URTeC: 47

Application of Machine Learning for Production Forecasting for Unconventional Resources

Cheng Zhan, Sathish Sankaran, Vincent LeMoine, Jeremy Graybill, Didi-Ooi Sher Mey, Anadarko Petroleum Corporation

This paper was prepared for presentation at the Unconventional Resources Technology Conference held in Denver, Colorado, USA, 22-24 July 2019.

Abstract

Decline curve analysis is often the *de facto* method for large scale production rate forecasting based on empirical relationships. Often, we face a number of practical problems for reliably estimating rates due to operational changes in the field (changing chokes, common surface network), curtailed production, limited available historical data, complex flow behavior, well interference etc. Our main objective here is to develop a new methodology to make robust and accurate oil rate prediction based on limited initial production data. We will show that the resulting model is useful for production forecasting, business planning and decision making in response to the fast pace development for unconventional resource plays.

A machine learning approach based on LSTM (Long Short Term Memory) is used to tackle the production forecasting problem. Compared with the modified hyperbolic approach, where the problem has been reduced to a pre-defined equation and essentially determined by a global curvature structure, the LSTM model is more dynamic and has a better chance of capturing non-linear events. In time series prediction, one main difficulty is how to stabilize the solution, as the error can easily accumulate over time. Besides modifying the objective function that aims for long term accuracy or incorporating physics-based modeling, one effective way to make the algorithm more robust is through feature engineering. By leveraging historical data from other wells, the prediction has been improved significantly. We also build another model in the accumulated curve domain, and ensemble multiple models to reduce the variance.

Forecasting is highly challenging in many domains with complex multivariate correlation structures and nonlinear dynamics. We have utilized existing data and built two prediction models, one from the decline curve domain, the other from the accumulated curve. Based on the observation, the first model is slightly over-predicted, and the second one moderately under-predicted, and through integrating these two models, the final result is more promising. We have conducted hindcasting for more than 300 wells, and the mean difference between the predicted and actual accumulated production of the first 2 year is less than 0.2%, with the variance less than 5%.

Many empirical production decline models have been proposed in the literature, but most fail to capture the complexity of forecasting and reduce the problem to an over-simplified curve. Our data-driven procedure is a unique and novel approach, which is more dynamic and has a better chance of capturing non-linear events. This method can also be applied to conventional reservoirs.

Introduction

Ever since the US unlocked the unconventional resource in early 2000s, shale has reshaped the global oil market (Webster, et al., 2018). One consequence is that current production analysis tools we relied on for traditional oil and gas forecasting might not be applicable for shale segment, as the well productivity is significantly driven by completion characteristics and the physical mechanisms of fluid flow in these prospects remains to be understood. In response to the massive amount of data from hundreds of wells in different states, innovative methods or workflows are needed for engineers to effectively predict the production.

Forecasting decline curve is an important component for E&P companies in business planning, asset evaluation and decision making. For example, asset management needs to be able to estimate accurately the net present value (NPV) of an oil drilling project, and the EUR (estimated ultimate recovery based on production forecasting) is an essential input to reach an accurate valuation of the potential oil reserve. Here we formulate the problem as a time series based supervised learning. In general, there is sequential information encoded within the time series event, and carefully selected algorithms are necessary to fully unlock the potential in the data.

On the other hand, with enormous volume of datasets (collected from all kinds of sensors at different stages) becoming available and rising power of cloud computing, big data analytics or artificial intelligence is gaining momentum in oil and gas industry. Few examples are deep learning application in seismic fault interpretation (Lu, et al., 2018), rapid development of real-time drilling analytics system, (Cao, et al., 2018), and implementing integrated production surveillance and optimization system (Sankaran, et al., 2017).

In this paper, we will be discussing a machine learning approach, to be more specific, long short-term memory (LSTM) method to predict oil rate. Very little historical data is needed, say only the initial 3 months production will be fed into the training model, and based on that we are able to make reasonable prediction for the first 2 years or even longer. As most of the shale wells decline rapidly, and the first few years' production could contribute around 70% or even more of the total EUR, so the prediction provides valuable insights for reservoir evaluation. Since the decline is rather steep, it might be difficult for one individual model to capture the trend completely, leading some estimate to fall into the category of over predicting, which motivates us to naturally introduce the ensemble learning to reduce the variance. More than 300 wells in an onshore unconventional asset has been analyzed through hindcasting, where the mean difference between the predicted and actual accumulated production is less than 0.2% and the variance is less than 5%.

Method

Our proposed method is based on recurrent neural networks (RNN), a class of neural network that can predict the future based on past values. They have been used to predict when to buy or sell in the stock market, anticipate car trajectories in autonomous driving systems, and sentiment analysis in natural language processing (NLP) (refer chapter 14 of Geron, 2017).

Humans don't start their thinking from scratch all the time, and thoughts have their persistence. One drawback in the traditional feedforward neural networks is that they are not able to capture sequential information well. Recurrent neural networks address this issue by building loops within the networks to allow information or memory to persist. Figure 1 displays the basic architecture of RNN that receives inputs, produces output, and sends that output back to itself. Since the output of RNN is a function of all the inputs from previous time steps, it makes sense to claim that the RNN has a form of memory. The component of neural network that preserves some state across time steps is called a memory cell.

$$h_t = f_W(h_{t-1}, x_t)$$

$$\downarrow$$

$$h_t = tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$

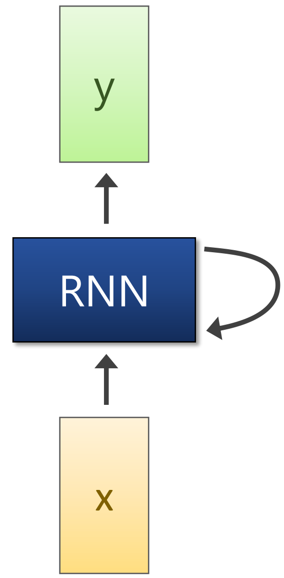
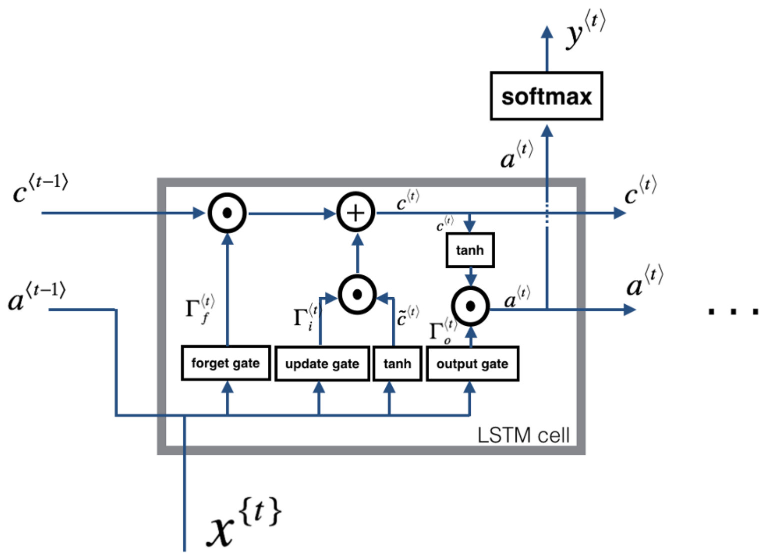


Figure 1. A recurrent neuron with mathematical formulation

However, there are two practical challenges in training an RNN model. One is for long sequences – many time steps need to be run and we end up with a very deep unrolled RNN network, which may suffer from vanishing or exploding gradients problem. The other is related to the memory, as the data goes through an RNN, some information is lost after every time step and the RNN's state might contain no trace of the early inputs at later stage. To solve this problem, various types of cells with memory have been introduced.

The Long Short-Term Memory (LSTM) was proposed in 1997 by Sepp Hochreiter and Jurgen Schmidhuber. At the cell level, it can be used like a basic RNN cell, and the benefit is that it will perform better mainly from two aspects – first, the training will converge faster and the other is it will be able to detect long-term dependencies. There are two state vectors in LSTM cells – a short-term state and a long-term state. Through training, LSTM cells learn to recognize an important input, store it in the long-term state, preserve it as long as needed, and extract it whenever necessary. There are three gates in the LSTM with different functionalities, and forget gate will control which parts of the long-term state should be erased, input gate which parts be added, output gate which parts be read and output at a specific time step. Figure 2 is the architecture of a basic LSTM cell.



$$\Gamma_f^{\langle t\rangle} = \sigma(W_f[a^{\langle t-1\rangle}, x^{\langle t\rangle}] + b_f)$$
$$\Gamma_u^{\langle t\rangle} = \sigma(W_u[a^{\langle t-1\rangle}, x^{\langle t\rangle}] + b_u)$$
$$\tilde{c}^{\langle t\rangle} = \tanh(W_C[a^{\langle t-1\rangle}, x^{\langle t\rangle}] + b_C)$$
$$c^{\langle t\rangle} = \Gamma_f^{\langle t\rangle} \circ c^{\langle t-1\rangle} + \Gamma_u^{\langle t\rangle} \circ \tilde{c}^{\langle t\rangle}$$
$$\Gamma_o^{\langle t\rangle} = \sigma(W_o[a^{\langle t-1\rangle}, x^{\langle t\rangle}] + b_o)$$
$$a^{\langle t\rangle} = \Gamma_o^{\langle t\rangle} \circ \tanh(c^{\langle t\rangle})$$

Figure 2. LSTM architecture (from https://github.com/aniruddhachoudhury)

Here we include the architecture implemented for this study. First, we feed the features (tubing pressure, latency information and etc.) into the first LSTM layers. The input data need to be shaped into a 3 dimension matrix, with the first dimension as number of samples, second as the time step or 'look back', and the third is the number of features (here is number of tubing pressure + number of latency which includes oil production rate for past 3 days + number of selected wells to build the prediction, and will explain how to select wells in details later). The purpose of stacking 3 recurrent layers here is to increase the representational power of the network, and all the immediate LSTM layers need to return the full sequence of outputs for each timestep (a 3D tensor of shape (batch_size, timesteps, features)), while the final LSTM layer only the last output for each input sequence (a 2D tensor of shape (batch_size, features)). In the end comes the fully connected layer to aggregate all the processed information to make the prediction. The structure can be found in figure 3.

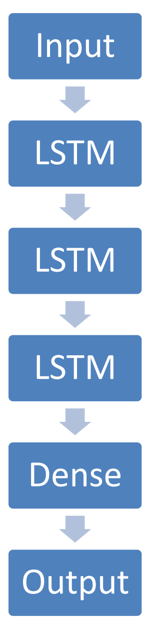


Figure 3. The underlying stacked long short-term memory architecture

For most practical data science problems, developing the right algorithm is only part of the story. Feature engineering is almost a must. The error accumulation is a fundamental problem in time series prediction, and we need to find a way to control the accumulated error. Besides the tubing pressure, existing oil production data in the neighboring region are extracted to guide prediction for robust results. In additional to that, we take advantage of ensemble learning. In many cases, an aggregated answer is usually better than an individual method.

We include additional features in building the model based on the initial 3 months data. In addition to the tubing pressure and latency oil rate, 12 similar wells are selected dynamically from the database and translated into derived features. A natural question to ask is how to pick those similar well to guide the prediction? Since wells are somehow encoded by their oil production rate, and we can define the similarity between wells using time series techniques, say correction or dynamic warping, or simply some commonly defined measurement, like Manhattan or Euclidian distance. The fundamental assumption here is that if two wells perform comparable in the first 3 months, so as the first 2 years, therefore the accumulated error over time could somehow be bounded by existing wells' production. For the following case, we use the correlation as the metric to rank and select those similar wells. See figure 4 for the workflow.

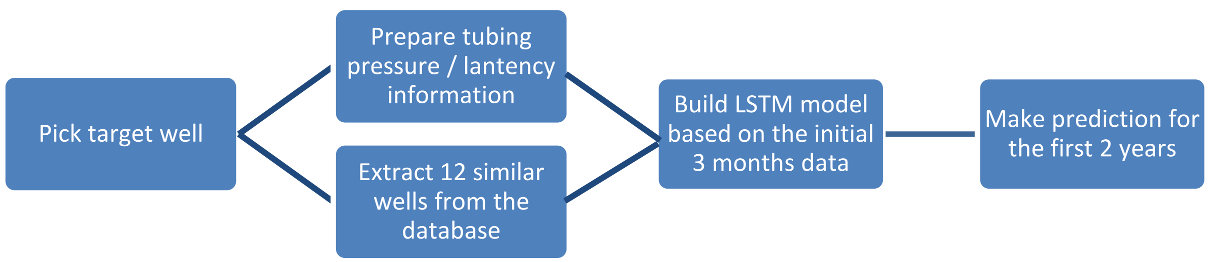
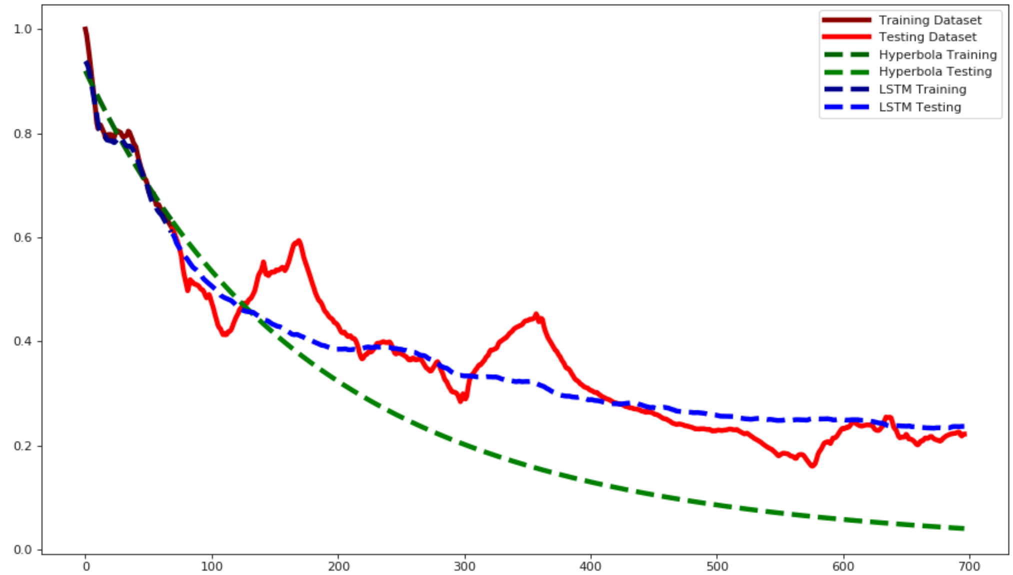


Figure 4. Schematic diagram for the prediction workflow

For unconventional resources, the sharp decline rate is one characteristic for majority of the wells, and the forecast could easily be over-predicting. So we incorporate another workflow to address this issue. In order to reduce the curvature for the decline curve, we leverage the idea in calculus, or to be more specific, change of variable. Instead of predicting the oil rate, we change the target variable to cumulative production. In the next section we will notice that such approach results in slightly under-predicting estimate, and integrating these two models makes the forecasting more reliable.

Results

More than 300 wells with more than 2 years of production history from an onshore unconventional asset were selected to conduct hindcasting. In Figure 5, we compare the hyperbolic decline curve analysis (current industry method) with LSTM (proposed approach), and we can tell the improvement from applying the LSTM method. Here the model is trained on the first 3 months of data, and we validate the time series prediction on the next 21 months. Both rate and cumulative plots are included that highlight the good match with the true production. In general, actual production data is impacted by operational changes that cause unnatural movements in the decline rate domain, and simple curve fitting could suffer greatly from those operational adjustments. Whereas, existing physics-based models that account for pressure variations are limited by the availability of consistent and reliable bottom-hole pressure and other basic reservoir information, our proposed approach is still data-driven and based on routinely available operating data.



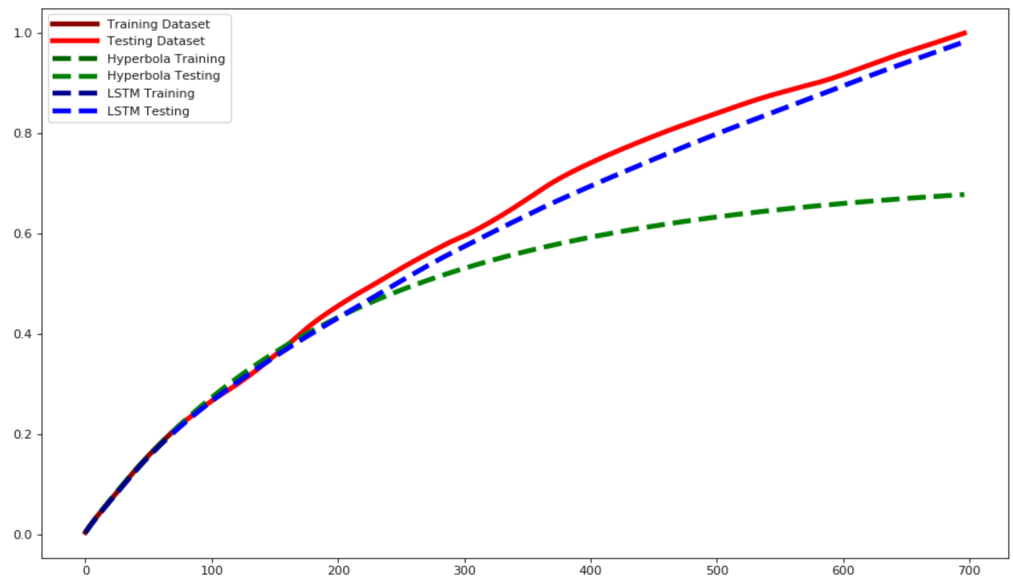


Figure 5. Oil Rate and Cumulative Oil Production

With the appropriate method and features, we are able to solve 80% of the problems. Based on the observations, the prediction for some wells are still far away from the ground truth. There could be many reasons for this, as LSTM attempts to capture dependency over long period of time, it is possible that such relationship happens to be weak for certain wells, if that is the case, the LSTM might not the most effective procedure. When it comes to features, there could be measurement errors in both oil rate and tubing pressure, and we might not have similar wells available to guide prediction from the existing database, meaning that it could be extremely challenging to overcome the continuously growing error in long run through introducing regularization from other wells. As mentioned earlier, the sharp decline is a prompt character for most of the shale resources, leading to an interesting observation that most of decline predictions are slightly over predicting for the current model. In order to compensate that, we develop another model that will be trained in the accumulated domain. Essentially, the accumulated production is the discrete summation (or continuous integral) of the oil rate. As we know, integral is a way to reduce or smooth the curvature, and we will see the new model turns out to be under predicting most of time, which happens to calibrate the existing model in the right direction. By integrating these two models, the prediction results is much better for most of situations.

In the following Figure 6, the first plot is the prediction model from the rate and we show the forecasting for the cumulative production, while the second is the other model built on the cumulating domain. The training process is the same, and the first 3 month of data is fed into the model, and the result is validated on the next 21 months. This case is very extreme, and both the models are far off from the true, meaning that both the LSTM and existing history data are not able to fully capture the trend for this particular well. However, by integrating these two models, those over-predicting and under-predicting can offset each other and would be able to generate a reasonable prediction for the remaining 20% wells, whose analysis requires more efforts.

Ensemble multiple models is a very common practice in most of data science projects and Kaggle competitions. For our particular problem, the rate forecasting model slightly suffers from over-predicting, and as mentioned earlier, this is mainly due to the sharp decay, which is a distinct nature for unconventional oil production. There are many methods to address this issue, one approach is simply applying transformation to change the underlying representation of the data structure. Many analytic methods are available for this purpose, and some of them could be constructive but lack of physics explanation, like Log transformation, for example. Eventually we narrow down and propose to build a forecasting model for cumulative oil production, not only because it addresses the over-predicting problem, but also incorporates physics into the predictive modeling. It might look purely coincidence that integrating these two models happen to generating better forecasting results, yet the second model is carefully designed in response to the specific issue (over-predicting) we encountered in the first model, and it turns out to be remarkably effective.

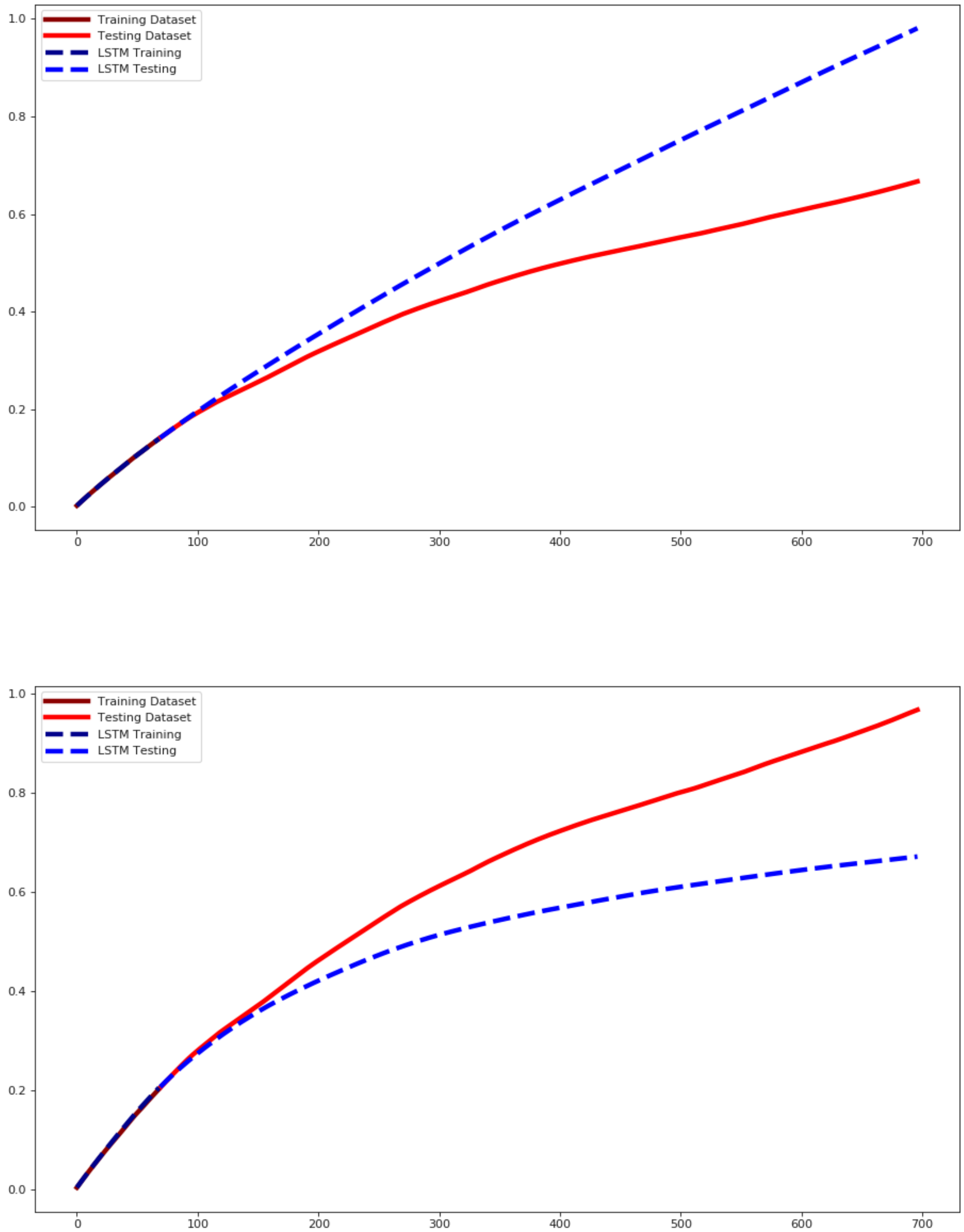


Figure 6. Two Prediction Models

In summary, (see the following ensemble workflow in figure 7), two individual models are constructed separately and independently, and we will combine them at later stage using the weighted average. The weight is calculated statistically by running the models on all existing wells to find the best ratio. As we increase the number of wells in the database (quite rapidly under current pace of development), it is recommended to update the ratio and adjust the weights to keep the prediction more dynamic, up-to-date and accurate. Figure 8 is a scatter plot of backtesting (or hindcasting) to compare the real production and prediction for the first two years for more than 300 wells, and with so little data (first 3 months) fed into the model for training, the performance is very promising.

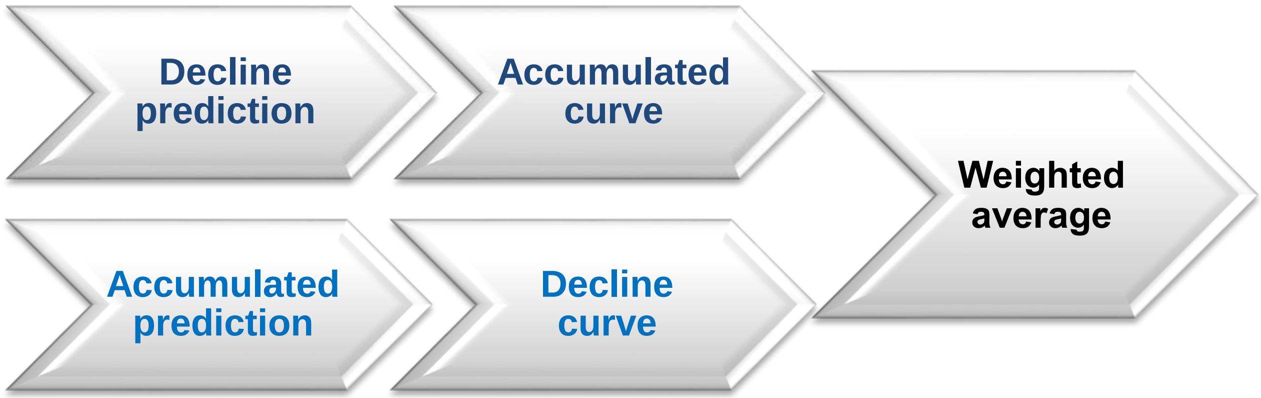


Figure 7. Ensemble Forecasting Workflows

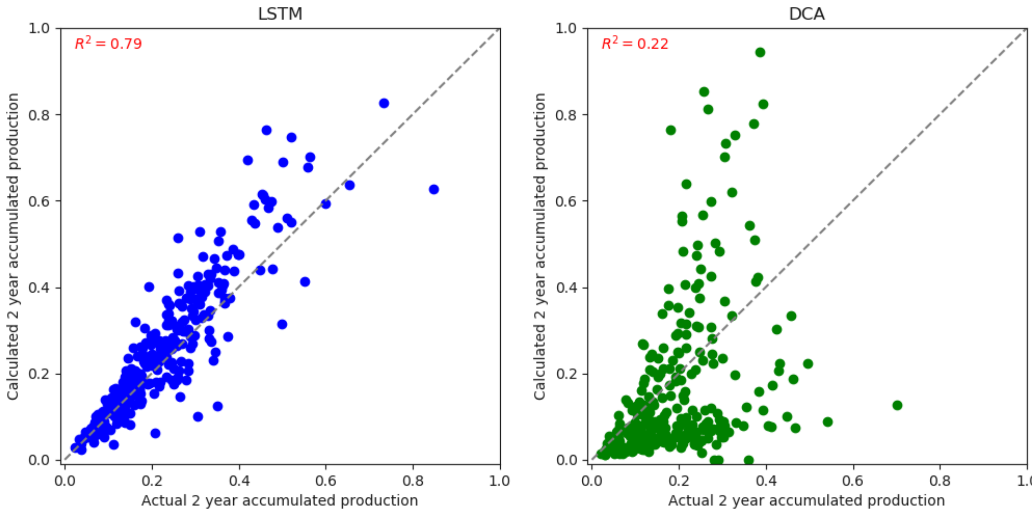


Figure 8. Predicted vs. Actual 2-year Prediction for LSTM on the left and DCA on the right

Limitations

Here we only utilize the dynamic features, without any static features, like the geological properties (e.g. porosity, permeability, formation, fluid etc.), well geometry (e.g. location, lateral length etc.) or completion properties (e.g. number of fractures, cluster spacing etc.). The intended use of the method is therefore limited to observing field data (production rates and well head pressure) and use it for forecasting.

Conclusions

Forecasting is one of the central function in unconventional field development, and here we establish an innovative workflow to address the oil rate prediction for unconventional resources. The solution is not straightforward as we face a number of practical problems due to operational changes, curtailed production, limited available historical data, complex flow behavior, well interference etc. The combination of a fit-for-purpose algorithm (LSTM to capture long-period dependency) with feature engineering (existing history data to control growth of error) and ensemble learning (reduce variance arising from individual experts) appear to be one of the most promising data-driven approaches to tackle the forecasting challenge.

Acknowledgement

We would like to thank the following people for their constructive discussions and suggestions for this project – Diego Molinari, Jackson Bi, Bin Xu, Sanjay Paranji, Yuxing Ben, Alex Bayeh, Dingzhou Cao, Ping Lu, Yanyan Zhang, Mustafa Kara, Michael Perrotte, Ingrid Tobar, Natalie Berestovsky, Ricardo Fernandez and Yuan Xiao.

References

1. Arps, J.J. 1944. Analysis of Decline Curves. Trans. AIME 160: 228-247.
2. Cao, Dingzhou, Chad Loesel, and Sanjay Paranji. 2018. "Rapid Development of Real-Time Drilling Analytics System." IADC/SPE Drilling Conference and Exhibition, Fort Worth, Texas, USA.
3. Duong, Ahn N. 2010. “An Unconventional Rate Decline Approach for Tight and Fracture-Dominated Gas Wells.” SPE 137748, Canadian Unconventional Resources & International Petroleum Conference, Calgary, Alberta, Canada.
4. Fetkovich, M. J. 1980. Decline curve analysis using type curves. Journal of Petroleum Technology, 32(06), 1-065.
5. Géron, A., 2017. Hands-on Machine Learning With Scikit-Learn and TensorFlow: Concepts Tools and Techniques to Build Intelligent Systems, Sebastopol, CA, USA:O’Reilly.
6. Lu, Ping, Matt Morris, Seth Brazell, Cody Comiskey, and Yuan Xiao. 2018. "Using generative adversarial networks to improve deep-learning fault interpretation networks." The Leading Edge 37: 578-583.
7. McNeil, R., O. Jeje, and A. Renaud. 2009. "Application of the power law loss-ratio method of decline analysis." Canadian International Petroleum Conference, Calgary, Alberta, Canada.
8. Ristanto, Tita, 2018. Deep Learning Application in Well Production. CS230, Stanford University, Stanford, California, USA.
9. Sankaran, Sathish, David Wright, Huan Gamblin, and Dhilip Kumar. 2017. "Creating Value by Implementing an Integrated Production Surveillance and Optimization System–An Operator's Perspective." SPE 187222, SPE Annual Technical Conference and Exhibition, San Antonio, Texas, USA
10. Webster, Jamie, Goydan Paul and Oudenot Eric, Profiting from Shale in Five Phases, 2018. BCG publications, USA.
11. Yu, Rose, Yaguang Li, Cyrus Shahabi, Ugur Demiryurek, and Yan Liu. 2017. "Deep learning: A generic approach for extreme condition traffic forecasting." SIAM International Conference on Data Mining, Atlanta, Georgia, USA.