Transfer Learning Approach to Prediction of Rate of Penetration in Drilling*

Felix James Pacis[1][0000−0002−7203−642X], Sergey Alyaev[2][0000−0002−2105−2067], Adrian Ambrus[2][0000−0002−4219−0370], and Tomasz Wiktorski[1][0000−0002−5940−8102]

[1] University of Stavanger, Stavanger, Norway
(felix.j.pacis,tomasz.wiktorski)@uis.no
https://stavanger.ai/
[2] NORCE Norwegian Research Centre, PB 22 Nygårdstangen, 5838 Bergen, Norway
(saly, aamb)@norceresearch.no
https://www.norceresearch.no/

Abstract. The rate of penetration (ROP) is a key performance indicator in the oil and gas drilling industry as it directly translates to cost savings and emission reductions. A prerequisite for a drilling optimization algorithm is a predictive model that provides expected ROP values in response to surface drilling parameters and formation properties. The high predictive capability of current machine-learning models comes at the cost of excessive data requirements, poor generalization, and extensive computation requirements. These practical issues hinder ROP models for field deployment. Here we address these issues through transfer learning. Simulated and real data from the Volve field were used to pretrain models. Subsequently, these models were fine-tuned with varying retraining data percentages from other Volve wells and Marcellus Shale wells.

Four out of the five test cases indicate that retraining the base model would always produce a model with lower mean absolute error than training an entirely new model or using the base model without retraining. One was on par with the traditional approach. Transfer learning allowed to reduce the training data requirement from a typical 70 percent down to just 10 percent. In addition, transfer learning reduced computational costs and training time. Finally, results showed that simulated data could be used in the absence of real data or in combination with real data to train a model without trading off model's predictive capability.

Keywords: Rate of Penetration model · Transfer Learning · Deep Learning.

* This work is part of the Center for Research-based Innovation DigiWells: Digital Well Center for Value Creation, Competitiveness and Minimum Environmental Footprint (NFR SFI project no. 309589, DigiWells.no). The center is a cooperation of NORCE Norwegian Research Centre, the University of Stavanger, the Norwegian University of Science and Technology (NTNU), and the University of Bergen, and funded by the Research Council of Norway, Aker BP, ConocoPhillips, Equinor, Lundin, TotalEnergies, and Wintershall Dea.

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

1 Introduction

According to a 2016 study by EIA [2], drilling constitutes 30-60% of the average cost per well, which varies from $4.9 MM to $8.3 MM for onshore wells and $120 MM to $230MM for offshore wells. Thus, a modest improvement in the duration of drilling a well results in significant monetary savings. Among other factors such as preventing a non-productive time due to equipment failure or poor weather conditions, choosing the optimal drilling parameters to increase ROP is essential in reducing drilling duration.

Many attempts have been made on predicting the ROP. Although with some success [32], traditional physics-based models require frequent recalibration depending on the auxiliary data such as facies types, bit design, and mud properties [5, 23, 16, 24]. This is challenging since facies types, in particular, are often unknown prior to drilling and would require correlation to data from nearby (offset) wells, if such wells exist.

Machine learning (ML) models try to address these challenges by using data to find correlations among many drilling variables. A study by Hegde et al. [17] showed an improvement in ROP prediction in accuracy from 0.46 to 0.84 when using random forest. Elkatatny et al. [11] also showed an improvement from 0.72 to 0.94 using an Artificial Neural Network (ANN).

Despite significant improvements in recent years, no ML approach has been widely used for ROP optimization to date [25]. The potential reason could be that the existing ML models are impractical for real-time ROP prediction tasks. Developing an ML ROP model is a multidimensional problem that does not revolve solely around prediction accuracy. Higher predictive capability comes at the cost of substantial data requirements, computational constraints, and generalization capability. From a practical perspective, tackling these constraints would be desirable for several reasons.

First, the need for large datasets for training a model for every well would limit the value creation. ANN training, such as Elkatatny et al. [11] and Abbas et al. [3], would require 70% of data for training; rendering these methods essentially not applicable in real scenarios since only a fraction of a well can benefit from such approach, see Figure 1.

Second, ML models presented by O'Leary et al. [25], Mantha et al. [21], and Hegde et al. [17] require a priori knowledge on the formations being drilled. However, this information is rarely available prior to drilling the hole. This is problematic for wells drilled in new areas where offset wells do not exist yet.

Third, ROP prediction is a real-time regression problem. Unlike physics-based models that only require pre-identification of parameters, the ML requires training before deployment. Hence, one should consider the online computation requirements.

Fortunately, ROP ML models' issues are not foreign in other domains. Deep Learning models, in general, suffer from overfitting due to insufficient training data [33]. Transfer learning (TL) is an active research field in Deep Learning that deals with reusing a model trained from a more general task, termed base model or pre-trained model, to another specific tasks, termed target model. TL

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

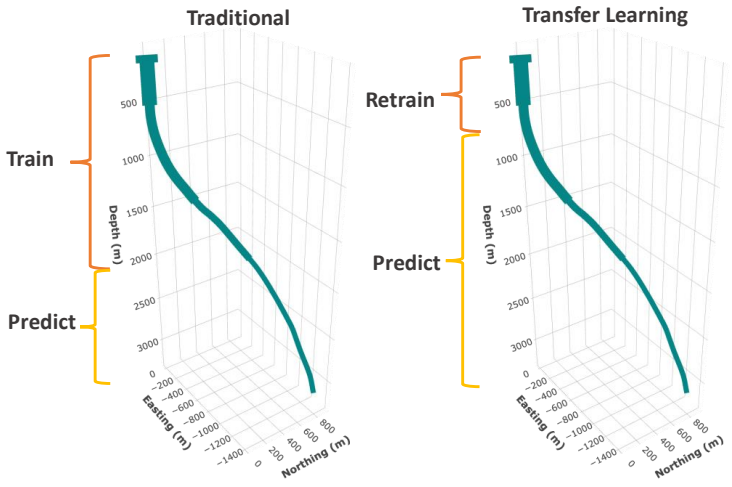

Fig. 1. Data utilization for traditional vs transfer learning approaches for well data.

techniques have been proven successful in many domains such as computer vision and natural language processing [26].

In this paper, we present the application of TL to ROP prediction. To our knowledge, this is the first application of TL in the context of drilling. We train base models using real, simulated, and combined data from previously drilled wells. Then, we reconfigure each model by freezing some model parameters in order to limit the number of trainable parameters. Each reconfigured base model is retrained using a small fraction of target-well data, yielding a target model. This way a high quality target model is available already from the early stage of drilling operation, see Figure 1. The performance of our TL models is compared to both the base models and models trained only for the data from the new well.

The paper is organized as follows. In Section 2, we briefly discuss the concept of TL. In Section 3, we describe the datasets and then proceed with the experimental setup, including the model architecture, input data, and the method for training and retraining. We also provide an end-to-end sample application of TL approach. Section 4 presents the results and lays out recommendations based on these. Section 5 concludes the paper.

2 Transfer learning

Following the notations by Pan and Yang [26], Transfer Learning mainly involves a domain D and Task T. The domain, denoted by $D = \{X, P^X\}$, includes two components: a feature space X and a marginal probability distribution P^X, where each input instance is denoted by $x \in X$. On the other hand, the task, denoted by $T = \{Y, f(\cdot)\}$, includes all possible labels Y and a predictive function $f(\cdot)$ that predicts a corresponding label using unseen instances $\{x*\}s$. For a

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

two domain scenario, given a source domain D_s and learning task T_s, a target domain D_t and learning task T_t, where $D_s \neq D_t$, or $T_s \neq T_t$, TL leverages learned knowledge from T_s to improve the T_t predictive function. Subscripts s and t here corresponds to source and target, respectively.

The most common TL technique is fine-tuning [29]. In the context of ANN, fine-tuning involves reusing the whole network or freezing certain hidden layers before updating the network weights during retraining for the target task. Fine-tuning works based on the premise that Deep Learning models learn different features at different layers. Thus, reusing a pre-trained model for a target task allows better performance with less training time by starting from "near truth" parameters than training a new model with randomly initialized parameters.

TL has been widely used both in computer vision and Natural Language Processing[26, 37]. This is apparent from the proliferation of pre-trained networks e.g.,VCG-16 [31], XLNet [38], GPT-3 [7] using large datasets e.g., ImageNet [3], Giga5 [4], and Common Crawl Dataset [5], and reused in domains where data is expensive or hard to obtain. For example in medical imaging, Shin et al. [30] fine-tuned AlexNet [20] - a pre-trained network using ImageNet dataset [9] with more than 14 million images belonging to around 20 thousand categories. They successfully achieved 85% sensitivity at 3 false positive per patient in thoraco-abdominal lymph node (LN) detection and interstitial lung disease (ILD) classification. Another successful application, Bird et al. [6] used a simulated scene from a computer game to train a model and resulted in an improvement for the real-world scene classification task.

Pre-trained networks also catalyzed the recent advances in Natural Language Processing (NLP). For example, Devlin et al. [10] introduced Bidirectional Encoder Representations from Transformers (BERT), which can be fine-tuned with adding an output layer to create state-of-the-art models for a wide range of tasks. Successful applications of BERT include text summarizing [10], modeling clinical notes and predicting hospital readmission [18], and machine reading comprehension [10].

The success of TL is apparent from its ubiquitous applications. This motivated websites, such as Hugging Face[6] and Model Zoo[7], which provide a platform to access many open-sourced pre-trained networks with ease.

TL has yet to be explored and applied broadly in the oil and gas domain. Since well-annotated datasets are expensive and difficult to obtain in the oil and gas industry, TL can be used to make rapid progress in this domain [15].

[3] https://www.image-net.org
[4] https://catalog.ldc.upenn.edu/LDC2011T07
[5] https://commoncrawl.org/the-data/
[6] https://huggingface.co
[7] https://modelzoo.co/

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

3 Experimental setup and data

3.1 Methodology

TL requires the base model to be trained from several unique wells to improve their generalization capability. We freeze selected layers in these base models to keep the original weights and allow some to be trainable. These reconfigured layers are then fine-tuned using a pre-determined percentage of data from target wells. Hyperparameters during fine-tuning are carefully chosen to prevent vanishing or exploding gradients. This happens when the distribution of retraining data is entirely different, and the learning rate is too high; this impairs the base model's performance. In addition, a new model is also trained using the same retraining data. All these models are then tested using the remaining data from the target well.

We performed all computations using 2.3 GHz Dual-Core Intel Core i5.

3.2 Datasets

We used well data from three sources namely, Volve field data, Marcellus shale field data, and synthetic data. Table 1 summarizes the datasets. The well name, hole size, hole depth range, and the data source type for each dataset are provided for reference.

In general, when drilling an oil and gas well, bigger holes are drilled first, followed by smaller holes until they reach the predefined target. This is done to maintain well integrity, particularly when transitioning to a new geologic formation. Drillers use different drill bits, bit designs, and drilling fluid properties at each new hole size. This is similar to drilling an entirely new well from an engineering perspective. Thus, we produce independent datasets by segregating each data according to hole size from each well. These datasets contain recorded real-time drilling parameters such as hookload, stand pipe pressure, hole depth, weight on bit, mud weight, and rotations per minute. These measurements' frequency varies for every well depending on the equipment used.

In 2018 Equinor publicly shared raw real-time drilling data from 20 wells found in the Volve field in the North Sea [12], together with well logging data, surveying data, drilling reports, and other auxiliary information. Pre-processed Volve drilling logs can be found in a public data repository [35]. For this paper, we selected drilling data from 7 wells and separated them according to the hole size. In total, we compiled 12 independent datasets for the experiment. Volve data has an average sampling frequency of 0.4 Hz, corresponding to a time step of 2.5 seconds.

Marcellus shale is the most prolific natural gas-producing formation from the Appalachian basin in the United States [34]. A site owned and operated by Northeast Natural Energy, LLC provides several horizontal wells drilled in the Marcellus shale [1]. A specific long horizontal well spanning 2431 meters, with an average measurement frequency of 0.176 Hz or 5.67 seconds time step, was chosen

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

for the current study. This well data allows testing the models' generalization and re-usability outside Volve data.

To provide additional training data and investigate the feasibility of using simulated training data for the TL application, we generated eight synthetic datasets using a state-of-the-art drilling simulator which includes advanced hydraulics, mechanics, and heat transfer models [13]. The well architecture, trajectory, drilling mud properties, drill string configuration, and formation properties were based on the drilling reports extracted from the Volve public database [12]. The drilling set points (top-drive rotary speed, weight on bit, and flow rate) used as input to the simulations were based on the values from the Volve recorded drilling logs compiled by Tunkiel [35]. The simulation outputs were stored as time-series with a time step of 1 second.

Table 1. Description of Datasets.

Well name	Hole size (in)	Depth range (m)	Dataset type	Dataset source type	Test Case #
F-1 A	8.5	2602-3682	Train & Val.	Sim. & Real	
F-1 B	12.25	2603-3097	Train & Val.	Sim. & Real	
F-1 B	8.5	3097-3465	Retrain & Test	Real	4
F-1 C	12.25	2662-3056	Retrain & Test	Real	2
F-1 C	8.5	3067-4094	Train & Val.	Sim. & Real	
F-11 A	8.5	2616-3762	Train & Val.	Sim. & Real	
F-11 B	12.25	2566-3197	Train & Val.	Sim. & Real	
F-11 B	8.5	3200-4771	Retrain & Test	Real	3
F-15 A	17.5	1326-2591	Retrain & Test	Real	1
F-15 A	8.5	2656-4095	Train & Val.	Sim. & Real	
F-9 A	12.25	489-996	Train & Val.	Sim. & Real	
F-9 A	8.5	1000-1202	Train & Val.	Sim. & Real	
Marcellus Shale	8.75	1974-4405	Retrain & Test	Real	5

3.3 Setup of experiments

Our model starts with an input layer, which receives four input parameters, followed by three successive pairs of dense and batch normalization layers. By embedding normalization as part of the model architecture, this prevents internal covariate shift [19] and causes a more predictable and stable behavior of the gradients [28], allowing higher learning rates without the risk of divergence [19, 28]. In addition, batch normalization eliminates the need for Dropout [33] for regularization [19]. We use rectified linear unit [14] as activation function. Finally, the output layer is a single-output dense layer with mean squared error as the loss function. A complete and detailed network structure is shown in figure 2.

To predict ROP, we used stand pipe pressure, weight on bit, mud weight, and rotations per minute (RPM). We based these inputs from the setup described in Ambrus et al.'s work [4]. In general, when choosing our input parameters,

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

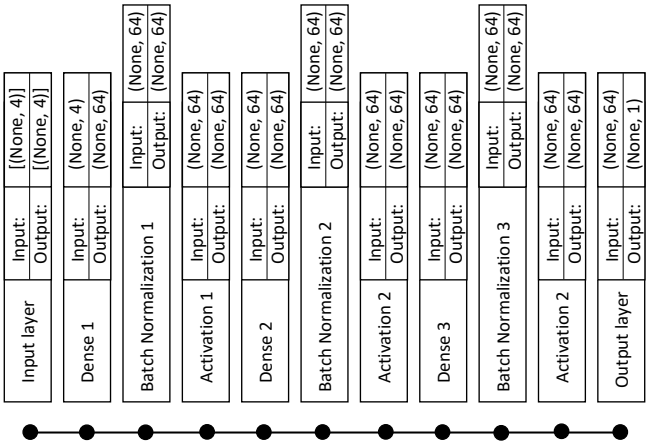

Fig. 2. Model Architecture.

two considerations were in place: first, despite using an ANN, the choice of input parameters should still reflect the physics of the drilling process. Second, selected inputs must always be available. During drilling, hundreds of parameters and metadata are recorded in real-time [36]. The inclusion of many drilling parameters as inputs to the ANN could be helpful but at the same time dangerous when one or more of these parameters are missing for the current well due to sensor failure or they were not necessarily recorded during the operation. Although one might infer the missing values, this would increase the model's prediction uncertainty when there are many inferred values.

Eight datasets were selected to build the base models out of the 12 available well sections from Volve. These were carefully selected to ensure that they contain values of the upper and lower boundaries of each input and output parameter. For example, the dataset with the highest ROP and lowest ROP values should be among the chosen eight.

To avoid overfitting the model, the first 80 percent of each well section is concatenated into the training dataset, whereas the remaining 20 percent are used for validation. This was done to all the three data source types - real, simulated, and combined. The shapes of concatenated training and validation data are shown in Table 2.

Data were scaled using a MinMaxScaler from Scikit-Learn [27] before passing to the model. This removes the harmful effects of having different value ranges for every input variable by scaling all of them to a (0,1) range.

Three separate runs for each data source type were conducted to build base models while keeping the model's hyperparameters the same. In particular, batch

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

Table 2. Traning and Validation Data Shapes.

Data Source Type	TrainX	TrainY	ValX	ValY
Real	(333400,4)	(333400,1)	(83350,4)	(83350,1)
Simulated	(649503,4)	(649503,1)	(162376,4)	(162376,1)
Combined	(982903,4)	(982903,1)	(245726,4)	(245726,1)

size was chosen to be 10000. This is relatively small, around 1 to 3 percent of each training dataset, to increase variation in batch statistics, thereby enabling better model generalization during the retraining process [22]. An early stopping Keras callback [8] was placed to cease training when the validation loss stops improving after 100 epochs. This allows us to generate two distinct base models for each data source type: one base model with the best validation loss and another based on the training loss. Altogether, we train six base models.

The four remaining well sections from real Volve data and the Marcellus shale horizontal well are used for retrain and test data. A sensitivity analysis was done by creating independent datasets with different retrain: test data ratios. These vary from 30:70, 20:80, 10:90, and 5:95, where the smaller partition corresponds to retraining data. Similar to the data preprocessing used for building the base model, each dataset is split sequentially and the values are scaled to a (0,1) range.

During retraining, we kept a similar model architecture to the base models, except that some layers were frozen. This allows us to retrain the model using smaller training datasets since fewer parameters are retrainable, and at the same time, model parameters are not initialized randomly. In addition, since models are pre-trained, a low learning rate is needed to reach the global minima. In our case, we used a learning rate of 0.0001 for all instances, with the exception of test case 4 that used 10^{-9}. Maximum epochs are set at 150000 for tests. Similar to training the base model, we set up an early stopping at 50 epochs based on the training loss.

These base models are reconfigured in three ways: freezing the first dense layer, first and second dense layers, and keeping all dense layers unfrozen. This gives us 18 reconfigured base models for retraining. All batch normalization layers were frozen in all these configurations to prevent the risk of vanishing or exploding gradients. In this context, freezing a layer means keeping the parameters learned during the initial training stage. Table 3 shows the number of trainable and non-trainable parameters for each configuration.

Table 3. Number of trainable and non trainable parameters.

Model configuration	Trainable Parameters	Non-trainable Parameters
Base Model	8897	384
Zero Frozen Layers	8513	768
One Frozen Layer	8257	1024
Two Frozen Layers	4161	5120

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

A randomly initialized new model with similar model architecture and hyperparameters was trained for every retraining data configuration. This is to compare the performance of fine-tuning a pre-trained model with that of a new model trained from scratch on the same dataset.

3.4 Model Evaluation

We have six unique base models from previous sections, wherein each was reconfigured in three configurations based on the number of frozen layers. This gives us 18 unique models on top of the base models plus an entirely new trained model. In total, for every retraining data configuration, e.g., one test well, with unique retrain:test ratio, we tested 25 different models.

Model performance is evaluated by computing the mean absolute error ($MAE := L_1$) and root-mean-square error ($RMSE := L_2$) for every test data configuration:

$$L_k = \left(\frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i|^k \right)^{\frac{1}{k}} \tag{1}$$

where N is the number for data points, $\hat{y}_i$ is the true value, and y_i is the predicted value. In addition, we kept a summary of a moving window MAE by dividing the test data into ten equal windows with the exact count of data points and computing MAE at each window. This enables us to measure prediction quality for various test data sizes. It is also important to emphasize that every well section data has a varying frequency of measurements and size, e.g., 30 percent of the well F-1C 12.25 in. section contains fewer data points than the F-11B 8.5 in. section.

3.5 Example of usage

As discussed in Section 3.3, we train six base models then reconfigure by freezing layers. Subsequently, we derive four different datasets from well F-15A 17.5 in data. Each dataset differs on the retrain and test ratio as described previously. Each of the 18 reconfigured base models is then fine-tuned using retraining data from every dataset. In addition, we train an entirely new model using similar retraining data. From here, we have a total of 25 distinct models - 6 base models, 18 reconfigured base models, and one new model. These 25 models are then used for predicting ROP on the test datasets. Having 4 data split ratios from well F-15A 17.5 gives us a total of 100 test runs. For every test case, MAE is recorded.

4 Results

In Section 4.1, we analyze the results on well F-15A 17.5 in and compare the best models from several methods, which includes fine-tuning, training of an entirely new model, and direct use of the base model. Data from the other wells are not presented due to space constraints of this paper. This well was selected because

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

results obtained on it were representative of both Volve and Marcellus shale test cases. In Section 4.2, we provide recommendations based on the results of five test cases. In Section 4.3, we provide results on the generalization capability of the approach.

4.1 3-way-comparison

After testing 100 models, we plot the predicted vs. expected ROP values plus a moving MAE window. In each plot, X-axis represents the hole depth, Y-axis to the left is the ROP with m/hr unit, and Y-axis to the right is the MAE. A, B, and C plots in Figure 3 show the best model among fine-tuned base models, base models, and new models, respectively. Model configurations and metadata of these models can be found in Table 4.

Table 4. Test Case 1: F-15A 17.5 in

Frozen layers	BM loss type	Source type	Retrain (%)	Train (%)	MAE	RMSE
0	Validation	Simulated	30	N/A	3.674	4.886
2	Validation	Simulated	5	N/A	4.104	5.238
1	Validation	Combination	30	N/A	4.165	5.656
2	Validation	Simulated	30	N/A	4.234	5.562
1	Validation	Simulated	30	N/A	4.268	5.87
*New Model	Training	Real	N/A	30	5.061	5.845
*Base Model	Validation	Simulated	N/A	80	9.091	10.767

Retraining the base model reduces the MAE by 59.6% and 27.4% vs. using the base model and training an entirely new model, respectively. A relatively close RMSE to MAE also indicates that the ROP error disperses equitably across the data. Despite the base model not being trained with the same 17.5-inch hole size, it outperforms other models by tuning with the current well data. This is also on top of the fact that model A has fewer trainable parameters. Although not seen on the plot, the second-best model overall has an MAE of 4.104, despite only using 5% retraining data and 55% fewer trainable parameters compared to training a new model. Furthermore, both of the best two models were pre-trained using simulated data.

4.2 Recommendations based on all test cases

Training data source type. Four out of the five test cases suggest that training with simulated data provides better result than training with real data in terms of MAE. One explanation for this could be that predictions are less noisy since the simulated data is deterministic; thus, it produces better results when re-trained on a small section of the test set.

Base Model loss type. Four out of the five test cases suggest that the best model should be based on the best validation loss rather than training

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

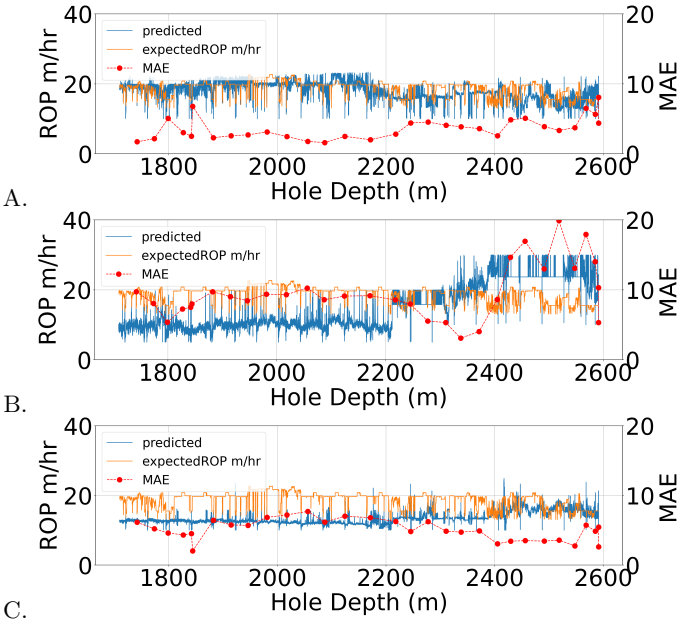

Fig. 3. ROP predicted by different models for well F-15A 17.5 in. A. Fine tuned model with TL. B. Base (pre-trained model) without fine-tuning. C. Newly trained model only using the data from the current well. Orange and blue lines refer to expected and predicted ROP values, respectively. Red markers are the computed MAE moving window e.g., one red marker is the MAE of the previous 2500 observations. All data are plotted against hole depth on the X-axis. Fine-tuned model performed best among other models with an MAE of 3.674.

loss. This is expected since the early stopping based on the validation loss helps reduce overfitting on the training data. Although the best retrained model in test case 3 was obtained using the training loss criterion, the base model using the validation loss criterion does not come far behind when considering the MAE.

Number of Frozen Layers. Four out of the five test cases suggest that fine-tuned models always perform the best. Case 4 performed just as well as the base model. Although, there was no clear relation between the number of frozen layers and the MAE. Paradoxically, increasing the number of frozen layers also increased the retraining time by 43 up to 247 percent. Thus, from retraining time perspective the ROP prediction problem benefits more from a pre-trained network without frozen layers. Another observation is that models with zero, one, and two frozen layers took an average retraining time of 7, 12, and 15 minutes, respectively, versus base models' 22 minutes.

Retraining data percent. As mentioned previously, every test well has a different length; therefore, even having the same retraining data percentage, the number of data points would still vary. There is no clear correlation between the number of data points and retraining data percentage for the best fine-tuned

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

model based on the five test cases, although one could say that there could be a slight trade-off between the accuracy of the model and the length of the well to be predicted.

During our experiments we also observed that TL was sensitive to the choice of base model training data and learning rate. However the detailed analysis is out of the scope of this paper.

4.3 Test outside Volve Data

We tested the approach on the Marcellus shale dataset to evaluate the generalization and re-usability of the TL approach. This well is entirely distinct from Volve data in terms of well profile (horizontal), type of formation (shale), location (onshore well), and equipment used. This is analogous to recognizing between breeds of dogs and breeds of cats for the computer vision domain. Clearly, the best retrained model reduced the MAE by 29% and 19% when compared with the newly trained model and base model, respectively. Relative to other test cases from Volve data, computed MAE is higher because of noise and lower measurement frequency in the Marcellus data. On top of improving the MAE, the retrained model only used 20% of retraining data while decreasing the trainable parameters by 10%. Furthermore, this result was achieved by training the base model with simulated data. This demonstrates the potential in using synthetic data generated with a high-fidelity drilling simulator for training the ANN ROP model that can be reconfigured for real operations with minimal amount of re-training.

5 Conclusions

We presented the application of TL for ROP prediction in oil and gas drilling. We trained, retrained, and tested a total of 100 models for each of the five test wells. Based on MAE evaluation, the TL approach for four out of five test wells outperforms both the newly trained model and the non-fine-tuned base model. For the fifth well the TL was on par with the traditional approach.

We explored the best model configurations based on the five test cases. In most cases the best results were obtained with the base models trained on the simulated data. Moreover the validation loss seems to be a good indicator of the model's performance on the new well. During fine-tuning, pre-trained models with zero frozen layers converge faster, although there was no clear relation between the MAE and the number of frozen layers. Despite uncertainty in the optimal number of frozen layers and retraining data percentage, results indicate that transfer learning is a valuable element in developing an adaptable, reusable, and more general ROP prediction model.

After successfully addressing the initial bottlenecks, new practical issues were identified. We noticed cases of negative transfer where some retrained models performed worse than their base model. The approach was also sensitive to the learning rate. Further work will focus on model optimization towards stability and improved accuracy while considering all practical bottlenecks.

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

References

1. Marcellus shale energy and environment laboratory. http://mseel.org, accessed: 2022-01-11
2. Trends in u.s. oil and natural gas upstream costs. https://www.eia.gov/analysis/studies/drilling/pdf/upstream.pdf, accessed: 2022-01-13
3. Abbas, A.K., Rushdi, S., Alsaba, M.: Modeling rate of penetration for deviated wells using artificial neural network. In: Abu Dhabi International Petroleum Exhibition & Conference. OnePetro (2018)
4. Ambrus, A., Alyaev, S., Jahani, N., Wiktorski, T., Pacis, F.J.: Rate of penetration prediction using quantile regression deep neural networks (2022)
5. Bingham, G.: A new approach to interpreting rock drillability. TECHNICAL MANUAL REPRINT, OIL AND GAS JOURNAL, 1965. 93 P. (1965)
6. Bird, J.J., Faria, D.R., Ekárt, A., Ayrosa, P.P.: From simulation to reality: Cnn transfer learning for scene classification. In: 2020 IEEE 10th International Conference on Intelligent Systems (IS). pp. 619–625. IEEE (2020)
7. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. arXiv preprint arXiv:2005.14165 (2020)
8. Chollet, F., et al.: Keras (2015), https://github.com/fchollet/keras
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
10. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
11. Elkatatny, S., Al-AbdulJabbar, A., Abdelgawad, K.: A new model for predicting rate of penetration using an artificial neural network. Sensors **20**(7), 2058 (2020)
12. Equinor: Volve field data (CC BY-NC-SA 4.0). URL https://www.equinor.com/en/news/14jun2018-disclosing-volve-data.html (2018)
13. Gravdal, J.E., Ewald, R., Saadallah, N., Moi, S., Sui, D., Shor, R.: A new approach to development and validation of artificial intelligence systems for drilling. In: 2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA). pp. 302–307. IEEE (2020)
14. Hahnloser, R.H., Sarpeshkar, R., Mahowald, M.A., Douglas, R.J., Seung, H.S.: Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. Nature **405**(6789), 947–951 (2000)
15. Hajizadeh, Y.: Machine learning in oil and gas; a swot analysis approach. Journal of Petroleum Science and Engineering **176**, 661–663 (2019)
16. Hareland, G., Rampersad, P.: Drag-bit model including wear. In: SPE Latin America/Caribbean Petroleum Engineering Conference. OnePetro (1994)
17. Hegde, C., Daigle, H., Millwater, H., Gray, K.: Analysis of rate of penetration (rop) prediction in drilling using physics-based and data-driven models. Journal of petroleum science and Engineering **159**, 295–306 (2017)
18. Huang, K., Altosaar, J., Ranganath, R.: Clinicalbert: Modeling clinical notes and predicting hospital readmission. arXiv preprint arXiv:1904.05342 (2019)
19. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning. pp. 448–456. PMLR (2015)

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44

20. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25**, 1097–1105 (2012)
21. Mantha, B., Samuel, R.: Rop optimization using artificial intelligence techniques with statistical regression coupling. In: SPE annual technical conference and exhibition. OnePetro (2016)
22. Masters, D., Luschi, C.: Revisiting small batch training for deep neural networks. arXiv preprint arXiv:1804.07612 (2018)
23. Maurer, W.: The perfect-cleaning theory of rotary drilling. Journal of Petroleum Technology **14**(11), 1270–1274 (1962)
24. Motahhari, H.R., Hareland, G., James, J.: Improved drilling efficiency technique using integrated pdm and pdc bit parameters. Journal of Canadian Petroleum Technology **49**(10), 45–52 (2010)
25. O'Leary, D., Polak, D., Popat, R., Eatough, O., Brian, T.: First use of machine learning for penetration rate optimisation on elgin franklin. In: SPE Offshore Europe Conference & Exhibition. OnePetro (2021)
26. Pan, S.J., Yang, Q.: A survey on transfer learning. ieee transactions on knowledge and data engineering. 22 (10): 1345 **1359** (2010)
27. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al.: Scikit-learn: Machine learning in python. Journal of machine learning research **12**(Oct), 2825–2830 (2011)
28. Santurkar, S., Tsipras, D., Ilyas, A., Madry, A.: How does batch normalization help optimization? In: Proceedings of the 32nd international conference on neural information processing systems. pp. 2488–2498 (2018)
29. Sarkar, D., Bali, R., Ghosh, T.: Hands-On Transfer Learning with Python: Implement advanced deep learning and neural network models using TensorFlow and Keras. Packt Publishing Ltd (2018)
30. Shin, H.C., Roth, H.R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., Summers, R.M.: Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. IEEE transactions on medical imaging **35**(5), 1285–1298 (2016)
31. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
32. Soares, C., Daigle, H., Gray, K.: Evaluation of pdc bit rop models and the effect of rock strength on model coefficients. Journal of Natural Gas Science and Engineering **34**, 1225–1236 (2016)
33. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research **15**(1), 1929–1958 (2014)
34. Statistics, I., Analysis, U.E.I.A.: Marcellus shale play: Geology review (2017)
35. Tunkiel, A.: Selected work repository. https://www.ux.uis.no/ atunkiel/ (2020)
36. Tunkiel, A.T., Wiktorski, T., Sui, D.: Drilling dataset exploration, processing and interpretation using volve field data. In: International Conference on Offshore Mechanics and Arctic Engineering. vol. 84430, p. V011T11A076. American Society of Mechanical Engineers (2020)
37. Weiss, K., Khoshgoftaar, T.M., Wang, D.: A survey of transfer learning. Journal of Big data **3**(1), 1–40 (2016)
38. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R.R., Le, Q.V.: Xlnet: Generalized autoregressive pretraining for language understanding. Advances in neural information processing systems **32** (2019)

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_44