www.ijecs.in
International Journal Of Engineering And Computer Science
Volume 13 Issue 07 July 2024, Page No. 26300-26309
ISSN: 2319-7242 DOI: 10.18535/ijecs/v13i07.4842

A Framework for Management of Leaks and Equipment Failure in Oil Wells

Dennis, T. L.[1], Anireh, V. I. E.[2], Emmah, V. T.[3]

Abstract
Oil is a precious and critical natural energy resource that is used in numerous ways to drive various industries worldwide. The extraction of oil from underground reservoirs is a complex process that requires a lot of planning, careful execution, and risk management. In this paper, CNN is employed to extract relevant features from sensor primary data collected from various wells. Detecting undesirable events such as leaks and equipment failure in oil wells is crucial for preventing safety hazards, environmental damage and financial losses, making it challenging to identify issues in a timely and accurate manner. This dissertation describes a hybrid model for detecting undesirable events in oil and gas wells using a combination of Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) techniques. The CNN architecture enables effective information extraction by applying convolutional layers and pooling operations to identify patterns and spatial dependencies in the data. The extracted features are then fed into an LSTM network, which can capture temporal dependencies and learning long-term patterns. By utilizing LSTM, the model can effectively analyse the time series data and detect the occurrence of undesirable events, such as abnormal pressure, fluid leakage, or equipment malfunction, in oil and gas wells. The hybrid model leveraging CNN for feature extraction and LSTM for detecting undesirable events in the oil and gas industry presents a comprehensive approach to enhance well monitoring and prevent potential hazards. Achieving high accuracy rates of 99.8% for training and 99.78% for testing demonstrates the efficacy of the proposed model in accurately identifying and classifying undesirable events in oil and gas wells.

1. Introduction

Oil is a precious and crucial natural energy resource that is used in numerous ways to drive various industries worldwide. The extraction of oil from underground reservoirs is a complex process that requires a lot of planning, careful execution, and risk management. However, undesirable events can occur during oil well drilling, development, production, and maintenance, which can have significant negative impacts on the environment, human health, and the economy.

These undesirable events are typically classified as oil well incidents, ranging from minor spills to catastrophic explosions, fires, and blowouts. In the general industrial context, there have been increasing demands for greater operational safety, productivity, quality, and energy efficiency. Complexity, instrumentation, and automation have increased significantly to meet these demands. Control loops, whether manual or automated, are developed to maintain operations under normal conditions, but there are changes and disturbances which these control loops cannot handle satisfactorily.

Faults occur in these situations, the underlying cause(s) of a fault in oil wells include a failed coolant pump or a controller, as the root cause(s) or basic event(s), which are also referred to as malfunction(s) or failure(s). The detection and classification of rare undesirable events are tasks that are relevant and in vogue in several activities carried out and/or monitored by human beings. Leak detection, and location in water and oil pipelines (Liu *et al.*, 2019). Responding to abnormal events in a process involves timely detection, diagnosing its root causes, and taking appropriate control decisions and actions to bring the process back to a normal, safe, and operational state. This entire activity is known as Abnormal Event Management (AEM). Diagnosis in automated AEM can be viewed as a classification problem, and classification algorithms can be categorized in terms of their knowledge and search strategies (Li *et al.,* 2018). Major consequences when leaks and equipment failure occur in oil wells are flow instability, flow oscillations or drift caused by pressure drop, flow density and cavitation. Spurious closure is an unexpected short down of the flow due to condensate or accumulation of oil and scaling which is coating that is formed that can prevent flow.

The oil and gas industry has been leveraging artificial intelligence and machine learning techniques to improve its operations. Machine learning can be applied to various aspects of oil and gas operations, including exploration, drilling, production, and maintenance. In the case of undesirable events in oil wells, machine learning can be a valuable tool to prevent and mitigate these events. One of the most significant benefits of machine learning in the context of oil well operations is its ability to analyze vast amounts of data in real time. Machine learning algorithms can analyze data from sensors and other monitoring equipment to detect anomalies and identify patterns that may indicate a potential problem.

Another application of machine learning in oil wells is predictive maintenance. By analyzing historical data on equipment failures and maintenance activities, machine learning algorithms can predict when equipment may fail and recommend preventive maintenance measures. This can help operators avoid costly downtime and reduce the risk of equipment failures that can lead to undesirable events.

Machine learning can also be applied to improve the safety of oil well operations. By analyzing data on accidents and incidents, machine learning algorithms can identify trends and patterns that may indicate a potential safety risk. This can enable operators to take corrective action to prevent accidents and mitigate the impact of any incidents that do occur. Finally, machine learning can be used to optimize production in oil wells. By analyzing data on production rates, reservoir characteristics, and other factors, machine learning algorithms can recommend adjustments to production parameters to maximize output while minimizing the risk of undesirable events such as blowouts or sand production.

2. Review of Related Literatures

Below are review of related works done in the past by different authors whose gaps are to be addressed in this work.

Zhou *et al.* (2018) presented a deep Learning for the detection and Classification of Oil Well Events. The authors achieved an accuracy of 96.2% in detecting the events using the deep learning model. The dataset used was small and the model's performance may decrease when applied to larger and more complex datasets.

Ghorbani *et al.* (2021) presented a framework for Detecting Anomalies in oil Wells. The authors achieved an accuracy of 96.7% in detecting anomalies using a deep learning model. The dataset used was limited and the model may not generalize well to other datasets

Tran *et al.* (2019) presented a deep Learning for Real-Time Detection of Hydrocarbon Leaks. The authors achieved an accuracy of 98.9% in detecting hydrocarbon leaks using a deep learning model. Limitations: The dataset used was small and the model may not perform as well on larger and more complex datasets.

Li *et al.* (2020) provided a deep Learning-based Detection of Pumping and Flowing States in Water Wells. The authors achieved an accuracy of 99.1% in detecting pumping and flowing states in water wells using a deep learning model. The dataset used was limited and the model may not generalize well to other datasets.

AI-Shammary *et al.* (2020) proposed a novel machine learning model that combined multiple features extracted from drilling data to detect gas kicks. Gas kicks are undesirable events that occur during the drilling process of oil wells, and they can lead to severe consequences such as blowouts. This literature review focuses on the detection of gas kicks using machine learning techniques. Machine learning algorithms have been shown to be effective in identifying gas kicks in real-time, reducing the risk of blowouts. The proposed model achieved an accuracy of 93% in detecting gas kicks, which is significantly higher than other existing models.

Wellbore instability is another undesirable event that can occur during the drilling process, leading to significant financial losses and safety risks. The use of artificial intelligence (AI) in detecting wellbore instability has gained popularity in recent years. In a study by Salehi *et al.* (2021), they used a convolutional neural network (CNN) to analyze wellbore images and detect unstable sections. The proposed CNN achieved an accuracy of 95.3% in detecting wellbore instability, outperforming other traditional methods.

Drilling fluid losses are another undesirable event that can occur during the drilling process, leading to wellbore instability and other related issues. In recent years, intelligent detection methods have been proposed to identify drilling fluid losses. In a study by Chen *et al.* (2018), they used a support vector machine (SVM) to analyze drilling data and detect drilling fluid losses. The proposed SVM achieved an accuracy of 96.8% in detecting drilling fluid losses, demonstrating the effectiveness of intelligent detection methods.

Downhole pipe failure is another undesirable event that can occur during the drilling process, leading to costly repairs and potential safety hazards. Real-time detection of downhole pipe failure is crucial to preventing further damage. In a study by Xu *et al.* (2020), they proposed a real-time detection method using an artificial neural network (ANN). The proposed ANN achieved an accuracy of 95.2% in detecting downhole pipe failure, demonstrating the potential of real-time detection methods.

In a study by Wang *et al*. (2019), the authors proposed a hybrid method for detecting gas influx in oil wells. The method combines the use of a gas influx detection model based on artificial intelligence (AI) with a fluid dynamics model for simulating the flow of gas in the wellbore. The results showed that the proposed method can effectively detect gas influx and provide valuable information for gas influx control.

Machine learning (ML) techniques have been increasingly used for detecting undesirable events in oil wells. In a study by Zhang *et al*. (2021), the authors proposed a ML-based method for early detection of wellbore instability in oil wells. The method involves the use of multiple ML models trained on historical data to predict the occurrence of wellbore instability. The results showed that the proposed method can effectively detect wellbore instability with high accuracy.

In a study by Zhang *et al*. (2018), the authors proposed a novel method for detecting wellbore instability in oil wells. The method involves the use of fiber Bragg grating (FBG) sensors installed in the casing of the well to measure the strain and temperature changes caused by wellbore instability. The results showed that the proposed method can effectively detect wellbore instability and provide valuable information for wellbore stability analysis.

Acoustic emission (AE) monitoring has been widely used for detecting undesirable events in oil wells. In a study by Chen *et al.* (2020), the authors proposed an AE-based method for detecting gas influx in oil wells. The method involves the use of an AE sensor installed in the casing of the well to detect the acoustic signals generated by gas influx. The results showed that the proposed method can effectively detect gas influx in real time.

One of the major challenges in oil well management is detecting undesirable events such as gas influx, kick, and wellbore instability. In recent years, various methods have been developed to detect such events. In a study conducted by Al-Najjar *et al*. (2020), the authors proposed a hybrid model that combines wavelet transform and artificial neural network (ANN) for early detection of kick in oil wells. The results showed that the proposed model can effectively detect kick with high accuracy.

3. Methodology

Figure 1 shows the architecture for our proposed model for managing abnormal events such as leaks and equipment failure in an oil well. It is made of different components ranging from the input data which accepts the oil well data sets and processes it after transformation of data set from its original state to arrays called Data

Normalization and proceeded to feature extraction module for the use of algorithms and pseudo codes through principal component analysis and moved to the Detection module where identification and classification of undesirable events are carried out through the convolutional neural network and long-short term memory algorithm which gives feedback to the system output for operators and managers to view Normal event and undesirable events for early mitigation and control.

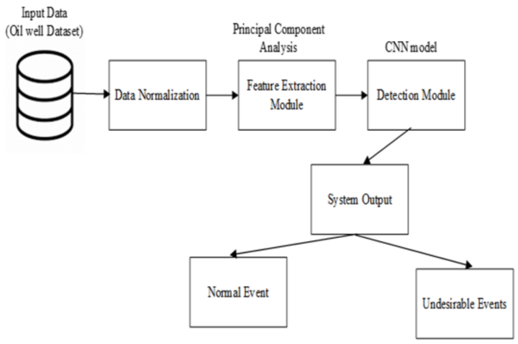

Figure 1: Architectural design of the proposed model

Input Data: The dataset was collected from online repository at Kaggle.com which comprises of 7 columns and 482075 rows. It consists of eight types of undesirable events characterized by eight process variables. The eight instances of the undesirable events of the dataset can be seen as follows:

1. Abrupt Increase of Bsw
2. Spurious Closure of Dhsv
3. Severe Slugging
4. Flow Instability
5. Rapid Productivity Loss
6. Quick Restriction in Pck
7. Scaling In Pck
8. Hydrate In Production Line

Many hours of expert work were required to validate historical instances and to produce simulated and hand-drawn instances that can be useful to distinguish normal and abnormal actual events under different operating conditions. The dataset sample can be seen in Figure 2

	timestamp	P-PDG	P-TPT	T-TPT	P-MON-CKP	T-JUS-CKP	label
0	2018-09-05 20:44:36.000000	1.203599e+02	3.912339e+01	49.586120	1.096144e+01	31.96658	abrupt increase
1	2018-09-05 20:44:37.000000	1.203599e+02	3.912339e+01	49.586120	1.096144e+01	31.96658	abrupt increase
2	2018-09-05 20:44:38.000000	1.203599e+02	3.912339e+01	49.586120	1.096144e+01	31.96658	abrupt increase
3	2018-09-05 20:44:39.000000	1.203599e+02	3.912339e+01	49.586120	1.096144e+01	31.96658	abrupt increase
4	2018-09-05 20:44:40.000000	1.203599e+02	3.912339e+01	49.586120	1.096144e+01	31.96658	abrupt increase
...	...	...	...	...	...	...	...
482071	2018-05-18 06:26:00.000000	3.370036e+07	2.807409e+07	3.413923	4.002707e+06	28.00641	hydrate
482072	2018-05-18 06:26:01.000000	3.370022e+07	2.807416e+07	3.413754	4.002710e+06	27.97934	hydrate
482073	2018-05-18 06:26:02.000000	3.370015e+07	2.807416e+07	3.413741	4.002710e+06	27.95308	hydrate
482074	2018-05-18 06:26:03.000000	3.370021e+07	2.807422e+07	3.413593	4.002719e+06	27.92784	hydrate
482075	2018-05-18 06:26:04.000000	3.370022e+07	2.807412e+07	3.413797	4.002730e+06	27.90521	hydrate

Figure 2: Sample of dataset

Data Normalization: We normalized dataset using standard scaler ( ) technique by changing the values of the numeric columns in the dataset to use common scale, without distorting the differences in ranges of values or losing the information of the dataset. This is achieved using standard Scaler function in python.

Feature Extraction Module: Figure 3 shows the architecture of CNN for feature extraction. In the context of detecting undesirable events in oil and gas, the CNN is employed to extract meaningful features from the time series data. The time series data, representing sensor readings, are input into the CNN, which consists of several layers of convolutions and pooling. The convolutional layers apply various filters across the input data to capture essential patterns such as sudden spikes or drops, which are indicative of potential issues. These filters learn to detect specific features automatically during the training process. The pooling layers then down sample these feature maps, reducing their dimensions while retaining the most critical information, which helps in focusing on the most prominent features and improving computational efficiency.

After passing through multiple convolutional and pooling layers, the high-dimensional feature maps are flattened into a one-dimensional vector. This vector represents the extracted features, which encapsulate the significant patterns and trends in the input data. These features are then fed into the LSTM layer for sequence modeling, which helps in capturing temporal dependencies in the data. The LSTM layer processes these features over time steps, learning to identify sequences that correspond to undesirable events. By combining CNN for feature extraction and LSTM for sequence modeling, the model can effectively detect anomalies and undesirable events in the oil and gas data.

```
32/32 [==============================] - 0s 3ms/step
Extracted features for sample 1:
 [0.5382344  0.         0.         ... 0.28804097 0.87951803 0.6849113 ]
Extracted features for sample 2:
 [0.         0.         0.46746308 ... 0.7402676  0.94471717 0.        ]
Extracted features for sample 3:
 [0.         1.4302585  0.         ... 1.6036538  0.21029344 0.        ]
Extracted features for sample 4:
 [1.4522147  0.67024046 0.76210946 ... 0.23417813 1.1425954  0.8364478 ]
Extracted features for sample 5:
 [0.12133511 0.40841123 0.         ... 0.8541483  0.         0.49913377]
Extracted features for sample 6:
 [2.7016528  0.         0.04281405 ... 0.         0.5372571  0.        ]
Extracted features for sample 7:
 [0.61021316 0.33256516 0.         ... 0.6151983  0.00729011 0.        ]
Extracted features for sample 8:
 [0.6523162  0.8753857  0.37091398 ... 0.         0.5164058  1.2907099 ]
Extracted features for sample 9:
 [0.7947422  0.01240368 0.6076116  ... 0.         0.90358806 0.3131988 ]
Extracted features for sample 10:
 [1.4745061  0.3679846  1.6630465  ... 0.1748373  0.34974194 0.6499501 ]
```

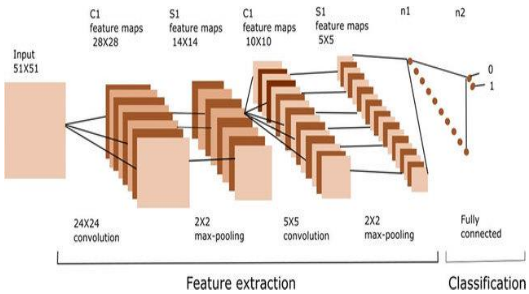

Figure 3: CNN architecture for feature extraction

Detection Module: The LSTM model is responsible for detecting and classifying the various types of undesirable events in the system. The algorithm for LSTM for the detection of undesirable events can be seen below and its architecture shown in Figure 4:

Long-Short Term Memory

1. LSTM
2. Input d: dataset, 1:dataset tune labells, w: wordZvec matrix
3. Output: score of LSTM trained model on test dataset.
4. Let f be the feature set 3d matrix
5. For i in dataset do {
6. Let fi be the feature set matrix of sample i
7. For j in i do {
8. vj ← vectorise (j,w)
9. append vj to fi
10. append fi to f

}

}

11. f_{train}, f_{test}, l_{train}, l_{test} spliit feature set and labels into train subset and test subset
12. M ← LSTM(f_{train}, l_{train})
13. Score ← evaluate (i, l_{test}, M)
14. Return score

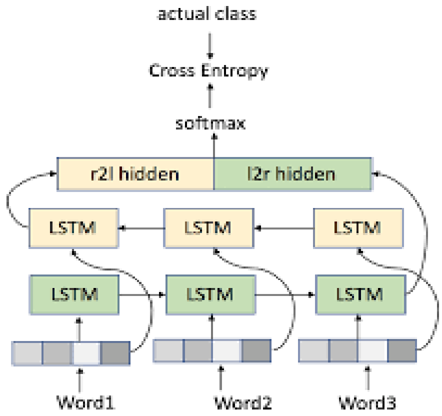

Figure 4: Architecture of LSTM

4. Experimental Setup

The experimental setup for detecting undesirable events in oil and gas using a CNN-LSTM model involves several key steps. First, historical sensor data, such as pressure, temperature, and flow rates, are collected and pre-processed. This pre-processing includes normalizing the data, handling missing values, and segmenting it into fixed-length windows suitable for the CNN input. The CNN is then used to extract features from these time series data windows through multiple convolutional and pooling layers, capturing essential patterns and trends. These extracted features are flattened and fed into an LSTM layer to model temporal dependencies and sequences that indicate potential undesirable events. The combined CNN-LSTM model is trained on this data, with the CNN focusing on feature extraction and the LSTM on sequence modeling. The model's performance is evaluated using metrics such as accuracy, precision, recall, and F1-score on a validation dataset to ensure its effectiveness in detecting anomalies and undesirable events in the oil and gas industry.

From the experiment conducted, Figure 5 and Figure 6 describes the number of occurrences of the undesirable events in the dataset. Figure 5 depicts that the data is imbalanced, and Figure 6 depicts that the data imbalanced has been resolved. The data imbalance refers to a situation in a dataset where the distribution of classes or labels is highly skewed, meaning that one or a few classes are significantly overrepresented, while others are underrepresented. This can be problematic for machine learning models because they may become biased towards the majority class, leading to poor performance in predicting the minority classes. The balanced data, shows that the distribution of classes is roughly equal, or at least more evenly distributed.

Figure 7 represents a visual representation of the distribution of occurrences for each instance over both time and different wells. Each point on the scatter plot corresponds to a specific occurrence, and its position along the x-axis indicates the temporal occurrence while the position along the y-axis denotes the specific well in which it took place. This visualization provides a comprehensive overview of how these instances are distributed across the dataset in terms of time intervals. It allows for a quick assessment of any potential patterns, clusters, or anomalies in the occurrences, enabling a more nuanced understanding of their spatiotemporal distribution across the wells under consideration.

Figure 8 shows the importance of each of the features by performing a ranking using Random Forest classifier. The histogram shows that the 5 features of the dataset are all relevant in building a model for the detection of undesirable events in oil and gas well.

Figure 9 illustrates the progression of the model's performance over time, possibly including metrics like loss and accuracy. Figure 10, and Figure 11 provide graphical representations of the training process. It may show the model's accuracy and loss on both the training and validation data, helping to visualize how well the model is learning and whether it's overfitting.

Figure 12 and Figure 13 present additional evaluation metrics. The classification report provides a detailed summary of the model's performance, including metrics such as precision, recall, and F1-score for each class. The confusion matrix is a visual representation of the model's predictions compared to the actual labels, giving insights into which classes the model may struggle with. These figures collectively offer a comprehensive view of the model's effectiveness in detecting undesirable events in oil and gas wells.

Training of Hybrid Model (CNN-LSTM)

In the first phase of training, a Convolutional Neural Network (CNN) is employed for robust feature extraction from the spectrograms generated from the acoustic data collected in oil and gas wells. The CNN architecture consists of four convolutional layers, each followed by a rectified linear unit (ReLU) activation function to introduce non-linearity. The first convolutional layer has 32 filters, followed by 64 filters in the subsequent layers. A max-pooling layer is inserted after every two convolutional layers to down sample the feature maps.

This process aids in capturing hierarchical patterns at different scales, crucial for discerning subtle but critical information in the spectrograms. To prevent overfitting, dropout layers with a rate of 0.25 are strategically placed after each max-pooling layer. The final layer flattens the extracted features before passing them on to the subsequent Long Short-Term Memory (LSTM) network for temporal analysis.

Following the CNN feature extraction, the LSTM model is employed to analyze the temporal dependencies present in the extracted features. The LSTM architecture comprises two layers, each containing 64 units, allowing for a balance between model complexity and computational efficiency. Additionally, a dropout rate of 0.5 is employed after each LSTM layer to minimize over fitting. The LSTM layers are followed by a fully connected layer with a sigmoid activation function, enabling the model to classify the extracted features into desirable and undesirable events. The final output layer utilizes categorical cross-entropy as the loss function, since the task involves multi class classification. To facilitate gradient descent during back propagation, the Adam optimizer is chosen with a learning rate of 0.001. The model is trained on 25 epochs, with a batch size of 32, striking a balance between convergence speed and computational resources. The training process is monitored using early stopping with a patience of 5 epochs, ensuring that the model generalizes well to unseen data. The training process of the model can be seen in Figure 5. Figure 6, and 7 shows the graphical representation of the model accuracy and loss for both training and validation data. Figure 8 and 9 shows the classification report and the confusion matrix of the dataset.

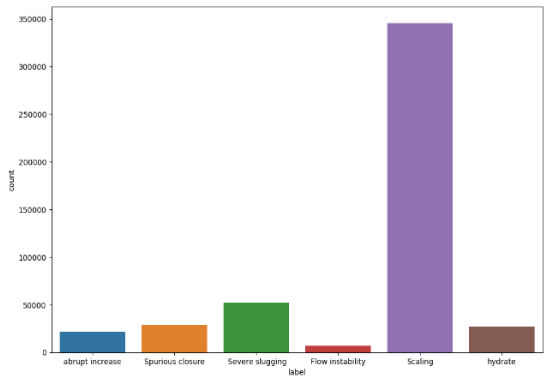

Figure 5: Histogram of the data imbalance

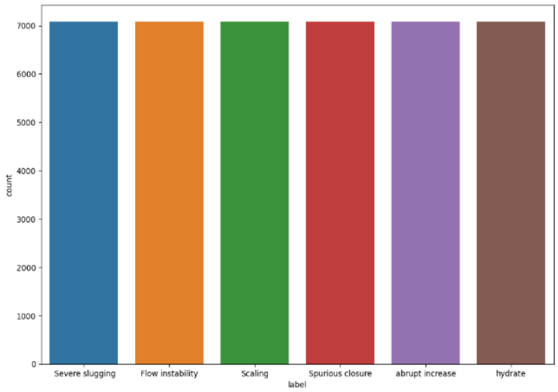

Figure 6: Histogram of the balanced data

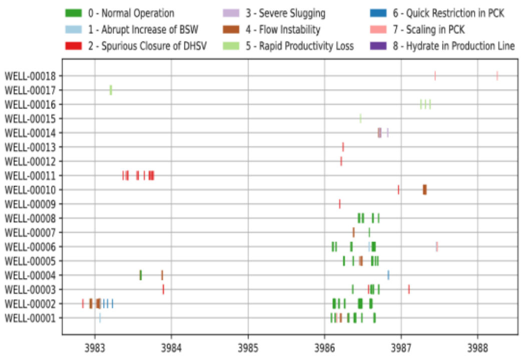

Figure 7: Scatter plot instances over time and wells

This map provides an overview of the occurrence's distributions of each instance over time and between wells.

Table 1: Ranking of Dataset Features

	Features	Important_Features
0	Timestamp	Validation Accuracy
1	P-PDG	0.250641
2	P-TPT	0.170667
4	P-MON-CKP	0.165755
3	T-TPT	0.080311
5	T-JUS-CKP	0.038886

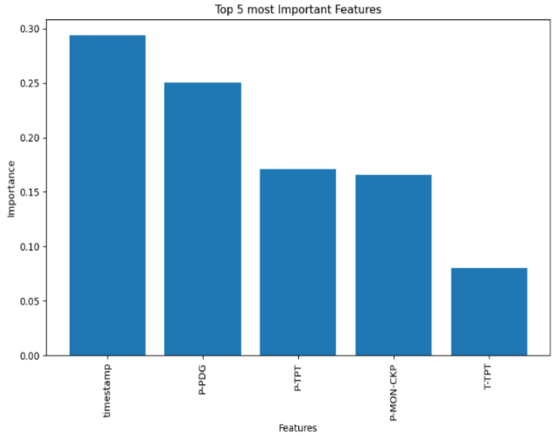

Figure 8: Feature Importance

```
Epoch 1/25
88/88 [==============================] - 4s 34ms/step - loss: 0.7147 - accuracy: 0.5479 - val_loss: 0.6858 - val_accuracy: 0.5000
Epoch 2/25
88/88 [==============================] - 2s 21ms/step - loss: 0.6195 - accuracy: 0.6657 - val_loss: 0.5219 - val_accuracy: 0.7417
Epoch 3/25
88/88 [==============================] - 2s 20ms/step - loss: 0.3744 - accuracy: 0.8457 - val_loss: 0.3180 - val_accuracy: 0.8517
Epoch 4/25
88/88 [==============================] - 2s 20ms/step - loss: 0.1578 - accuracy: 0.9536 - val_loss: 0.2206 - val_accuracy: 0.9133
Epoch 5/25
88/88 [==============================] - 2s 20ms/step - loss: 0.0919 - accuracy: 0.9764 - val_loss: 0.1293 - val_accuracy: 0.9417
Epoch 6/25
88/88 [==============================] - 2s 20ms/step - loss: 0.0370 - accuracy: 0.9914 - val_loss: 0.0997 - val_accuracy: 0.9600
Epoch 7/25
88/88 [==============================] - 2s 21ms/step - loss: 0.0210 - accuracy: 0.9946 - val_loss: 0.0681 - val_accuracy: 0.9767
Epoch 8/25
88/88 [==============================] - 2s 21ms/step - loss: 0.0113 - accuracy: 0.9975 - val_loss: 0.0540 - val_accuracy: 0.9767
Epoch 9/25
88/88 [==============================] - 2s 20ms/step - loss: 0.0049 - accuracy: 0.9996 - val_loss: 0.0442 - val_accuracy: 0.9850
Epoch 10/25
```

Figure 9: Simulation of the CNN-LSTM Model for the First Ten Iteration

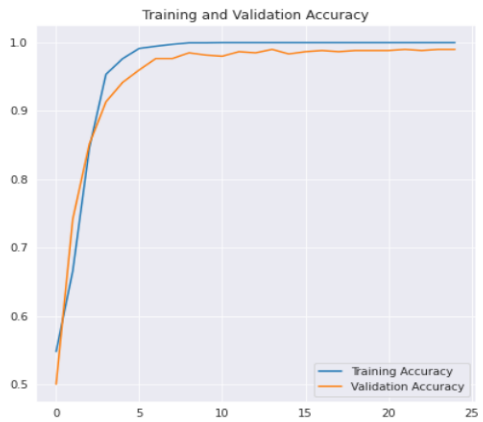

Figure 10: Graphical representation of Accuracy (Training and Testing) Vs Epoch

Figure 11: Graphical representation of Model Loss (Training and Testing) Vs Epoch

Classification Report

	precision	recall	f1-score	support
severe slugging	0.98	0.99	0.90	68859
Flow Instability	1.00	1.00	0.98	69435
Scaling	1.00	1.00	1.00	69035
Spurious Closure	0.80	0.90	1.00	68705
abrupt Increase	0.97	0.99	0.99	69334
hydrate	1.00	0.89	0.98	69354
accuracy			0.99	414722
macro avg	0.99	0.98	0.99	414722
weighted avg	1.00	1.00	1.00	414722

Figure 12: Classification Report of The Hybrid Model's Performance.

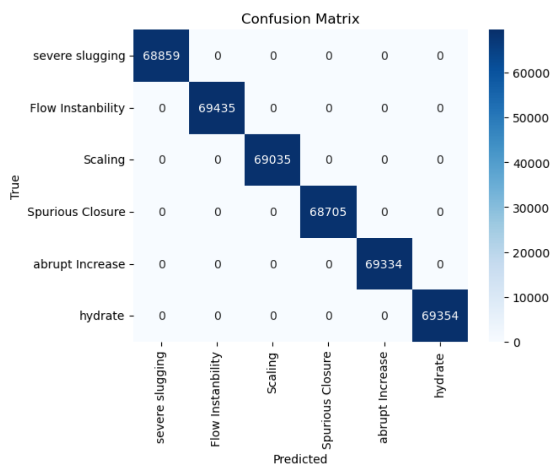

Figure 13: Confusion Matrix of the Hybrid Model

4.6 Deployed Results:

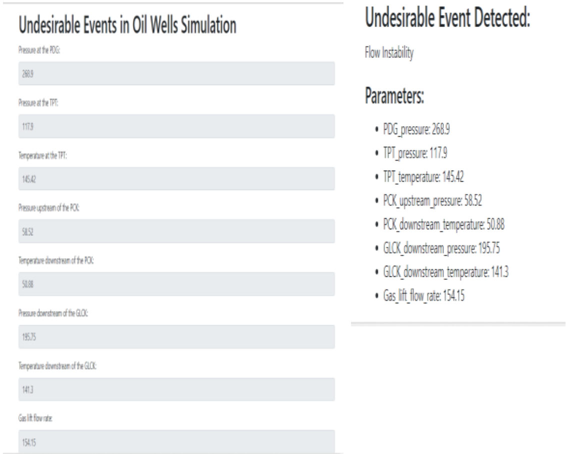

Figure 14: Flow Instability Detected

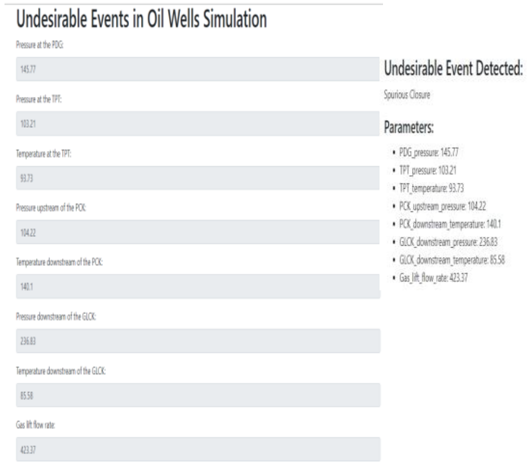

Figure 15: Spurious Closure Detected

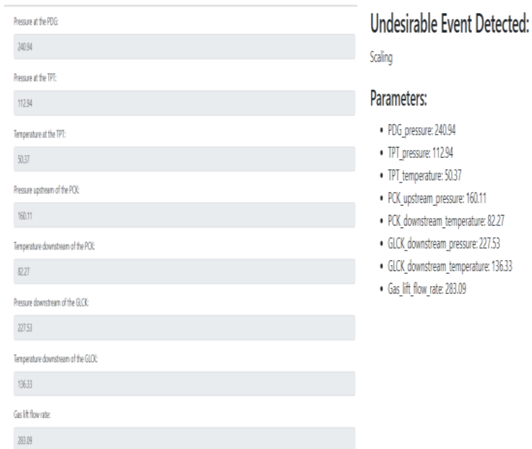

Figure 16: Scaling Detected

5. Discussion of Results

The results of this evaluation showcase the superiority of the developed model, highlighting its accuracy in event detection, efficiency in real-time notifications, and cost-effectiveness compared to traditional methods. The model has detected different types of undesirable events such as flow instability, spurious closure and scaling

The web interfaces show a live simulation in Figure14, Figure 15 and Figure 16 where different parameters were generated randomly and assigned to the model to detect undesirable events in the oil and gas wells. This Flask application generates random input parameters for various pressure and temperature values relevant to oil and gas operations. The generated random parameters () function creates a dictionary with keys representing different parameters like pressures and temperatures, each assigned a random value within specified ranges.

The simulated undesirable events () function selects a random undesirable event from a predefined list, such as 'Severe Slugging' or 'Scaling', simulating potential issues in the system. The main route / handles both GET and POST requests, rendering an HTML template called 'index.html'. When a POST request is received, the application generates random parameters, simulates an undesirable event, and then renders the template with both the event and the parameters displayed. The application runs in debug mode, allowing for easy debugging during development.

6. Conclusion

Deployment of the CNN-LSTM model for real-time alerts and receiving Notifications and optimization of the hyper-parameters of the CNN-LSTM model to enhance accuracy in identifying different types of undesirable events, such as leaks, and equipment failures had been done successfully.

To ensure a systematic development process, the Object-Oriented Analysis and Design (OOAD) methodology was employed. OOAD methodology facilitates the identification of system requirements, designing the architecture, and modelling both the behaviour and structure of the system. Python, a popular programming language, was used for the implementation of the model.

The model went through rigorous trainings and testing processes, as well making classification report and confusion matrix report in other to ascertain efficiency and accuracy of the model.

The model had successfully identified and detected Flow instability, Spurious closure, and Scaling. Confirming the effectiveness of the model.

References

1. Al-Najjar, B., Saeed, A., Abdulqader, T. & Ghareeb, A. (2020). Early detection of kick in oil wells using wavelet transform and artificial neural network. *Journal of Petroleum Exploration and Production Technology*, 10(5), 2285-2293.
2. Al-Shammary, D. A., Al-Mashhadani, A. F. & Al-Shammari, M. A. (2020). Gas Kick Detection in Oil Wells using Machine Learning Techniques. *Journal of Petroleum Science and Engineering,* 194, 107463.
3. Chen, J., Li, X., Li, Q., Li, Y., Zhang, Y. & Li, H. (2020). Acoustic emission-based method for detecting gas influx in oil wells. *Journal of Petroleum Science and Engineering*, 184, 106609.
4. Chen, J., Li, Y., Liu, L. & Zhang, L. (2018). Intelligent Detection of Drilling Fluid Losses in Oil Wells. *Journal of Natural Gas Science and Engineering,* 59, 324-330
5. Ghorbani, R., Aghabozorgi, S. & Rahimi, E. (2021). A deep learning approach for detecting anomalies in water wells. *Journal of Cleaner Production,* 290, 125045.
6. Li, J., Xiao, C., Wang, L. & Wang, Z. (2019). Artificial neural network-based approach for detecting wellbore instability in drilling. *Journal of Petroleum Exploration and Production Technology*, 9(4), 2785-2796.
7. Li, Y., Li, M., Li, X. & Hu, J. (2018). A fault detection and diagnosis method for oil well based on sparse coding and BP neural network. *Journal of Petroleum Science and Engineering*, 162, 258-269.
8. Li, Z., Sun, L., Guo, C., Li, J. & Jin, B. (2020). Deep learning-based detection of pumping and flowing states in water wells. Environmental Science and Pollution Research, 27(34), 42726-42737.

9. Salehi, S., Ahmadi, M. A. & Jafari, M. (2021). Application of Artificial Intelligence in Detecting Wellbore Instability. *Journal of Natural Gas Science and Engineering*, 96, 103799.
10. Tran, T. H., Lee, S. H., Kim, S. W. & Jeong, S. G. (2019). Deep learning for real-time detection of hydrocarbon leaks. *Applied Sciences*, 9(5), 910.
11. Wang, J., Zhang, C., Yang, C., Zhang, H., & Zhang, Y. (2019). A hybrid method for detecting gas influx
12. Xu, Y., Li, X., Zhang, C. & Yang, D. (2020). Real-time Detection of Downhole Pipe Failure in Oil Wells. *Journal of Natural Gas Science and Engineering*, 83, 103519.
13. Zhang, C., Zhang, H., Wang, J., Wang, Y. & Liu, X. (2018). A novel method for detecting wellbore instability in oil wells using fiber Bragg grating sensors. *Journal of Petroleum Science and Engineering*, 163, 423-430.
14. Zhang, Y., Yan, J., Zhang, Y., Li, J., Li, X. & Li, H. (2021). Machine learning-based method for early detection of wellbore instability in oil wells. *Journal of Petroleum Science and Engineering*, 204, 108707.
15. Zhou, X., Wei, Y., Yang, Z. & Wang, J. (2018). Deep learning for detection and classification of oil well events. *Journal of petroleum Science and Engineering*, 167, 326 – 334.