engineering proceedings

MDPI

Proceeding Paper

Multimodal Model Based on LSTM for Production Forecasting in Oil Wells with Rod Lift System [†]

David Esneyder Bello Angulo [‡,§] and Elizabeth León Guzmán [*,‡,§]

Department of Systems Engineering, Universidad Nacional de Colombia, Bogotá D.C. 111321, Colombia; dabelloa@unal.edu.co

[*] Correspondence: eleonguz@unal.edu.co; Tel.: +57-(601)-316-5000 (ext. 14084 or 14011)

[†] Presented at the 10th International Conference on Time Series and Forecasting, Gran Canaria, Spain, 15–17 July 2024.

[‡] Current address: Av Cra 30 # 45-3-Building 453-Office 101, Bogotá D.C. 111321, Colombia

[§] These authors contributed equally to this work.

Abstract: This paper presents a novel multimodal recurrent model for time series forecasting leveraging LSTM architecture, with a focus on production forecasting in oil wells equipped with rod lift systems. The model is specifically designed to handle time series data with diverse types, incorporating both images and numerical data at each time step. This capability enables a comprehensive analysis over specified temporal windows. The architecture consists of distinct submodels tailored to process different data modalities. These submodels generate a unified concatenated feature vector, providing a holistic representation of the well's operational status. This representation is further refined through a dense layer to facilitate non-linear transformation and integration. Temporal analysis forms the core of the model's functionality, facilitated by a Long Short-Term Memory (LSTM) layer, which excels at capturing long-range dependencies in the data. Additionally, a fully connected layer with linear activation output enables one-shot multi-step forecasting, which is necessary because the input and output have different modalities. Experimental results show that the proposed multimodal model achieved the best performance in the studied cases, with a Mean Absolute Percentage Error (MAPE) of 8.2%, outperforming univariate and multivariate deep learning-based models, as well as ARIMA implementations, which yielded results with a MAPE greater than 9%.

Keywords: multimodal time series forecasting; oil production; machine learning; deep learning; neural networks

check for updates

Citation: Bello Angulo, D.E.; León Guzmán, E. Multimodal Model Based on LSTM for Production Forecasting in Oil Wells with Rod Lift System. *Eng. Proc.* **2024**, *68*, 31. https://doi.org/10.3390/engproc2024068031

Academic Editors: Olga Valenzuela, Fernando Rojas, Luis Javier Herrera, Hector Pomares and Ignacio Rojas

Published: 10 July 2024

1. Introduction

In an era defined by the proliferation of data from diverse sources and modalities, the predictive power of time series analysis has become indispensable across numerous industries. This study seeks to advance the capabilities of this field, particularly in the domain of multimodal time series forecasting. Traditionally, time series forecasting has focused on univariate and multivariate data; however, our research introduces an innovative approach by harnessing multimodal neural network models.

The novelty of our work lies in the development of a multimodal encoder architecture tailored to address the complexities of multimodal temporal phenomena. At each time point, our architecture integrates information from various modalities, such as numerical data and images. By doing so, it not only captures the individual behaviors of each data source but also elucidates the intricate inter-relationships among them. This holistic approach enables our model to achieve a deeper understanding of the phenomenon under study, recognizing that the whole is greater than the sum of its parts.

Our dataset revolves around the production dynamics of oil wells employing rod lift systems as an artificial lift method. This dataset encompasses multimodal information, including numerical data, images, and text, captured at each time step.

In our forecasting experiments, multimodal models featuring LSTM layers demonstrate superior performance over non-multimodal neural network models relying solely on numerical data, yielding a mean absolute percentage error of 8.2%. However, we note instances where the ARIMA model outperforms the multimodal approach, particularly in cases where the production time series exhibits values proximate to the mean with low dispersion.

Furthermore, we explore alternative model configurations by substituting the LSTM recurrent layer with a transformer encoder layer. However, given the nature of the data and the limited number of training examples, this specific architecture utilizing a transformer encoder fails to produce satisfactory results for this use case. This underscores the necessity of further research to enhance the performance of transformer-based architectures in the context of multimodal time series analysis.

2. Related Work

This section offers a comprehensive review of current state-of-the-art methodologies in time series forecasting, focusing on both traditional models and advanced deep learning approaches.

2.1. Traditional Models

Regression-Based Models: Commonly used methods such as Support Vector Machine (SVM), Linear Regression (LR), and Random Forest (RF) were initially developed for tabular data but have been adapted for time series forecasting. These models learn a mapping function from extracted features of time series, although they may overlook the critical temporal dimension inherent in such data [1].

Functional Linear Models (FLMs): FLMs extend multiple linear regression to functional data by employing basis functions such as Functional Principal Components (FPCs) or B-spline functions, addressing the continuous nature of time series [2].

Interval-Based Algorithms: Approaches like time series forest extract features from specific intervals of the time series, often outperforming models utilizing the entire series.

Dictionary-Based Algorithms: These methods build a "dictionary" of frequent patterns in time series. Some examples include Bag of Patterns (BOP) and Symbolic Aggregation Approximation Vector Space (SAXVSM) [3].

2.2. Deep Learning-Based Models

Residual Networks (ResNet): Particularly influential in univariate time series analysis, ResNet features an innovative structure with residual blocks that mitigate gradient vanishing problems, making it highly effective [4].

Fully Convolutional Neural Networks (FCN): Composed solely of convolutional layers, FCN is well-suited for regression and classification tasks in time series analysis.

Inception-Based Networks: These networks represent a significant advancement in deep learning for time series, incorporating funnel layers and filters of varying lengths, along with MaxPooling operations [5].

Transformer-Based Architectures: Recent studies have introduced architectures based on transformers, achieving better results than Multi-Layer Perceptron (MLP) models for certain data types [6].

2.3. Multimodal Approaches

While traditional models primarily handle univariate or multivariate numerical data, models like CNN-LSTM are utilized for time series of images. However, few state-of-the-art models effectively handle truly multimodal data.

One approach integrates recurrent and convolutional networks to merge time series and image data using a CNN-BiLSTM model. Another significant work employs a two-stage model with a multimodal autoencoder followed by an LSTM network for forecasting [7].

3. Data Description

This section offers a detailed description of the dataset used in this study, encompassing monthly measurements of various variables associated with 200 oil wells in a Colombian oil field. Each time series (one for each well) has between 100 and 300 observations of numerical variables and images related to the behavior of the system.

3.1. Numerical Data

Monthly measurements are recorded for 47 numerical variables related to production and the state of the artificial lift system for each well. These variables include total production (of all fluids), oil production, and water and solids contents, among others.

However, not all variables contribute equally to the production phenomenon, and correlations exist among them. To address this, the following two approaches were employed for dimensionality reduction, resulting in two distinct datasets: expert judgment-based variable selection and dimensionality reduction using Principal Component Analysis (PCA).

For expert judgment-based selection, the following variables were chosen:

- Oil production;
- Total production;
- Water and solids contents in oil;
- Liquid level above the pump;
- Pump filling percentage;
- Top hole pressure;
- Casing pressure;
- Salinity;
- Maximum load in the middle of the polished rod cycle;
- Peak load in the middle of the polished rod cycle;
- Maximum load of the polished rod;
- Peak load on the polished rod;
- Pump displacement;
- Pump volumetric fillage (%);
- Unaccounted friction.

Alternatively, employing principal component analysis (PCA) reduced the dimensionality to 30 variables, retaining 94.9% of the variance.

For further preprocessing, all selected numerical variables were scaled using a standard scaler, transforming each variable to have a mean of 0 and a standard deviation of 1, promoting consistency and facilitating model convergence during training.

3.2. Dynacard Images

Dynacards, also known as dynamometer cards, are visual representations of the load on the pumping unit over a pump cycle. These images are collected monthly and provide valuable insights into the mechanical state of rod lift systems. However, it is important to note that unlike numerical data, dynacard images do not provide direct numerical values of the variables. Instead, they offer a graphical depiction of the load dynamics throughout the pump cycle. Therefore, in our modeling approach, we treat dynacard images as valuable visual data, aiming to train the model to understand the patterns and deformations within these images rather than focusing on specific numerical values. Example Dynacard images are depicted on the left side of Figure 1.

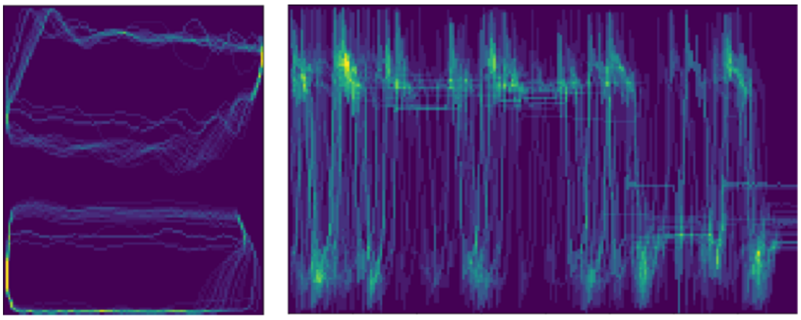

Figure 1. Representative images distributed over time. Dynacard image of 120 × 100 pixels on the left and valve test image of 120 × 200 pixels on the right. Each image source represents a different modality of the measured data.

3.3. Valve Test Images

Similarly, monthly-collected valve test images offer crucial insights into the behavior of the valve throughout a pump cycle. These images help assess valve performance and system efficiency. As with dynacard images, valve test images do not directly provide numerical data; rather, they visually illustrate the load behavior over time. Therefore, in our modeling framework, we leverage these images as informative visual data, training the model to interpret and understand the patterns and deformations within the images rather than relying on specific numerical values. Example valve test images are depicted on the left side of Figure 1.

3.4. Dataset Structure

The structure of the dataset is determined by the nature of the data and the specific requirements of the business context. Given the limited number of measurements for each independent time series, a structured dataset is essential for enabling the model to learn the general behavior of the series and predict the future behavior of each well.

To address the prediction problem, where the input and output do not share the same dimensionality, a model architecture is designed with an input consisting of a certain number of time steps from the past to make a one-shot multi-step prediction. After careful consideration, a window size of 36 steps (equivalent to 3 years) from the past was chosen to predict the next 24 steps (equivalent to 2 years). This decision was made based on an understanding of the data dynamics and the specific forecasting needs of the business context.

4. Baseline Models

Various baseline models were developed to compare results with the proposed multimodal model. The following models were developed:

- Auto_ARIMA: An ARIMA model for each time series was created using the auto-arima package from pmdarima [8].
- ARIMA model with the lowest AIC: The ARIMA model with the lowest Akaike Information Criterion (AIC) values according to the implementation by Bello-Angulo et al. [9].
- Univariate model: LSTM model that takes only the production window as input in making predictions. This model was trained on all time series, and its performance was assessed with and without fine tuning on the series to predict.
- Multivariate variable model: LSTM model that takes the window of all numerical variables as input in making predictions. One model was developed for variables selected by expert judgment and another for variables obtained with PCA. This model was trained on all time series, and its performance was assessed with and without fine tuning on the series to be predicted.

5. Multimodal Model Architecture

Our multimodal time series forecasting model is designed to receive a multimodal input at each time step and predict the specified number of steps. The architecture comprises several crucial components customized to handle and integrate various data modalities before making predictions. A schematic representation is depicted in Figure 2.

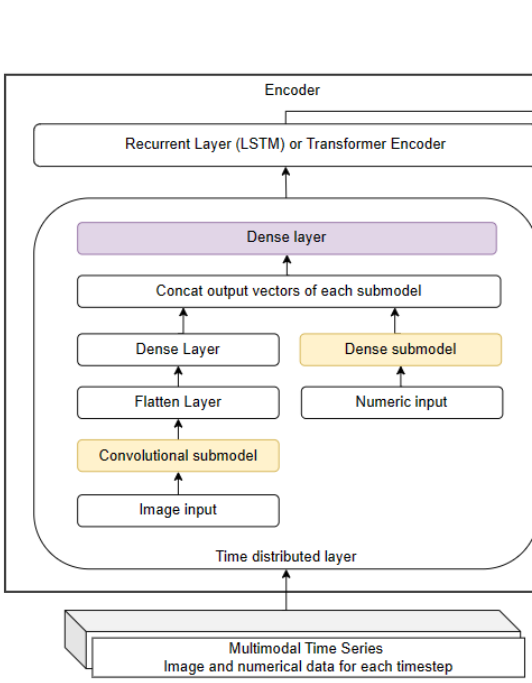
Forecasting of Ysteps for the interest variable
Encoder
Recurrent Layer (LSTM) or Transformer Encoder
Dense layer
Concat output vectors of each submodel
Dense Layer
Dense submodel
Flatten Layer
Numeric input
Convolutional submodel
Image input
Time distributed layer
Dense layer with linear activation function for Multi Output Regression
units: Ysteps
Decoder
Multimodal Time Series
Image and numerical data for each timestep

Figure 2. Schematic representation of the multimodal time series forecasting model, demonstrating the concatenation of submodels inside the time-distributed layer corresponding to each modality in the dataset. In this specific case, two modalities are utilized, namely images and numerical data.

5.1. Submodel Architectures for Heterogeneous Data

Numerical data submodel: For structured time series data, such as production test and sensor data, we employed a fully connected layer with a sigmoid activation function consisting of four units.

Dynacard and Valve Test Image Data Submodel: Convolutional Neural Networks (CNNs) were utilized for the image data from dynacards and valve tests. Each CNN comprises a series of two convolutional layers with ReLU activations paired with max pooling followed by a flatten and a dense layer for feature extraction. These CNNs transform the raw image pixels into a compact and informative representation of visual features. The submodel has 112 trainable parameters for dynacards and 120 for valve tests. Each submodel outputs a unidimensional vector of four positions.

5.2. Integration of Modalities

To amalgamate insights from all modalities, we concatenated the outputs of submodels into a unified feature vector, offering a holistic snapshot of the well's operational status at each time step. The length of this vector was fine-tuned via hyperparameter optimization, where various sizes are tested through a grid search to identify the most informative fusion of features from each modality. Following this optimization process, the concatenated output was fed through a dense layer. For this specific use case, the dense layer comprises eight units, a pivotal component in our model's architecture. This dense layer plays a critical role in facilitating further non-linear transformation and integration of features from each modality. Its significance lies in empowering the model to discern intricate patterns and relationships among the modalities, thereby enhancing its predictive capacity for future production outcomes.

5.3. Temporal Analysis

The core temporal analysis was conducted via a long short-term memory (LSTM) layer, which was specifically designed for the processing of sequential data. The unified feature vector for each time step was fed into the LSTM, allowing it to learn and retain information across the entire temporal window, employing a time-distributed Layer as shown in Figure 3.

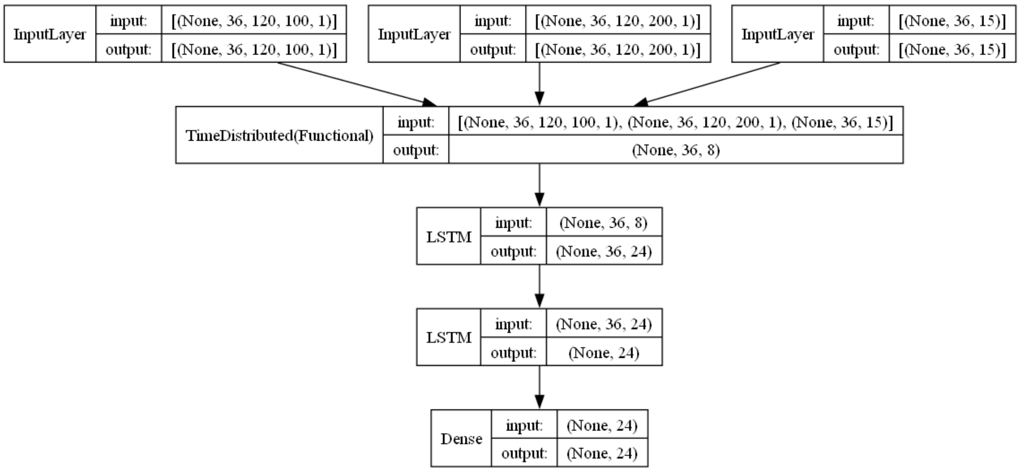
InputLayer
input: [(None, 36, 120, 100, 1)]
output: [(None, 36, 120, 100, 1)]
InputLayer
input: [(None, 36, 120, 200, 1)]
output: [(None, 36, 120, 200, 1)]
InputLayer
input: [(None, 36, 15)]
output: [(None, 36, 15)]
TimeDistributed(Functional)
input: [(None, 36, 120, 100, 1), (None, 36, 120, 200, 1), (None, 36, 15)]
output: (None, 36, 8)
LSTM
input: (None, 36, 8)
output: (None, 36, 24)
LSTM
input: (None, 36, 24)
output: (None, 24)
Dense
input: (None, 24)
output: (None, 24)

Figure 3. Time-distributed framework used to feed the submodels for each time step, then pass the output to the LSTM layers.

Moreover, we explored alternative configurations by substituting the LSTM recurrent layer with a transformer encoder layer. However, due to the nature of the data and the limited number of training examples, this specific architecture based on a transformer encoder did not yield satisfactory results for this use case.

5.4. Prediction

The prediction phase involves the decoder, which directly consists of a dense layer output with a linear activation function. The decoder features as many units as there are steps to predict, aligning with the dimensional requirements of the forecasting task. Since the dependent and independent variables exhibit a dimensional mismatch, rendering the use of a sequence-to-sequence model unfeasible, a multiple-output layer was employed. This configuration allows for the simultaneous prediction of multiple time steps in a single iteration.

In summary, the proposed model architecture offers a cohesive structure capable of capturing the complex multimodal interactions within time series data, specifically for oil production forecasting in this case.

6. Model Training

For model training, a computer with dual Xeon E2625-V3 processors, 40 GB of RAM, and an Nvidia 980 TI graphics card with 6 GB of VRAM was utilized. Details regarding the number of trainable parameters, the training time for each model, and the final training MSE for each model can be found in Table 1. The models were trained using a dataset consisting of 4322 training examples. Notably, the training data indicate that the transformer architecture exhibits slightly faster training times, which may prove advantageous when scaling up to larger datasets. However, it is worth mentioning that the LSTM architecture achieved superior results on the training data.

Table 1. Training data for the deep learning models.

Model	Trainable Params	Train Time (s)	MSE
Multimodal LSTM	8872	1858	0.34
Multimodal LSTM-PCA	8932	1899	0.32
Multimodal Transformer	8072	1832	0.39
Multimodal Transformer-PCA	8132	1758	0.36

7. Results

Table 2 displays the results of the tests conducted across 10 oil wells. These results are quantified in terms of MAPE (mean absolute percentage error), as shown in Equation (1). The findings indicate that the multimodal model exhibited the most favorable performance, with a mean absolute percentage error of 8.2%. Nevertheless, it is noteworthy that the ARIMA model with the lowest AIC outperformed the multimodal model in some specific cases.

$$\text{MAPE} = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{Y_i - \hat{Y}_i}{Y_i} \right| \tag{1}$$

Table 2. MAPE results of production forecasting for 10 oil wells for each model.

	MAPE for Each Oil Well (%)											
Model	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	Avg. **	Std. Dev
ARIMA with lowest AIC	6.4	7.1	7.9	6.2	2.5	15.9	17.4	9.2	12.6	5.0	9.0	4.8
Auto ARIMA	12.8	5.2	7.4	18.4	2.3	7.5	17.4	18.5	17.2	5.7	11.2	6.3
LSTM model for only one well	78.6	22.3	26.5	45.3	59.3	11.2	41.4	39.9	46.6	7.6	37.9	21.8
Multimodal LSTM	6.2	2.7	8.1	6.6	5.3	9.3	13.4	5.1	18.0	7.8	8.2	4.5
Multimodal LSTM-Fine-T *.	12.4	4.3	7.9	5.0	8.8	3.9	17.4	5.7	19.4	8.8	9.4	5.4
Multimodal LSTM-PCA	5.6	3.3	13.0	6.0	24.4	7.6	17.7	8.6	21.2	6.6	11.4	7.3
Multimodal LSTM-PCA-Fine-T *.	10.7	15.7	6.1	10.6	12.3	28.5	18.3	8.2	20.5	6.4	13.7	7.1
Multimodal Transformer	18.2	19.2	11.0	16.0	17.7	7.4	25.9	7.5	41.1	12.4	17.6	10.0
Multimodal Transformer-Fine-T *.	30.4	24.8	8.5	16.2	17.7	11.4	34.1	7.7	39.1	15.2	20.5	11.0
Multimodal Transformer-PCA	17.1	22.3	9.3	8.7	31.4	9.4	18.5	9.3	32.2	4.6	16.3	9.8
Multimodal Transformer-PCA-Fine-T*.	13.6	24.1	12.3	19.3	41.1	8.4	19.7	11.4	23.0	5.8	17.9	10.2
Univariate	68.5	34.1	46.1	43.8	47.6	28.9	55.1	39.2	56.5	15.0	43.5	15.2
Univariate-Fine-T *.	60.3	29.0	15.9	25.9	17.7	20.2	38.3	29.2	35.9	12.4	28.5	14.1
Multivariate	6.8	8.9	8.3	5.3	73.9	19.6	34.9	6.9	27.9	7.0	20.0	21.5
Multivariate-Fine-T *.	6.1	7.4	8.5	8.8	68.2	29.0	27.0	5.8	31.0	5.8	19.8	19.9
Multivariate–PCA	5.9	24.8	6.3	3.6	20.5	9.5	26.1	9.2	25.7	6.2	13.8	9.3
Multivariate-PCA-Fine-T *.	16.1	41.9	6.7	3.3	20.5	16.1	26.7	7.0	24.8	6.8	17.0	12.0

* Fine-T indicates that the model was fine-tuned using the time series data specific to the case. ** The Avg column presents the average of the individual results with the corresponding standard deviations.

Figure 4 illustrates an example of the prediction, showcasing the superior fit of the multimodal models compared to other developed models. Analyzing the cases where ARIMA fits better, it is observed that the ARIMA model performs better than the multimodal model when the production time series data are close to the mean and exhibit low dispersion (see Figure 5).

Figure 4 illustrates an example of the prediction, highlighting the superior performance of the multimodal models compared to other developed models. Upon closer examination of cases where ARIMA outperforms other models, it becomes evident that the ARIMA model excels when the production time series data are close to the mean and exhibit low dispersion, as depicted in Figure 5.

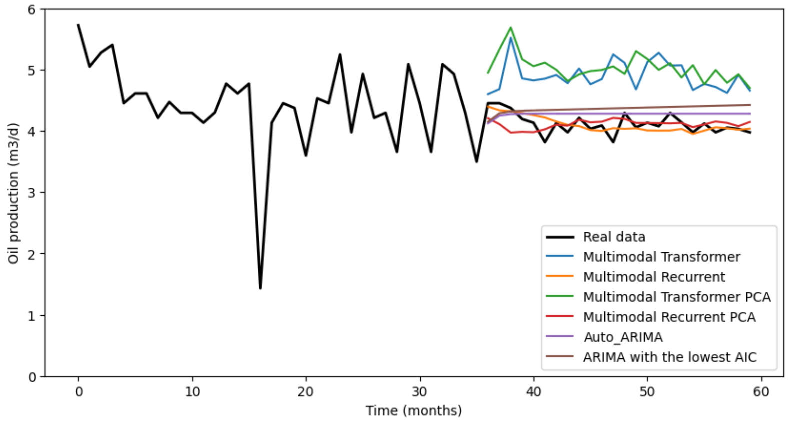

Figure 4. Example of prediction in a well. It is observed that the best result is obtained with the multimodal LSTM model.

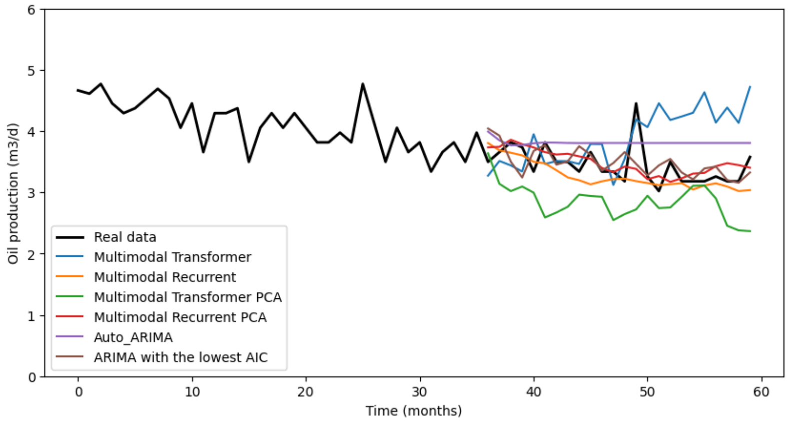

Figure 5. Example of prediction in a well with low dispersion. A better fit of the ARIMA model is observed.

8. Discussion

This study introduces an innovative multimodal time series forecasting model applied to predict the production of oil wells with rod lift systems. The key contribution of this research lies in the architectural design of the proposed model, which integrates separate submodels for various data modalities at each time step. This integration facilitates the effective processing and merging of image-based and numerical data, all of which are temporally distributed. The processed data seamlessly feed into a recurrent layer, with the output layer being a dense layer capable of making a one-shot multi-step forecast.

To assess the effectiveness of our approach, we compared it with non-multimodal univariate and multivariate models, as well as ARIMA models, leveraging state-of-the-art techniques for time series forecasting. Our findings are summarized as follows:

- The multimodal model demonstrates strong performance across the studied cases, exhibiting a mean absolute percentage error of 8.2%.
- Detailed analysis reveals that the ARIMA model outperforms the multimodal model when the production time series data are close to the mean and exhibit low dispersion.
- The multimodal model based on transformers did not yield satisfactory results for the studied use cases, possibly due to an insufficient number of training examples for this neural network architecture to learn patterns associated with the studied phenomena.

- Multimodal models exhibit a better fit for the numerical dataset with variables selected by expert judgment. Conversely, in the case of the multivariate model, the fit was better for the model trained on the dataset resulting from PCA.

Further research is warranted on the implementation of the multimodal model architecture based on transformers. Evaluating different positional encoding schemes proposed in the literature could better exploit the potential shown by this architecture in the fields of multivariate time series forecasting and extend it to the realm of multimodal time series.

We recommend implementing the proposed multimodal architecture in the study of other phenomena with temporally distributed multimodal information sources to validate its performance in different fields.

Author Contributions: Conceptualization, D.E.B.A. and E.L.G.; methodology, D.E.B.A. and E.L.G.; software, D.E.B.A.; validation, E.L.G.; formal analysis, D.E.B.A.; validation, E.L.G.; investigation, D.E.B.A.; validation, E.L.G.; resources, D.E.B.A.; validation, E.L.G.; data curation, D.E.B.A.; writing—original draft preparation, D.E.B.A.; writing—review and editing, E.L.G.; visualization, D.E.B.A.; supervision, E.L.G.; project administration, E.L.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to it belongs to a private company and has been shared solely for the purposes of this study under a confidentiality agreement.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Tan, Y. Time Series Extrinsic Regression. *Data Min. Knowl. Discov.* **2020**, *35*, 1032–1060. https://arxiv.org/abs/2006.12672v3. [CrossRef] [PubMed]
2. Goldsmith, J.; Scheipl, F. Estimator selection and combination in scalar-on-function regression. *Comput. Stat. Data Anal.* **2014**, *70*, 362–372. [CrossRef]
3. Lin, C. Rotation-invariant similarity in time series using bag-of-patterns representation. *J. Intell. Inf. Syst.* **2012**, *39*, 287–315. [CrossRef]
4. Fawaz, H.I.; Lucas, B.; Forestier, G.; Pelletier, C.; Schmidt, D.F.; Weber, J.; Webb, G.I.; Idoumghar, L.; Muller, P.; Petitjean, F. InceptionTime: Finding AlexNet for Time Series Classification. *Data Min. Knowl. Discov.* **2019**, *34*, 1936–1962. https://arxiv.org/abs/1909.04939v3. [CrossRef]
5. Xue, W.; Zhou, T.; Wen, Q.; Gao, J.; Ding, B.; Jin, R. Make Transformer Great Again for Time Series Forecasting: Channel Aligned Robust Dual Transformer. *arXiv* **2023**, arXiv:2305.12095. https://arxiv.org/abs/2305.12095v3.
6. Xian, W. A Multi-modal Time Series Intelligent Prediction Model. In Proceeding of 2021 International Conference on Wireless Communications, Networking and Applications—942 LNEE, Berlin, Germany, 17–19 December 2021; Lecture Notes in Electrical Engineering; pp. 1150–1157. [CrossRef]
7. Zhu, Q.; Zhang, S.; Zhang, Y.; Yu, C.; Dang, M.; Zhang, L. Multimodal time series data fusion based on SSAE and LSTM. In Proceedings of the 2021 IEEE Wireless Communications and Networking Conference (WCNC), Nanjing, China, 29 March–1 April 2021. [CrossRef]
8. Smith, T.G. pmdarima: ARIMA Estimators for Python. 2017. Available online: http://www.alkaline-ml.com/pmdarima (accessed on 13 December 2023).
9. Bello-Angulo, D.; Mantilla-Duarte, C.; Montes-Paez, E.; Guerrero-Martin, C. Box–Jenkins Methodology Application to Improve Crude Oil Production Forecasting: Case Study in a Colombian Field. *Arab. J. Sci. Eng.* **2022**, *47*, 11269–11278. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.