Adaptive Well Log Interpretation: A Hybrid Supervised and Sequential Modeling Framework for Extrapolation

Moses Falowo, Charlie Bergdall
CIS 730/530

Abstract

Subsurface characterization, crucial for energy industries and geological storage, relies heavily on interpreting well log data to predict properties like lithology, fluid type, and porosity. Traditional methods often struggle with noisy data and, critically, extrapolating predictions to deeper, unmeasured zones. This study develops and evaluates a machine learning framework using a single-well dataset from the Kansas Geological Survey (~8,700 depth points) to predict these three key properties, with a primary focus on extrapolation performance. We compare baseline supervised models, ZeroR, Random Forest (RF), and k-Nearest Neighbors (k-NN), against a sequential Long Short-Term Memory (LSTM) network on a standard 80/20 randomized split. Subsequently, RF and a deeper LSTM architecture are retrained on a chronological top 80% split and evaluated on their ability to extrapolate to the bottom 20% of the well. Results from the standard split show RF significantly outperforming k-NN and the initial LSTM for lithology (RF Acc: 0.7700) and fluid type (RF Acc: 0.9514) classification, and providing better porosity regression (RF R^2: 0.2019). In the more challenging extrapolation task, the retrained RF generally maintained better performance (e.g., Lithology Acc: 0.5498; Fluid Type Acc: 0.7134) than the deeper LSTM, though both models struggled significantly with porosity extrapolation (RF R^2: -0.0073; LSTM R^2: -0.2425). Visual analysis of extrapolated logs highlighted challenges in geological plausibility for the LSTM. The study underscores the difficulty of extrapolation from single-well data and suggests RF's robustness in this context, while highlighting areas for future improvement using multi-well data and advanced sequential architectures.

1. Introduction

The accurate characterization of subsurface geological formations is paramount for various energy-related industries, including oil and gas exploration, geothermal resource assessment, and the long-term security of carbon capture and storage (CCS) initiatives. Well logs, which are continuous measurements of various physical properties recorded along the depth of a borehole, serve as a primary source of data for inferring critical rock and fluid properties such as lithology (rock type), fluid content (e.g., oil, gas, water), and porosity (the void space within the rock). However, interpreting these logs is often complex due to inherent data noise, abrupt geological transitions between different formations, and the significant challenge of making reliable predictions for zones deeper than where direct measurements are available or economically feasible, a process known as extrapolation.

This project aims to develop and evaluate a machine learning (ML) framework to predict these three key geological properties using data from a single, publicly available well log dataset from the Kansas Geological Survey (KGS). The dataset comprises approximately 8,739 depth points, sampled every 0.5 meters, offering a rich sequence of multivariate measurements. While predicting properties within the sampled range is valuable, a core focus of this study is to rigorously assess the extrapolation capabilities of different ML models: how well can they predict into deeper, unmeasured sections of the well, simulating a common real-world exploration scenario?

Our approach involves a comparative analysis. We establish baseline performance using traditional supervised learning models: ZeroR (a simple majority/mean predictor), Random Forest (RF), and k-Nearest Neighbors (k-NN). These are compared against a sequential model, the Long Short-Term Memory (LSTM) network, hypothesized to better capture depth-dependent contextual patterns. The evaluation is twofold: first, a standard 80/20 randomized train-test split is used to assess general model performance. Second, and more critically for our objectives, a chronological data split is employed where models (RF and a more complex, deeper LSTM architecture) are trained on the upper 80% of the well data and their ability to extrapolate to the lower, unseen 20% is meticulously evaluated. This includes analyzing performance degradation with increasing extrapolation depth and assessing the geological plausibility of the predicted log profiles.

Initial findings indicate that on the standard data split, Random Forest generally provides superior performance for all three target properties compared to k-NN and an initial LSTM configuration. In the more challenging extrapolation task, while all models face difficulties, particularly with porosity, the retrained Random Forest demonstrates more robust and geologically plausible extrapolations compared to the deeper LSTM within the constraints of this single-well dataset. This study highlights the significant challenges of subsurface extrapolation and provides insights into model selection for such tasks.

2. Background and Related Work

The interpretation of well log data to infer subsurface properties has a long history, traditionally relying on empirical petrophysical models, chart-based lookups, and significant expert geological judgment (Mavko et al., 2009). These methods, while foundational, often face limitations in handling data inconsistencies, noise, the complex non-linear relationships between log responses and geological properties, and particularly, in robustly extrapolating predictions to un-cored or unmeasured intervals.

With the advent of machine learning, various techniques have been applied to automate and enhance well log interpretation. Supervised learning algorithms have been widely explored for their ability to learn from labeled.

For tasks involving tabular data, as is common with processed log suites at discrete depth points, Random Forest (Breiman, 2001) has emerged as a powerful and popular algorithm. Its ensemble nature, built on multiple decision trees, provides robustness against overfitting, handles high-dimensional data well, and can capture non-linear relationships without requiring extensive feature scaling. It serves as a strong baseline for many classification and regression problems in geosciences. Similarly, k-Nearest Neighbors (k-NN), another non-parametric method, offers a simpler approach by classifying or regressing based on the properties of the closest samples in the feature space, though it can be sensitive to feature scaling and the "curse of dimensionality.

Well log data, however, possesses an inherent sequential characteristic due to its ordered acquisition along depth. Measurements at one depth are often correlated with those at adjacent depths, and geological formations exhibit vertical continuity and transitions. This sequential nature suggests the applicability of models designed for ordered data. Long Short-Term Memory (LSTM) networks (Hochreiter and Schmidhuber, 1997), a type of Recurrent Neural Network (RNN), are specifically architected to learn long-range dependencies and contextual patterns in sequences. They have found significant success in various time-series forecasting and sequence modeling tasks, and their application to well log interpretation is a logical extension, with the hypothesis that they can better model depth-dependent variations and transitions than purely atemporal models.

The challenge of extrapolation in geosciences remains particularly acute. Models trained on data from a certain depth range or geological setting may not generalize well when asked to predict properties in a significantly different, deeper regime. This is because the statistical properties of the data and the underlying geological processes can change with depth (non-stationarity). While LSTMs might capture local sequential patterns, robustly extrapolating these patterns far beyond the training distribution is a non-trivial task, especially with data from a single well which provides a limited view of geological variability.

Our initial project proposal also considered the exploration of Reinforcement Learning (RL) (Sutton & Barto, 2018) for the extrapolation task. RL frameworks, involving an agent learning to make optimal decisions in an environment to maximize a cumulative reward, have shown promise in sequential decision-making problems. For well log extrapolation, this could involve an agent deciding whether to "continue drilling" into an unknown zone based on predicted geology and receiving rewards for plausible or valuable discoveries. However, applying RL to this domain faces significant practical challenges, primarily in defining a realistic environment that can simulate the "next state" (i.e., future log readings) and in formulating a reward function that accurately reflects "good geological extrapolation" in the absence of ground truth for deeper, unmeasured zones. Due to these complexities and the desire to provide a focused and robust comparison of well-established supervised and sequential techniques within the project's scope, the RL component was strategically deferred. This allows for a more thorough investigation of the comparative strengths and weaknesses of Random Forest and LSTM in the context of single-well log extrapolation.

3. Methodology

This study employed a multi-stage methodology to preprocess the well log data, derive target geological properties, and subsequently train and evaluate various machine learning models for both standard prediction and extrapolation tasks. All analyses were conducted using Python, leveraging libraries such as Pandas for data manipulation, Scikit-learn for classical machine learning algorithms and preprocessing, and TensorFlow/Keras for building and training LSTM networks.

3.1 Data Acquisition and Preprocessing

The primary dataset for this project is a single-well log dataset obtained from the Kansas Geological Survey (KGS). The raw data, provided as a CSV file (log.csv), contains approximately 8,739 depth points, typically sampled at 0.5-meter intervals. Each depth point includes measurements from various logging tools.

Data Loading: The dataset was loaded into a Pandas DataFrame. Column names were standardized, for instance, renaming 'DEPT' to 'Depth' and 'GAMMA_RAY' to 'GR' if necessary, based on the raw file headers.

Handling Missing Values: An initial inspection revealed the dataset to be remarkably complete, with no missing values in the numeric log columns prior to explicit imputation.

Target Variable Derivation: Three key geological properties were derived from the existing log data to serve as our prediction targets (Y1, Y2, Y3). This derivation was based on common petrophysical thresholds and rules:

Y1: Lithology (Rock Type): A multi-class categorical variable derived primarily from the Gamma Ray (GR) log. The GR log is sensitive to the natural radioactivity of formations, which often correlates with shale content. Specific GR thresholds (50, 80, 100, 150 API units) were used to define five lithological classes: 'CleanSand', 'ShaleSand', 'ShalySandstone', 'Shale', and 'Carbonate/Other'.

Y2: Fluid Type: A multi-class categorical variable indicating the dominant fluid in the formation pores (Gas, Oil, Water). This was derived using thresholds applied to the Deep Induction Resistivity (RILD) log (8 and 20 ohm-meters), as formation resistivity is strongly influenced by the type and saturation of pore fluids.

Y3: Porosity: A continuous numerical variable representing the fraction of void space in the rock. This was derived from the Density Porosity log (DPOR). If DPOR values were in percentage units (as indicated by values > 1.5), they were converted to a fractional range (0.0 to 1.0). Values were clipped to ensure they remained within these physical bounds.

Data Integration: The newly derived target variables (Y1_Lithology, Y2_FluidType, Y3_Porosity) were joined back to the main processed DataFrame. Any rows where target derivation resulted in an undefined or missing value were subsequently dropped to ensure a clean dataset for model training. After these steps, the processed dataset typically contained around 8,739 samples and 17 columns.

3.2 Feature Engineering & Selection (Blinding)

To prepare the data for model training and prevent target leakage, specific input features (X) were selected for each prediction task, and a "blinding" process was applied.

Commonly Excluded Features: The 'Depth' column, while defining the sequence, was not used as a direct input feature value for the models. The 'CNLS' (Compensated Neutron Log Shaliness) log was also excluded based on initial setup decisions (as per presentation slide feedback). If an original 'Lithology' string column existed (from which Y1 might have been inspired), it was also dropped.

Blinding for Y1 (Lithology): In addition to common exclusions, the 'GR' log was removed from the input features for predicting Y1, as GR was the primary log used to derive the Y1_Lithology classes.

Blinding for Y2 (Fluid Type): Similarly, the 'RILD' log was removed when predicting Y2_FluidType.

Blinding for Y3 (Porosity): The 'DPOR' log (the source for Y3_Porosity) and the 'RHOB' (Bulk Density) log, which is highly correlated and often used in porosity calculations, were excluded. The 'NPHI' (Neutron Porosity) log, if considered a direct porosity indicator, was also blinded for this task.
After blinding, the feature set for predicting Y1 and Y2 typically consisted of 11 features ('RxoRt', 'RLL3', 'SP', 'MN', 'MI', 'MCAL', 'DCAL', 'RHOB', 'RHOC', 'DPOR', and 'GR' (for Y2 only)), while for Y3, it consisted of 10 features (excluding DPOR, RHOB, NPHI, CNLS, Depth).

3.3 Standard Evaluation Setup (Randomized 80/20 Split)

For an initial assessment of general model performance, a standard randomized train-test split was employed.

Data Split: The processed dataset (after target derivation and feature selection for each task) was split into an 80% training set and a 20% testing set. For the classification tasks (Y1 Lithology, Y2 Fluid Type), this split was stratified to maintain the proportional representation of each class in both the training and testing subsets. A random_state=42 was used for reproducibility.

Justification: This standard split allows for direct comparison of different model architectures on their ability to learn from a representative sample of the data, assuming data points are independent and identically distributed. While k-fold cross-validation can provide a more robust estimate of performance, a single 80/20 split was chosen for this phase to manage computational time, especially for the LSTM models, and because the primary focus of the project was on the subsequent extrapolation analysis which requires a different splitting strategy.

Models Implemented:

ZeroR: DummyClassifier(strategy='most_frequent') for Y1 and Y2, and DummyRegressor(strategy='mean') for Y3.

Random Forest (RF): RandomForestClassifier and RandomForestRegressor with n_estimators=100 and random_state=42.

k-Nearest Neighbors (k-NN): KNeighborsClassifier with n_neighbors=5. Since k-NN is sensitive to feature scales, it was implemented within a Pipeline that included StandardScaler for feature normalization.

Initial LSTM: A sequential model with an architecture comprising two LSTM layers (50 units each), dropout layers (0.2), a dense hidden layer (32 units, ReLU), and an appropriate output layer (Softmax for classification, Linear for regression). Adam optimizer was used.

3.4 Extrapolation Analysis Setup (Chronological Split)

To specifically evaluate the models' ability to extrapolate into deeper, unseen geological zones, a chronological data split was performed.

Chronological Data Split: The entire processed dataset was ordered by the 'Depth' column. The top 80% of this depth-ordered data was designated as the new training/validation set, representing known, shallower geology. The bottom 20% was strictly held out as the extrapolation test set, representing deeper, unknown geology. This ensures that the models are tested on data that is "in the future" in terms of depth relative to their training data.

i. **Model Retraining:**

Random Forest: An RF model (with n_estimators=100) was retrained *only* on the features and targets from this chronological top 80% training set.

Deeper LSTM: Recognizing that extrapolation might require a more complex model to capture subtle trends, a "Deeper LSTM" architecture was defined and trained *only* on sequences derived from the chronological top 80% training data. This architecture featured three KerasLSTM layers (64 units, 64 units, 32 units), dropout (0.2), a dense hidden layer (32 units, ReLU), and an Adam optimizer with a lower learning rate (0.0005). More training epochs (up to 100) and increased patience (15) for early stopping were used for this more demanding task.

Evaluation: The performance of these retrained RF and Deeper LSTM models was then evaluated exclusively on the bottom 20% extrapolation test set.

3.5 Feature Scaling for LSTMs

For all LSTM models (both initial and deeper), the input features were scaled to a range of 0 to 1 using MinMaxScaler from Scikit-learn. The scaler was fit *only* on the respective training data (either the 80% from the standard split or the top 80% from the chronological split) and then used to transform both the training and corresponding test/extrapolation sets. This prevents data leakage from the test set into the training process.

3.6 Sequence Generation for LSTMs

To prepare data for the LSTM models, the time-series or depth-series data was converted into overlapping sequences. A window_size of 10 was used. This means that for each prediction, the LSTM model considered a sequence of the 10 preceding depth measurements (and all their associated features) to predict the target property at the 11th depth point. For classification targets (Y1, Y2), the target variables for the LSTM sequences were further processed using OneHotEncoder.

3.7 Evaluation Metrics

The performance of the models was assessed using standard metrics appropriate for classification and regression tasks:

The performance of the models was assessed using standard metrics appropriate for classification tasks, specifically Accuracy, Precision, Recall, F1-score, and Confusion Matrix for Lithology (Y1) and Fluid Type (Y2), and for the regression task of Porosity (Y3) using R-squared (R2), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE)

4. Experimental Results and Discussion

This section presents the performance of the implemented machine learning models on both the standard 80/20 randomized data split and the more challenging chronological extrapolation task. Results are discussed in terms of quantitative metrics and qualitative visual assessments.

4.1 Model Performance on Standard 80/20 Randomized Split

The initial phase of evaluation focused on assessing the general learning capability of the models on a dataset where training and testing samples were randomly drawn from the entire well log.

4.1.1 Lithology Classification (Y1)

The models were tasked with classifying the rock type into five categories: 'Carbonate/Other', 'CleanSand', 'Shale', 'ShaleSand', and 'ShalySandstone'.

ZeroR Baseline: The ZeroR classifier, predicting the majority class ('ShaleSand'), achieved an accuracy of 0.3072 and a macro F1-score of 0.0940. Its confusion matrix (Figure 1) visually confirms this, showing predictions concentrated along the 'ShaleSand' row/column.

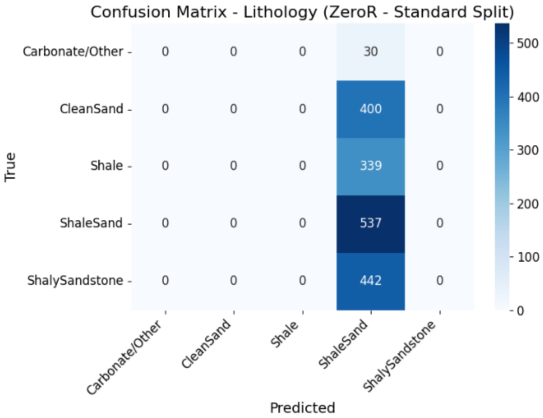

***Figure 1**: Confusion Matrix for ZeroR Lithology classification on the standard 80/20 test split.*

Random Forest: The Random Forest model demonstrated significantly better performance, achieving an accuracy of 0.7700 and a macro F1-score of 0.7405. The confusion matrix (Figure 2) shows good diagonal concentration, though with some confusion between 'ShaleSand' and 'ShalySandstone'.

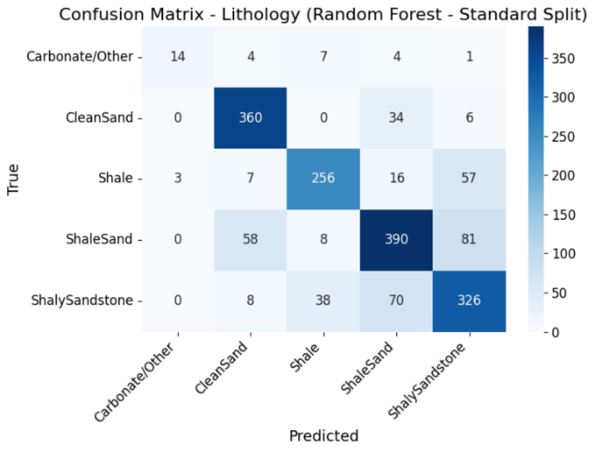

Figure 2: Confusion Matrix for Random Forest Lithology classification on the standard 80/20 test split.

k-Nearest Neighbors (k-NN): The k-NN classifier (k=5, with StandardScaler) achieved an accuracy of 0.6436 and a macro F1-score of 0.6053. Its performance, detailed in its confusion matrix (Figure 3), was intermediate between ZeroR and Random Forest.

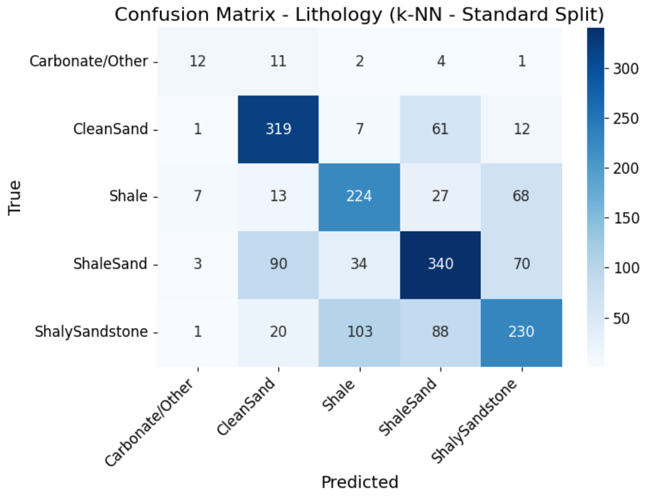

Figure 3: Confusion Matrix for k-NN Lithology classification on the standard 80/20 test split.

Initial LSTM: The initial LSTM configuration struggled on this task, yielding an accuracy of 0.3055 and a macro F1-score of 0.0936. This performance is comparable to, and slightly worse than, the ZeroR baseline. The confusion matrix (Figure 4) reflects this difficulty in distinguishing classes.

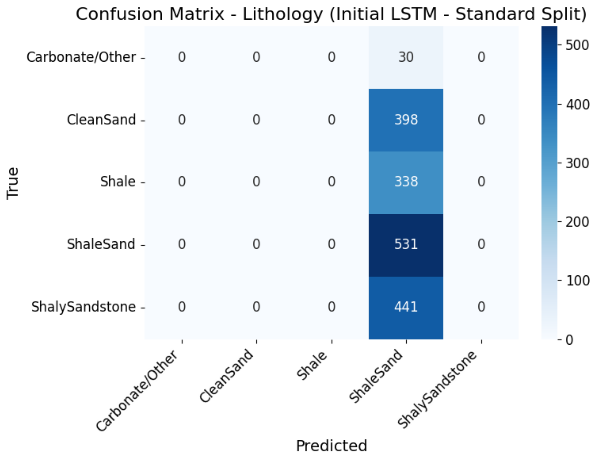

Figure 4: Confusion Matrix for Initial LSTM Lithology classification on the standard 80/20 test split.

Summary Table: Table 1 summarizes these results.

Table 1: Lithology Classification Performance (Standard Split)

Model	Accuracy	Macro F1	Weighted F1
ZeroR	0.3072	0.0940	0.1444
Random Forest	0.7700	0.7405	0.7689
k-NN (k=5)	0.6436	0.6053	0.6403
Initial LSTM	0.3055	0.0936	0.1430

4.1.2 Fluid Type Classification (Y2)

The models aimed to classify fluid type into 'Gas', 'Oil', and 'Water'.

ZeroR Baseline: Predicting the majority class ('Water'), ZeroR achieved an accuracy of 0.5069 and a macro F1-score of 0.2242 (Figure 6).

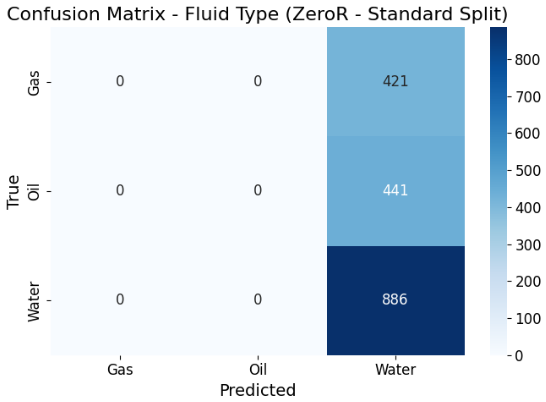

Figure 6: Confusion Matrix for ZeroR Fluid Type classification on the standard 80/20 test split.

Random Forest: RF performed exceptionally well, with an accuracy of 0.9514 and a macro F1-score of 0.9458. Its confusion matrix (Figure 7) shows high accuracy across all three classes.

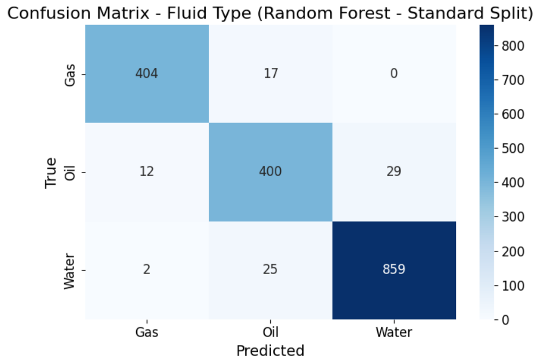

Figure 7: Confusion Matrix for Random Forest Fluid Type classification on the standard 80/20 test split.

k-Nearest Neighbors (k-NN): k-NN also performed strongly, with an accuracy of 0.8965 and a macro F1-score of 0.8828 (Figure 8).

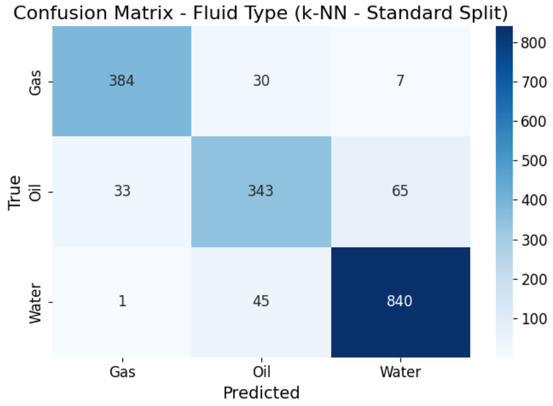

Figure 8: Confusion Matrix for k-NN Fluid Type classification on the standard 80/20 test split.

Initial LSTM: The LSTM showed modest performance with an accuracy of 0.5069 (same as ZeroR) and a macro F1-score of 0.2243, again comparable to the ZeroR. The confusion matrix (Figure 9).

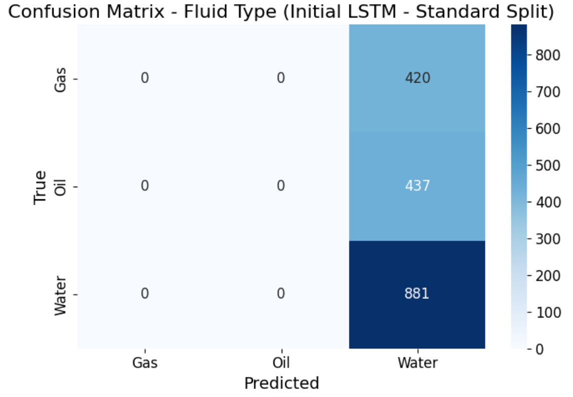

Figure 9: Confusion Matrix for Initial LSTM Fluid Type classification on the standard 80/20 test split.

Summary Table: Table 2 summarizes these results.

Table 2: Fluid Type Classification Performance (Standard Split)

Model	Accuracy	Macro F1	Weighted F1
ZeroR	0.5069	0.2242	0.3410
Random Forest	0.9514	0.9458	0.9514
k-NN (k=5)	0.8965	0.8828	0.8955
Initial LSTM	0.5069	0.2243	0.3410

4.1.3 Porosity Regression (Y3)

Porosity was predicted as a continuous fractional value.

ZeroR Baseline: The ZeroR regressor (predicting the mean porosity of the training set) resulted in an R² of -0.0003 and an RMSE of 0.060766.

Random Forest: RF achieved an R² of 0.2019 and an RMSE of 0.054278, showing some predictive capability (Figure 10).

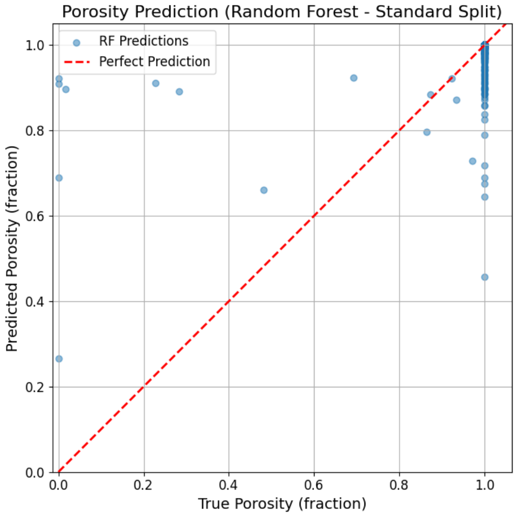

Figure 10: True vs. Predicted Porosity for Random Forest on the standard 80/20 test split.

Initial LSTM: The LSTM performed poorly, with a negative R² of -0.1407 and an RMSE of 0.065075 (worse than ZeroR). The scatter plot (Figure 11) shows a poor correlation.

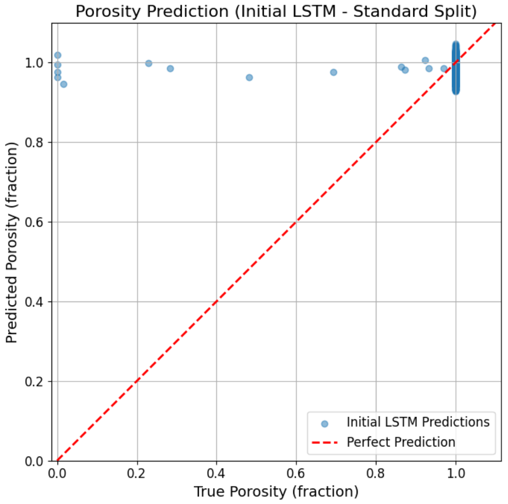

Figure 11: True vs. Predicted Porosity for Initial LSTM on the standard 80/20 test split.

Summary Table: Table 3 summarizes these results.

Table 3: Porosity Regression Performance (Standard Split)

Model	R²	RMSE	MAE
ZeroR	-0.0003	0.060766	0.009808
Random Forest	0.2019	0.054278	0.007362
k-NN (k=5)	-0.0101	0.061061	0.008093
Initial LSTM	-0.1407	0.065075	0.025240

4.2 Model Performance on Extrapolation Test Set (Chronological Split)

This phase evaluated models retrained on the top 80% of depth-ordered data and tested on the bottom 20% to assess true extrapolation capability. The retrained Random Forest was compared against a Deeper LSTM architecture designed for this more complex task.

Summary Metrics: The overall performance metrics for the retrained RF and Deeper LSTM on the extrapolation test set are presented in Table 4.

Table 4: Summary of Extrapolation Performance Metrics

Task	Metric	Random Forest (Retrained)	Deeper LSTM (Retrained)
Y1: Lithology	Accuracy	0.5498	0.1226
	F1(Macro)	0.3463	0.0755
Y2: Fluid Type	Accuracy	0.7134	0.1611
	F1(Macro)	0.6994	0.0925
Y3: Porosity	R²	-0.0073	-0.2425
	RMSE	0.1364	0.1519

4.2.1 Performance Degradation vs. Depth in Extrapolation Zone

To understand how performance varied with increasing extrapolation depth, metrics were calculated in chunks of 50 steps into the extrapolation zone.

Figure 12 presents these trends for accuracy and F1-score (for Y1, Y2) and MSE/MAE (for Y3) for both the retrained RF and Deeper LSTM.

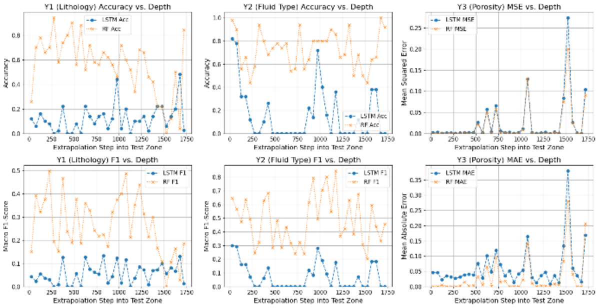

Figure 12: Performance degradation of retrained RF and Deeper LSTM in the extrapolation zone.

4.2.2 Visual Log Comparison and Geological Plausibility

Visual inspection of the predicted logs against the true logs in the extrapolation zone provides qualitative insights into model behavior.

Figure 13 shows the true logs for Lithology, Fluid Type, and Porosity in the extrapolation zone, overlaid with predictions from the retrained RF and Deeper LSTM.

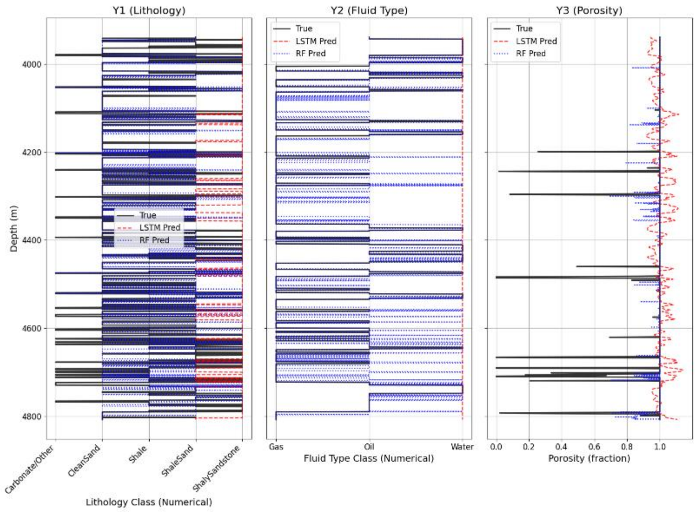

Figure 13: Visual comparison of true logs with retrained Random Forest and Deeper LSTM predictions in the extrapolation zone.

4.3 Overall Discussion of Results

Across both standard and extrapolation evaluations, Random Forest consistently demonstrated more robust performance than the k-NN and LSTM architectures tested. On the standard split, RF's ability to handle tabular data effectively was evident. The initial LSTM's poor performance, often comparable to or worse than ZeroR, suggests that for randomly sampled data from this single well, the sequential processing with a fixed window did not confer an advantage and may have even hindered learning generalizable patterns.

The extrapolation analysis, which was the primary focus, highlighted the significant difficulty of predicting into unseen geological domains. While the retrained Random Forest's metrics also degraded, it generally outperformed the Deeper LSTM. The Deeper LSTM, despite its increased complexity and design for sequential data, struggled significantly, exhibiting issues with metric performance, limited prediction diversity, and geologically implausible outputs (e.g., porosity > 1). This may suggest that with only a single well for training, the LSTM might have overfit to patterns in the shallower 80% that were not representative of the deeper 20%, or it lacked sufficient diverse examples to learn truly generalizable depth-dependent relationships necessary for robust extrapolation. The blinding of primary predictor variables, especially for porosity, undoubtedly exacerbated the challenge for all models.

5. Summary and Future Work

5.1 Summary of Key Findings

This study investigated the application of machine learning models for predicting lithology, fluid type, and porosity from single-well log data, with a particular emphasis on evaluating their extrapolation capabilities.

Standard Evaluation (Randomized 80/20 Split):

Random Forest (RF) consistently and significantly outperformed k-Nearest Neighbors (k-NN) and an initial Long Short-Term Memory (LSTM) network configuration across all three target tasks. For lithology and fluid type classification, RF achieved high accuracy and F1-scores. For porosity regression, RF showed modest predictive capability (R² = 0.2019), while k-NN (not applied to regression) and the initial LSTM performed poorly, with the LSTM often failing to surpass even a simple ZeroR baseline.

Extrapolation Evaluation (Chronological Split - Top 80% Train, Bottom 20% Test):

When models (RF and a Deeper LSTM architecture) were retrained on the chronological upper portion of the well and tested on the deeper, unseen portion, the retrained RF generally maintained better performance metrics for lithology and fluid type classification compared to the Deeper LSTM.

Both models struggled profoundly with porosity extrapolation, yielding negative R-squared values, indicating their predictions were less accurate than a simple mean predictor for the unseen deeper zone. This was attributed largely to the necessary blinding of primary porosity-indicating logs (DPOR, RHOB).

Visual analysis of extrapolated logs revealed that the Deeper LSTM sometimes produced geologically implausible predictions, such as porosity values exceeding physical limits (e.g., >1.0) and a lack of diversity in predicted lithological and fluid classes. Random Forest's extrapolated predictions, while also imperfect, often appeared more varied and physically constrained.

Challenges of Single-Well Extrapolation: The results underscore the inherent difficulty of extrapolating geological properties from a single well, especially when primary predictive features are blinded. The limited geological variability captured in a single borehole makes it challenging for models, particularly complex sequential ones like LSTMs, to learn patterns that robustly generalize to significantly different, deeper geological regimes.

5.2 Limitations of the Study

This study, while providing valuable insights, has several limitations:

Single-Well Dataset: The primary limitation is the use of data from only one well. This restricts the geological variability available for model training, making it difficult for models to learn truly generalizable relationships applicable to diverse subsurface conditions or different wells.

Model Scope and Tuning: The exploration of LSTM architectures and hyperparameter tuning was constrained by project timelines. More extensive tuning, particularly for the Deeper LSTM in the extrapolation context, or the exploration of other advanced sequential architectures (e.g., Transformers, attention mechanisms), might yield different results but would require significant computational resources and potentially more data.

Feature Engineering: While essential blinding was performed, more sophisticated domain-specific feature engineering (e.g., creating petrophysical indices, ratios, or depth-trend features) was not extensively explored but could potentially improve model performance.

5.3 Future Work

Based on the findings and limitations, several avenues for future work are recommended:

Incorporate Multi-Well Data: This is arguably the most critical next step. Training models on data from multiple wells, ideally sampling a wider range of geological settings and depths, would expose them to greater variability and significantly enhance their potential for generalization and robust extrapolation.

Advanced Sequential Architectures: Explore more sophisticated sequence models such as Gated Recurrent Units (GRUs), Transformers with self-attention mechanisms, or hybrid Convolutional-Recurrent Neural Networks (CRNNs), which might be better suited for capturing long-range dependencies and complex patterns in well log data, especially with larger datasets.

Systematic Hyperparameter Optimization: Conduct rigorous hyperparameter tuning (e.g., using grid search, random search, or Bayesian optimization techniques like Optuna or KerasTuner) for the LSTM models, specifically tailoring the optimization process for the chronological extrapolation task.

Enhanced Feature Engineering: Incorporate advanced petrophysical knowledge by engineering features like Archie's saturation exponents, shale volume corrections, or features that explicitly model depth trends and geological context.

Addressing Porosity Prediction: Investigate alternative modeling strategies or feature sets specifically for improving porosity prediction, perhaps by exploring transfer learning from models trained on datasets where direct porosity indicators are available, or by using physics-informed neural networks if applicable physical constraints can be formulated.

In conclusion, while machine learning models show promise for well log interpretation, robust extrapolation from limited data remains a significant hurdle. Future efforts should focus on leveraging more diverse datasets and advanced modeling techniques to improve the reliability of predictions in unexplored geological domains.

Acknowledgements

We would like to express our gratitude to Professor William Hsu for his guidance and insightful feedback throughout this project. We also acknowledge the Kansas Geological Survey (KGS) for making the well log dataset publicly available. Additionally, Google Gemini was consulted to assist with coding.

References

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation, 9*(8), 1735-1780.

Kansas Geological Survey (KGS). Project dataset provided by KGS

Mavko, G., Mukerji, T., & Dvorkin, J. (2009). *The rock physics handbook: Tools for seismic analysis of porous media* (2nd ed.). Cambridge University Press.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction* (2nd ed.). MIT Press.