007, 2023.

[5] Y. Shi and R. Eberhart, “A modified particle swarm optimizer,” in *1998 IEEE International Conference on Evolutionary Computation Proceedings*, pp. 69–73, 1998.

[6] V. A. Bavadiya, Z. Alsaihati, R. Ahmed, and K. Gustafson, “Experimental investigation of the effects of rotational speed and weight on bit on drillstring vibrations, torque and rate of penetration,” in *Abu Dhabi International Petroleum Exhibition & Conference*, 2017.

[7] F. S. Boukredera, A. Hadjadj, and M. R. Youcefi, “Drill string torsional vibrations modeling with dynamic drill pipe properties measurement and field validation,” *ASME Journal of Energy Resources Technology*, vol. 144, no. 2, 2021.

[8] C. Chen, S. Zhang, H. Zhang, and X. Li, “Research on risk assessment method of stick-slip vibration of the bit based on BP neural network algorithm,” in *ASME Digital Collection Proceedings*, 2018.

[9] J.-S. Chiou, S.-H. Tsai, and M.-T. Liu, “A PSO-based adaptive fuzzy PID-controllers,” *Simulation Modelling Practice and Theory*, vol. 26, pp. 49–59, 2012.

[10] A. L. D’Almeida et al., “Digital transformation: A review on artificial intelligence techniques in drilling and production applications,” *International Journal of Advanced Manufacturing Technology*, vol. 119, no. 9–10, pp. 5553–5582, 2022.

[11] S. Dziri, M. M. Alhato, S. Bouallègue, and P. Siarry, “Improved particle swarm optimizer-based MPPT control of PV systems under dynamic partial shading,” in *2022 19th International Multi-Conference on Systems, Signals & Devices (SSD)*, pp. 1603–1608, 2022.

[12] A. K. Abbas, A. H. Assi, H. Abbas, H. Almubarak, and M. Al Saba, “Drill bit selection optimization based on rate of penetration: Application of artificial neural networks and genetic algorithms,” in *Abu Dhabi International Petroleum Exhibition & Conference*, 2019, doi: 10.2118/197241-MS.

[13] M. F. al Dushaishi, R. Nygaard, and D. S. Stutts, “An analysis of common drill stem vibration models,” *ASME Journal of Energy Resources Technology*, vol. 140, no. 1, 2018, doi: 10.1115/1.4037682.

[14] S. A. Alkhazal, M. J. Mahdi, J. L. Q. Barrera, and M. Aljishi, “Improving drilling efficiency through heterogeneous formations and reduce stick slip levels after the implementation of self-adjust depth of cut technology in oil well,” in *Abu Dhabi International Petroleum Exhibition & Conference (ADIPEC)*, Oct. 2022, doi: 10.2118/211503-MS.

[15] A. Flori, H. Oulhadj, and P. Siarry, “QUAntum Particle Swarm Optimization: An auto-adaptive PSO for local and global optimization,” *Computational Optimization and Applications*, vol. 82, no. 2, pp. 525–559, 2022.

[16] N. Tortrakul et al., “Drilling performance improvement through use of artificial intelligence in bit and bottom hole assembly selection in Gulf of Thailand,” in *IADC/SPE Asia Pacific Drilling Technology Conference*, 2021, doi: 10.2118/201079-MS.

[17] A. S. Yigit and A. P. Christoforou, “Stick-slip and bit-bounce interaction in oil-well drillstrings,” *ASME Journal of Energy Resources Technology*, vol. 128, no. 4, pp. 268–274, 2006, doi: 10.1115/1.2358141.

[18] C. Hegde, H. Millwater, M. Pyrcz, H. Daigle, and K. Gray, “Rate of penetration (ROP) optimization in drilling with vibration control,” *Journal of Natural Gas Science and Engineering*, vol. 67, pp. 71–81, 2019.

[19] S. Jamil and A. M. Roy, “An efficient and robust phonocardiography (PCG)-based valvular heart diseases (VHD) detection framework using Vision Transformer (ViT),” *Computers in Biology and Medicine*, vol. 158, p. 106734, 2023.

[20] P. Jangir, P. Manoharan, S. Pandya, and R. Sowmya, “MaOTLBO: Many-objective teaching-learning-based optimizer for control and monitoring the optimal power flow of modern power systems,” *International Journal of Industrial Engineering Computations*, vol. 14, no. 2, pp. 293–308, 2023.

[21] M. R. Youcefi, A. Hadjadj, A. Bentriou, and F. S. Boukredera, “Rate of penetration modeling using hybridization extreme learning machine and whale optimization algorithm,” *Earth Science Informatics*, vol. 13, no. 4, pp. 1351–1368, 2020.

[22] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of ICNN’95 - International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995.

[23] N. Kouka et al., “EEG channel selection-based binary particle swarm optimization with recurrent convolutional autoencoder for emotion recognition,” *Biomedical Signal Processing and Control*, vol. 84, p. 104783, 2023.

[24] A. Koulidis and S. Ahmed, “Advisory system for drilling optimization using artificial intelligence and in-cutter sensing data,” in *SPE Symposium: Leveraging Artificial Intelligence to Shape the Future of the Energy Industry*, 2023, doi: 10.2118/214468-MS.

[25] E. Millan, M. Ringer, R. Boualleg, and D. Li, “Real-time drillstring vibration characterization using machine learning,” in *SPE/IADC Drilling Conference and Exhibition*, 2019. SPE/IADC-194061-MS.

[26] S. B. Pandya, S. Ravichandran, P. Manoharan, P. Jangir, and H. H. Alhelou, “Multi-objective optimization framework for optimal power flow problem of hybrid power systems considering security constraints,” *IEEE Access*, vol. 10, pp. 103509–103528, 2022.

[27] Y. Zhou, W. Zhang, I. Gamwo, and J.-S. Lin, “Mechanical specific energy versus depth of cut in rock cutting and drilling,” *International Journal of Rock Mechanics and Mining Sciences*, vol. 100, pp. 287–297, 2017.

[28] G. S. Payette et al., “Mitigating drilling dysfunction using a drilling advisory system: Results from recent field applications,” in *International Petroleum Technology Conference*, 2015.

[29] J. C. R. Pllacido, H. M. R. Santos, and Y. D. Galeano, “Drillstring vibration and wellbore instability,” *ASME Journal of Energy Resources Technology*, vol. 124, no. 4, pp. 217–222, 2002, doi: 10.1115/1.1501302.

Received: July 16, 2025

[30] C. Zhou, H. Tao, Y. Chen, V. Stojanovic, and W. Paszke, “Robust point-to-point iterative learning control for constrained systems: A minimum energy approach,” *International Journal of Robust and Nonlinear Control*, vol. 32, no. 18, pp. 10139–10161, 2022.

[31] Z. Zhuang, H. Tao, Y. Chen, V. Stojanovic, and W. Paszke, “An optimal iterative learning control approach for linear systems with nonuniform trial lengths under input constraints,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2022.

[32] S. Hbaieb, J. Converset, J. Foster, and Y. Arevalo, “Technology update: Well construction performance service bridges planning-execution gap to boost efficiency,” *Journal of Petroleum Technology*, vol. 70, no. 4, pp. 20–22, 2018.

[33] B. Jiang, S. Chen, B. Wang, and B. Luo, “MGLNN: Semi-supervised learning via multiple graph cooperative learning neural networks,” *Neural Networks*, vol. 153, pp. 204–214, 2022.

Received: July 16, 2025