

Applying deep learning method to develop a fracture modeling for a fractured carbonate reservoir using geologic, seismic and petrophysical data

Fatemeh Heydarpour ^a, Abbas Bahroudi ^{a,*}

^a School of Mining Engineering, College of Engineering, University of Tehran, Tehran, Iran.

Article History:

Received: 26 April 2023.

Revised: 19 June 2023.

Accepted: 15 July 2023.

ABSTRACT

Fractures are one of the most important geological features that affect production from most carbonate reservoirs. A large amount of the world's hydrocarbon resources are located in fractured reservoirs and the identification of fractures is one of the important steps in reservoir development. Due to the high cost of tools that are used in the petroleum industry to identify fractures such as image logs, and their inaccessibility in most of the studied areas, it is often tried to use other available data to identify fractures. Due to the ever-increasing progress of data-driven methods such as neural networks and machine learning, this study has tried to apply 1D-Convolutional Neural Network (1D-CNN) which is one of the deep learning algorithms on well-logging data and seismic attributes in a carbonate reservoir to identify the existing fractures in the investigating area. The approach used in this research is a binary classification which is applied first in the well location. To validate the method, results are compared with the reports obtained from image logs. Finally, the fracture density map is drawn in the entire reservoir area.

Keywords: Carbonate reservoirs, Deep learning, Fracture detection, Machine learning, Seismic attributes, Well logging data.

1. Introduction

The term 'fracture' refers to any secondary physical break or discontinuity in the rock caused by stresses exceeding the rock resistance threshold [1, 2]. Identification of fractures and their density in different areas of the reservoir, especially fractured reservoirs, is one of the most important stages of reservoir studies [3, 4].

The results of fracture modeling are used as input data for flow simulation and analysis. Therefore, it is very important to study the network of fractures to know how they spread in fractured reservoirs and create a model for it [4, 5]. The main information used for identifying fractures are seismic data, well logging data, well test data, mud loss information, and analysis of cores taken from reservoir horizons.

In the middle of 1980 with the advent of image logs the process of identifying fractures and their characteristics such as slope, aperture, and fracture density improved significantly [5,6]. Because geological phenomena have various patterns and the presence of anomalies in the collected data increases its complexity, human-based methods are very complicated in using the above-mentioned data to identify fracture zones and characteristics, [7]. Also, direct tools for identifying fractures like the image logs and coring have limitations such as not always being available, time-consuming, and economic inefficiency [8]. Therefore, to analyze subsurface data to investigate fractures in the reservoirs it's important to apply intelligent methods that can perform complex calculations and provide comprehensive algorithms.

Artificial intelligence is a field of science that can produce an intelligent machine to be able to perform tasks that require human intelligence [9]. Today, artificial intelligence systems are programmed with the help of machine learning and deep learning, to be trained and

used for the intended purposes [10]. In recent years researchers have used artificial methods to identify faults and fractures in some reservoirs. Ozkaya et al. (2008), used a probabilistic decision tree to detect fracture corridors in a mature oil field. In their research, the images log, production logs and injector/producer shortcuts were selected as the training set [11].

Tokhmechi et al. (2009), used a combined method to detect fractures. They used all available petro physical logs for training the Parzen algorithm, they also used wavelet transform to preprocess the input logs before training the algorithm [12]. Jafari et al. (2012), suggested a model to estimate fracture density from conventional well logs using an Adaptive-Nero fuzzy inference system. They used image logs from two wells to verify the results of the model [13]. Asghari Nejad et al. (2014), have used the Parzen-wavelet approach to detect the vuggy zone in a carbonate reservoir. The combined Parzen-wavelet-based algorithm was developed for identifying vuggy zones in the wavelet coefficient domain using gamma ray (GR), neutron porosity (NPHI), bulk density (RHOB), and sonic (DT) logs. Compatibility between core tests and the results of the developed method revealed the capability of the method [14]. Zheng et al. (2014), applied an artificial neural network in seismic object detection successfully by combining multiple attributes into a single object-sensitive attribute. Their final results showed that neural network-based fault detection contributes the better structural interpretation, especially in an area with a complex fault system [15].

Bayat et al. (2015), used a neural network-based method for 3D fracture modeling. They used seismic attributes, petrophysical logs, and reservoir data to train the artificial network. The fracture model obtained from the neural network agreed with the transmissibility map

* Corresponding author: Tel: +98 2141868218, E-mail address: bahroudi@ut.ac.ir (A. Bahroudi).

by about 82% [16]. Cruz et al. (2017), applied a deep-learning method to improve automatic fracture detection in borehole images [17]. Xiong et al. (2018), used supervised convolutional deep learning algorithms to identify faults in 3D seismic data [18]. Haibin et al. (2018), used a deep convolutional neural network to delineate seismic salt bodies from 3D seismic data [19]. Udegbe et al. (2019), used data science analysis to identify faults in the scale of seismic data. The combination of the pseudo-HAR algorithm, which is used for face recognition, and the cascade AdaBoost algorithm was one of the main approaches of their research [20]. Ashraf et al. (2020), used an unsupervised machine learning algorithm and ant colony optimization to recognition of fracture networks [21]. To recognize fractured zones, Tian et al. (2020), proposed an integrated workflow, which took fracture identification as an end-to-end project by combining boundary detection and deep learning classification (CPD-DL) [22].

Previous studies have been carried out to identify large-scale fractures (on the fault scale) using seismic data, machine learning, and neural network algorithms. Considering the development of algorithms and special applications that deep learning methods have, this research has tried to identify small-scale fractures (well scale) to achieve more effective results from the One-dimensional convolution neural network algorithm. To compare the result with conventional works that have been done before to identify fractures. This study helps to introduce the various advantages of using deep learning methods in the field of exploration. In this paper, we try to apply supervised deep learning methods to identify fractures in the scale of wells using well-log data and seismic data such as attributes relating to fault and fractures. The research consists of two parts. In the first section, well-logging data is used as an input to the algorithm for classifying fractured and non-fractured zones, and in the next section, fracture-related attributes are used for classification. Finally, the fracture density map obtained from the training parameters is displayed in the part of the reservoir. It should be noted that due to the lack of available data in the research area, part of the data from two wells is used for training and the rest is set aside as the test data.

2. Geological setting and data acquisition

The study area is located in the southwest of Iran within the Zagros fold-thrust belt resulting from the continental convergence of Arabian and Iranian-Turkish plates. The belt hosts about 10 percent oil and 15 percent gas of worldwide reservoirs. At the middle part of the belt, the Dezful Embayment as a tectonic re-entrant part hosts 45 oil fields and many fault-related folded structures. In the extreme north of the Dezful embayment, the study oil field is located in Ilam province, 22 km southwest of Dehloran and 180 km south of Kermanshan cities. This field is located between Cheshmeh Kush, Musian, and Changuleh fields and is hosted in a northwest-southeast trending anticline with 45 km length and 8 km wide on top of the Bangestan horizon. The oil field was discovered in 1972.

According to available geological data, the stratigraphic succession in the study area is normal from surface to depth. Agahjari formation is the youngest sedimentary sequence and is exposed at the surface. It consists of sandstone, cherty conglomerate, medium grain, reddish brown with thick layers of red silty marl, gypsiferous marl, and thin beds of limy sandstone. The Aghajari formation overlies the Gachsaran formation which consists mainly of grey-red marl, anhydrite in alternation with limestone, and salt beds. Gachsaran formation is the most important seal for hydrocarbon fields in the Zagros basin. Asmari formation, as the most important reservoir unit with the Zagros basin, is overlain by the Gachsaran formation and consists of limestone, cream-light brown mudstone, dolomitic limestone, partly shale, and silt. To the depth, the Pabdeh and Gurpi formation consists of dark grey argillaceous limestone, and greenish grey marl, overlying the alternation of yellowish-brown light grey to brownish argillaceous limestone and greenish grey Marl of the Ilam formation. These units overly on the Sarvak formation as the second most important reservoir unit in the Zagros basin. This formation is one of the geological formations of the Bangestan Group in

Zagros with the Middle Cretaceous (Albian-Tronian) age and is an alternation of off-white to light cream limestone to dolomitic limestone and some green to pale green pyritic shale beds.

In this study, geophysical data of two wells from the study oil field were used to model the fractured reservoir at the level of the Sarvak formation. These wells with an average depth of 4394 (-4170.7mc) and 4518 (-3831.2mc) meters were drilled in 1972 and 2003. The wells are located near the crest of the anticlinal structure and were drilled with a 5.875-inch bit across the Sarvak formation. The deviation of the well in the logged interval reached a maximum of 2 degrees. The well was drilled using oil-based mud. By checking the geological information of the area and the image logs reports, the fractures are observed in the Sarvak formation. In addition to open-hole logs, an Ultrasonic Borehole Imager (UBI*) was logged over the interval from 3938m to 4385m on 1st January 2007. The main objectives of logging the UBI* were to characterize natural fractures and to analyze the borehole condition.

To study the fractures in the target area, image log data from two wells are available. In well No. 1, the report from the FMI image tool and in well No. 2 the report from the UBI tool were provided to study the location and type of fractured intervals. In Table 1, fracture information in two wells is summarized.

Table 1. Information on the type of fractures in different depth intervals.

| Well number | Fracture interval | Tool | Type of fracture |
|-------------|-------------------|------|----------------------|
| 1 | 3918-4105 | SFMI | Open/possible |
| 2 | 3938-4385 | UBI | Open/possible/closed |

3. Deep learning

Deep learning is a subset of machine learning algorithms and in recent years, its computational model is considered a gold standard in the field of machine learning. This computational tool has achieved outstanding results in adaptive and complex tasks performed by humans [23]. The term deep learning refers to an artificial neural network with a deep structure. Deep learning networks are a set of algorithms that provide the best performance in solving critical problems such as voice perception, image recognition, and language processing [24]. Deep learning allows computational models composed of multiple processing layers to learn representations of data with multiple levels of abstraction. This branch of learning science includes numerous algorithms that can be used in both supervised and unsupervised approaches [25]. The most important deep learning algorithms include convolutional networks, Auto-encoder networks, deep belief networks, and recurrent neural networks. Many deep architectures are growing every day and it is difficult to compare them according to their performance and application in different fields [26]. In this research, to identify fractures in the field of study it has been tried to apply the deep learning classification method, which is a type of supervised approach. For this purpose, a one-dimensional convolutional algorithm has been used. In addition, to increase the accuracy of the results and compare it with deep model, cumulative learning algorithms have been used too. In the following, these items have been described.

3.1. Convolutional neural network

The convolutional neural network algorithm (CNN) is one of the most used and famous algorithms in the field of deep learning [23]. These networks are specialized types of neural network algorithms that use convolution math operations in their layers. These networks are specifically designed to process pixel data and are used in image recognition and processing [27]. By using this algorithm, it is possible to process one-dimensional and two-dimensional data. This algorithm can automatically extract features [23]. The input x of each layer in a CNN model is organized in three dimensions: height, width, and depth, or $m \times m \times r$, where the height (m) is equal to the width. The depth is also referred to as the channel number. Several kernels (filters) available

in each convolutional layer are denoted by k and also have three dimensions ($n \times n \times q$), similar to the input image, the kernels are the basis of the local connections, which share similar parameters (bias b_k and weight W_k) for generating k feature maps h_k with a size of $(m - n - 1)$ each and are convolved with input, as mentioned above. The convolution layer calculates a dot product between its input and the weights as in formula 1, similar to NLP, but the inputs are undersized areas of the initial image size. Next, by applying the nonlinearity or an activation function to the convolution-layer output, we obtain the following [28]:

$$f_h(1)k = f(W_k * x + b_k) \quad (1)$$

In the following, each layer in the network architecture is explained:

3.2. CNN architecture

A convolutional network consists of an input layer, hidden layers, and an output layer. In a convolutional network, hidden layers perform the convolution operation. In a convolution structure, the input is a tensor that has three components: length, width, and height. After passing the convolutional layer, the image is abstracted into a feature map, called an activation map, in which the presence of features detected in the input is summarized [29]. In each convolutional layer, a kernel or filter is slid over the input data and scan it. As the convolution kernel slides along the input data matrix, it creates a feature map that in turn contributes to the input to the next layer. The next layers include: a) pooling layers: These layers, which are usually used after the convolutional layers, have the task of reducing the dimensions of the feature map obtained from the convolutional layer by sampling. Pooling can be done in two ways: maximum and averaging. In the maximum method, pooling is a predetermined window that scans the input or image and selects the highest value in each window. In the averaging method, the average values are selected [30]. b) Fully connected layers: Each neuron in one layer is connected to another neuron in a different layer. When the features are extracted from the convolutional layer and resampled by the pooling layer, they are mapped to the final outputs of the network by a subset of fully connected layers. The final fully connected layer usually has the same number of output nodes as the number of available classes and each fully connected layer is followed by a non-linear function [31].

As mentioned, this research has been done with the supervision approach. Therefore, the fractured zones obtained from image logs are the areas where the network should extract and train parameters and features by considering the input values corresponding to each depth of the fracture. In fact, the values obtained from well logs or seismic attributes are 1-dimensional inputs that are processed with Convolutional, pooling, and fully connected layers to lead to a binary output of zero and one.

4. Majority voting algorithms

Cumulative learning refers to methods of learning algorithms that use the combination of several models to predict and achieve higher accuracy. Cumulative learning consists of a limited set of alternative models and follows more flexible structures [32]. Cumulative learning is done in two ways; bagging and boosting methods. Each has been described below.

4.1. Bagging methods

In this method, several weak learning models are used homogeneously. These models learn from each other independently and concurrently. The result of each model is combined as a kind of deterministic averaging process [33]. Figure 1 shows the bagging method.

4.2. Boosting method

The boosting method is a type of cumulative learning algorithm that

consists of combining several weak classifiers to lead to a strong learning model same as the bagging algorithms [34]. In this method, learning is done sequentially, in other words, the base model depends on the previous models and combines the result to achieve a definite result [35]. Figure 2 shows the boosting method.

5. Methodology

The main goal of this research is to use deep learning algorithms for classifying fractured and non-fractured zones within the scope of two investigated wells. For this reason, after obtaining the required data and loading them into the software system, seismic and petro-physical data are processed and checked in two separate sections. First, the petro-physical data are used as the input of deep learning algorithms for classifying fractured and non-fractured zones by using the supervised model, and in the next step, seismic data, especially seismic attributes such as three-dimensional curvature, dip deviation, instantaneous frequency, relative acoustic impedance, attenuation, are used as inputs to the deep learning algorithm.

5.1. Applying 1D-CNN on well-logging data

Before implementing the algorithms on any type of data, it is necessary to introduce the basic knowledge of the investigated problem properties of the network. Therefore, in the first step, to identify the points and intervals with fracture characteristics and extract the labels 1, meaning the presence of fracture, and 0 meaning the absence of fracture, image logs were analyzed. In this research, among the analyzed data to identify fractures, only two wells had image log information. In well number 1, FMI log, and in well number 2, UBI log. After completing the mentioned Phase, applying the deep learning algorithm, and performing the training and testing processes, image logs are used in the final step to validate the results. By checking the logs of the driven wells, the conventional logs that exist in the two wells are extracted, and by placing the corresponding depth intervals from the structural logs next to the labels extracted from the image logs in the previous step, the desired data set for the algorithm is obtained. Common logs in two wells include neutron porosity, density, sonic, formation photoelectric index, and measured values from gamma-ray logs such as potassium, uranium, and thorium. Therefore, the values corresponding to each log will be introduced to the network as features from the Sarvak depth Formation. Table 2 shows the inputs of the algorithm.

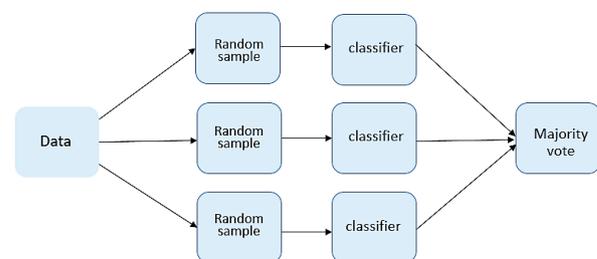


Figure 1. Structure of bagging method algorithm [34].

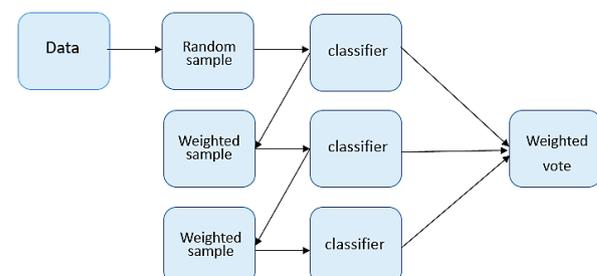


Figure 2. Structure of boosting method algorithm [34].

Table 2. Well-logging data as input of the algorithm.

| Gamma-ray | sonic | photoelectric | density | Neutron |
|-----------|-------|---------------|---------|---------|
| URAN | | | | |
| POTA | DT | PEF | RHOB | NPHI |
| THUR | | | | |

Before the process of training and applying the algorithm, it is necessary to preprocess the data so that the data acquires the appropriate capability and form to feed into the network. These steps include a) data normalization: When different numerical features are available for each sample, and since each of these features has a different scale, it is necessary to bring all the features in a common interval between zero and one, so the features with larger values won't affect the performance of the network. [36]. If we call the desired data sample X and denote the normalized value of the sample by N, each sample can be placed between zero and one, using the following formula.

$$N = \frac{X - \text{Min}}{\text{Max} - \text{Min}} \quad (2)$$

b) Feature selection: In any machine learning algorithms prediction, it is necessary to select the related input features according to the objective variable before the training process so that the network can train and learn in the most optimal mode. The presence of inappropriate features will decrease the performance of the network and will cause the model to overfit [37]. In this research, different algorithms were selected and examined to find suitable features, each of which acted independently in introducing important features. Also, due to the type of problem investigated and the lack of direct relationship between the considered features (well log values and the presence of fractures), the lack of a clear trend in distinguishing the features of both classes in visual evaluations and since the algorithms used in this research has the deep structure, the extraction of suitable features can be done automatically by the deep learning algorithm during the training phase [38]. c) Data balancing: The unbalanced number of samples belonging to each class is one of the common issues in most classification problems [39]. Various algorithms have been presented by programmers to overcome this problem. In this research according to the evaluation of results obtained from various algorithms, the algorithm that ultimately led to the highest possible accuracy in the evaluation was the random sampling algorithm. In any classification problem where the number of samples of each class is unbalanced, we face two types of samples, the majority class and the minority class. These methods are based on the examples in the minority class, which are the important category for learning in the target problem, the reproduction and simulation of the data samples belonging to the minority class lead to an increase in the desired examples [40]. In Tables 3 and 4 the number of samples in each class before and after applying the balance method is shown. d) Splitting data in train and test set: In every process of machine learning and deep learning, it is necessary to determine the training and test data. In this research, after applying balancing algorithms, the number of available samples belonging to both categories increased to 4958 samples in two wells. Due to the small and limited number of available wells and the lack of enough samples for the desired class, which is a fractured type, the training and learning of the algorithm are done on a part of the data of both wells. A part of both wells is also set aside randomly to be used in the test process for validating the trained network. 70 percent of the samples were for training, 20 percent for testing, and 10 percent for validation. In the test dataset which has been set aside before, the number of samples that belong to each class is shown in table 5.

Table 2. The number of samples in each class before applying the random sample method.

| Number of samples | Class |
|-------------------|----------------|
| 2479 | Non- fractured |
| 662 | Fractured |

Table 3. The number of samples in each class after applying the random sample method.

| Number of samples | Class |
|-------------------|----------------|
| 2479 | Non- fractured |
| 2479 | Fractured |

Table 4. The final number of sample data for the testing algorithm.

| Number of samples | Class |
|-------------------|----------------|
| 486 | Non- fractured |
| 506 | Fractured |

In previous Phases, the training data were processed and prepared to feed the deep learning algorithm. The structure used in the network consists of two convolution layers, two Max Pooling layers, and connected layers. In addition, the hyper-parameters such as the size of the kernels, padding, strides, dropout rate, and activation function were selected. The specification of each parameter is explicated below:

a) Kernel: It specifies the size of the convolutional window. Its optional arguments can be set in the network [41]. b) Padding: the number of pixels added to the image during processing by the convolution layer [42]. c) Stride: Refers to the step number of each convolutional layer [41]. d) Dropout rate: It is a technique that is added on layers to prevent model overfitting and it means to invalidate some neurons randomly in the neural network. In other words, to avoid gaining the importance of, some neurons in the model they can be left out [43]. e) Activation function: In artificial neural networks, the activation function of a node defines the output of that node given an input or set of inputs. By making changes or processing the data, they produce the output of the neural network. In some sources, they are also referred to as transfer functions [44]. Activation functions have different types such as sigmoid, rectified linear, and tangential [45]. In this study tangential activation function gave the best results.

In this study, the network performs learning and training operations with 512 input neurons and 200 training steps. It should be noted that, during the trial and error process, the desired parameters were used to adjust the network. After reaching the optimal level and increasing the accuracy of the results, the optimal values of each parameter were determined and hyper-parameter tuning was done by the greed search algorithm. Figure 3 shows the architecture of a one-dimensional convolutional neural network on the well-logging data for classification.

5.2. Applying the 1D-CNN on Seismic data

After loading the seismic data in PETREL software, the steps related to loading the well data, well survey curves, and transferring the well data to the time domain were done. By uploading the labels extracted from the image log reports for each depth interval in binary form, which was done before, the fractured area was identified. Because the fractures are located at the Sarvak formation and considering the coordinates from the highest and lowest depths of the Sarvak formation to extract seismic attributes in this range, the time slice on Seismic data was applied. In Figure 4, the matching of the well tops data in the middle of the Sarvak horizon is shown next to the binary diagram resulting from the fracture density. The red areas represent the presence of fractures and the blue areas mean the absence of fractures in the target range.

After determining the limitations related to the presence of fractures in wells No. 1 and 2 in the seismic data and creating a time slice, in the desired range, the attributes of dip deviation, envelop, variance, chaos, amplitude attenuation, three-dimensional curvature, instantaneous frequency, which were introduced in the previous section, were extracted and used as input of the learning algorithm in the second part of the research. More information about these attributes is explained below:

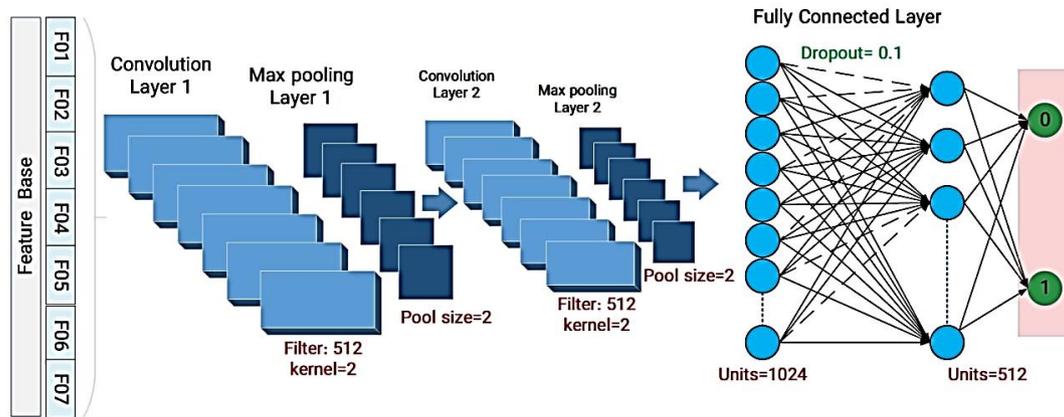


Figure 3. The architecture of the 1D-CNN algorithm for this research.



Figure 4. Depth matching for Sarvak well tops and binary diagram extracted from fracture density in Sarvak horizon.

a) Variance: Variance is an edge detection technique, it calibrates the dissimilarities from a mean value to produce computationally proficient results that are much sharper than coherency [21, 46]. The variance attribute can detect the discontinuities sharply. b) Chaos: The term chaoticness is used for highlighting the reflector discontinuities using differences in azimuth and dip [21, 47]. c) curvature: measured the bending extent of a surface at a specific point using volumetric methods. A curvature is an effective tool for detecting small-scale faults and fractures with low displacement [48]. d) envelope: it's the representation of the amplitude $E(t)$ of an oscillatory function $f(t)$. It is used to highlight the main seismic features such as discontinuities, lithology changes, faults, variations in sedimentary deposits, tuning effects, and sequence boundaries.[49] The envelope formula is shown below:

$$E(t) = \sqrt{(Re s(t))^2 + (Im s(t))^2} \quad (3)$$

e) The dip deviation attribute is one of the edge detection methods that identifies sudden changes in the local slope. This attribute can distinguish features such as faults and channel margins from the slope of the reflections [50]. f) Instantaneous frequency: This physical attribute, which is defined to be the time rate of instantaneous phase change, is used to identify specific events, such as abnormal events and thin layer tuning based on its frequency content [51]. The formula is as follows:

$$freq(x, t) = \frac{d[Ph(x, t)]}{dt} = \frac{f(x, t) \cdot \frac{dg}{dt} - g(x, t) \cdot \frac{df}{dt}}{f^2(x, t) + g^2(x, t)} \quad (4)$$

g) Attenuation: Theoretically, the attenuation is associated with the movement of fluid in the pore; hence the existence of hydrocarbon in the pore will be represented by the attenuation attribute directly. It can be formulated as:

$$Q = \frac{2\pi E}{\Delta E} \quad (5)$$

Q represents the attenuation or rock quality factor; the numerator of the upper fraction represents the energy of the seismic wave and the denominator of the fraction represents the energy lost during one wave cycle [50].

After applying the seismic attributes, in the next step, it is necessary to extract the time interval of attributes where the fractures have been determined for feeding into algorithms. Performing this part requires the application of preliminary steps to extract seismic attributes at specific time intervals corresponding to pre-loaded tags representing fractured and non-fractured depth intervals. These steps include gridding, building the horizon in the desired range, layering, scaling seismic data and binary diagram extracted from the image logs, and finally extracting attributes from the area of two wells and the Sarvak zone.

Just like the part of applying the algorithm on the data obtained from well logging, the process of training the network on new inputs also requires initial processing. These steps include removing missing values, normalizing data, determining features and objective functions, checking input features, balancing the number of samples belonging to both classes and finally selecting the training and the test data.

To use deep learning algorithms, it is necessary to have a sufficient number of training samples. owing to the limitation of the number of training samples in the upcoming research due to the limitation in the number of available wells, in this section, to achieve the desired accuracy and implement the 1D-CNN algorithm, which is an example of deep networks, the combined algorithms of machine learning and Deep learning were also used. It should be noted that 70% of the data was used for training, the remaining 10% was used as algorithm validation data, and 20% was used as test data. As in the previous section, where the input of the network was the outputs obtained from the well-logging tools, first, the 1D-CNN algorithm was used for data training and classification. This network performs the training process with an optimal amount of 512 input neurons and three convolutional layers. The optimal values were obtained by trial and error method and reaching the minimum desired accuracy, and then by optimizing the parameters using the greed search algorithm. Also, to increase network efficiency, Max pooling, Flatten, and Dense layers were added and finally, the values that lead to an increase in network performance were used with the optimal amount of 100 training steps.

5.3. Applying deep-majority voting algorithm

Since the results of the previous stage did not show appreciable accuracy for this algorithm, to achieve the desired accuracy, the cumulative algorithm was used along with the 1D-CNN algorithm. In other words, in this approach, feature extraction by the deep algorithm and classification by cumulative algorithm done, which is a combination of LGBM, random forest, and decision tree algorithms, perform the learning process with the majority voting approach. Figure 5 shows the structure of the combined algorithm.

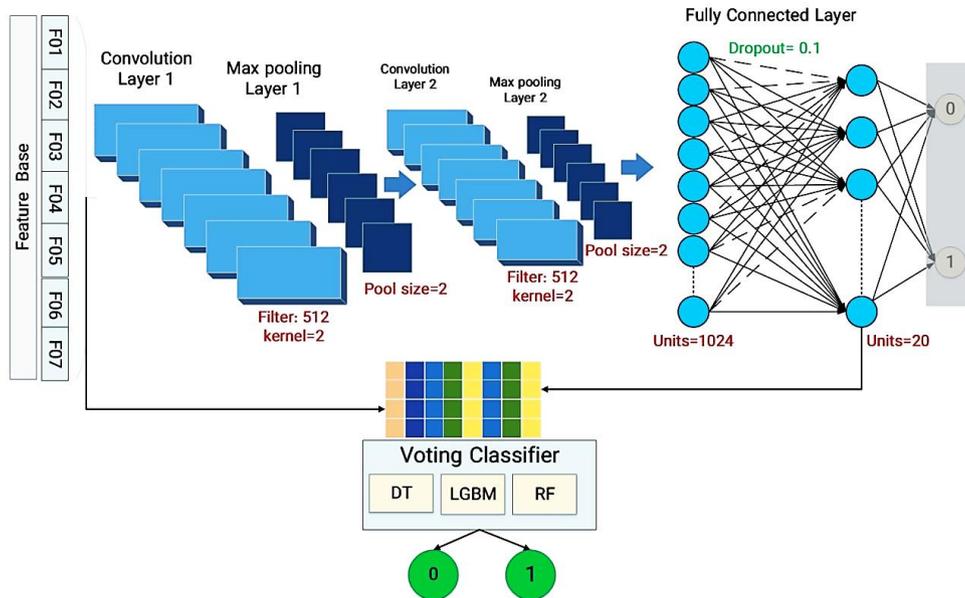


Figure 5. The structure of the combined algorithm.

6. Results and discussions

In this study, the ROC curve and the report of the confusion matrix were used to evaluate the results. The final results have been explained below.

The training and test accuracy values provide important information because they give us a better insight into how the learning performance changes over the number of epochs and help us diagnose any problems with learning that can lead to an underfit or an overfit model. As seen in Figure 6, the training process of the algorithm through the training steps is shown in these curves the good fit is the goal of the learning algorithms and is identified by a training and validation loss that decreases to a point of stability with a minimal gap between the two final loss values [52]. Figure 6 shows the training process of the algorithm through the training steps. By setting the optimal parameters for the network, the accuracy of the test data reaches 94%.

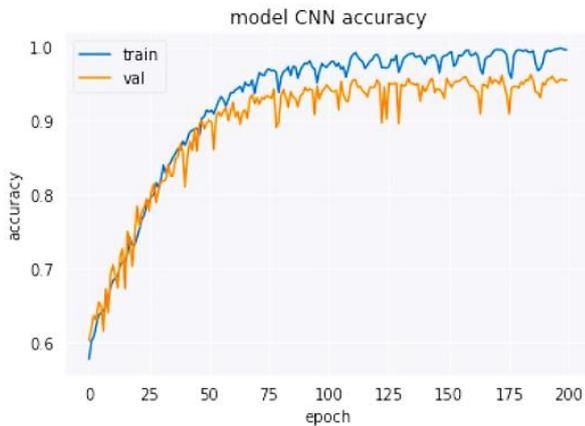


Figure 6. Training curve of 1D-CNN on petrophysical data, the blue curve for the train set, and the orange curve for the validation set.

Figure 7 shows the results of the confusion matrix for the test data. A confusion matrix is a table that is used to define the performance of a classification algorithm. A confusion matrix visualizes and summarizes the performance of a classification algorithm [53]. each row of

the matrix represents the instances in an actual class while each column represents the instances in a predicted class. As can be seen in the figure, among the 506 samples that belonged to class 1, 487 samples were fractured and were correctly classified as class 1 or fractured samples. In the same way, among the 486 samples that belonged to the zero class or un-fractured class, 425 samples were correctly classified in this category. As shown in the confusion matrix the Y-axis shows the true values and X-axis shows the predicted values, so the true positive rate is equal to 487, the true negative rate is 425, the false positive rate is 61, and the false negative is equal to 19. So, the result shows that this algorithm can classify the samples correctly.

Another evaluation metric that was used in this research is the receiver operating characteristic curve (ROC), which is shown in Figure 8. roc_auc_score is defined as the area under the ROC curve, which is the curve having False Positive Rate on the x-axis and True Positive Rate on the y-axis at all classification thresholds. According to this curve, in a binary classification, an algorithm with the best performance is closer to number one, in other words, it has the largest area under the curve.

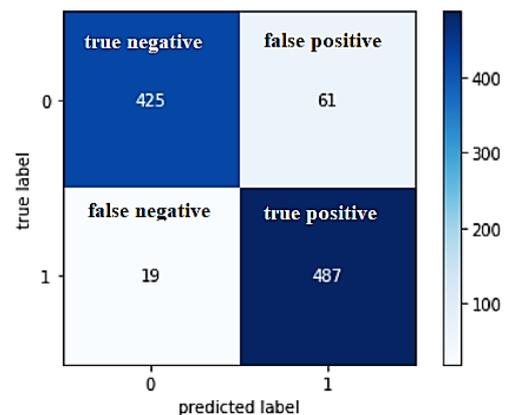


Figure 7. Confusion matrix of the test set. the darker color of the block, the stronger the class recognition.

This graph shows how well the algorithm can distinguish false positives from true positives. As can be seen in the figure, the level under the curve approaches the number one and has a maximum of 0.94, which is an acceptable percentage.

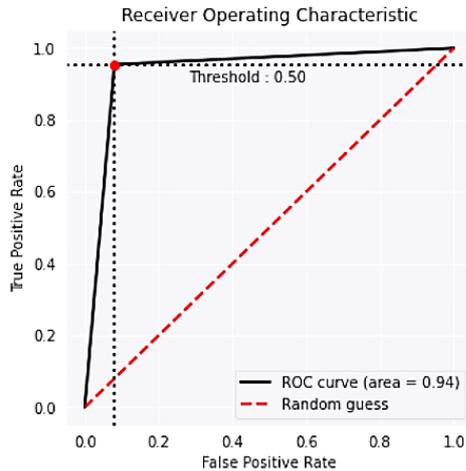


Figure 8. roc_auc_score as the area under the ROC curve.

6.1. Validation of the results with image log

For each depth sample in the test data, the corresponding predicted label was also obtained by the 1D-CNN algorithm in the previous section. At this stage, the validation of the obtained results is done by putting together the real labels that were initially extracted from the image logs and the labels predicted by the algorithm for the test set. In other words, the corresponding depth of the test set was already available. In the reports presented from the image logs for these depths, in which we considered the corresponding conventional log values as the test samples, the presence or absence of fracture was determined, therefore, a column titled real Label was created. The algorithm also evaluated the total final test samples according to the training and test that it had done before and presented the predicted labels. Figure 9 shows the results related to the depth range of the test. The white lines are the depth intervals with fractures and the black lines are the depth intervals without the presence of the fracture.

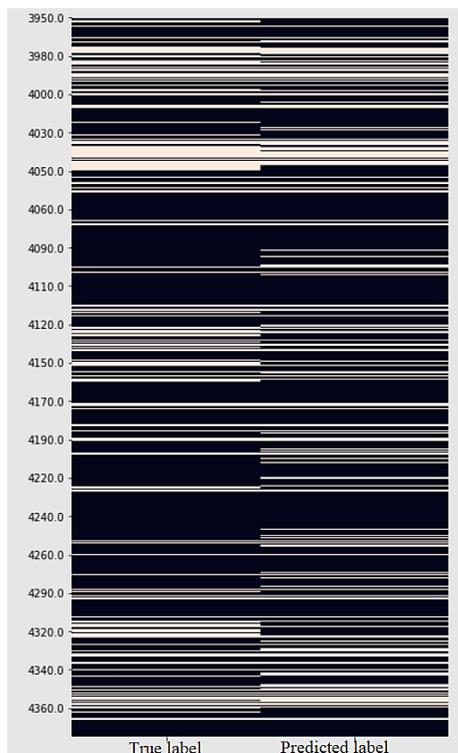


Figure 9. Validation of the predicted labels from the 1D-CNN algorithm on the right side and the real labels from image logs on the left side.

6.2. The results of 1D-CNN vs. cumulative 1D-CNN majority voting algorithm on seismic data

Using the confusion matrix evaluation criteria and the receiver operating characteristic diagram, the results of the 1D-CNN network are evaluated. As seen in Figure 10, the algorithm training graph is shown on the validation data. The test curve shown in orange has strong fluctuations and it shows that the training process is not done well. Also, in Figure 11 from the confusion matrix, it can be concluded that the performance of the network is not accurate enough to distinguish test samples belonging to both classes. According to the figure in the test data, among the 55 samples in class 1, the algorithm has correctly classified 39 samples as class 1, and among the 221 samples in class 0, 151 samples have been correctly classified as zero, and 76 samples were classified wrongly. Also, Figure 12 shows the ROC. The area under the curve shows a value of 0.69, which indicates an average level of accuracy and is not suitable enough.

As mentioned in the previous section, a combination of deep and cumulative algorithms was used to increase the accuracy of the model.

The main benefit of CNN compared to its predecessors is that it automatically identifies the relevant features without any human supervision. These algorithms are suitable for when we have a lot of data, in other words, having a lot of data is one of the requirements to use them. Since in the second part of the research, which is the use of seismic data, few data were available, in such cases, the use of shallow algorithms can obtain higher accuracy. By comparing the results obtained from the so-called machine learning algorithms and the combination of the majority and deep voting algorithms, it is concluded that using deep algorithms and majority voting sequentially has an acceptable effect in increasing accuracy.

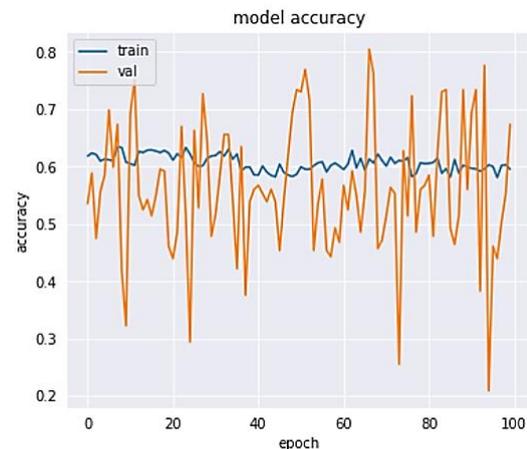


Figure 10. Training curve for applying 1D-CNN on seismic data.

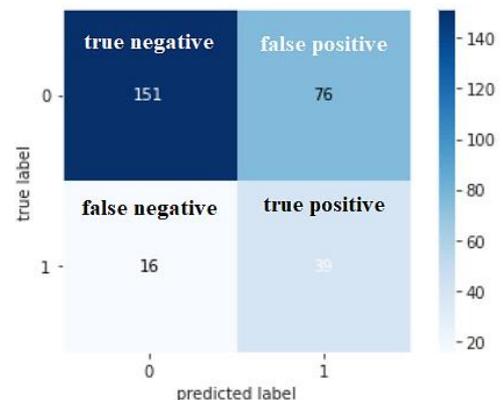


Figure 11. Confusion matrix of the 1D-CNN algorithm on seismic data. There is no balance in distinguishing samples that belong to class one and zero.

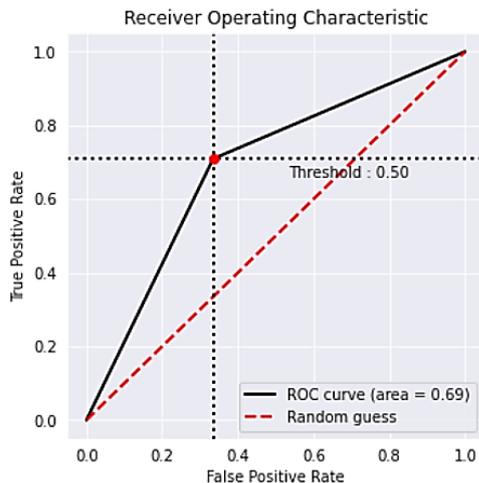


Figure 12. ROC of applying 1D-CNN on the test set of seismic data as shown in the area under the curve is near 0.5 and it shows a moderate level of accuracy.

Considering the seismic attributes as features and fracture labels at the well scale, and the inability of the algorithm to find the relationship between the features and labels, as a solution A one-dimensional convolutional algorithm was used to automatically extract suitable patterns from existing features and create the feature itself. In the following, the features created by the one-dimensional algorithm are fed into three algorithms: random forest, LGBM, and decision tree, strictly using the majority vote method for each input sample to classify it. Figure 13 shows a view of the process of the majority vote algorithm. As seen in the figure, each sample enters the different classifier separately, according to the training that each category has had on the training samples, it predicts a class for the new sample. Finally, the new sample belongs to the class that gets the most votes based on the prediction of existing classification algorithms is known.

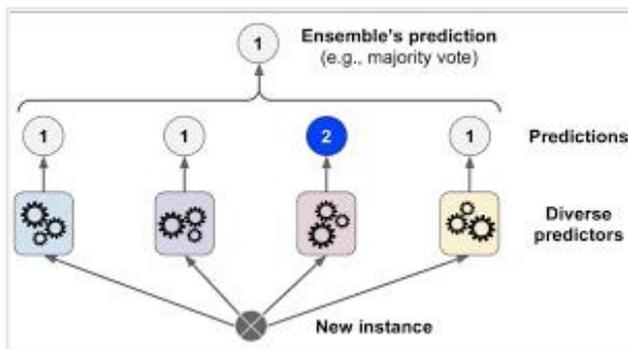


Figure 13. Majority voting algorithm structure.

According to the final results, the cumulative algorithm gives better accuracy compared to the implemented deep algorithm. As seen in Figure 14, among the 55 samples in the fractured class, the algorithm has classified 48 samples correctly and only 7 samples have classified wrongly as the non-fractured category. Also, out of 221 samples belonging to the non-fractured class, the algorithm correctly has classified 163 samples as the zero class and made a mistake in only 64 samples.

Also, Figure 15 shows the ROC of the combined algorithm. As it is clear from the area under the curve, this algorithm can classify the samples in the test data with 80% accuracy.

In Figure 16, the validation of the predicted labels for the time zones in the test data has been extracted. Similar to the validation of network results on well-log data, the actual time zone labels of the test data were already available. This information was placed alongside the results obtained from network classification (predicted labels) on seismic data.

The white color indicates the presence of the fracture and the black color indicates the absence of it among test data.

6.3. Generalizing the training algorithm on the entire reservoir cube attributes

By coding on the desired ranges and obtaining the accuracy of the results related to the test data, the attributes were extracted from the entire Sarvak zone, so that this time, the algorithm was applied to the entire seismic cube limited area. In other word, the network was previously trained on the time zones that had zero and one labels and was tested with acceptable accuracy on the test data. Now, this trained network was applied to the entire reservoir area, which means that this time the test data is the entire reservoir area and the network should provide coordinates that can contain fractures or not. In this section, entering the entire coordinates of the reservoir zone as the input to the algorithm was a challenge. Due to the large volume of input data, which were coordinate samples of the entire reservoir, applying the algorithm on them would cause problems in the network's performance, and the process would face bugs. Therefore, the samples were fed to the network in separate sections, so the 1D-CNN algorithm extracts the features, and classification algorithms vote the related label by evaluating the characteristics of each sample, which are the seismic attributes corresponding to each coordinate. Therefore, the cumulative algorithm predicts the label of each coordinate node, and the predicted output of the entire reservoir cube is created.

Figure 17 shows the fracture density map in the total reservoir area. It should be noted that the blue color indicates the absence of the fracture and the red color indicates its presence.

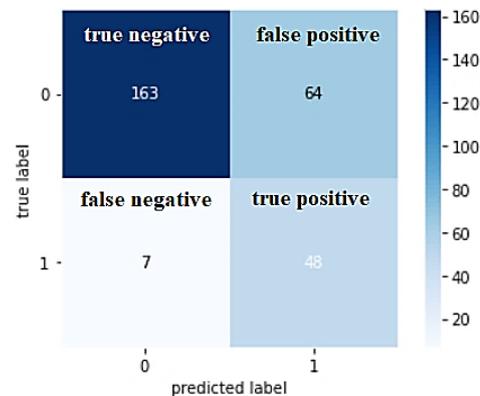


Figure 14. The result of a confusion matrix for applying cumulative deep-majority voting classifier on a test set of seismic data.

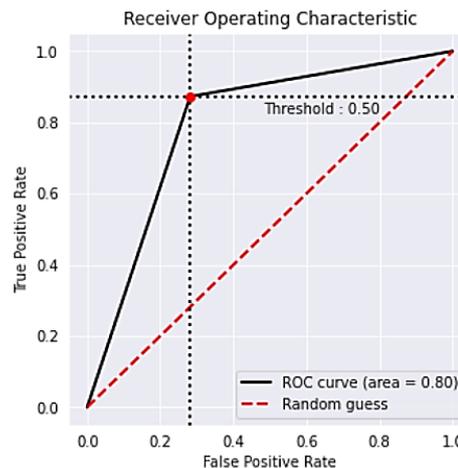


Figure 15. ROC of the test set driven from cumulative deep-majority voting algorithm, as shown the area under the curve is near to one in comparison of previous ROC from 1D-CNN.

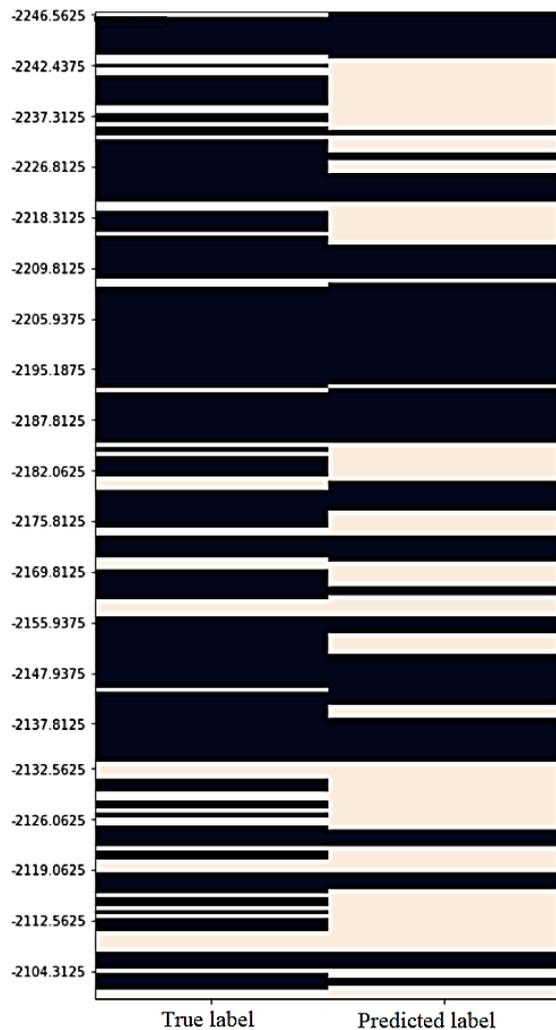


Figure 16. Validation of predicted labels from combining 1d_CNN and majority voting classifiers on the test set, on the left side, and real labels on the left side.

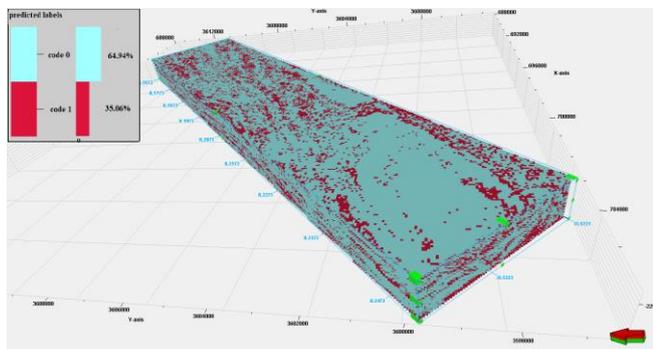


Figure 17. Fracture density map obtained from deep-cumulative algorithm on Sarvak reservoir cube.

7. Conclusion

In this research, it was tried to use deep learning algorithms to identify fractures in a carbonate reservoir. According to the previous studies, the fractures were on the fault scale, the present research was investigated to identify the fractures on the well scale. Considering the importance of carbonate reservoirs due to fracture and permeability and

the uncertainty of the tools used, using deep networks with high precision can help to identify fracture, which is one of the most complex and effective parameters in reservoirs. According to the analysis, the results are as follows:

- 1) Detection of fractured and non-fractured samples with 94% accuracy by validation data using a one-dimensional convolutional algorithm.
- 2) Achieving higher accuracy equivalent to 80% in the application of the cumulative-deep algorithm on seismic data compared to the application of the deep learning algorithm individually.
- 3) Using a one-dimensional convolutional algorithm to extract features has a significant effect on increasing the accuracy.
- 4) Using seismic data with higher representation quality in fractured intervals is suggested.
- 5) Since in this study there were a small number of wells, a part of the data from both wells was used as blind data. To achieve more effective results, it is recommended an area with a large number of wells with more fracture characteristics for predicting the zones of at least one well.
- 6) Using drilling, exploitation, and reservoir data as fracture presence indicator besides the seismic and well log data is helpful.
- 7) Majority voting algorithms are beneficial and the novelty of this research is applying these algorithms with deep learning methods.

Acknowledgment

The authors would like to declare their sincere thanks to the National Iranian Central Oil Fields Company (NICOFC), Mr. Kamrani, and Mrs. Ghorbani of the NICOFC for their support in providing the information and their technical input regarding this work.

REFERENCES

- [1] Blès, J.-L. and B. Fuega (1986). "The fracture of rocks."
- [2] Fossen, H. (2016). Structural geology, Cambridge University Press.
- [3] Nelson, R. (2001). Geologic analysis of naturally fractured reservoirs, Elsevier.
- [4] Schlumberger" FMI full bore formation micro imager" , Houston: Schlumberger Educational services, 1994.
- [5] AGHLI, G., et al. (2016). "DETECTION OF FRACTURE USING PETROPHYSICAL LOGS BY NEW METHOD AND ITS COROLATION WITH IMAGE LOGS."
- [6] Ezati, M., et al. (2019). "Petrophysical Evaluation of a Severely Heterogeneous Carbonate Reservoir; Electrical Image Logs Usage in Porosity Estimation." Open Journal of Geology 9(03): 193.
- [7] Bates, C., et al. (1999). "The study of a naturally fractured gas reservoir using seismic techniques." AAPG bulletin 83(9): 1392-1407.
- [8] Martinez, L. P., et al. (2002). "Identification and characterization of naturally fractured reservoirs using conventional well logs." The University of Oklahoma: 1-23.
- [9] Glikson, E. and A. W. Woolley (2020). "Human trust in artificial intelligence: Review of empirical research." Academy of Management Annals 14(2): 627-660.
- [10] Perrotta, P. (2020). "Programming Machine Learning: From Coding to Deep Learning." Programming Machine Learning: 1-300.

- [11] Ozkaya, S. I. (2008). "Using probabilistic decision trees to detect fracture corridors from dynamic data in mature oil fields." *SPE Reservoir Evaluation & Engineering* 11(06): 1061-1070.
- [12] Tokhmechi, B., et al. (2009). "A novel approach proposed for fractured zone detection using petrophysical logs." *Journal of Geophysics and Engineering* 6(4): 365-373.
- [13] Ja'fari, A., et al. (2012). "Fracture density estimation from petrophysical log data using the adaptive neuro-fuzzy inference system." *Journal of Geophysics and Engineering* 9(1): 105-114.
- [14] Yousef, A. N., et al. (2014). "A combined Parzen-wavelet approach for detection of vuggy zones in fractured carbonate reservoirs using petrophysical logs." *Journal of Petroleum Science and Engineering* 119: 1-7.
- [15] Zheng, Z. H., et al. (2014). Multi-attributes and neural network-based fault detection in 3D seismic interpretation. *Advanced Materials Research, Trans Tech Publ.*
- [16] Bayat, A. A., et al. (2015). "Continuous Fracture Modeling in Carbonate Reservoir by Integration of Seismic, Geological, and Petrophysical Data." *Journal of Petroleum Research* 25(83): 96-107.
- [17] Cruz, R. A. Q., et al. (2017). Improving accuracy of automatic fracture detection in borehole images with deep learning and GPUs. 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), IEEE.
- [18] Xiong, W., et al. (2018). "Seismic fault detection with convolutional neural network." *Geophysics* 83(5): O97-O103.
- [19] Di, H., et al. (2018). "Deep convolutional neural networks for seismic salt-body delineation."
- [20] Udegbe, E., et al. (2018). Big Data Analytics for Seismic Fracture Identification, Using Amplitude-Based Statistics. *SPE Annual Technical Conference and Exhibition, OnePetro.*
- [21] Ashraf, U., et al. (2020). "Application of unconventional seismic attributes and unsupervised machine learning for the identification of fault and fracture network." *Applied Sciences* 10(11): 3864.
- [22] Tian, M., et al. (2021). "Deep learning assisted well log inversion for fracture identification." *Geophysical Prospecting* 69(2): 419-433.
- [23] Alzubaidi, L., et al. (2021). "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions." *Journal of big Data* 8(1): 1-74.
- [24] Goodfellow, I., et al. (2016). *Deep learning*, MIT Press.
- [25] LeCun, Y., et al. (2015). "Deep learning." *nature* 521(7553): 436-444.
- [26] Sengupta, S., et al. (2020). "A review of deep learning with special emphasis on architectures, applications and recent trends." *Knowledge-Based Systems* 194: 105596
- [27] Albawi, S., et al. (2017). Understanding of a convolutional neural network. 2017 international conference on engineering and technology (ICET), Ieee.
- [28] Alzubaidi, L., et al. (2021). "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions." *Journal of big Data* 8(1): 1-74.
- [29] Santosh, K., et al. (2021). *Deep Learning Models for Medical Imaging*, Academic Press.
- [30] Yamaguchi, K., et al. (1990). A neural network for speaker-independent isolated word recognition. *ICSLP*.
- [31] Yamashita, R., et al. (2018). "Convolutional neural networks: an overview and application in radiology." *Insights into imaging* 9(4): 611-629.
- [32] Sagi, O. and L. Rokach (2018). "Ensemble learning: A survey." *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8(4): e1249.
- [33] Bühlmann, P. (2012). Bagging, boosting and ensemble methods. *Handbook of computational statistics*, Springer: 985-1022.
- [34] Peterson, K. D. (2018). "Resting heart rate variability can predict track and field sprint performance." *OA Journal-Sports* 1(1).
- [35] Zhou, Z.-H. (2012). *Ensemble methods: foundations and algorithms*, CRC press.
- [36] Kamani, G., et al. (2019). "Data normalization in data mining using graphical user interface: A pre-processing stage." *Gujarat Journal of Extension Education* 30(2): 106-109.
- [37] Forman, G. (2003). "An extensive empirical study of feature selection metrics for text classification." *J. Mach. Learn. Res.* 3(Mar): 1289-1305.
- [38] Katz, G., et al. (2016). *Explorekit: Automatic feature generation and selection*. 2016 IEEE 16th International Conference on Data Mining (ICDM), IEEE.
- [39] Chawla, N. V., et al. (2002). "SMOTE: synthetic minority over-sampling technique." *Journal of artificial intelligence research* 16: 321-357.
- [40] Ganganwar, V. (2012). "An overview of classification algorithms for imbalanced datasets." *International Journal of Emerging Technology and Advanced Engineering* 2(4): 42-47.
- [41] Kalyani, G. and B. Janakiramaiah (2021). "Deep learning-based detection and classification of adenocarcinoma cell nuclei." *Trends in Deep Learning Methodologies*: 241-264.
- [42] Dwarampudi, M. and N. Reddy (2019). "Effects of padding on LSTMs and CNNs." *arXiv preprint arXiv:1903.07288*.
- [43] Rala Cordeiro, J., et al. (2021). "Neural Architecture Search for 1D CNNs—Different Approaches Tests and Measurements." *Sensors* 21(23): 7990
- [44] Nwankpa, C., et al. (2018). "Activation functions: Comparison of trends in practice and research for deep learning." *arXiv preprint arXiv:1811.03378*.
- [45] Wang, Y., et al. (2020). "The influence of the activation function in a convolution neural network model of facial expression recognition." *Applied Sciences* 10(5): 1897.
- [46] Pigott, J.D.; Kang, M.-H.; Han, H.-C. First order seismic attributes for clastic seismic facies interpretation: Examples from the East China Sea. *J. Asian Earth Sci.* 2013, 66, 34–54.
- [47] Fang, J.; Zhou, F.; Tang, Z. Discrete fracture network modelling in a naturally fractured carbonate reservoir in the Jingbei oilfield, China. *Energies* 2017, 10, 183.
- [48] Mai, H.T.; Marfurt, K.J.; Chávez-Pérez, S. Coherence and volumetric curvatures and their spatial relationship to faults and folds, an example from Chicotepec basin, Mexico. In *Proceedings of the 2009 SEG Annual Meeting*, Houston, TX, USA, 25–30 October 2009; Society of Exploration Geophysicists: Houston, TX, USA, 2009; pp. 1063–1067

- [49] Taner, M. T., et al. (1994). Seismic attributes revisited. SEG technical program expanded abstracts 1994, Society of Exploration Geophysicists: 1104-1106.
- [50] Koson, S., et al. (2013). "Seismic attributes and their applications in seismic geomorphology." *Bulletin of Earth Sciences of Thailand* 6(1): 1-9.
- [51] TANER, M. T. (2000). "Attribute Revisited, Rock Solid Images." http://www.rocksolidimages.com/pdf/attrib_revisited.htm.
- [52] Yingge, H., et al. (2020). Deep neural networks on chip-A survey. 2020 IEEE International Conference on Big Data and Smart Computing (BigComp), IEEE.
- [53] [Singh, P.a.S., Dr and Singh, Krishna Kant and Singh, Akansha], Diagnosing of disease using machine learning. 10.1016/B978-0-12-821229-5.00003-3. Vol. 10.1016/B978-0-12-821229-5.00003-3. 2021.