Journal of Theoretical and Applied Information Technology
31st May 2023. Vol.101. No 10

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ABUNASER - A NOVEL DATA AUGMENTATION ALGORITHM FOR DATASETS WITH NUMERICAL FEATURES

BASEM S. ABUNASSER[1], SALWANI MOHD DAUD[2], IHAB S. ZAQOUT[3], SAMY S. ABU-NASER[4]

[1,2]University Malaysia of Computer Science & Engineering (UNIMY), Cyberjaya, Malaysia

[3,4]Faculty of Engineering and Information Technology, Al-Azhar University, Gaza, Palestine

Email: [1]p05210002@student.unimy.edu.my, [2]salwani.daud@unimy.edu.my, [3]i.zaqout@alazhar.edu.ps, [4]Abunaser@alazhar.edu.ps

ABSTRACT

This research paper introduces Abunaser, a novel data augmentation algorithm for numerical datasets. Abunaser is designed to address the challenge of overfitting in machine learning models when working with small numerical datasets. We evaluate the effectiveness of Abunaser in improving the performance of machine learning models on numerical datasets and compare it with other commonly used data augmentation techniques. Our results show that Abunaser can effectively increase the size of the dataset and improve the performance of machine learning models across different types of tasks, including classification, regression, and clustering. We also investigate the sensitivity of Abunaser to different parameters, such as the size of the dataset and the number of features. Additionally, we provide insights into the underlying mechanisms of Abunaser and how it affects the distribution and structure of the augmented data. However, we acknowledge some limitations of our research, including the dataset characteristics and computational requirements of Abunaser. Overall, our study suggests that Abunaser is a promising data augmentation algorithm for numerical datasets and has the potential to improve the performance of machine learning models in various applications.

Keywords: *Dataset, Augmentation, Machine learning, supervised models, Deep Learning*

1. INTRODUCTION

As a result of the great progression of Artificial Intelligence (AI) [1–10] in the recent years; the significance of data has developed vastly. The bottleneck for AI in today's world is Lack of Data. Lack of Data remains a persistent problem in numerous areas where AI can be employed. In some circumstances, even though the dataset is existing and sufficient big, then labeling is a great problem when one is working with supervised learning tasks. Manual labeling is possible nevertheless is extremely burdensome when dealing with big datasets. For instance, in the works of [1], the authors have attempted to automatically label images by a suggested label proliferation agenda based on "Kernel Canonical Correlation Analysis". They built a semantic space in a way that the correlation of visual attributes are well-kept in an embedded semantic. In order for the researchers to avoid the need for big datasets, they applied the Transfer Learning idea. In Transfer Learning case, a model was trained on a dissimilar dataset and then retrained on the new dataset, connected with the task being solved by adjusting some of the weights of the network being trained. However this idea totally fails when the current dataset is not associated to the dataset on which the model was trained on.

Another major issue related to the data augmentation is the data balancing. The dataset sometimes big enough; but the classes within the dataset are not balanced. In classification problems, one may encounter this situation where the target class label is not equally distributed. This kind of a dataset is called Imbalanced dataset. Imbalance in dataset can be a blocker to train a deep learning model. In the situation of imbalance class tasks, the deep learning model is trained largely on the majority class and the deep learning model come to be biased to the majority class classification.
Therefore management of imbalance class is crucial beforehand continuing to the model pipeline. There are several class balancing methods that resolve the

task of class balancing by either creating a new sampling of the minority class or by removing some majority class samples. Treatment of class balancing methods can be generally categorized into two classes:

1- Over-sampling techniques: It refers to creating artificial new samples for the minority class to become balanced with majority class. Example for oversample technique is SMOTE.
2- Under-sampling techniques: it refers to removing majority class samples to become balanced with minority class. Example for under sample technique is SMOTE.

One disadvantage of employing under sampling method is that one is losing out a lot of majority class data samples for balancing the class. Oversampling method overcome that disadvantage; however, creating multiple samples within the minority class may result in overfitting during the training of the deep learning model.

Synthetic Minority Oversampling Technique (SMOTE) is one of the most popular oversampling technique researchers that create artificial minority data samples within the cluster of minority class. SMOTE works by selecting examples that are close in the feature space, drawing a line between the examples in the feature space and drawing a new sample at a point along that line. Specifically, a random example from the minority class is first chosen. Then k of the nearest neighbors for that example are found (typically $k=5$). A randomly selected neighbor is chosen and a synthetic example is created at a randomly selected point between the two examples in feature space.

Augmentation of data is the way of creating synthetic data using the given dataset. There are different techniques through which you can do data augmentation. When the data involve images, augmentation methods can be like rotation, flipping, scaling, cropping, translation, shearing, etc. Furthermore, advanced techniques can also be performed images is Generative Adversarial Networks (GANs). GANs provide a way to learn deep representations without extensively labeling training data. They attain this through deriving backpropagation signals through a competitive process involving a pair of networks. The representations that can be learned by GANs may be used in a range of applications, including image synthesis, semantic image editing, image super-resolution and classification

In our proposed methodology (Abunaser), we have used somewhat similar to SMOTE algorithm to generate synthetic data with CSV files. Abunaser is a model which studies the distribution of the features of dataset, and then samples out the artificial examples based on this distribution. Abunaser algorithm detailed methodology is described in Section. 6.

2. PROBLEM STATEMENT

Numerical datasets are commonly used in machine learning tasks such as classification, regression, and clustering. However, when the dataset is small, the performance of machine learning algorithms may suffer due to overfitting, which occurs when the model learns the noise in the data rather than the underlying patterns. Data augmentation is a technique that can be used to increase the size of the dataset and prevent overfitting. While there are several data augmentation methods available for image datasets, there are relatively fewer methods available for datasets with numerical features. This presents a challenge for researchers and practitioners working with numerical datasets. Therefore, there is a need for a novel data augmentation algorithm specifically designed for datasets with numerical features. The purpose of this research paper is to introduce and evaluate Abunaser, a novel data augmentation algorithm that can effectively augment numerical datasets, thereby improving the performance of machine learning models.

3. OBJECTIVES

- To develop a novel data augmentation algorithm, Abunaser, for numerical datasets.
- To evaluate the effectiveness of Abunaser in improving the performance of machine learning models on numerical datasets.
- To compare the performance of Abunaser with other commonly used data augmentation techniques for numerical datasets.
- To analyze the impact of Abunaser on different types of machine learning tasks, such as classification, regression, and clustering.
- To investigate the sensitivity of Abunaser to different parameters, such as the size of the dataset and the number of features.
- To provide insights into the underlying mechanisms of Abunaser and how it affects the distribution and structure of the augmented data.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. LIMITATIONS

- The performance of Abunaser may be influenced by the specific characteristics of the dataset used in the evaluation, such as the distribution of the features, the size of the dataset, and the noise level.
- The comparison of Abunaser with other data augmentation techniques may depend on the specific evaluation metrics used and the machine learning models employed.
- Abunaser may not be suitable for all types of numerical datasets, such as those with highly irregular or non-linear distributions.
- The evaluation of Abunaser may be limited to a specific set of machine learning tasks, and the results may not generalize to other domains.
- The impact of Abunaser on the interpretability of machine learning models may need to be further investigated.
- The computational requirements of Abunaser may be higher compared to other data augmentation techniques, and this may limit its practical utility in some applications.

5. RELATED WORK

Random erasing [10] is another interesting Data Augmentation technique that was developed. It was motivated by the dropout regularization techniques. It can be viewed as analogous to dropout except in the level of input data space instead of being embedded into the architecture of the network. This method was precisely designed to fight image recognition contests due to obstruction. Obstruction refers to when some parts of the object are unclear. Random erasing stops this by making the model to learn more expressive features about an image, stopping it from overfitting to a certain pictorial feature in the image. Apart from the visual challenge of obstruction, in precise, random erasing is an encouraging technique to assurance a network pays care to the whole image, instead of just a part of the image.

Random erasing works by randomly selecting an $n \times m$ patch of an image and masking it with either 0 s, 255 s, mean pixel values, or random values. On the CIFAR-10 dataset this resulted in an error rate reduction from 5.17 to 4.31%. The best patch fill method was found to be random values. The fill method and size of the masks are the only parameters that need to be hand-designed during implementation. Random erasing is a Data Augmentation method that pursues to directly stop overfitting by modifying the input space. By eliminating certain input patches, the model is enforced to discover other descriptive features. This augmentation method can also be stacked on top of other augmentation techniques such as flipping or color filtering. Random erasing produced one of the highest accuracies on the CIFAR-10 dataset. Authors in [11] conducted a similar study called Cutout Regularization. Like the random erasing study, they experimented with randomly masking regions of the image.

Authors in [12] presented an interesting idea to combine random erasing with GANs designed for image in-painting. Image in-painting describes the task of filling in a missing piece of an image. Using a diverse collection of GAN in-painters, the random erasing augmentation could seed very interesting extrapolations. It will be interesting to see if better results can be achieved by erasing different shaped patches such as circles rather than $n \times m$ rectangles. An addition of this will be to parameterize the geometries of random erased patches and learn an optimal erasing configuration.

A disadvantage to random erasing is that it will not always be a label-preserving transformation. In handwritten digit recognition, if the top part of an ‘8’ is randomly cropped out, it is not any different from a ‘6’. In many fine-grained tasks such as the Stanford Cars dataset, randomly erasing sections of the image (logo, etc.) may make the car brand unrecognizable. Therefore, some manual intervention may be necessary depending on the dataset and task [13].

Authors in [3] proposed to use Generative Adversarial Networks (GANs) as a novel way to extract more information from a medical dataset, by generating synthetic samples very similar in appearance to the real images [3].

In [4], they proposed using a GAN-based model for synthetic medical image augmentation for increasing the performance of the Convolutional Neural Network (CNN) in liver lesion classification [4].

In study [5], they offered Balancing GAN (BAGAN) as a tool for augmentation that returns balance to imbalanced data. In some cases, the data is even not enough to generate more by training GANs. They demonstrated that BAGAN generates more realistic images in imbalanced datasets in comparison to other stated GANs.

In [6] they proposed to use EmbNum+, a numerical embedding for learning both discriminant representations and a similarity metric from numerical columns, to do the attribute augmentation. Attribute augmentation generates

Journal of Theoretical and Applied Information Technology
31st May 2023. Vol.101. No 10

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

samples by changing the size of the attributes and randomly choose the numerical values in the original attributes. However, in numerical datasets, there has not been enough advancement and there are still more rooms to work on.

6. ABUNASER DATA AUGMENTATION ALGORITHM

The proposed algorithm starts with choosing a numerical feature from the dataset, for each category in the output class, determine the N value that makes that class is balanced, determine the max and min for that category in the output class. Generate a random number between min and max and replace the feature value with this random value. Add the new sample to the new dataset. Keep generating random values and replacing the feature value and adding to the new dataset until N is reached.

1. Study the numerical Features in the Dataset
2. Select one of the features ***f***
3. For each label ***l*** in the output Class Do
   - 3.1 Determine N the number of samples to make the dataset balanced for ***l***
   - 3.2 Determine the Min^f and Max^f of the selected feature ***f*** of label ***l***
   - 3.3 Generate a random number between Min^f and Max^f
   - 3.4 Keep all other features as is and replace the new random value in place of feature ***f***
   - 3.5 Add the new sample to the new dataset
   - 3.6 Repeat the process of 3.3 -3.5 ***N*** times so the number of samples in the feature become balanced or the number samples reach s specific count.
4. Return the new balanced dataset

Figure1. Abunaser data augmentation algorithm

6.1 Process of evaluating Abunaser Algorithm

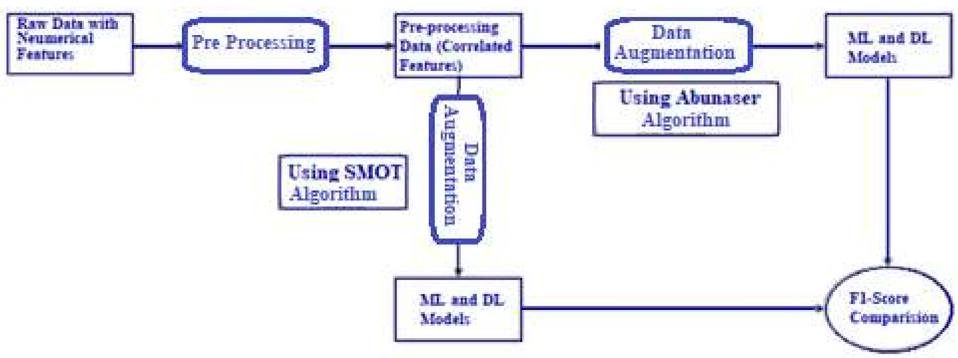

Figure. 2. The process of evaluating Abunaser Data Augmentation

6.2 The First Experiment (Cirrhosis Dataset)

Cirrhosis is a late stage of scarring (fibrosis) of the liver caused by many forms of liver diseases and conditions, such as hepatitis and chronic alcoholism. The dataset contains the information collected from the Mayo Clinic between 1974 and 1984[7].

The dataset consists of 644 samples, 20 features (19 input features and one output feature).

Table1: Description of the Cirrhosis Dataset features

Feature	Description	Input/output
ID:	unique identifier	Input
N_Days:	number of days between registration and the earlier of death, transplantation, or study analysis time in July 1986	Input
Status:	status of the patient C (censored), CL (censored due to liver tx), or D (death)	Input
Drug:	type of drug D-penicillamine or placebo	Input
Age:	age in [days]	Input
Sex:	M (male) or F (female)	Input
Ascites:	presence of ascites N (No) or Y (Yes)	Input
Hepatomegaly:	presence of hepatomegaly N (No) or Y (Yes)	Input
Spiders:	presence of spiders N (No) or Y (Yes)	Input
Edema:	presence of edema N (no edema and no diuretic therapy for edema), S (edema present without diuretics, or edema resolved by diuretics), or Y (edema despite diuretic therapy)	Input
Bilirubin:	serum bilirubin in [mg/dl]	Input
Cholesterol:	serum cholesterol in [mg/dl]	Input
Albumin:	albumin in [gm/dl]	Input
Copper:	urine copper in [ug/day]	Input

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Alk_Phos:	alkaline phosphatase in [U/liter]	Input
SGOT:	SGOT in [U/ml]	Input
Triglycerides:	triglicerides in [mg/dl]	Input
Platelets:	platelets per cubic [ml/1000]	Input
Prothrombin:	prothrombin time in seconds [s]	Input
Stage:	histologic stage of disease (1, 2, 3, or 4)	Output

We have pre-processed the dataset, converted the categorical features to numeric values, and standardized the numeric values.
The dataset is not balanced as can be seen in Figure 3.

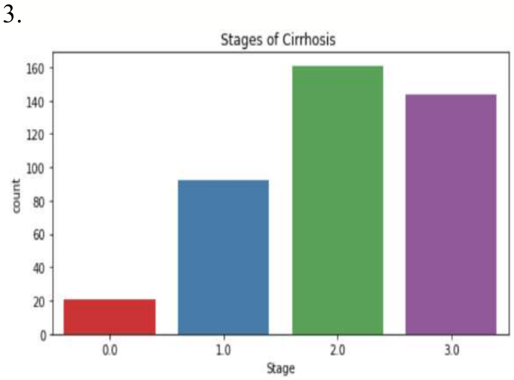

Figure 3. Distribution of the output class (Stage)

We used SMOTE technique to balance the dataset once and the Abunaser algorithm to balance and generate new samples of the dataset. After balancing the dataset, we have spit the dataset to 70% x 15% x15% (Training, Validation, and Testing). We have set the learning rate = 0.001, batch-size = 50, epoch = 100.

After we have trained and validated Deep Neural Network using the architecture used in Figure 4.

```
inputs = tf.keras.Input(shape=(x2.shape[1],))
x = tf.keras.layers.Dense(32, activation='relu')(inputs)
x = tf.keras.layers.Dense(64, activation='relu')(x)
x = tf.keras.layers.Dense(128, activation='relu')(x)
x = tf.keras.layers.Dense(128, activation='relu')(x)
x = tf.keras.layers.Dense(256, activation='relu')(x)
x = tf.keras.layers.Dense(256, activation='relu')(x)
x = tf.keras.layers.Dense(512, activation='relu')(x)
x = tf.keras.layers.Dense(512, activation='relu')(x)
outputs = tf.keras.layers.Dense(4, activation= 'softma
x')(x)
model = tf.keras.Model(inputs, outputs)
model.compile(Adam(lr=0.001), loss='sparse_categor
ical_crossentropy', metrics=['accuracy'])
model.summary()
```

Figure 4: architecture of the proposed DNN for evaluating Smote and Abunaser technique using Cirrhosis Dataset

The Abunaser history of the training and validation accuracy and loss are shown in Figure 5; while the SMOTE history of the training and validation accuracy and loss are shown in Figure 6.

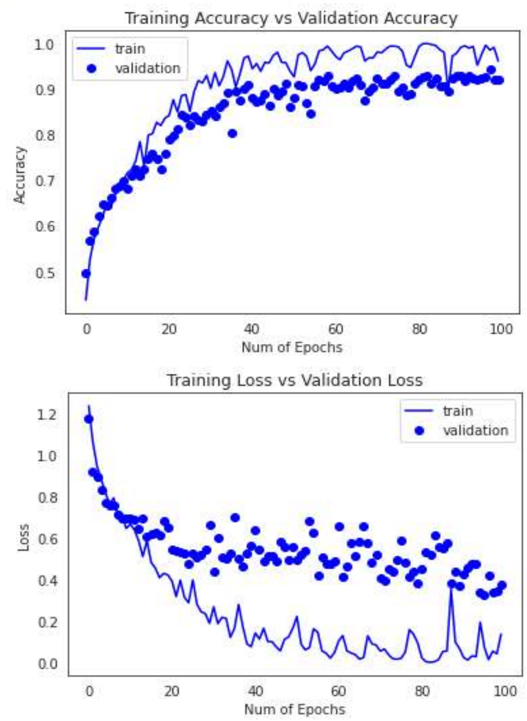

Figure 5 History of training, validation accuracy and loss using Abunaser algorithm

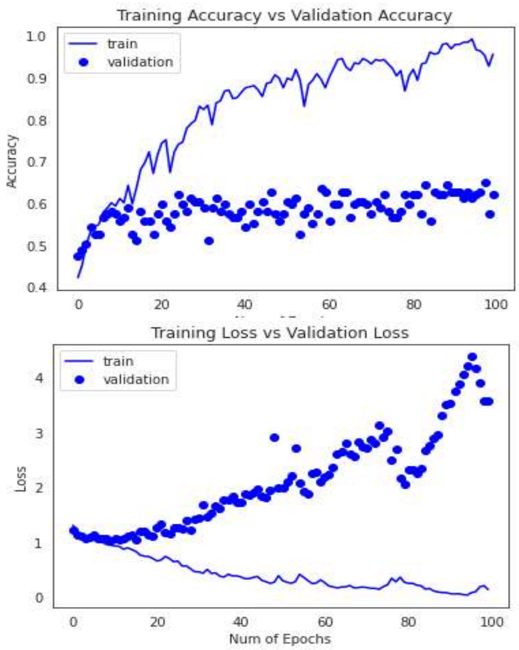

Figure 6: History of training, validation accuracy and loss using SMOTE technique

A comparison between SMOTE and Abunaser algorithms (Seen in Table 2) in terms of accuracy, precision, Recall, F1-Score, and Time need in second. In the experiment we employed nine classical ML and one DL methods: LGBM Classifier, Random Forest Classifier, Extra Tree Classifier, Bagging Classifier, Gradient Boosting Classifier, Decision Tree Classifier, Label Propagation, KNeighbors Classifier, MLP Classifier, and DNN model.

Table 2: A comparison between SMOTE and Abunaser algorithms using Cirrhosis Dataset

ML Model-Name	Accuracy		Precision		Recall		F1_score		Time-in-Sec	
	Abunaser	SMOTE	Abunaser	SMOTE	Abunaser	SMOTE	Abunaser	SMOTE	Abunaser	SMOTE
LGBM Classifier	99.67%	65.89%	99.67%	63.80%	99.67%	65.89%	99.67%	64.16%	0.57	0.20
Random Forest Classifier	95.67%	62.02%	95.76%	60.14%	95.67%	62.02%	95.63%	60.39%	0.34	0.20
Extra Tree Classifier	94.00%	48.06%	94.19%	47.85%	94.00%	48.06%	93.95%	47.79%	0.01	0.02
Bagging Classifier	93.00%	55.81%	93.00%	53.99%	93.00%	55.81%	92.99%	53.61%	0.12	0.06
Gradient Boosting Classifier	91.67%	60.47%	91.65%	59.30%	91.67%	60.47%	91.64%	59.76%	1.71	0.73
Decision Tree Classifier	89.00%	50.39%	89.03%	48.83%	89.00%	50.39%	88.86%	49.41%	0.01	0.01
Label Propagation	85.00%	52.71%	86.13%	51.61%	85.00%	52.71%	84.61%	50.70%	0.19	0.02
KNeighbors Classifier	74.33%	50.39%	74.15%	49.10%	74.33%	50.39%	73.52%	48.29%	0.04	0.01
MLP Classifier	67.67%	51.94%	66.37%	49.22%	67.67%	51.94%	66.23%	49.62%	2.63	0.62
DNN model	94.00%	51.19%	0.9400	54.71%	94.00%	51.16%	93.94%	50.25%	1.09	0.82

6.3The second Experiment (Breast Cancer)

This dataset of breast cancer patients was obtained from the 2017 November update of the SEER Program of the NCI, which provides information on population-based cancer statistics. The dataset involved female patients with infiltrating duct and lobular carcinoma breast cancer (SEER primary cites recode NOS histology codes 8522/3) diagnosed in 2006-2010. Patients with unknown tumor size, examined regional LNs, positive regional LNs, and patients whose survival months were less than 1 month were excluded; thus, 4024 patients were ultimately included [8].

Table3: Descriptions Of The Features Of Breast Cancer Dataset

Features	Description	Input/output
Race:	0 = represent white Race , 1 = represent Black Race and 2 = Represent Other Race	Input
Marital Status:	0 = Married, 1 = Divorced , 2= Single, 3 = Widowed and 4 = Separated	Input
T Stage:	The T refers to the size and extent of the main tumor. The main tumor is usually called the primary tumor.T1, T2, T3, T4: Refers to the size and/or extent of the main tumor. The higher the number after the T, the larger the tumor or the more it has grown into nearby tissues. T-T0: No evidence of primary tumor. T1 (includes T1a, T1b, and T1c): Tumor is 2 cm (3/4 of an inch) or less across. T2: Tumor is more than 2 cm but not more than 5 cm (2 inches) across. T3: Tumor is more than 5 cm across	Input
N Stage:	The main tumor is usually called the primary tumor. The N refers to the number of nearby lymph nodes that have cancer. The M refers to whether the cancer has metastasized. This means that the cancer has spread from the primary tumor to other parts of the body.N1, N2, N3: Refers to the number and location of lymph nodes that contain cancer. The higher the number after the N, the more lymph nodes that contain cancer.	Input
6th Stage:	0 = IIA , 1= IIIA, 2 = IIIC , 3=IIB and 4 = IIIB Stage groups for breast cancer, Doctors assign the stage of the cancer by combining the T, N, and M classifications (see above), the tumor grade, and the results of ER/PR and HER2 testing.	Input
Stage IIA	The tumor is 20 mm or smaller and has spread to 1 to 3 axillary lymph nodes (T1, N1, M0). The tumor is larger than 20 mm but not larger than 50 mm and has not spread to the axillary lymph nodes (T2, N0, M0).	Input
Stage IIB	Either of these conditions: 1. The tumor is larger than 20 mm but not larger than 50 mm and has spread to 1 to 3 axillary lymph nodes (T2, N1, M0). 2. The tumor is larger than 50 mm but has not spread to the axillary lymph nodes (T3, N0, M0).	Input
Stage IIIA	The tumor of any size has spread to 4 to 9 axillary lymph nodes or to internal mammary lymph nodes. It has not spread to other parts of the body (T0, T1, T2, or T3; N2; M0). Stage IIIA may also be a tumor larger than 50 mm that has spread to 1 to 3 axillary lymph nodes (T3, N1, M0).	Input
Stage IIIB	The tumor has spread to the chest wall or caused swelling or ulceration of the breast, or it is diagnosed as inflammatory breast cancer. It may or may not have spread to up to 9 axillary or internal mammary lymph nodes. It has not spread to other parts of the body (T4; N0, N1, or N2; M0).	Input
Stage IIIC	A tumor of any size that has spread to 10 or more axillary lymph nodes, the internal mammary lymph nodes, and/or the lymph nodes under the collarbone. It has not spread to other parts of the body (any T, N3, M0).	Input
differentiate	0 =Poorly differentiated, 1 = Moderately differentiated, 2= Well differentiated and 3 = Undifferentiated	Input
Grade:	Grade 1 looks most like normal breast cells and is usually slow growing Grade 2 looks less like normal cells and is growing faster Grade 3 looks different to normal breast cells and is usually fast	Input

Journal of Theoretical and Applied Information Technology
31st May 2023. Vol.101. No 10

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

	growing	
A Stage:	0 =Regional and 1 = Distant	Input
Estrogen Status:	0 =Estrogen positive and 1 = Estrogen negative	Input
Progesterone Status:	0 = Progesterone positive and 1 = Progesterone negative	Input
Status:	0 = Alive and 1 = dead	Output

We have pre-processed the breast cancer dataset, converted the categorical features to numeric values, and standardized the numeric values.
The dataset is not balanced as can be seen in Figure 7.

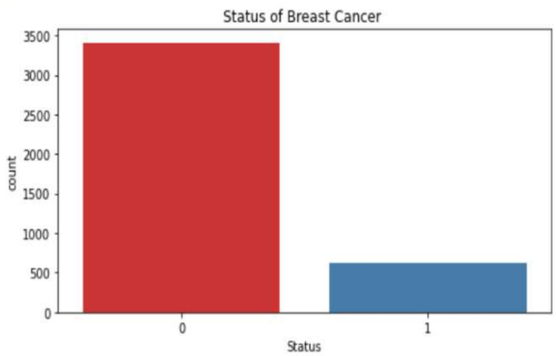

Figure 7: Distribution of the output class (Status)

We used SMOTE algorithm to balance the dataset once and the Abunaser algorithm to balance and generate new samples of the dataset. After balancing the dataset, we have spit the dataset to 70% x 15% x15% (Training, Validation, and Testing). We have set the learning rate = 0.0001, batch-size = 50, epoch = 100. After we have trained and validated Deep Neural Network using the architecture used in Figure 8.

```
from tensorflow.keras.optimizers import Adam
inputs = tf.keras.Input(shape=(x2.shape[1],))
x = tf.keras.layers.Dense(32, activation='relu')(inputs)
x = tf.keras.layers.Dense(64, activation='relu')(x)
x = tf.keras.layers.Dense(64, activation='relu')(x)
x = tf.keras.layers.Dense(128, activation='relu')(x)
x = tf.keras.layers.Dense(128, activation='relu')(x)
x = tf.keras.layers.Dense(256, activation='relu')(x)
x = tf.keras.layers.Dense(256, activation='relu')(x)

outputs = tf.keras.layers.Dense(2, activation= 'softmax')(x)
model = tf.keras.Model(inputs, outputs)
model.compile(Adam(lr=0.0001), loss='sparse_categorical
_crossentropy', metrics=['accuracy'])
model.summary()
```

Figure 8: Architecture of DNN used in Breast Cancer Dataset

The Abunaser history of the training and validation accuracy and loss are shown in Figure 9; while the SMOTE history of the training and validation accuracy and loss are shown in Figure 10.

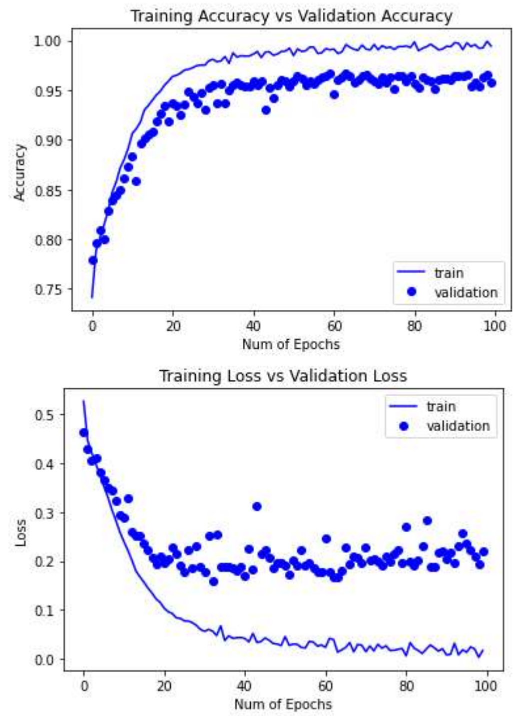

Figure 9: History breast cancer training, validation accuracy and loss using Abunaser algorithm

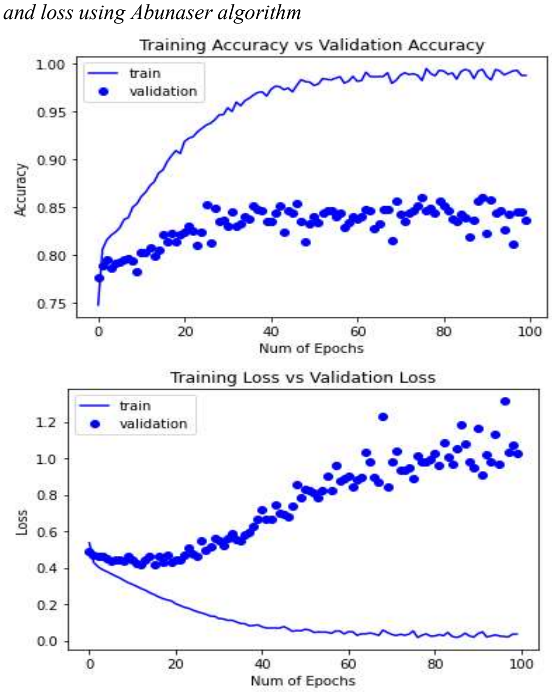

Figure 10: History Of Breast Cancer Training, Validation Accuracy And Loss Using SMOTE Technique

Journal of Theoretical and Applied Information Technology
31st May 2023. Vol.101. No 10

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A comparison between SMOTE and Abunaser algorithms (Seen in Table 4) in terms of accuracy, precision, Recall, F1-Score, and Time need in second. In the experiment we employed nine classical ML and one DL methods: LGBM Classifier, Random Forest Classifier, Extra Tree Classifier, Bagging Classifier, Gradient Boosting Classifier, Decision Tree Classifier, Label Propagation, KNeighbors Classifier, MLP Classifier, and DNN model.

Table 4: A Comparison Between SMOTE And Abunaser Algorithms Using Breast Cancer Dataset

ML Model-Name	Accuracy		Precision		Recall		F1_score		Time-in-Sec	
	Abunaser	**SMOTE**	**Abunaser**	**SMOTE**	**Abunaser**	**SMOTE**	**Abunaser**	**SMOTE**	**Abunaser**	**SMOTE**
Random Forest Classifier	99.22%	90.91%	99.23%	90.94%	99.22%	90.91%	99.22%	90.91%	0.87	0.59
Bagging Classifier	98.67%	88.27%	98.70%	88.41%	98.67%	88.27%	98.67%	88.26%	0.28	0.20
Decision Tree Classifier	96.83%	83.48%	96.92%	83.59%	96.83%	83.48%	96.83%	83.46%	0.04	0.03
Extra Tree Classifier	95.94%	84.16%	95.99%	84.17%	95.94%	84.16%	95.94%	84.17%	0.03	0.02
LGBM Classifier	94.22%	88.76%	94.22%	88.89%	94.22%	88.76%	94.22%	88.75%	0.19	0.17
Label Propagation	88.56%	83.58%	88.58%	84.13%	88.56%	83.58%	88.55%	83.52%	3.37	1.12
KNeighbors Classifier	83.94%	84.16%	84.36%	84.17%	83.94%	84.16%	83.89%	84.17%	0.21	0.10
Gradient Boosting Classifier	83.83%	85.63%	84.04%	86.00%	83.83%	85.63%	83.81%	85.60%	1.16	0.69
MLP Classifier	82.17%	83.19%	82.32%	83.58%	82.17%	83.19%	82.15%	83.15%	12.54	7.47
DNN model	95.28%	83.97%	95.29%	84.01%	95.28%	83.97%	95.28%	83.97%	2.19	2.07

3.4 The Third Experiment (Boston Housing)

We will be attempting to predict the median price of homes in a given Boston suburb in the mid-1970s, given a few data points about the suburb at the time, such as the crime rate, the local property tax rate, etc. The dataset has very few data points, only 506 in total, split between 404 training samples and 102 test samples, and each "feature" in the input data (e.g. the crime rate is a feature) has a different scale. For instance some values are proportions, which take a values between 0 and 1, others take values between 1 and 12, others between 0 and 100 [9].

As you can see, we have 404 training samples and 102 test samples. The data comprises 13 features. The 13 features in the input data are as in Table 5.

Table5: Descriptions Of The Features Of Boston Housing Dataset

Feature	Description	Input/output
F1	Per capita crime rate.	Input
F2	Proportion of residential land zoned for lots over 25,000 square feet.	Input
F3	Proportion of non-retail business acres per town.	Input
F4	Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).	Input
F5	Nitric oxides concentration (parts per 10 million).	Input
F6	Average number of rooms per dwelling.	Input
F7	Proportion of owner-occupied units built prior to 1940.	Input
F8	Weighted distances to five Boston employment centers.	Input
F9	Index of accessibility to radial highways.	Input
F10	Full-value property-tax rate per $10,000.	Input
F11	Pupil-teacher ratio by town.	Input
F12	1000 * (Bk - 0.63) ** 2 where Bk is the proportion of Black people by town	Input
F13	% lower status of the population	Input
Price	Target	Output

The targets are the median values of owner-occupied homes, in thousands of dollars: The prices are typically between $10,000 and $50,000. If that sounds cheap, remember this was the mid-1970s, and these prices are not inflation-adjusted.

After the Generation of the new samples using Abunaser algorithm, we have split the dataset to 70% x 15% x15% (Training, Validation, and Testing). We have set the learning rate = 0.0001, batch-size = 16, epoch = 500. After we have trained and validated Deep Neural Network using the architecture used in Figure 11.

```
from keras import models
from keras import layers
def build_model():
   model = models.Sequential()
   model.add(layers.Dense(64, activation='relu',
                      input_shape=(X_train.shape[1],)))
   model.add(layers.Dense(128, activation='relu'))
   model.add(layers.Dense(256, activation='relu'))
   model.add(layers.Dense(1))
   model.compile(optimizer='rmsprop', loss='mse', metrics
=['mae'])
   return model
```

Figure 11: Architecture Of DNN Used In Bosting Housing Dataset

A comparison between Abunaser algorithm and without Abunaser (Seen in Table 6) in terms R^2-score, MAE, and Root Mean Squared Error. In the experiment we employed six classical ML regression and one DL method: Linear regression, Decision tree regression, Random forest regression, Ridge regression, Lasso regression, Polynomial regression, and DNN model.

Table 6: A Comparison Between Abunaser Algorithm And Without Abunaser Algorithm In Boston Housing Dataset

Regression	**R^2-score**		**MAE**		**Root Mean Squared Error**	
	Abunaser	**Without Abunaser**	**Abunaser**	**Without Abunaser**	**Abunaser**	**Without Abunaser**
Linear regression	0.7638	0.6464	2.9154	3.3356	4.0165	5.1257
Decision tree regression	0.7630	0.6535	2.9005	3.2393	4.0227	5.0734
Random forest regression	0.7630	0.6535	2.9005	3.2393	4.0227	5.0734
Ridge regression	0.7630	0.6535	2.9005	3.2393	4.0227	5.0734
Lasso regression	0.2613	0.2480	5.0336	5.2108	7.2926	7.4746
Polynomial regression	0.7630	0.6535	2.9005	4.6085	4.0227	5.1257
DNN model	0.9050	0.8476	1.8553	2.49507	2.5025	3.3645

7. RESULTS AND DISCUSSION

We have carried out 3 experiments two of which of type classification problems and the third is regression problem. In first two experiments we used Abunaser algorithm for generating new data one and the other using the classical SMOTE technique. During the first two experiments we employed 9 machine learning algorithms and one deep neural network algorithm for the testing and comparing the results of using Abunaser and SMOTE algorithms. The machine learning algorithms used for testing were: LGBM Classifier, Random Forest Classifier, Extra Tree Classifier, Bagging Classifier, Gradient Boosting Classifier, Decision Tree Classifier, Label Propagation, KNeighbors Classifier, MLP Classifier. The measure we used includes: Accuracy, Recall, Precision, F1-score and time performance. From Table 2 and Table 4, Abunaser algorithm output perform the SMOTE algorithm.

In the third experiment the dataset was of type regression. The SMOTE does not work with regression; however, Abunaser algorithm works fine with regression, we you can generate new data easily. In this experiment we used six regissors: Linear Regressor, Random Forest Regressor, Decision Tree Regressor, Polynomial Regressor, and Ridge Regressor, and Lasso Regressor. All the regressors have been implemented using the python Scikit-learn library. Since it is a regression problem, we have used Mean Squared Error (MSE) as the loss function. We have also made a comparison using the R^2-squared metric, which is a statistical measure of how good the data is fitted to the regression model. The R^2-squared value lies between 0 and 1. Table 6 corresponds to the statistics when the model is trained without incorporating Abunaser samples once and once with Abunaser algorithm for generating new samples. In total, 2000 examples were considered for training and 300 for testing. Testing Mean Absolute Error (MAE), Root Mean Squared Error, and R^2-squared are shown for each of the 6 regressors. Again, using Abunaser for generating new examples improved the results.

8. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a methodology of augmenting the data of CSV files called Abunaser algorithm. For experimentation purposes, we have used only three datasets and evaluated the performance of our algorithm (Abunaser) using 10 different ML and DNN algorithms. Results demonstrated that Abunaser algorithm is performing better for all the 3 datasets when trained on augmented data generated by Abunaser. So, for our future work, we can evaluate our algorithm on different datasets, where data is structured in CSV files, like in this case.

ACKNOWLEDGEMENT.

This work was developed in the Labs of Google (Goole colab), were Google provided us with hardware and Software with GPU power and high RAM and high Hard disk storage.

REFERENCES

[1]. Burdescu, D., Mihai, G., Stanescu, L., Brezovan, M.: Automatic image annotation and semantic based image retrieval for medical domain. Neurocomputing 109, 33–48 (2016). https://doi.org/10.1016/j.neucom.2012.07.030
[2]. Zhong, Z., Zheng, L., Kang, G., Li, S., Yang, Y.: Random erasing data augmentation. arXiv preprint arXiv:1708.04896 (2017)
[3]. Bowles, C., Chen, L., Guerrero, R., Bentley, P., Gunn, R., Hammers, A., Dickie, D.A., Hernández, M.V., Wardlaw, J., Rueckert, D.: GAN augmentation: augmenting training data using generative adversarial networks. arXiv preprint arXiv:1810.10863 (2018).
[4]. Frid-Adar, M., Diamant, I., Klang, E., Amitai, M., Goldberger, J., Greenspan, H.: GANbased synthetic medical image augmentation for increased CNN performance in liver lesion classification. Neurocomputing 321, pp. 321–331 (2018).
[5]. Mariani, G., Scheidegger, F., Istrate, R., Bekas, C., Malossi, C.: BAGAN: data augmentation with balancing GAN. arXiv preprint arXiv:1803.09655 (2018)
[1]. Nguyen, P., Nguyen, K., Ichise, R., Takeda, H.: EmbNum+: effective, efficient, and robust semantic labeling for numerical values. New Gener. Comput. 37(4), 393–427 (2019)
[2]. https://www.kaggle.com/datasets/fedesoriano/cirrhosis-prediction-dataset
[3]. https://www.kaggle.com/code/amitmittalamit140/boston-house-price-prediction/data
[4]. https://www.kaggle.com/datasets/reihanenamdari/breast-cancer
[5]. Zhun Z, Liang Z, Guoliang K, Shaozi L, Yi Y. Random erasing data augmentation. ArXiv e-prints. 2017.
[6]. Terrance V, Graham WT. Improved regularization of convolutional neural networks with cutout. arXiv preprint. 2017.
[7]. Agnieszka M, Michal G. Data augmentation for improving deep learning in image classification problem. In: IEEE 2018 international interdisciplinary Ph.D. Workshop, 2018.
[8]. Jonathan K, Michael S, Jia D, Li F-F. 3D object representations for fine-grained categorization. In: 4th IEEE Workshop on 3D Representation and Recognition, at ICCV 2013 (3dRR-13). Sydney, Australia. Dec. 8, 2013.
[9]. Alrakhawi, H. A., Jamiat, N., Abu-Naser, S. S. Intelligent Tutoring Systems in Education: A Systematic Review of Usage, Tools, Effects and Evaluation. Journal of Theoretical and Applied Information Technology, 2023, Vol. 101. No. 4.
[10]. Zarandah, Q. M. M., Daud, S. M., Abu-Naser, S. S. A Systematic Literature Review Of Machine and Deep Learning-Based Detection And Classification Methods for Diseases Related To the Respiratory System, Journal of Theoretical and Applied Information Technology, 2023, Vol. 101. No. 4.
[11]. Alkayyali, Z. K. D., Idris, S. A. B, Abu-Naser, S. S. A Systematic Literature Review of Deep and Machine Learning Algorithms in Cardiovascular Diseases Diagnosis, Journal of Theoretical and Applied Information Technology, 2023, Vol. 101. No. 4.
[12]. Abunasser, B. S. Daud, S. M., Zaqout, I., Abu-Naser S. S. Convolution Neural Network For Breast Cancer Detection And

Classification - Final Results. Journal of Theoretical and Applied Information Technology, 2023, Vol. 101. No. 1, pp. 315-329.

[13]. Taha, A. M. H., Ariffin, D. S. B. B., Abu-Naser, S. S. A Systematic Literature Review of Deep and Machine Learning Algorithms in Brain Tumor and Meta-Analysis, Journal of Theoretical and Applied Information Technology, 2023, Vol. 101. No. 1, pp. 21-36.

[14]. Abu Ghosh, M.M., Atallah, R.R., Abu Naser, S.S. Secure mobile cloud computing for sensitive data: Teacher services for palestinian higher education institutions. International Journal of Grid and Distributed Computing, 2016, vol. 9, no. 2, pp. 17–22

[15]. Abu Naser, S.S. Evaluating the effectiveness of the CPP-Tutor, an intelligent tutoring system for students learning to program in C++. Journal of Applied Sciences Research, vol. 5, no. 1, pp. 109-114, 2009.

[16]. Abu-Naser, S.S., El-Hissi H., Abu-Rass, M., & El-khozondar, N. An expert system for endocrine diagnosis and treatments using JESS. Journal of Artificial Intelligence; vol. 3, no. 4, pp. 239-251, 2010.

[17]. Abunasser, B. S., AL-Hiealy, M. R. J., Zaqout, I. S. and Abu-Naser, S. S. "Breast Cancer Detection and Classification using Deep Learning Xception Algorithm" International Journal of Advanced Computer Science and Applications(IJACSA), 13(7),223-228, 2022. http://dx.doi.org/10.14569/IJACSA.2022.0130729

[18]. Abunasser, B.S., AL-Hiealy, M.R. J., Barhoom, A. M. Almasri A. R. and Abu-Naser, S. S. "Prediction of Instructor Performance using Machine and Deep Learning Techniques" International Journal of Advanced Computer Science and Applications(IJACSA), 13(7), 78-83, 2022. http://dx.doi.org/10.14569/IJACSA.2022.0130711

[19]. Alayoubi, M.M., Arekat, Z.M., Al Shobaki, M.J., Abu-Naser, S.S. The Impact of Work Stress on Job Performance Among Nursing Staff in Al-Awda Hospital. Foundations of Management, 2022, 14(1), pp. 87–108

[20]. Albatish, I.M., Abu-Naser, S.S. Modeling and controlling smart traffic light system using a rule based system. Proceedings - 2019 International Conference on Promising Electronic Technologies, ICPET 2019, 2019, pp. 55–60, 8925318

[21]. Almasri, A., Obaid, T., Abumandil, M.S.S., ...Mahmoud, A.Y., Abu-Naser, S.S. Mining Educational Data to Improve Teachers' Performance. Lecture Notes in Networks and Systems, 2023, 550 LNNS, pp. 243–255

[22]. Almasri, A.R., Yahaya, N.A., Abu-Naser, S.S. Instructor Performance Modeling For Predicting Student Satisfaction Using Machine Learning - Preliminary Results. Journal of Theoretical and Applied Information Technology, 2022, 100(19), pp. 5481–5496

[23]. Alsharif, F. Safi S., AbouFoul T., Abu Nasr, M., Abu Nasser S. Mechanical Reconfigurable Microstrip Antenna. International Journal of Microwave and Optical Technology, vol. 11, no. 3, pp.153-160, 2016.

[24]. Arqawi, S., Atieh, K.A.F.T., Shobaki, M.J.A.L., Abu-Naser, S.S., Abu Abdulla, A.A.M. Integration of the dimensions of computerized health information systems and their role in improving administrative performance in Al-Shifa medical complex, Journal of Theoretical and Applied Information Technologythis link is disabled, 2020, vol. 98, no. 6, pp. 1087–1119

[25]. Arqawi, S.M., Abu Rumman, M.A., Zitawi, E.A., ...Abunasser, B.S., Abu-Naser, S.S. Predicting Employee Attrition And Performance Using Deep Learning. Journal of Theoretical and Applied Information Technology, 2022, 100(21), pp. 6526–6536

[26]. Arqawi, S.M., Zitawi, E.A., Rabaya, A.H., Abunasser, B.S., Abu-Naser, S.S., "Predicting University Student Retention using Artificial Intelligence", International Journal of Advanced Computer Science and Applications , 2022, vol. 13, no. 9, pp. 315–324

[27]. Barhoom, A.M.A., Al-Hiealy, M.R.J., Abu-Naser, S.S. Bone Abnormalities Detection and Classification Using Deep Learning-VGG16 Algorithm. Journal of Theoretical and Applied Information Technology, 2022, 100(20), pp. 6173–6184

[28]. Barhoom, A.M.A., Al-Hiealy, M.R.J., Abu-Naser, S.S. Deep Learning-Xception Algorithm for Upper Bone Abnormalities Classification. Journal of Theoretical and Applied Information Technology, 2022, 100(23), pp. 6986–6997

[29]. Buhisi, N. I., & Abu-Naser, S. S. Dynamic programming as a tool of decision supporting. Journal of Applied Sciences Research. Vo. 5, no. 6, pp. 671-676, 2009.
[30]. El-Habil, B.Y., Abu-Naser, S.S. Global Climate Prediction Using Deep Learning. Journal of Theoretical and Applied Information Technology, 2022, 100(24), pp. 4824–4838
[31]. Elzamly, A., Hussin, B., Naser, S.A., ...Selamat, A., Rashed, A. A new conceptual framework modelling for cloud computing risk management in banking organizations. International Journal of Grid and Distributed Computing, 2016, vol. 9, no. 9, pp. 137–154
[32]. Elzamly, A., Messabia, N., Doheir, M., ...Al-Aqqad, M., Alazzam, M. Assessment risks for managing software planning processes in information technology systems. International Journal of Advanced Science and Technology, 2019, vol. 28, no. 1, pp. 327–338
[33]. Eneizan, B., Obaid, T., Abumandil, M.S.S., ...Arif, K., Abulehia, A.F.S. Acceptance of Mobile Banking in the Era of COVID-19. Lecture Notes in Networks and Systems, 2023, 550 LNNS, pp. 29–42
[34]. Alzamily, J. Y. I., Ariffin, S. B., Abu-Naser, S. S. Classification of Encrypted Images Using Deep Learning –Resnet50. Journal of Theoretical and Applied Information Technology, 2022, 100(21), pp. 6610–6620
[35]. Mady, S.A., Arqawi, S.M., Al Shobaki, M.J., Abu-Naser, S.S. Lean manufacturing dimensions and its relationship in promoting the improvement of production processes in industrial companies. International Journal on Emerging Technologies, 2020, vol. 11, no. 3, pp. 881–896
[36]. Naser, S. S. A. Developing an intelligent tutoring system for students learning to program in C++. Information Technology Journal, vol. 7, no. 7, pp. 1051-1060, 2008.
[37]. Naser, S. S. A. Developing visualization tool for teaching AI searching algorithms. Information Technology Journal, vol. 7, no. 2, pp. 350-355, 2008.
[38]. Naser, S. S. A. Intelligent tutoring system for teaching database to sophomore students in Gaza and its effect on their performance. Information Technology Journal, vol. 5, no. 5, pp. 916-922, 2006.
[39]. Naser, S. S. A. JEE-Tutor: An intelligent tutoring system for java expressions evaluation. Information Technology Journal, vol. 7, no. 3, pp. 528-532, 2008.
[40]. Obaid, T., Eneizan, B., Naser, S.S.A., ...Abualrejal, H.M.E., Gazem, N.A. Factors Contributing to an Effective E- Government Adoption in Palestine. Lecture Notes on Data Engineering and Communications Technologies, 2022, 127, pp. 663–676
[41]. Obaid, T., Eneizan, B., Abumandil, M.S.S., ...Abu-Naser, S.S., Ali, A.A.A. Factors Affecting Students' Adoption of E-Learning Systems During COVID-19 Pandemic: A Structural Equation Modeling Approach. Lecture Notes in Networks and Systems, 2023, 550 LNNS, pp. 227–242
[42]. Saleh, A., Sukaik, R., Abu-Naser, S.S. Brain tumor classification using deep learning. Proceedings - 2020 International Conference on Assistive and Rehabilitation Technologies, iCareTech 2020, 2020, pp. 131–136, 9328072
[43]. Abunasser, B., AL-Hiealy, M. R., Zaqout, I., Abu-Naser, S. Convolution Neural Network for Breast Cancer Detection and Classification Using Deep Learning. Asian Pacific Journal of Cancer Prevention, 2023; vol. 24, no. 2, pp. 531-544. doi: 10.31557/APJCP.2023.24.2.531