iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** V **Month of publication:** May 2023

DOI: https://doi.org/10.22214/ijraset.2023.51592

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue V May 2023- Available at www.ijraset.com

Detection and Classification of Leukemia And Myeloma Using Soft Computing Techniques

Kriti Shrivastav[1], Clenzila De Sousa[2], Nidhi Desai[3], Pratibha Patil[4], Nausheen Sayyed[5], Manisha Fal Dessai[6]
[1, 2, 3, 4, 5, 6]*Don Bosco College of Engineering Goa, India*

Abstract: ***Blood cancer is a type of cancer that affects the blood cells. Medical image processing technology is essential in both early disease identification and cancer cell analysis. Blood cancer can impact the lymph nodes, bone marrow, blood cells, lymph nodes, and other lymphatic system components. A primary cause of blood cancer is an unusual and excessive amount of white blood cellular proliferation. Traditional cancer cell detection is time-consuming and inaccurate to a large extent, hence an automated approach based on soft computing techniques is presented to predict cancer cell presence and identify two types of blood cancer which are leukemia and myeloma. The dataset for Myeloma is acquired from TCIA (The Cancer Imaging Archive) repository and the Leukemia dataset comes from Kaggle-Blood Cell Images, both of which are available to the public. The datasets are already pre-processed. In our study, we have compared different hybrid models like DenseNet with XGBoost, InceptionResNet with SVM, etc from which the combination of VGG-19 for feature selection and SVM for classification gives the best performance. We have achieved Classification Accuracy of 96.4%(0.964), Precision(0.964), F1 Score(0.964) and Recall(0.964) for SVM.***

Keywords: Data augmentation, classification algorithms, Feature selection, Train and test ratio, hybrid models

I. INTRODUCTION

The majority of blood cancers, also known as hematologic cancers, begin in the bone marrow, where blood is produced. Blood cancers arise when abnormal blood cells begin to proliferate uncontrollably, interfering with the function of normal blood cells, which is to fight off infection and produce new blood cells.

Blood cancer can be difficult to identify. Cancer diagnosis usually begins with a physical examination in which a doctor reviews your medical history and examines your lymph nodes. Tests depend on the type of blood cancer suspected. E.g. biopsy, imaging scans, and blood tests.

The major types of blood cancer include Leukemia, Lymphoma, Myeloma, etc. which are further divided into subtypes. Leukemia is a condition that is caused by an increase in the number of white blood cells in your body, which interferes with the ability of your bone marrow to produce red blood cells.

Although the exact cause is unknown, a combination of genetic and environmental factors are thought to be involved. According to the Leukemia & Lymphoma Society[1], Leukemia accounts for 26.1 percent of all cancer-related deaths among children below 20 years of age.

Myeloma or multiple myeloma is a cancer of the plasma cells. It is the most common type of plasma cell tumor, which develops in the bone marrow and spreads throughout the body. An estimated 138,415 people in the United States (US) are living with or in remission from myeloma.

In our paper, we aim to detect two types of blood cancer, leukemia, and myeloma. We utilize a hybrid model by combining various feature selection algorithms and classifying algorithms.

II. METHODOLOGY

The system will work in 4 stages:

1) Obtaining dataset
2) Augmentation and balancing dataset
3) Feature Extraction
4) Classification Algorithm

The figure below shows the overview of the methodology being used in this paper.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue V May 2023- Available at www.ijraset.com

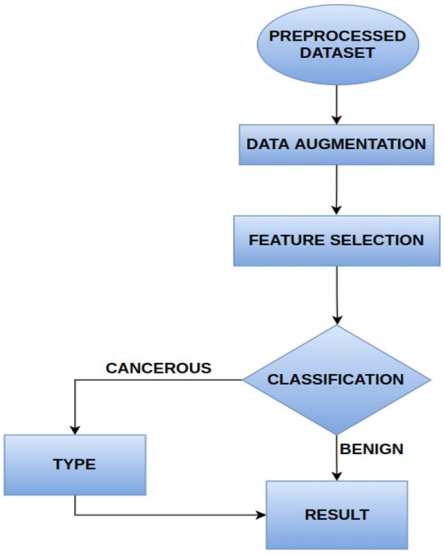


Fig. 1. Methodology being used in the paper.

A. Obtaining Dataset

This paper uses two types of datasets obtained from Kaggle i.e. Leukemia(Blood Cell Images) and multiple myeloma(TCIA), (website) [12], [13]. The obtained microscopic images were captured from bone marrow aspirate slides of patients diagnosed with Multiple Myeloma (MM), a type of white blood cancer. All images are in BMP format with 24-bit colour depth and a resolution of 450 x 450 Pixels. The Training Dataset consists of a total of 298 images. The validation dataset consists of a total of 200 images. The Test Dataset consists of a total of 277 images. The Leukemia dataset consists of 2478 train images and 620 test images. Lastly, the Normal dataset consists of 2483 images. The datasets are not balanced, hence they are balanced using data augmentation.



(a) (b)

Fig. 2. Sample Images (a) Leukemia (b) Myeloma

B. Augmentation and balancing dataset

Augmentation is increasing the size of the image dataset by rotation, resizing, or applying some other random (but realistic) image processing transformations in order to increase the diversity of the training set. Having a balanced dataset for a model would generate higher accuracy and bias towards one class will be avoided. Making use of augmentation, each class in the training set is balanced with 1500 images each. The model is then trained using different ratios of train and test datasets i.e.75/25, 60/40, and 50/50

TABLE I. The total of 2000 samples was divided into train and test data.

Data split	train	test
75/25	1500	500
60/40	1200	800
50/50	1000	1000

©IJRASET: All Rights are Reserved

C. Feature Extraction

The main issue when dealing with image datasets is a large number of attributes, most of which are not used for the training of the model. The data needs to be processed specially for dimensionality reduction and feature extraction so that we only work with data that will give us precise results. Failing to do so will only waste computing power and training time because there are few schemes of representation and a particular image has a lot of variations. When processing of the dataset is done by the algorithm, a lot of useless computation might be done unless we provide the necessary features. Therefore, by doing feature extraction, the dataset will be condensed to its bare minimum of essential variables or dimensions. The feature extraction approach refers to the extraction of meaningful features from the images in the dataset. The goal of feature extraction and selection approaches is to extract the most important information from the source data and express it in a space with reduced dimensionality. The result of this process, which starts with a set of data, are values that are more informative and non-redundant. It will be advantageous to reduce the dimensions of images and turn them into a set of necessary features. Reduced dimensionality results in less redundant and more accurate data. Many feature selection algorithms were used to extract the features. These models can be adjusted and utilized for prediction.

D. Algorithms Used For Feature Extraction Are

1) VGG-16

VGG - Visual Geometry Group, a multi-layered deep Convolutional Neural Network (CNN) architecture. The number refers to how deep the layers are, with VGG-16 or VGG-19 having 16 or 19 convolutional layers, respectively.

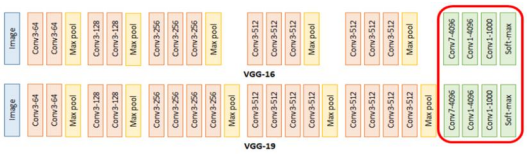


Fig. 3. Architecture of VGG16 and VGG19

Vgg16 has 144 million parameters and has 16 convolutional layers with very small receptive fields (3x3), five max-pooling layers of size 2x2 for spatial pooling, three fully connected layers, and a soft-max layer. All hidden layers are activated by ReLU. Dropout regularization is also used in the fully connected layers of the model. Vgg16 was trained using over a million photos from the ImageNet collection. The network can categorize photos into 1000 different object types, such as keyboards, mouse, and pencils. The Vgg16 model requires a 224*224*3 input picture (RGB image).

2) VGG19

Vgg19 is a CNN that has been trained on millions of images from the ImageNet database. The network can categorize images into 1000 different object types. The Vgg19 model requires a 224*224*3 input picture (RGB image). Vgg19 has 19 deep neural network layers. The Vgg19 network carries more weight (138M weights and 15.5M MACs). Figure 4 displays a schematic of the Vgg16 and Vgg19 architecture trained on the ImageNet database[3].

3) DenseNet201

DenseNet-201 is a 201-layer convolutional neural network. You may load a trained version of the network from the ImageNet database. DenseNet is a type of classic network. This image depicts a 5-layer dense block with a k = 4 growth rate and the conventional ResNet structure[4].

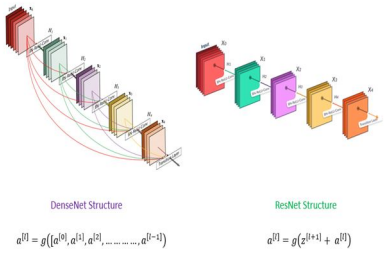


Fig 4. DenseNet Structure and ResNet Structure

Using the composite function operation, an output from the previous layer serves as an input to the second layer. The convolution layer, pooling layer, batch normalization, and non-linear activation layer are all part of this composite operation.

4) InceptionResNetV2

The Inception-ResNet-v2 convolutional neural network was trained on over a million images from the ImageNet collection. The 164-layer network can categorize images into 1000 object categories, including the keyboard, mouse, pencil, and many animals. As a result, the network has learned detailed feature representations for a diverse set of images. The network takes a 299-by-299 picture as input and returns a list of estimated class probabilities as output.

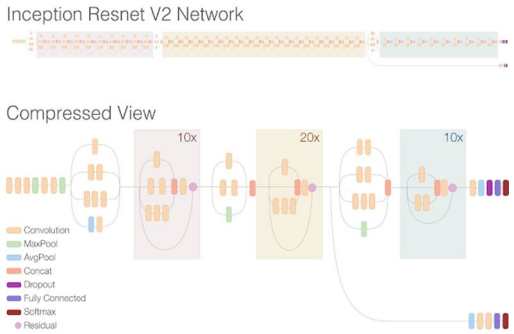


Fig 5. Inception ResnetV2 Network

It is formed by combining the Inception structure and the Residual connection. Multiple-sized convolutional filters are mixed with residual connections in the Inception-Resnet block. The introduction of residual connections not only solves the degradation issue caused by deep structures, but it also shortens the training time. Figure 7 depicts the fundamental network architecture of Inception-Resnet-v2 [5].

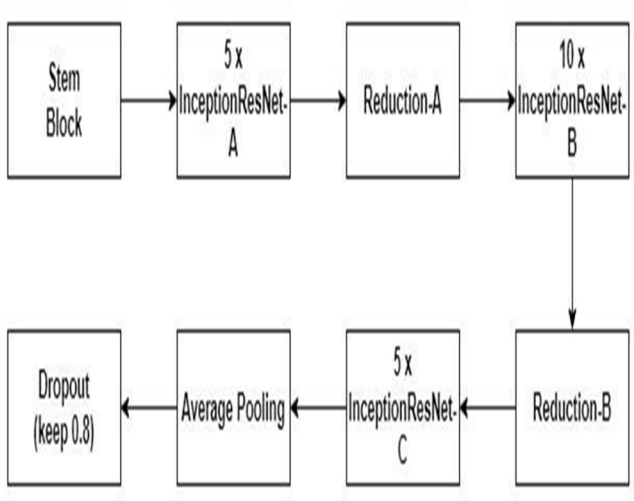


Fig 6. Architecture of Inception-Resnet-V2

D. Classification Algorithm

Three classifier algorithms were applied to the data. Those three classifiers are as follows:

1) XGBoost

XGBoost(Extreme Gradient Boosting) is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. It provides a machine learning library for regression and classification problems. XGBoost is a method where new models are added to predict and correct the errors made by existing models, then the final prediction is made by adding the models together. While adding models in order to minimize the loss it used a gradient descent algorithm. [6]

The model is trained iteratively by predicting errors of the prior tree. To make the final prediction, the prior trees are then combined with the existing trees.

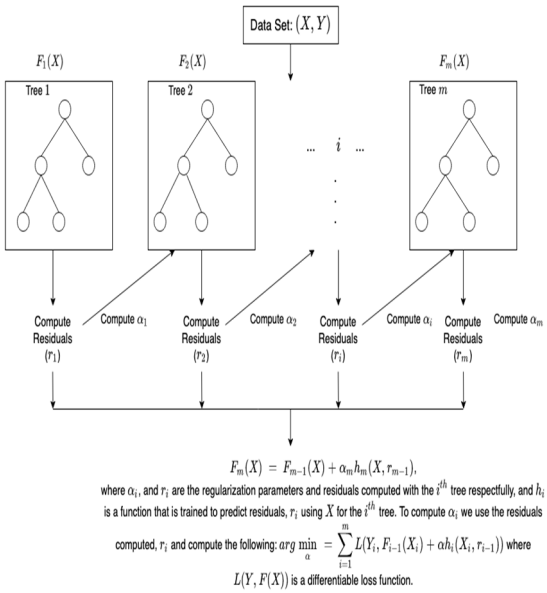


Fig. 7. Working of XGBoost

2) SVM

Support vector machines (SVMs) are supervised learning techniques that are employed in applications such as classification, regression, and outlier detection. The preprocessed dataset is utilized to train the model, and the SVM technique is utilized to categorize the images. SVM works well in cases when there are more features than available data points. Its decision function is memory-efficient because it uses a subset of training points known as support vectors. Recent research demonstrates that SVM can perform better in terms of accuracy while solving classification challenges.[7]

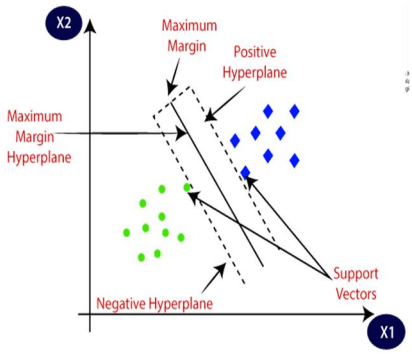


Fig 8. SVM working

3) Decision tree

Decision Tree is a Supervised learning method. It is simple to predict the outcome for upcoming records using the tree model created from historical data. Both classification and regression are handled by the Decision tree. Each node represents a feature (attribute) and each leaf node represents an outcome. The main advantage of using a decision tree in machine learning is its simplicity.

How it works [11]. A model needs to comprehend the characteristics that classify a data point into the various class labels in order to solve a classification problem. The entire dataset is divided into smaller subsets before the classification tree is incrementally created. When categorical or discrete target variables are involved, branching often takes place by binary partitioning.

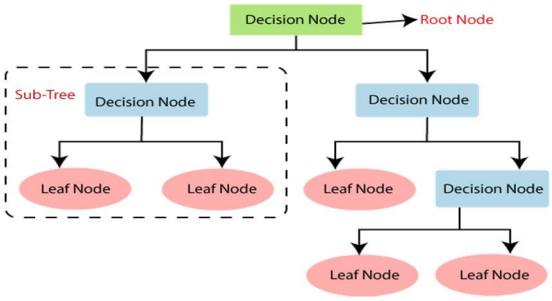


Fig. 9. Decision tree

III. EXPERIMENT

These classification algorithms were tested out with different CNN architectures, which were used for feature selection. These combinations are:

1) XGBoost and VGG-16
2) XGBoost and VGG-19
3) XGBoost and DenseNet201
4) SVM and VGG-16
5) SVM and VGG-19
6) SVM and InceptionResNetV2
7) Decision Tree and InceptionResNet

These classifier algorithms were applied to preprocessed data. Out of all the combinations mentioned above, SVM with VGG19 showed the best performance in terms of classification accuracy, precision, recall and F1 score.

TABLE II. Performance comparison table of all the algorithms and also the data splits

Metrics		Accuracy	Precision	Recall	F1 Score
Algorithm	DataSplit				
VGG16 + XGBoost	75 - 25	0.8606	0.8606	0.8606	0.8606
	60 - 40	0.8987	0.8987	0.8987	0.8987
	50 - 50	0.9016	0.9016	0.9016	0.9016
VGG19 + XGBoost	75 - 25	0.9	0.9	0.9	0.9
	60 - 40	0.9195	0.9195	0.9195	0.9195
	50 - 50	0.926	0.926	0.926	0.926
VGG16 + SVM	75 - 25	0.8446	0.8446	0.8446	0.8446
	60 - 40	0.9004	0.9004	0.90	0.900

Metrics		Accuracy	Precision	Recall	F1 Score
				04	4
	50 - 50	0.92	0.92	0.92	0.92
VGG19 + SVM	75 - 25	0.928	0.928	0.928	0.928
	60 - 40	0.9554	0.9554	0.9554	0.9554
	50 - 50	0.964	0.964	0.964	0.964
Inception ResNet + SVM	75 - 25	0.8313	0.8313	0.8313	0.8313
	60 - 40	0.8858	0.8858	0.8858	0.8858
	50 - 50	0.898	0.898	0.898	0.898
Densenet + XGBoost	75 - 25	0.9366	0.9366	0.9366	0.9366
	60 - 40	0.9358	0.9358	0.9358	0.9358
	50 - 50	0.926	0.926	0.926	0.926
Inception ResNet + Decision Tree	75 - 25	0.7453	0.7453	0.7453	0.7453
	60 - 40	0.755	0.755	0.755	0.755
	50 - 50	0.7626	0.7626	0.7626	0.7626

The results of these algorithms are given in the table below which consists of four parameters i.e. accuracy, precision, f1score, and recall. These values are obtained for all the combinations as well as for all the data splits. We can see that when combined with VGG-19, SVM gives the best accuracy (96.4%) for a 50/50 data split.
Below are the confusion matrices obtained for each algorithm that gives the highest accuracy:

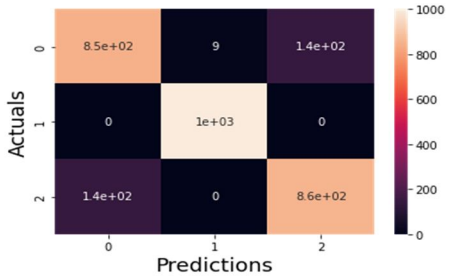


Fig. 10. Confusion Matrix for VGG-16 & XGBoost (50/50 split)

The figure above shows the confusion matrix for 50/50 data split. The accuracy calculated was 90.16%.

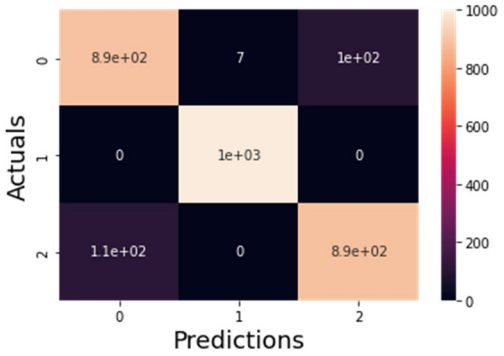


Fig. 11. Confusion Matrix for VGG-19 and XGBoost (50/50 split)

The figure above shows the confusion matrix for 50/50 data split. The accuracy calculated was 92.6%

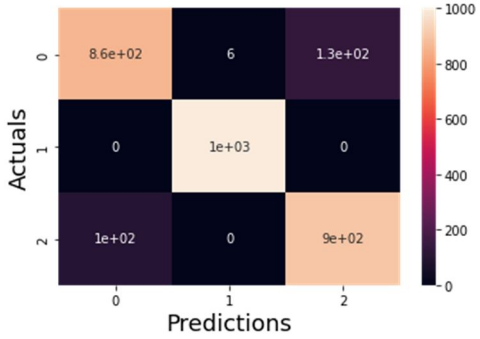


Fig. 12. Confusion Matrix for VGG-16 and SVM (50/50 split)

The figure above shows the confusion matrix for 50/50 data split. The accuracy calculated was 92%.

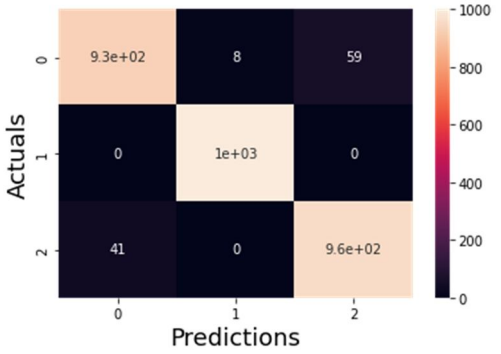


Fig. 13. Confusion Matrix for VGG-19 and SVM(50/50 split)

The figure above shows the confusion matrix for 50/50 data split. The accuracy calculated was 96.4%. This is the highest accuracy that we obtained.



Fig. 14. Confusion Matrix for InceptionResNetV2 and SVM(50/50 split)

The figure above shows the confusion matrix for 50/50 data split. The accuracy calculated was 89.8%.

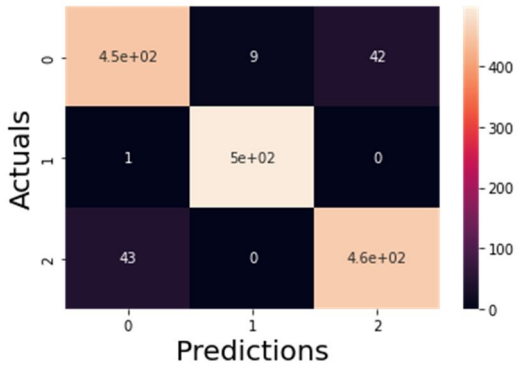


Fig. 15. Confusion Matrix for DenseNet201 and XGBoost (75/25 split)

The figure above shows the confusion matrix for 75/25 data split. The accuracy calculated was 92.6%.

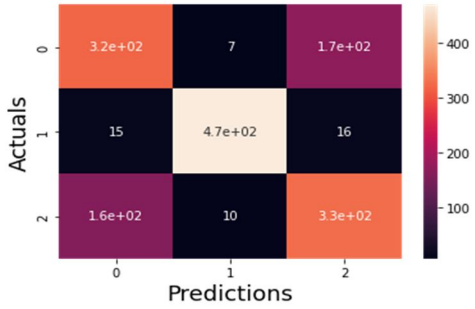


Fig. 16. Confusion Matrix for InceptionResNet and Decision Tree (50/50 split)

The figure above shows the confusion matrix for 50/50 data split. The accuracy calculated was 76.26%.

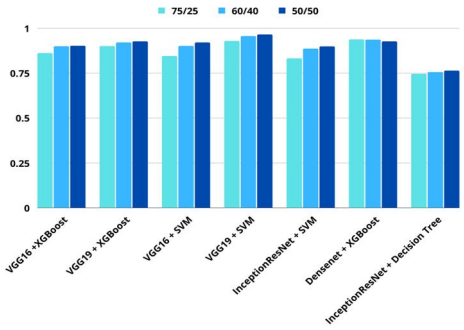


Fig. 17. Graph comparing the performance of all the algorithms for the data splits. The combination of VGG-19 and SVM performs well throughout with the accuracies of 92.8% for 75/25 split, 95.54% for 60/40 split, and 96.4% for 50/50 split.

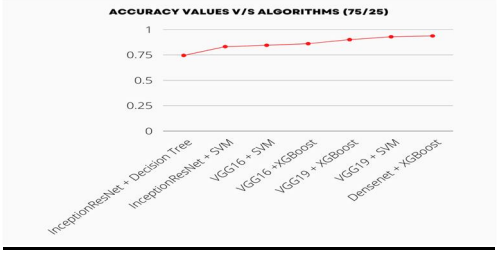


Fig. 18. Comparison between the accuracies of all the algorithms for 75/25 split. DenseNet201 and XGBoost perform the best, giving an accuracy of 93.66%. InceptionResNetV2 and Decision Tree give us the lowest accuracy of 74.53%

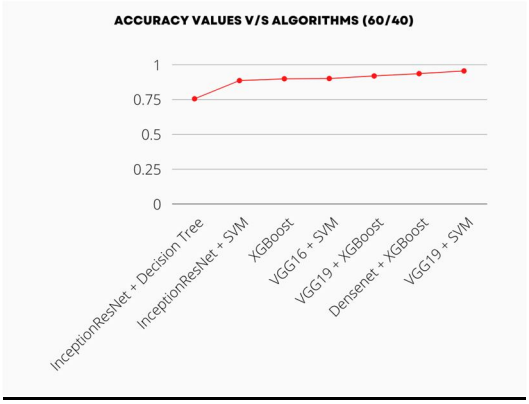


Fig. 19. Comparison between the accuracies of all the algorithms for 60/40 split.VGG-19 and SVM perform the best with an accuracy of 95.54%. InceptionResNetV2 and Decision Tree give us the lowest accuracy of 75.5%

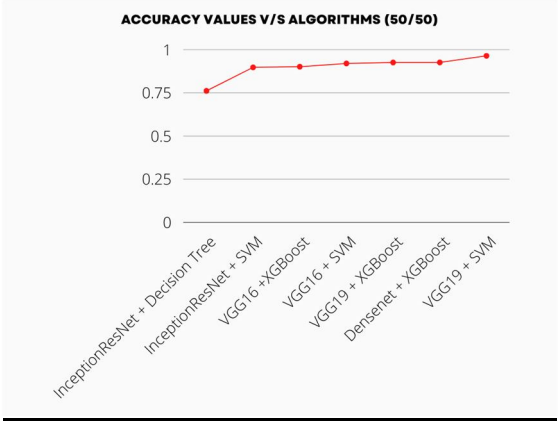


Fig. 20. Comparison between the accuracies of all the algorithms for 50/50 split.VGG-19 and SVM perform the best with an accuracy of 96.4%. InceptionResNetV2 and Decision Tree give us the lowest accuracy of 76.26%

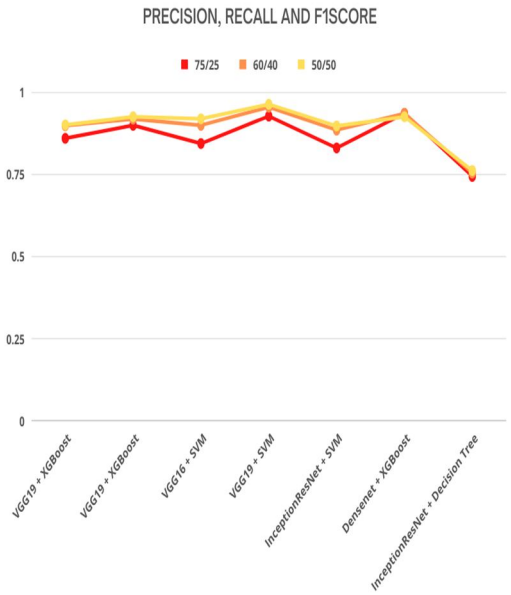


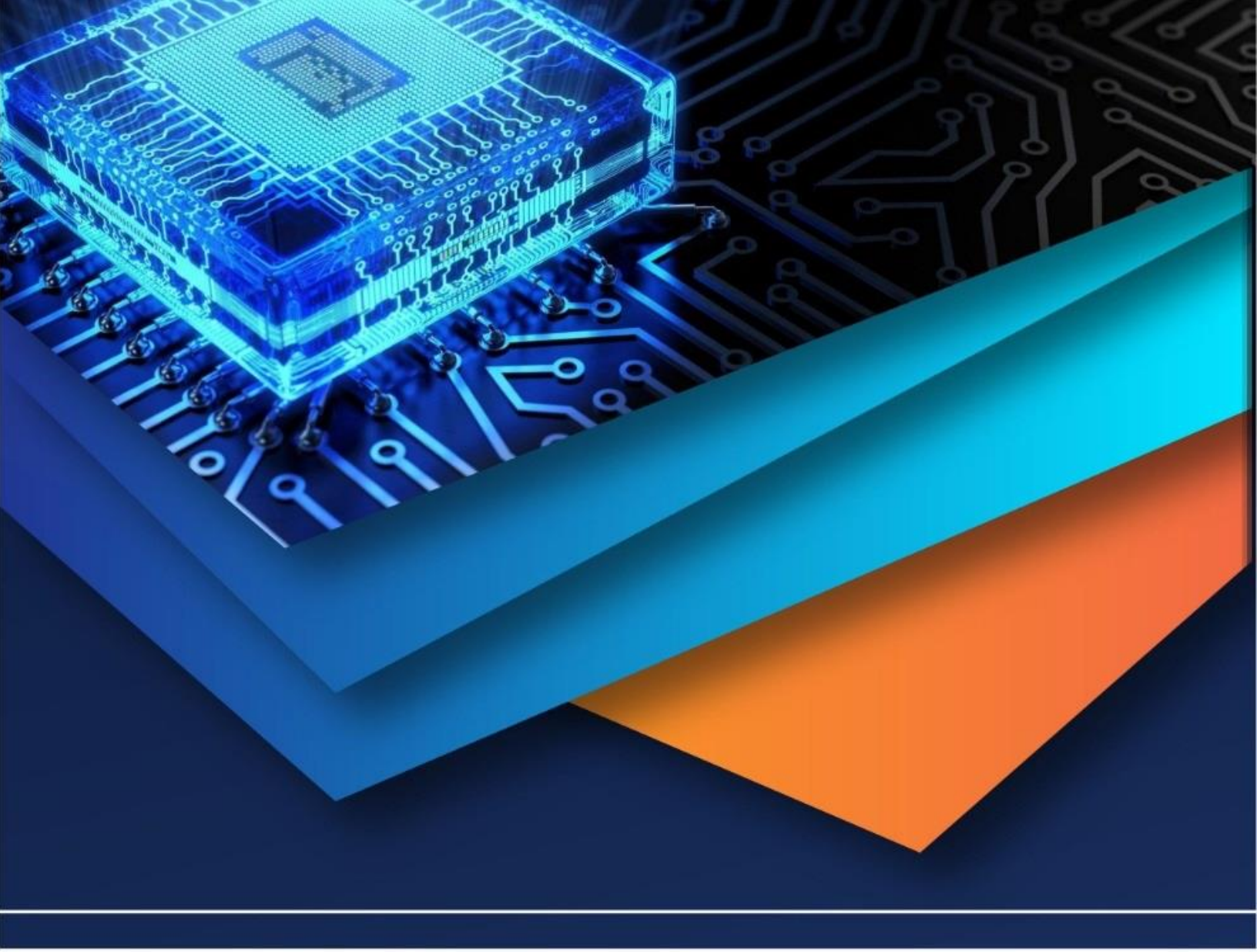
Fig. 21. Comparison between the Precision, Recall, and f1score of all the algorithms for all the data splits.

IV. CONCLUSION

In this paper, we try to analyze leukemia and myeloma datasets to predict whether a cell is cancerous or not and further classify it as leukemia or myeloma. Both the datasets were already preprocessed but augmentation was done to the myeloma dataset. In our study, multiple hybrid models were trained and tested for the different data splits as shown in Table II for various parameters like classification accuracy, precision, recall and F1 score. It is observed that SVM gives the best accuracy when used with VGG-19 for feature selection with an accuracy of 96.4%.

REFERENCES

[1] Leukemia & Lymphoma Society: https://www.lls.org/
[2] https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/
[3] https://www.researchgate.net/publication/343048987_Automatic_Medical_Images_Segmentation_Based_on_Deep_Learning_Networks#pf4
[4] https://www.pluralsight.com/guides/introduction-to-densenet-with-tensorflow
[5] https://medium.com/@zahraelhamraoui1997/inceptionresnetv2-simple-introduction-9a2000edcdb6
[6] Adeola Ogunleye and Qing-Guo Wang, "XGBoost Model for Chronic Kidney Disease Diagnosis", IEEE/ACM TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS, VOL. 17, NO. 6, NOVEMBER/DECEMBER 2020
[7] Manus Ross, Corey A. Graves, John W. Campbell, Jung H. Kim, "Using Support Vector Machines to Classify Student Attentiveness for the Development of Personalized Learning Systems", 2013 12th International Conference on Machine Learning and Applications
[8] https://docs.aws.amazon.com/sagemaker/latest/dg/xgboost-HowItWorks.html
[9] https://saturdays.ai/category/2022/
[10] Nilkanth Mukund Deshpande1,2, Shilpa Gite3,4 and Rajanikanth Aluvalu5, "A review of microscopic analysis of blood cells for disease detection with AI perspective", Deshpande et al. (2021), PeerJ Comput. Sci., DOI 10.7717/peerj-cs.460
[11] https://www.seldon.io/decision-trees-in-machine-learning
[12] Leukemia dataset https://www.kaggle.com/datasets/paultimothymooney/blood-cells
[13] Multiple myeloma dataset https://www.kaggle.com/datasets/sbilab/segpc2021dataset



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)