A Comparative Study of Different Deep Learning Models for the Prediction of Natural Gas Demand and Price in the United States

Vidhyadhar Manee, Jorge Chebeir, Jose A. Romagnoli*

Louisiana State University, Cain Department of Chemical Engineering, 3307 Patrick F. Taylor Hall, Baton Rouge, LA, U.S.A
jose@lsu.edu

In this work, the impact of network architecture on the natural gas demand and price forecasts task is evaluated. Recurrent models such as GRU (Gated Recurrent Units) and LSTM (Long Short-Term Memory networks) are investigated to verify that their impressive performance on sequence modeling tasks like audio synthesis can be transferred over to this challenging domain. The effect of data decomposition using Empirical Mode Decomposition technique is explored. Finally, the effect of the encoder-decoder architecture on model performance is evaluated in this work. The results obtained by the different approaches are then compared to identify the most appropriate model for the case studies analyzed.

1. Introduction

Natural gas represents a highly efficient and low energy intensive fossil fuel in comparison to other sources like coal. This energy commodity not only plays a key role as fuel in the residential and commercial heating market but also represents an important feedstock for industry and power generation. Over the past decade, the combination of different technological advancements including extended-reach directional drilling and multi-stage horizontal fracturing has unleashed the enormous potential of the United States as a producer of natural gas from shale rocks. The production of dry gas from shale gas and tight oil is expected to continue growing in the future and eventually account for more than three-quarters of the natural gas production in 2050. Long-term consumption is also expected to increase in this period, mostly driven by the industrial and electric power demands (EIA, 2018). In this context, natural gas is expected to continue playing a critical role in the energy consumption structure of the country in the future.

From the perspective of gas companies and government, the forecast of natural gas demand is essential to plan the long-term operations of the energy sector and to define the most appropriate policies. Incorrect demand forecasts can provoke a detrimental effect on the economy of the different players involved in the gas supply networks. It could force gas producers to make costly strategic mistakes or consumers to plan erroneously. To avoid these unpleasant scenarios, an accurate forecasting methodology needs to be developed in order to minimize the errors associated with mismatches between supply and consumption.

Similarly, the prediction of the natural gas price represents another fundamental decision-making tool for the different players involved in the oil and gas industry of the United States. It is highly challenging to get accurate predictions of gas price owing to the rapid and unpredictable movements of this exogenous variable in the energy market. During the period following the deregulation of gas markets in the late 1980s, prices fell to nearly US$ 2.3/MMBtu and remained at that level throughout 1990s. After 2000, gas prices started to increase rapidly and continued to be highly volatile. By 2008, the prices averaged $8/MMBtu (four times the levels of 1990s). After 2008, it entered a period of decline until the prices dropped to US$2.64/MMBtu (Power and Council, 2016). Therefore, a volatile behaviour is a characteristic feature of the natural gas price in the country. Although numerous statistical techniques have been developed for the prediction of time series (Box and Jenkins, 1970; Cox and Ross, 1976; Bollerslev, 1986; Zhang et al., 1998), machine learning models have emerged in the last years as serious contender in this field (Werbos, 1974; Werbos, 1988). Throughout this

work, different machine learning algorithms are explored for the prediction of the demand and price of natural gas in the United States. In the particular case of gas price, the Empirical Mode Decomposition (Huang et al., 2003) transformation is incorporated to improve the predictions of the algorithm.

2. Empirical-Mode Decomposition (EMD)

In time series analysis, the data/signal varies as a function of time. It carries useful information that can be used to forecast the future by studying the patterns in the signal. Over the years, several algorithms have been developed to model the signal by exploiting their inherent temporal structure. Some of the most successful algorithms like the ARMA class of models capture these patterns only if the data conforms to certain expectations. Data generated in the real world do not always meet these expectations and alternative algorithms are necessary to model their dynamics.

One way to simplify the task is to deconstruct the data into sub-parts that can be modeled individually (Hsieh et al, 2011). Mathematically, there can be infinite representations of the data but only some of them are informative. The best representations are those that help in identifying the underlying trends in the data which can later be exploited.

One of the more informative representations of the data is the frequency-time plot. The Fourier analysis is a technique that deconstructs the data into sine and cosine functions each of which has a unique frequency and amplitude. This helps to reveal patterns that might not be obvious in the raw data. But these algorithms require the data to be stationary and linear which cannot always be satisfied.

Several techniques have been developed to address the drawbacks of Fourier analysis. All of them try to fit the data to predetermined basis functions which fails if the data generating process is complex. A more effective approach to this problem is to extract the basis functions directly from the data. The Empirical mode decomposition (EMD) (Huang et al., 2003) model does just this. It deconstructs the time series data into subcomponents called Intrinsic Mode Functions which are adaptively computed from the data. An intrinsic mode function is a signal that satisfies the following two constraints.

i). the number of extrema and the number of zero crossings must either equal or differ at most by one

ii). At any point, the mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero.

The intrinsic mode functions are computed through a sifting process, the details of which are described here (Huang et al., 2003). Once the IMFs have been determined, the time series models can be applied to the IMFs directly to model the trends in the data (Figure 1). In this work, we use EMD to decompose the price of natural gas into 10 different IMFs and a residual signal (Figure 2).
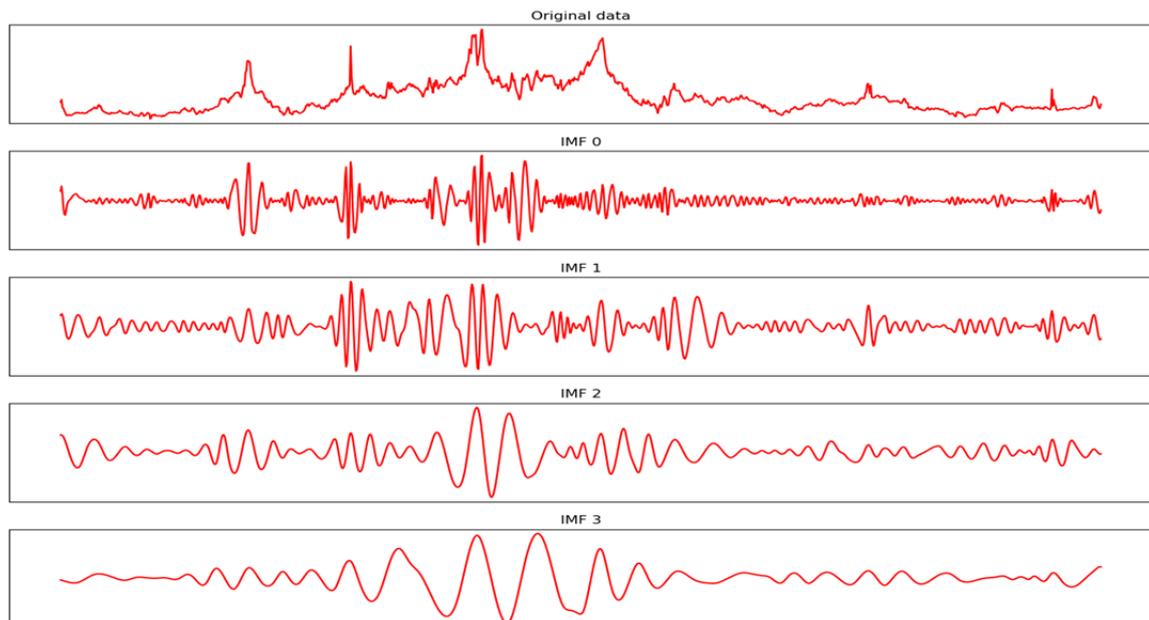

Figure 1: The results of empirical mode decomposition on the natural gas price data. Only the top 4 IMF's are shown here.

3. Recurrent Neural Networks

Neural networks (NN) are a class of self-learning algorithms that can be trained to identify patterns from data. In the last few years, they have greatly advanced research in areas related to Image object detection, Natural language processing, Self-driving cars, Medicine etc. One of the primary assumptions in these models is that the data samples used to train them are independent of one another. While this assumption holds true for a lot of applications, there are important exceptions. In tasks like image captioning, natural language processing, time series forecasting etc. the samples in the dataset are interdependent and assuming otherwise will lead to loss of valuable information.

The recurrent neural networks are a sub-class of neural networks that were built to model the long-range dependencies inherent between data samples. While the ordinary NN does not respect the temporal order of the input data, the recurrent neural network avoids this problem by having the notion of time built into it. Similar to other neural network architectures, the RNNs have a hidden state. But, unlike other models, the RNN updates its hidden state after processing each timestep in the input. This ensures that the temporal structure of the input sequence is respected.

3.1 LSTM/GRU

Although RNNs have the right framework to model temporal data, they suffer from some inadequacies. NNs learn by backpropagating the error to modify the network weights. If the network is too large or the number of timesteps is too long, the distance over which the error must be transmitted becomes infeasible and the network stops learning. This problem is referred to in literature as the vanishing gradient problem.

The LSTM (Hochreiter et al., 1997) is a modified RNN designed to overcome vanishing gradients. In addition to the hidden state, the LSTM carries a cell state that preserves long term information. The LSTM solves the problem by having a shortcut path to transmit the gradients back. This unique architecture involves three gates. The forget gate decides the contents of the cell state that are relevant to the problem and discards everything else. The input gate's function is to ensure that relevant information in the input is stored in the cell state. Finally, the output gate adds the relevant input to the cell state and passes it on to the next timestep. This architecture ensures that the gradients are propagated back into the earlier timesteps in the network.

The GRU (Cho et al., 2014) is another modified RNN cell that is similar to the LSTM in many ways. It differs from the LSTM in that it has two gates in place of three. The update gate and the reset gate preserve long term information in the data. While the reset gate determines the information that is important to hold over the long term, the update gate ensures that this information is added to the hidden state. This architecture ensures that the GRU has fewer parameters to train than an LSTM. It saves memory and take less time to train. However, on small sequences this typically does not make much of a difference.

4. Methodology

The experiments are designed to explore the effect of different RNN cells, architectures and data transformation on the forecasting performance of neural networks. The Vanilla RNN cell is the basic structure that is used as the baseline for our results. Both GRU and LSTM have been shown in literature to produce competitive results in a wide range of fields. Their performance on this time-series task is compared.

The simplest architecture used in our experiments is the stacked RNN model (Figure 2). Here, the RNN cells are layered one over the other with each cell connected to itself and the layer above it. The other architecture explored in this work is the Encoder-Decoder (ED) architecture (Cho et al., 2014) model popular in the field of natural language processing (Figure 2). Here, the cells are layered one over the other like the basic architecture. However, in the central layer, the cells are forced to withhold their connection with the layer above them until the last timestep. This ensures that the output of the layer is a summary of the contents of all the input timesteps. This summary vector is then fed as an input to all the timesteps in the layer above. Finally, the last variable in our experiments is data transformation. We test our models on both the raw data and the decomposed signals from EMD. The results of our experiments are shown in Figure 3.

5. Data Preprocessing

After the EMD was applied to the original time series, the resulting IMFs were processed using the following steps. First, the data was normalized into the [0, 1] range using the min-max scaling formula. Next, it was organized into input and output sequences using the lookback window approach. The length of the lookback window and the forecast horizon was set to 24. In order to ease the optimization of the RNNs, there was partial overlap between the input and the output sequences. The resulting sequences were finally split into training, validation and test sets.
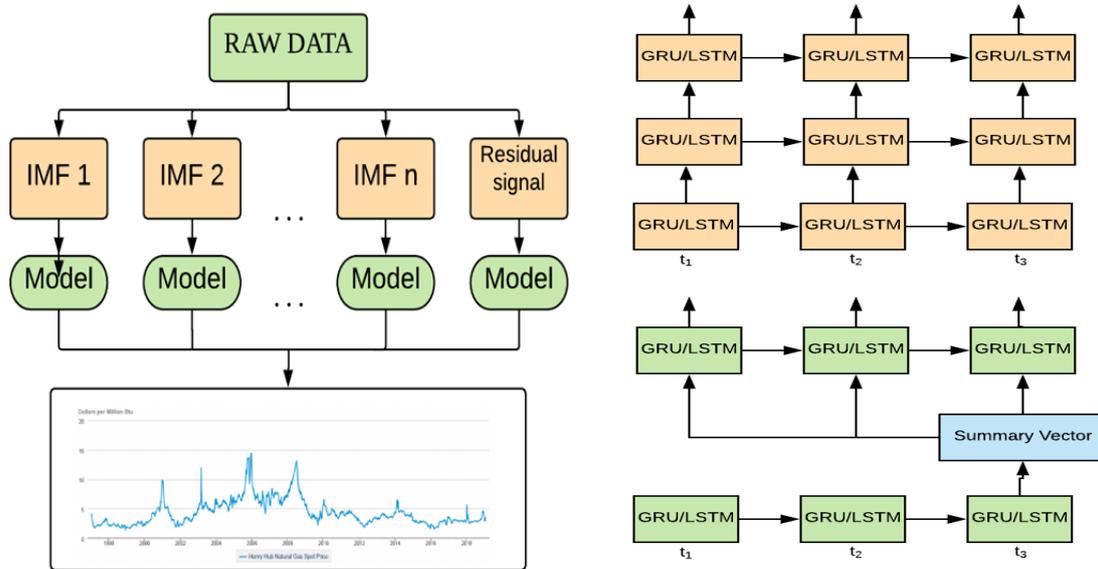
Figure 2: (left) Overall model structure incorporating empirical mode decomposition (top right) regular stacked RNN architecture (bottom right) Encoder-Decoder architecture with summary vector.

6. RNN Results

While predicting the natural gas demand, all the models used in the experiments converged to satisfactory optimums. This is evident by the low mean squared error (MSE) of the models on the test set predictions (Figure 3). They were able to pick up the dynamics of the signal and forecast accurately into the future. The mean squared error of the model predictions is shown in figure 3 while the predictions are shown in figure 4. The accuracy in predictions can be explained by the regular seasonality of the data and the absence of noise or irregular spikes. The slow long-term increase in the mean value of the demand was also picked up by the models.

On the other hand, the prediction of the natural gas price was more challenging. In terms of training time, the Vanilla RNN cell took the least time to train. This can be attributed to the fewer weights in the Vanilla RNN cell. The GRU has 3x weights as a Vanilla RNN while the LSTM has 4x the weights. However, the training of the Vanilla RNN was very unstable with the optimization converging at poor local optimums regularly. The cell was unable to capture long term dependencies in the model which is shown in the high MSE produced on the test set (Figure 3).

The LSTM cell exhibited better performance than the Vanilla RNN as expected. It was able to model the low frequency IMFs well, but it struggled with the higher frequency IMFs. On the higher frequency IMFs, the training was unstable, and the model took several epochs to converge to an acceptable optimum. This was surprising because the LSTM has been shown to outperform other models in a variety of tasks and was expected to do the same here.

Finally, the best performance was achieved by the GRU cell. It was able to pick up patterns in all the IMFs consistently and the training was stable. It also converged to the optimum faster than the other cells. A common theme among all the model predictions was that the predictions were accurate in the initial time steps of the test set but grew increasingly erratic as time progressed. This can be attributed to the accumulation of error over the prediction horizon.

In terms of overall architecture, the basic architecture performed poorly on all our experiments. Increasing the depth of the model did not improve performance. The ED architecture with the summary vector, on the other hand, was able to capture the dynamics of the data. The summary vector is very important to this architecture since it ensures that the entire sequence of input is processed before the model makes its predictions. This can also explain why the basic architecture failed to model the data properly.

A notable observation is that the dynamics of the highest frequency IMFs were essentially random and was impossible to capture by any of the models. The model predictions reverted to the mean of this IMF to minimize the loss function.
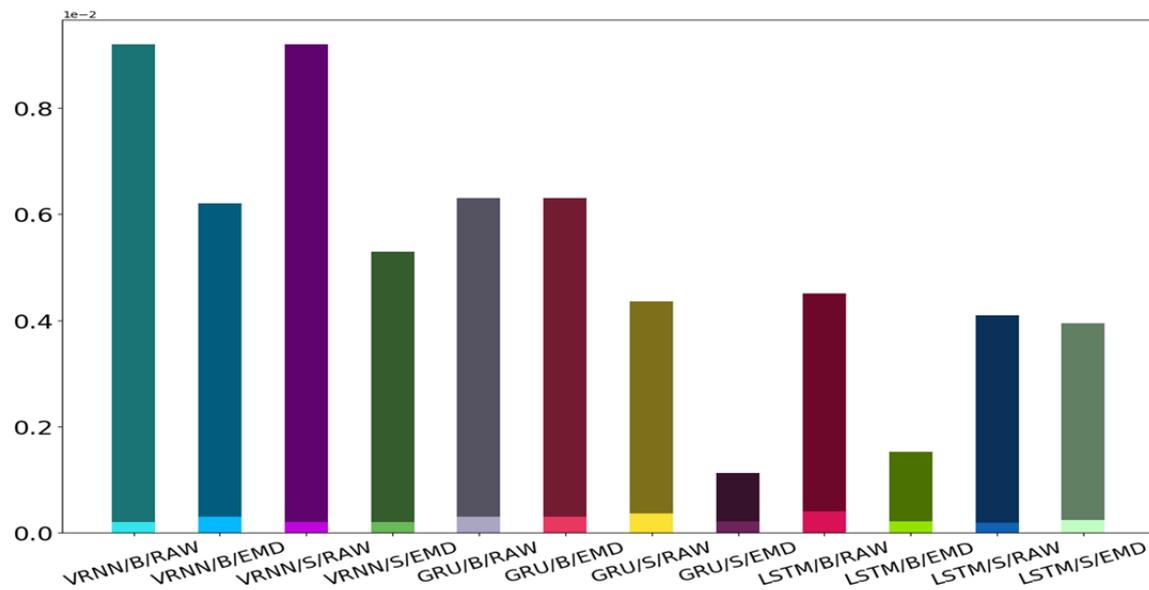
Figure 3: Mean squared error of the different models. Each model name is in the "RNN cell/architecture/data transformation" format with B denoting basic and S denoting summary vector". The bottom bar of each model corresponds to the demand prediction error while the top bar corresponds to the price prediction error.
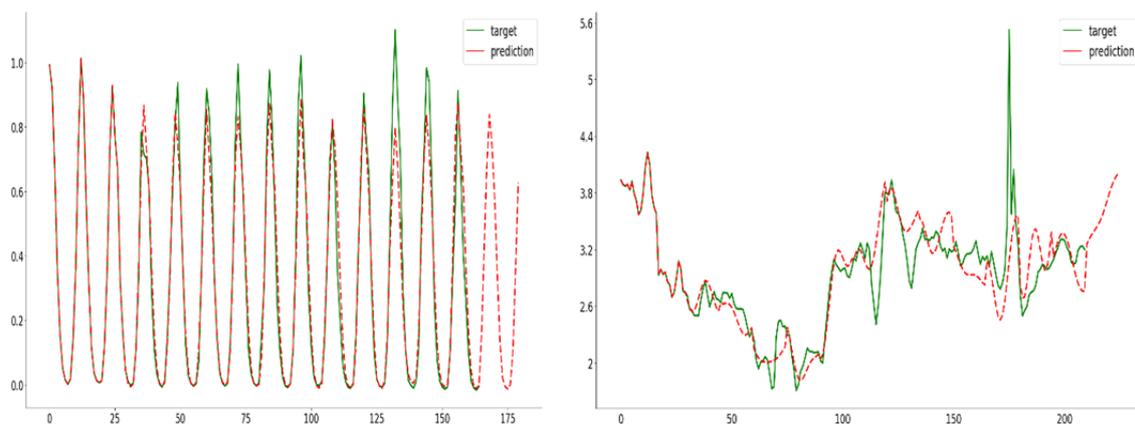

Figure 4: Predictions of the best performing model on the test set and future forecast for (left) natural gas demand and (right) natural gas price.

7. Conclusion

In this work, the prediction of natural gas demand and price using recurrent neural network architecture is presented. It is shown that the demand can be predicted accurately by all the models since the pattern is uniformly repeated. On the other hand, the price of natural gas was accurately predicted by only two models. The GRU cell with the ED architecture gave the best results while the LSTM cell with the basic architecture came second. Both models performed better when the data was deconstructed using EMD. The accuracy of the model predictions decreased with an increase in forecast horizon, which shows that there is still large scope for improvements in these models. Future work will be aimed at testing alternative neural network structures like the convolutional neural network and attention-based networks to improve the forecasting accuracy. In addition, the effect of pre-training on the current models will be investigated.

References

Bollerslev T., 1986, Generalized autoregressive conditional heteroscedasticity, Journal of Econometrics, 31, 307-327.

Box G.E.P, Jenkins G.M., 1970, Time series analysis, forecasting, and control, Francisco Holden-Day.

Cox J.C., Ross S.A., 1976, A survey of some new results in financial option pricing theory, The Journal of Finance, 31(2), 383-402.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. and Bengio, Y., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

Energy Information Administration (EIA), 2018, Annual energy outlook with projections to 2050, Government Printing Office.

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

Hsieh, T.J., Hsiao, H.F. and Yeh, W.C., 2011. Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial bee colony algorithm. Applied soft computing, 11(2), pp.2510-2525.

Huang N.E., Wu M. L.C., Long S.R., Shen S.S., Qu W., Gloersen P., Fan K.L., 2003, A confidence limit for the empirical mode decomposition and Hilbert spectral analysis, In Proceedings of the royal society of London: Mathematical, physical and engineering sciences.

Power N., Council C., 2016, Seventh northwest conservation and electric power plan, Portland, Oregon, USA.

Werbos, P., 1974, Beyond regression: New tools for prediction and analysis in the behavioral sciences, Ph. D. dissertation, Harvard University.

Werbos P.J., 1988, Generalization of backpropagation with application to a recurrent gas market model. Neural networks, 1(4), 339-356.

Zhang G., Patuwo B.E., Hu M.Y., 1998, Forecasting with artificial neural networks: The state of the art, International Journal of Forecasting, 14, 35-62.