REPUBLIC OF TURKEY
YILDIZ TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

IMPLEMENTATION OF DEEP LEARNING BASED APPROACHES IN THE DIAGNOSIS OF DIABETES AND DIABETIC RETINOPATHY

Özge Nur ERGÜN

MASTER OF SCIENCE THESIS
Department of Computer Engineering
Program of Computer Engineering

Supervisor
Asst. Prof. Dr. Hamza Osman İLHAN

March, 2023

REPUBLIC OF TURKEY
YILDIZ TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

IMPLEMENTATION OF DEEP LEARNING BASED APPROACHES IN THE DIAGNOSIS OF DIABETES AND DIABETIC RETINOPATHY

A thesis submitted by Özge Nur ERGÜN in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE** is approved by the committee on 07.03.2023 in Department of Computer Engineering, Program of Computer Engineering.

Asst. Prof. Dr. Hamza Osman İLHAN
Yildiz Technical University
Supervisor

Approved By the Examining Committee

Asst. Prof. Dr. Hamza Osman İLHAN, Supervisor
Yildiz Technical University _____

Assoc. Prof. Dr. Ali Can KARACA, Member
Yildiz Technical University _____

Asst. Prof. Dr. Nihal ALTUNTAŞ, Member
Istanbul Gelisim University _____

I hereby declare that I have obtained the required legal permissions during data collection and exploitation procedures, that I have made the in-text citations and cited the references properly, that I haven't falsified and/or fabricated research data and results of the study and that I have abided by the principles of the scientific research and ethics during my Thesis Study under the title of Implementation of Deep Learning Based Approaches in the Diagnosis of Diabetes and Diabetic Retinopathy supervised by my supervisor, Asst. Prof. Dr. Hamza Osman İLHAN. In the case of a discovery of false statement, I am to acknowledge any legal consequence.

Özge Nur ERGÜN

Signature

Dedicated to my sister Bengü...
This work would not have been completed without her.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my advisor Asst. Prof. Dr. Hamza Osman İLHAN for the continuous support and guidance during this research.

Besides my advisor, I would like to thank my mom and dad for their endless trust. The courage they gave me helped a lot to complete this work.

Also, thanks to my sister Bengü for her unconditional love and support.

Özge Nur ERGÜN

TABLE OF CONTENTS

LIST OF SYMBOLS vii

LIST OF ABBREVIATIONS viii

LIST OF FIGURES ix

LIST OF TABLES xi

ABSTRACT xii

ÖZET xiv

1 INTRODUCTION 1

1.1 Problem Definition 1

1.2 Literature Review 2

1.3 Objective of the Thesis 6

1.4 Hypothesis 7

2 DATASET INFORMATION AND DATA PREPROCESSING STEPS 9

2.1 Early Stage Diabetes Risk Prediction Dataset 9

2.2 APTOS 2019 Dataset 11

2.3 Splitting Data for Model Validation 14

3 EARLY DIAGNOSIS OF DIABETES USING MACHINE LEARNING METHODS 16

3.1 Decision Tree 18

3.2 Random Forest 19

3.3 Support Vector Machine 21

3.4 XGBoost 21

3.5 K- Nearest Neighbor 23

3.6 Gaussian Naive Bayes 24

3.7 Artificial Neural Network 25

3.8 Convolutional Neural Network 26

4 PREDICTING THE SEVERITY LEVEL OF DIABETIC RETINOPATHY USING ENSEMBLE OF DEEP NETWORKS 31

4.1 Base Learners: Deep Learning Models 33
4.2 Ensemble Models 37
4.2.1 Hard Voting 39
4.2.2 Soft Voting 39
4.2.3 Stacked Generalization 40
4.3 Performance Metrics 41

5 RESULTS AND DISCUSSIONS **44**
5.1 Programming Environment 44
5.2 Early Stage Diabetes Risk Prediction Results 46
5.3 Diabetic Retinopathy Severity Level Predicting Results 48

REFERENCES **59**

PUBLICATIONS FROM THE THESIS **65**

LIST OF SYMBOLS

P(A)	Probability of Event A Occurring
$\sum$	Sum of Multiple Terms
σ	Variance

LIST OF ABBREVIATIONS

API	Application Programming Interface
APTOS	Asia Pacific Tele-Ophthalmology Society
CLAHE	Contrastive Limited Adaptive Histogram Equalization
CNN	Convolutional Neural Network
DR	Diabetic Retinopathy
XGBoost	Extreme Gradient Boosting
JPEG	Joint Photographic Experts Group
1D-CNN	One Dimensional Convolutional Neural Network
PNG	Portable Network Graphics
RGB	Red, Green and Blue
SVM	Support Vector Machine
WHO	World Health Organization

LIST OF FIGURES

Figure 2.1 Distribution of the data over classes for early stage diabetes risk prediction dataset 9
Figure 2.2 Encoding process of early stage diabetes risk prediction dataset for categorical columns 11
Figure 2.3 The distribution of the images over the classes in APTOS 2019 dataset 12
Figure 2.4 Image preprocessing steps 13
Figure 2.5 Train test split mechanism 14
Figure 2.6 K fold cross validation working mechanism over a 5 fold example [51] 14
Figure 3.1 Flowchart of the diabetes prediction stage 17
Figure 3.2 Calculated feature importances using decision tree method for early stage diabetes risk prediction dataset 18
Figure 3.3 Created decision tree after pruning 19
Figure 3.4 Voting mechanism of random forest [55] 20
Figure 3.5 Calculated feature importances using random forest method for early stage diabetes risk prediction dataset 20
Figure 3.6 SVM working logic [57] 21
Figure 3.7 XBGoost working principle [58] 22
Figure 3.8 Calculated feature importances using XGBoost method for early stage diabetes risk prediction dataset 23
Figure 3.9 K nearest neighbor working principle [59] 24
Figure 3.10 Sigmoid activation function [62] 26
Figure 3.11 An example of the convolution operation [63] 27
Figure 3.12 An example of the maximum pooling operation [64] 28
Figure 3.13 An example of the flattening operation [65] 28
Figure 3.14 An example of dense layer [66] 29
Figure 3.15 Convolutional neural network architecture [67] 29
Figure 4.1 Flowchart of the proposed approach 32
Figure 4.2 General architecture of the EfficientNet models [71] 34
Figure 4.3 Performance of EfficientNets compared to other models [71] 35
Figure 4.4 VGG16/19 model performances compared to other models [72] . . 35

Figure 4.5 VGG16 layers [73] 36
Figure 4.6 VGG19 layers [74] 36
Figure 4.7 Importance of independent base learners to tolerate mispredictions 38
Figure 4.8 Ensemble learning approach [75] 38
Figure 4.9 The implementation of hard voting method over an example retinal image 39
Figure 4.10 The implementation of soft voting method over an example retinal image 40
Figure 4.11 Stacked generalization flowchart 41
Figure 4.12 The implementation of stacked generalization method over an example retinal image 41
Figure 4.13 Structure of the confusion matrix 42
Figure 5.1 Popularities of programming languages [83] 44
Figure 5.2 Deep learning libraries according to how often they are used [92] 46
Figure 5.3 Confusion matrix of the VGG16 base model 49
Figure 5.4 Confusion matrix of the VGG19 base model 49
Figure 5.5 Confusion matrix of the EfficientNetB5 base model 50
Figure 5.6 Confusion matrix of the EfficientNetB7 base model 50
Figure 5.7 Confusion matrix of the EfficientNetV2S base model 51
Figure 5.8 Confusion matrix of the hard voting ensemble model 51
Figure 5.9 Confusion matrix of the soft voting ensemble model 52
Figure 5.10 Confusion matrix of the stacked generalization ensemble model . . 52

LIST OF TABLES

Table 2.1 Explanation of the attributes of early stage diabetes risk prediction dataset 10
Table 2.2 Published datasets on the field of diabetic retinopathy 12
Table 3.1 Optimized model parameters 30
Table 4.1 Training parameters of the models 37
Table 5.1 Model performances over early stage diabetes risk prediction dataset 47
Table 5.2 Performance comparision with state of art studies 48
Table 5.3 The averaged performance scores of individual models and ensemble approaches 53
Table 5.4 Class-wise performance scores of ensemble approaches 55
Table 5.5 Performance comparison of proposed framework and present studies 57

ABSTRACT

Implementation of Deep Learning Based Approaches in the Diagnosis of Diabetes and Diabetic Retinopathy

Özge Nur ERGÜN

Department of Computer Engineering
Master of Science Thesis

Supervisor: Asst. Prof. Dr. Hamza Osman İLHAN

Diabetes is an incurable and fatal disease reasoned by inconsistent levels of blood sugar. More than 400 million people have diabetes. In addition to being a fatal and chronic illness, diabetes causes other issues by damaging several parts of the body. Since early stage symptoms of diabetes can not be noticed by some of the patients, severe damage to various parts of the body may occur when the disease is diagnosed. Therefore, around 1.5 million deaths are related to diabetic reasons each year.

Diabetes related eye disease called diabetic retinopathy (DR), is caused by damage to retinal vessels due to high sugar levels in the blood, is one of the diseases that is related to diabetes. As the disease progresses, possibility of blurry vision, intermittent vision loss and blindness increases. Severe symptoms that occur as the diseases progress can be minimized with early detection.

In this study, methods for diagnosing diabetes and diabetic retinopathy diseases are investigated. Both machine learning and deep learning approaches such as K nearest neighbor, support vector machine, naive Bayes, decision tree, random forest, extreme gradient boosting and convolutional neural network are applied for predicting early stage diabetes. Deep learning based ensemble methods such as hard voting, soft voting and stacked generalization are studied to predict the severity level of diabetic retinopathy using retinal images. In terms of both early stage diagnosis of diabetes and diabetic retinopathy severity level prediction, accurate results are observed. 99.04% and 95.55% accuracies are obtained, respectively.

Keywords: Retinal image analysis, medical data analysis, machine learning, deep learning, ensemble learning

ÖZET

Diyabet ve Diyabetik Retinopati Teşhisinde Derin Öğrenmeye Dayalı Yaklaşımların Uygulanması

Özge Nur ERGÜN

Bilgisayar Mühendisliği Anabilim Dalı
Yüksek Lisans Tezi

Danışman: Dr. Öğr. Üyesi Hamza Osman İLHAN

Diyabet kan şekeri seviyesindeki düzensizliklere bağlı olarak gelişen, tedavisi olmayan ve ölümcül bir hastalıktır. Dünya çapında 400 milyondan fazla insanı etkilediği bilinmektedir. Kronik ve ölümcül olmasının yanı sıra, vücuttaki çeşitli bölgelerde hasara yol açıp farklı hastalıklara da sebep olabilmektedir. Ek olarak, erken evre semptomları hastaların bir kısmı tarafından fark edilemediğinden dolayı teşhis konduğunda vücutta ağır bir hasarın oluşması söz konusu olabilmektedir. Her yıl 1.5 milyon civarı ölümün diyabete bağlı olarak gerçekleştiği bilinmektedir.

Diyabetik retinopati (DR), kan şekeri seviyesindeki düzensizliklere bağlı olarak retinadaki damarların zarar görmesi ile oluşan bir hastalık olup doğrudan şeker hastalığına bağlı olarak gelişmektedir. Hastalık ilerledikçe bulanık görme, geçici veya sürekli körlük yaşanma riski oluşmaktadır. Diyabet ve diyabetik retinopati hastalıkları vücutta ilerledikçe ortaya çıkan şiddetli belirtilerin erken evrede teşhis konarak minimuma indirilmesi mümkündür.

Bu çalışmada, diyabeti ve diyabete bağlı gelişen diyabetik retinopatiyi erken teşhis etmek için makine öğrenmesi ve derin öğrenme yöntemleri uygulanmıştır. Diyabetin erken evre teşhisi için K en yakın komşu, destek vektör makinesi, naive Bayes, karar ağacı, rastgele orman, XGBoost ve evrişimli sinir ağları yöntemleri kullanılmıştır. Diyabetik retinopatinin hangi evrede olduğunun saptanması için ise retina görüntüleri kullanılarak derin öğrenme bazlı kolektif öğrenme mimarileri geliştirilmiştir. Tekil modellerin ürettiği sınıf atamalarını ve sınıf olasılık değerlerini baz alan kolektif öğreniciler geliştirilmiştir. Hem diyabet hem de diyabetik retinopati hastalığı için

başarılı sonuçlar elde edilmiş olup sırasıyla %99.04 ve %95.55 oranında doğruluk değeri elde edilmiştir.

Anahtar Kelimeler: Retina analizi, medikal veri analizi, makine öğrenmesi, derin öğrenme, kolektif öğrenme

1
INTRODUCTION

1.1 Problem Definition

Diabetes is a chronic and fatal illness that seen in relation to the blood sugar level irregularities. It causes damages at numerous parts of human body such as heart, blood vessels, eyes and nerves. Diabetes may be seen in 2 types. Type 1 diabetes is a disease that pancreas is unable to produce enough insulin to balance the blood sugar level. In type 2 diabetes, in addition to the lack of insulin that is seen in type 1 diabetes, insulin resistance critically affects the body too. According to the reports of World Health Organization (WHO) [1], more than 400 million people suffers from diabetes and 1.6 million deaths that are related to diabetic issues are recorded each year. Also, the number of diabetics is increasing at a growing rate every year. Therefore, early stage detection has crucial importance for reducing the effects of diabetes.

Former reserach states that diabetes may be progressed for 4-12 years before diagnosis [2]. Diabetes related damage have already occured in half of the patients when diagnosis is made. Former research state that early diagnosis of diabetes will help to prevent diseases that are related to heart, eye, kidney and vascular system [3, 4]. So, early diagnosis is very important to reduce the effects of the disease and improve life quality of patients.

Diabetic retinopathy (DR), which is a disease that is related to diabetes, that happens due to the high sugar levels in the blood. It causes damages at retinal vessels. DR has 2 phases as proliferative and non-proliferative which results in intermittent vision loss, blurred vision, or blindness. Non-proliferative is the early stage of the DR. It causes blurry vision due to the leaks of tiny retinal vessels. Proliferative is the advanced stage of the DR. If it is not diagnosed at the non-proliferative stage, retina starts growing new vessels that may bleed easily. The greater the bleeding causes the greater the possibility of intermittent or complete vision loss [5].

The most dangerous and critical part of DR is the lack of obvious early symptoms [6].

Diabetic eye screening is recommended to diabetics once a year in order to diagnose DR early. Eye screening involves examining the vessels at the back of the eyes by taking photographs. It helps early diagnosis of DR before it effects vision of the patient [7]. Early diagnosed patients may live without side effects of DR by changing their life standards.

Machine learning is a subdomain of artificial intelligence which aims to use data to create predictive models for complex problems [8]. It can be used for analyzing data and retrieving critical information. Moreover, the estimation for the unseen data using the existing data is possible by using the machine learning methodologies for many different domains. Based on this information, machine learning methodologies can be also applied in the domain of medicine for pattern understanding and information retrieving from existing data to predict the future trends of health problems. For instance, some symptoms like polydipsia, polyuria, itching, obesity, polyphagia and delayed healing are known as the early symptoms of diabetes. However, healthy people may also suffer from some of these problems. By using machine learning methodologies, deciding if a person is diabetic or healthy may be possible.

For diagnosis of DR, diabetic eye screening photographs are examined by doctors. Retinal features and blood vessels are examined for diagnosis. Examining retinal images by doctors is a long and laborious task. This process can be automated with deep learning methods. Deep learning is a subbranch discipline of computer science that attempts to mimic the human brain activities to solve complex problems [9]. Existing eye screening results may be examined using deep learning methods to make predictions for unseen eye screening photographs.

1.2 Literature Review

In literature, there are several studies that investigates early prediction of the diabetes. Early stage diabetes risk prediction dataset [10] is a commonly used dataset in the field of machine and deep learning studies.

Oladimeji et. al [11] analyzed the dataset to discover the most informative features for making predictions. They implemented several attribute evaluating methods such as SymmetricalUncert Attribute Evaluator, InfoGain Attribute Evaluator, GainRatio Attribute Evaluator and Correlation Attribute Evaluator. They found that itching, obesity and delayed healing are redundant attributes so, they ignored these features. After feature selection step, they implemented Random forest, K nearest neighbor, decision tree and naive Bayes methods by using a machine learning toolbox named as WEKA. Results are evaluated by using a 10 fold cross validation schema. The highest

accuracy observed at this study is 98.31% which is obtained by using random forest.

In another study [12] on the same dataset, a feature selection algorithm is implemented to reduce the number of features to optimize the Multilayer Perceptron Classifier. Grey Wolf Optimization and Adaptive Particle Swarm Optimization methods are utilized. Obtained results are compared to several machine learning methods such as decision tree, random forest, support vector machine, naive bayes, k-nearest neighbor and logistic regression. An 80:20 ratio train and test split is used for model evaluation and 97% accuracy is reported by using the proposed optimization based multilayer perceptron classifier.

Sadhu and Jadli [13] studied the same dataset. They applied random forest method which includes 100 decision trees. They reported an accuracy of 98.08% and ROC score of 99.79% using 10 fold cross validation schema.

Oleiwi et. al [14] applied 10 fold cross validation over the radial basis function network model which provided 98.80% of accuracy and 100% of sensitivity scores. In another research [15] an accuracy of 98,08% is obtained by implementing the K nearest neighbor algorithm over early stage diabetes risk prediction dataset. Additionally, several more machine learning methods such as multilayer perceptron, decision tree, support vector machine, linear discriminant analysis and ensemble learning algorithms were also applied and an early diagnosis of diabetes kit is developed. Different from the commonly used machine learning applications, a time domain specific deep learning approach, LSTM, is also validated over the same data set by Özer [16]. Özer reported an average F1 score of 98.9% by implementing LSTM networks and tested the result of the created model using 10 fold cross validation schema.

Addition to determination of diabetic status of patient, several datasets have been published for the analysis of diabetic retinopathy such as APTOS 2019 [17], EyePACS [18], IDRiD [19], Messidor [20] and Messidor-2 [21]. Each of these datasets includes the retinal images with different severity levels of DR.

According to the results of the previous studies that were performed over APTOS 2019 dataset, transfer learning approach provides accurate results. In this context, Sheikh and Qidwai [22] used pre-trained four networks as DenseNet, VGG, ResNet and Inception models. They reported that using PNG image format gives better performance than JPEG format. They also emphasized that applying normalization makes predictions more accurate and consistent. They reported 90.5%, 79.3%, 84.7% and 85.8% classification accuracies for DenseNet121, VGG16, Resnet50 and InceptionV3 models on APTOS 2019 dataset, respectively. Similarly, Rahman and

Dola [23] performed data augmentation techniques to the original RGB format images for class balancing. Then they implemented three deep learning networks in order to apply transfer learning on APTOS 2019 dataset. They reported 96.54%, 90.93% and 90.93% classification accuracies for DenseNet169, DenseNet121 and ResNet50 models, respectively.

Since analyzing retinal images is a computer vision problem, image processing methods are helpful to enhance the quality of image and build more accurate models. In this regard, Sugeno et al. [24] removed blurry or duplicate images from the dataset for data pre-processing. Also, they extracted unnecessary background content, applied Laplacian filter, and implemented a masking method for red or white lesions. After these pre-processing steps, they implemented EfficientNet-B3 in a transfer learning concept and obtained an accuracy of 84% on APTOS 2019 dataset. Similarly, Gangwar and Ravi [25] worked on both APTOS 2019 and Messidor datasets to determine the severity level of DR. They applied background elimination, blurring and data augmentation techniques for the pre-processing step. This architecture consists of a pre-trained Inception-ResNet-v2 model that is customized by adding extra convolutional layers. 72.33% and 82.18% test accuracies of the proposed model are reported for Messidor and APTOS 2019 datasets, respectively. Islam et. al [26] applied Contrastive Limited Adaptive Histogram Equalization (CLAHE) method for the enhancement of the images in the APTOS 2019 dataset. Then, they implemented supervised contrastive learning approach, which using the two augmented version of the same input to determine the similar feature sets to use in the detection process of a different input image. They also performed a margin hyperparameter tuning for optimization. They analyzed the dataset in two stage as detection of DR cases and determination of severity of DR. This architecture was used for both DR detection and DR severity level prediction. 98.36% and 84.36% classification accuracies are obtained for DR and non DR classification and the prediction of DR severity, respectively.

In several research, feature vectors were used for extracting the retinal information. Classification of the extracted retinal features may lead to better results. In this regard, Bodapati et al. analyzed feature maps of the pre-trained VGG16 model [27]. Retinal lesion information feature maps were combined using feature level fusion method. 84.31% classification accuracy was reported for the classification of APTOS 2019 dataset in their study. Shaik and Cherukuri implemented CLAHE to RGB channels of retinal images, separately [28]. They extracted features using VGG16 network. Then, they created an architecture named hinge attention network to classify images using these extracted features. They reported 85.54% and 66.41% classification accuracies for APTOS 2019 and IDRiD datasets, respectively. In another study prepared by Shaik

and Cherukuri, they also used Xception architecture to extract lesion-specific features of retinal images [29]. They trained a neural SVM using the lesion-specific features. On IDRiD dataset, an accuracy of 63.24% is reported. The same approach is resulted in 84.31% accuracy over APTOS 2019 dataset.

Cahoon et al. [30] emphasized that APTOS 2019 dataset has highly imbalanced in terms of class distribution. They suggested data augmentation for class balancing. After class balancing, they created a three-stage architecture that divides DR grading problem into three parts. First, a model with three outputs as no DR, little DR or much DR was designed. Images are classified again with the customized another model for the determination of less DR or more DR in case of the detection for DR according to the output of the first model. 80% of accuracy is reported by this three stage architecture.

The idea of ensembling the models or datasets were performed in several studies to increase the prediction performance of the severity level of DR. Karki and Kulkarni [31] combined two DR datasets namely APTOS 2019 and EyePACS to create an ensemble of deep learning models. They trained EfficientNetB1, EfficientNetB2, EfficientNetB3 and EfficientNetB5 models and achieved 92.43% kappa score.

Sikder et al. extracted features of retinal images using histogram method and built an ensemble of decision trees using bagging [32]. They stated 91.07% classification accuracy for APTOS 2019 dataset. Later, Sikder et al. [33] implemented another ensemble learning approach over the same APTOS 2019 dataset to improve the previous model performance. They removed noisy and duplicated images and adjusted contrast levels of images for enhancing image quality. Textural features were extracted using Gray-Level Co-Occurrence Matrix. Addition to these features, they also used histogram features to fed a tuned XGBoost model. 94.20% accuracy score for APTOS 2019 dataset was reported by the proposed ensemble architecture.

In several studies, ensembling idea was performed over deep neural networks. Bodapati et al. [34] implemented multi-model fusion idea over pre-trained Inception and VGG16 models. They fused outputs of these two models using gate attention blocks within a new dense layer. 97.82% of DR detection accuracy and 82.54% of DR severity prediction accuracy is reported for APTOS 2019 dataset.

Matsoukas et al. [35] compared the classification performances of Convolutional Neural Networks (CNN) and transformers over APTOS 2019 dataset. They used vision transformer architecture. The results indicate that the transformer which are pretrained on ImageNet is as accurate as CNNs. Kappa scores are reported as 89.6% and 89.4% for transformer and Resnet50 architectures on APTOS 2019 dataset,

respectively.

Some studies focus on DR detection instead of determining the severity level of DR. For that purpose, datasets are reorganized as DR and no DR. DR class includes all the severity levels of DR as a single class. Çinarer and Kiliç [36] applied Gaussian filter and data augmentation on the reorganized APTOS 2019 dataset (DR and no DR). Then, VGG16 and AlexNet models were utilized to detect if DR exists. Classification accuracies were measured as 95.26% and 98.17% for AlexNet and VGG16 models, respectively. Similarly, Dwivedi and Attry implemented several models using transfer learning approaches for binary classification of DR on APTOS 2019 dataset. Maximum accuracy is reported as 96.73% obtained by using MobileNetV2 model. Sanjana et al. used InceptionResNetV2, Xception, MobileNetV2, NASNetMobile and DenseNet121 on APTOS 2019 dataset to detect DR [37]. Accuracies are reported as 86.25%, 96.25%, 93.75%, 81.25%, and 80% respectively.

Zhang et al. [38] utilized a source-free transfer learning approach. They used a huge retinopathy dataset, namely EyePACS dataset. They aimed to train the models from scratch to apply the transfer learning idea. After the training phase, they used the models over the classification of APTOS 2019 dataset. 91.2% accuracy is obtained on detection of DR.

In another DR detection study, Adriman et al. [39] extracted Local Binary Patterns (LBP) of retinal images in the APTOS 2019 dataset. Then, ResNet, DenseNet and DetNet models were trained by the extracted LBP feature sets. The highest accuracy was measured as 96.35% using ResNet34 model. On the other hand, Densenet121 and DetNet59 models resulted in 84.05% and 93.99% detection results. Li et al. implemented graph network for binary DR detection. They used auto-encoder feature learning for extracting retinal features and implemented a graph convolutional network (GCN) by using these features [40]. Accuracy of 94.4% is achieved over APTOS 2019 dataset.

1.3 Objective of the Thesis

The objective of the thesis is to investigate machine learning and deep learning methods in the determination of diabetic status of patient and severity level of one of the diabetes related disease, diabetic retinopathy. It is aimed to analyze a dataset about diabetes, analyze its features to see which ones are associated with diabetes and develop an early stage diagnosis kit to minimize the damage that the patient will suffer.

A diagnosis kit is also aimed to be developed for diabetic retinopathy, which is an eye disease that develops due to diabetes. In this way, these outcomes may be used to reduce the damage by providing early diagnosis of the disease.

The study that is carried out is presented in the following sections as below.

- Introduction: Fatality of diabetes and how easily it spreads through other parts of the body is explained and the importance of early diagnosis is stated. Former studies are detailed and discussed with their contributions and shortcomings. Then our hypothesis to solve this problem is explained briefly.
- Dataset Information and Data Processing Steps: Early stage diabetes risk prediction and APTOS 2019 datasets are explained, their data distribution are shown, applied preprocessing steps and model validation schema are detailed.
- Early Diagnosis of Diabetes Using Machine Learning Methods: Carried out study for detection of the diabetes is explained in detail. Applied machine learning methods are explained.
- Predicting the Severity Level of Diabetic Retinopathy Using Ensemble of Deep Networks: Ensemble learning approach is explained. Carried out study for detection of the severity of the diabetic retinopathy is explained in detail. Built model architectures are shown and applied ensemble ideas are discussed.
- Results and Discussion: Programming language and tools are detailed and reasons of why they have chosen are explained. Obtained results are explained using different performance metrics. Results of different models are compared and their differences are discussed.

1.4 Hypothesis

Machine learning is a discipline that allows computers to predict an upcoming outcome using the existing outcomes. If there are data of people who already suffered diabetes, it would be possible to discover which features are associated with diabetes. Also, making accurate predictions if a subject is diabetic or will be diabetic may be possible too.

Deep learning is a subdiscipline of machine learning. It may be used to process image, video, audio or text data. In this work, image data are processed in order to analyze retinal images. Using the existing data, it may be possible to predict if a subject has

diabetic retinopathy (which is a disease that develops due to the diabetes), if so, what is the severity level of it.

The hypotheses are given in the following items.

- Using existing data, it is possible to predict future outcomes.
- Data analysis is helpful to uncover patterns and correlations.
- Machine learning methods such as k-nearest neighbor, support vector machine, naive Bayes, XGBoost, decision tree, random forest are proven methods to make accurate predictions.
- Artificial neural networks and 1 dimensional convolutional neural networks are proven methods to make predictions on numerical and encoded categorical features.
- Deep learning models such as VGG16, VGG19 and EfficientNet can be used to analyze retinal images and make predictions on them.
- Ensemble methods are helpful to improve the accuracy of predictions.
- Base models' class predictions and prediction probabilities can be used to build ensemble models.
- Hard voting, soft voting and stacked generalization ensemble models are more robust then base learner models.

2
DATASET INFORMATION AND DATA PREPROCESSING STEPS

In this study, 2 datasets are used. For predicting early stage diabetes, Early Stage Diabetes Risk Prediction dataset is used. For predicting the severity level of diabetic retinopathy, APTOS 2019 dataset is used. These datasets are explained in the following subsections.

2.1 Early Stage Diabetes Risk Prediction Dataset

Early stage diabetes risk prediction dataset is prepared by Islam et al. [41]. Patient data have been collected from the records of Sylhet Diabetes Hospital in Bangladesh. There are 520 data and 16 attributes that can be related to diabetes. Data distribution among classes are shown in Figure 2.1.

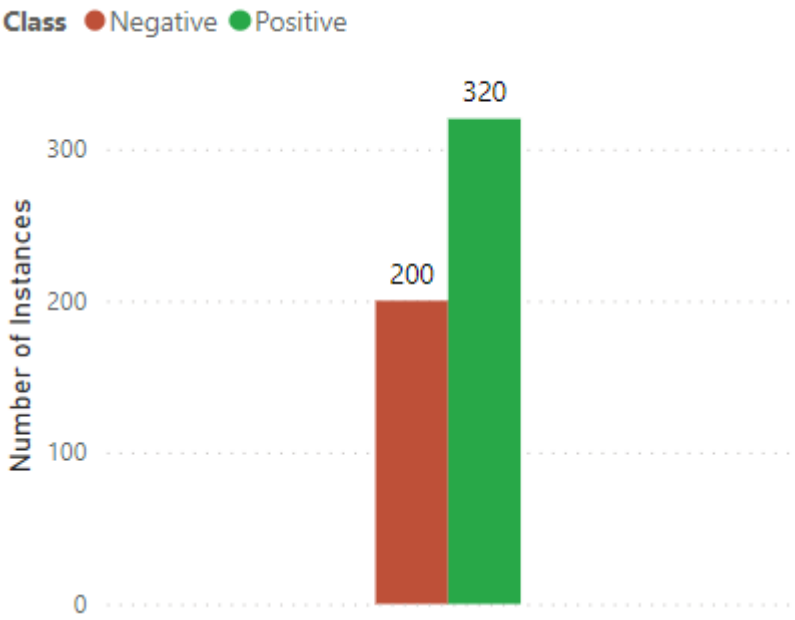


Figure 2.1 Distribution of the data over classes for early stage diabetes risk prediction dataset

15 features are categorical and 1 feature is continuous. Features are explained in the Table 2.1.

Table 2.1 Explanation of the attributes of early stage diabetes risk prediction dataset

Attribute	Explanation
Age	Age of the given instance
Sex	Gender of the given instance
Polyuria	Urinating more often than usual
Polydipsia	Getting thirsty more often than usual [42]
Sudden Weight Loss	Losing a noticeable amount of weight suddenly [43]
Weakness	Getting tired easily
Polyphagia	Getting hungry more often than usual [44]
Genital thrush	A yeast infection that can occur due to diabetes [45]
Visual blurring	Loss of sharpness of vision [46]
Itching	Itching more often than usual
Irritability	Being annoyed more often than usual
Delayed Healing	Wounds heal slower than a healthy person
Partial Paresis	The condition of weakness of voluntary [47] movement
Muscle Stiffness	The inability of the muscles to relax [48] normally.
Alopecia	Alopecia leads to hair loss on any part of the body [49]
Obesity	Being unhealthyly overweight
Class	1 for diabetic, 2 for healthy

There are no missing values at the Early Stage Diabetes Risk Prediction dataset so, encoding categorical values is enough for the data preprocessing step. For yes/no labeled attributes 1 for yes and 0 for no is used. For gender column, 1 for female and 0 for male is used for the encoding step. This encoding process over a few samples are shown in Figure 2.2.

Age	Gender	Polyuria	Polydipsia	Sudden Weight Loss	Weakness	Polyphagia	Genital Thrush	Visual Blurring	Itching	Irritability	Delayed Healing	Partial Paresis	Muscle Stiffness	Alopecia	Obesity	Class
40	Male	No	Yes	No	Yes	No	No	No	Yes	No	Yes	No	Yes	Yes	Yes	Positive
48	Female	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No	Yes	No	No	No	Positive
66	Male	Yes	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No	Yes	Yes	No	No	Positive
64	Male	No	No	No	Yes	Yes	No	Yes	Yes	Yes	Yes	No	Yes	Yes	No	Negative
32	Female	No	No	No	Yes	No	No	Yes	Yes	No	Yes	No	No	Yes	No	Negative

Age	Gender	Polyuria	Polydipsia	Sudden Weight Loss	Weakness	Polyphagia	Genital Thrush	Visual Blurring	Itching	Irritability	Delayed Healing	Partial Paresis	Muscle Stiffness	Alopecia	Obesity	Class
40	0	0	1	1	0	1	0	0	0	1	0	1	0	1	1	1
48	1	1	1	1	1	1	1	0	0	0	0	0	1	0	0	1
66	0	1	1	1	1	1	0	0	1	1	1	0	1	1	0	1
64	0	0	0	0	1	1	1	0	1	1	1	1	0	1	0	0
32	1	0	0	0	1	0	0	0	1	1	0	1	0	0	1	0

Figure 2.2 Encoding process of early stage diabetes risk prediction dataset for categorical columns

2.2 APTOS 2019 Dataset

For predicting the severity of diabetic retinopathy, there are some datasets that are published and studied by several researchers. These datasets are summarized in Table

2.2.

Table 2.2 Published datasets on the field of diabetic retinopathy

Dataset	Published Year	Number of Samples	Number of Classes
APTOS 2019	2019	3662	5
EyePacs	2015	35126	5
Messidor	2014	1058	4
Messidor-2	2018	1748	4
IDRID	2018	516	2

Since it is the most recent dataset on the topic of diabetic retinopathy detection, APTOS 2019 dataset is used in the experiments of the thesis. In 2019, Asia Pacific Tele-Ophthalmology Society (APTOS) created DR dataset by collecting the retinal images in order to predict severity level and presence of the diabetic retinopathy. APTOS 2019 dataset is publicly available and can be downloaded from Kaggle [17]. APTOS 2019 dataset contains 3662 retinal images that are collected from multiple clinics by using different camera types. Images are classified into 5 classes by experts as no DR, mild, moderate, severe and proliferative DR. The distribution of the images over the classes are given in Figure 2.3.

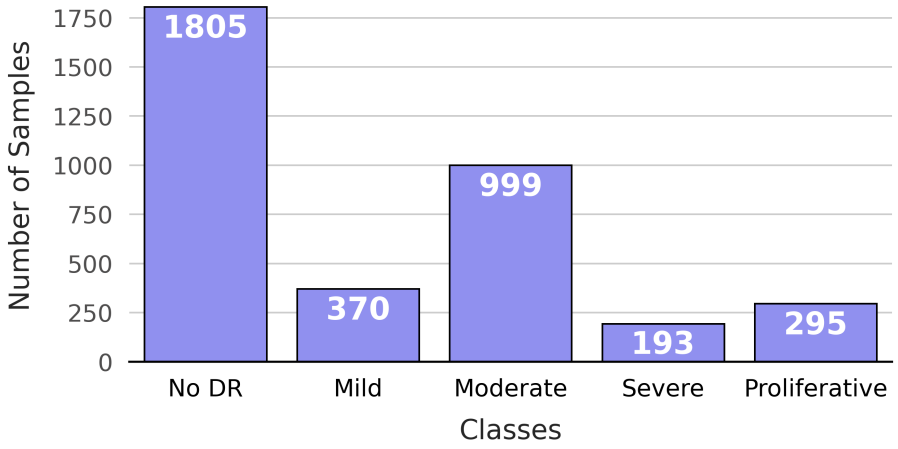


Figure 2.3 The distribution of the images over the classes in APTOS 2019 dataset

APTOS 2019 dataset is prepared using retinal images that are taken under different conditions. There are 3662 train images that have 5 labels. 0 for no DR, 1 for mild, 2 for moderate, 3 for severe and 4 for proliferative DR. From 0 to 4, severity of DR increases. Proliferative DR is the stage that patient suffers from vision loss.

In terms of data preprocessing, some image processing techniques are applied in order to build more robust models. Data preprocessing steps are explained below.

- Images are resized to 224x224 to get a standard shape and speed up the training phase.
- Gaussian filter using 5x5 kernel is applied to remove the salt and pepper noise.
- A histogram based automatic cropping background removing process is applied for isolating the retinal part of the images. Columns or rows that only contain black pixels were removed. Since retinal images include black background which is uninformative, it is a necessary step to remove them.
- Data augmentation methods are applied to the dataset for obtaining more accurate and consistent models.
  - Rotating the images up to 60 degrees
  - Flipping images horizontally
  - Flipping images vertically
  - Image zooming by the factor of 0.2 scale
  - Grayscaling operations

  were selected to use in the spatial domain based data augmentation step.

These data preprocessing steps are summarized in Figure 2.4.

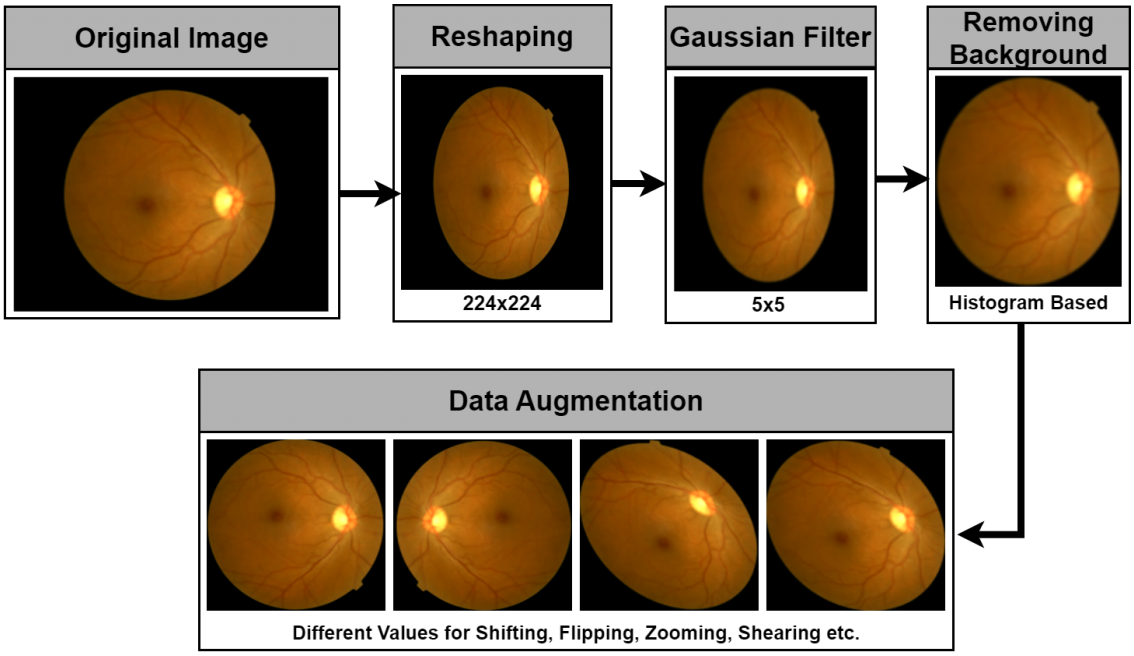


Figure 2.4 Image preprocessing steps

2.3 Splitting Data for Model Validation

Train-test split and k fold cross validation are common two methods for model validation. In train-test split approach, a part of data will be saved for testing and it is not used during training. Train-test split approach is shown in Figure 2.5 [50]. It is important to split data considering class distribution, otherwise test data would not reflect dataset correctly.

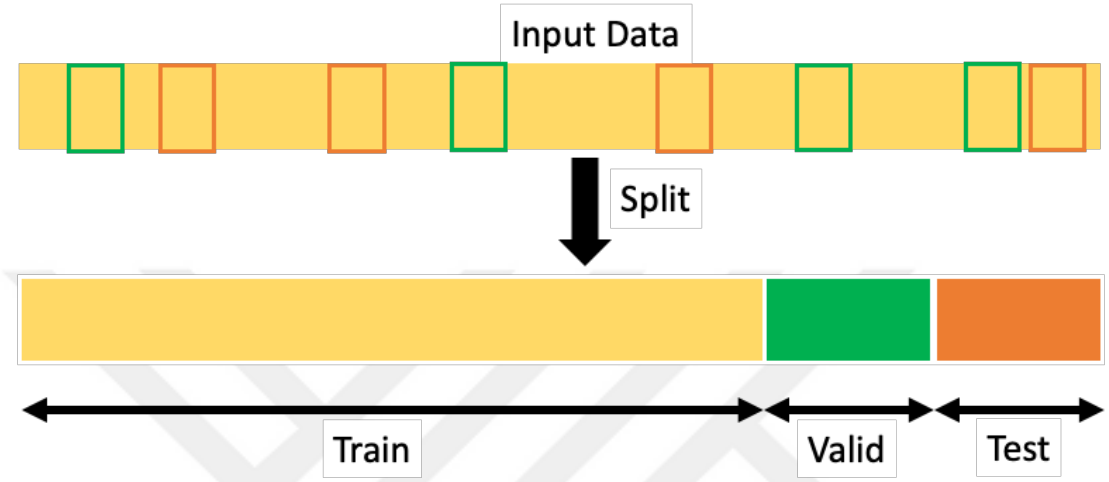


Figure 2.5 Train test split mechanism

K fold cross validation is another method to validate model. It repeats the training phase k times and at each repetition, it uses different part of dataset for testing. By doing so, it tests the behavior of model using all of the dataset. Working mechanism of k fold cross validation schema is shown in Figure 2.6 [51].

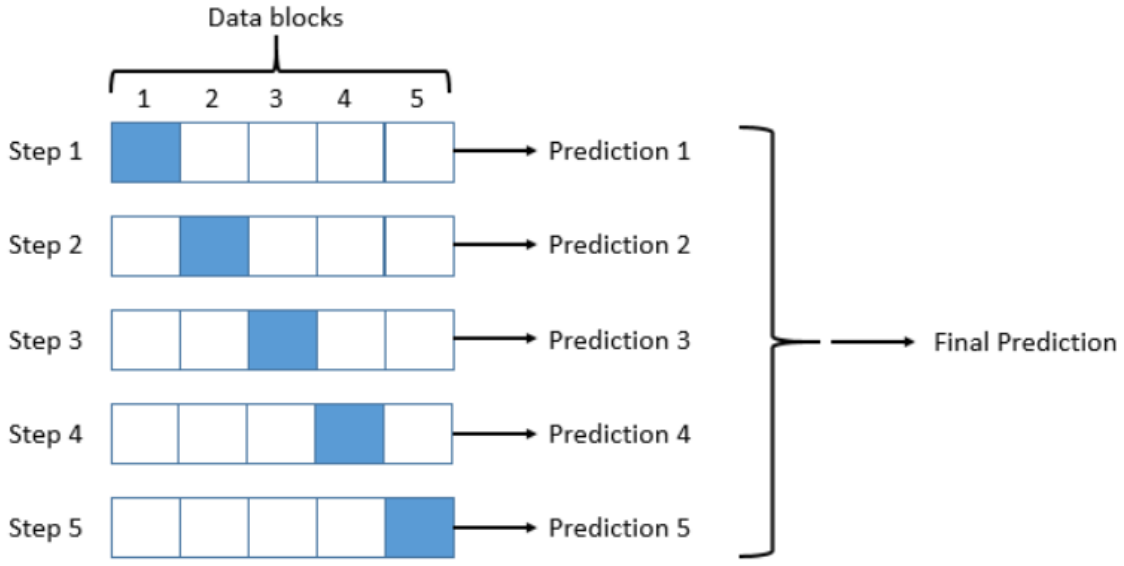


Figure 2.6 K fold cross validation working mechanism over a 5 fold example [51]

Since k fold validation schema provides more generic and objectiveness in the results of the trained model [52], 5 fold cross validation was applied in the presented work. 5 fold cross validation is applied instead of 10, in order to avoid long training time for building each ensemble. While 10 fold cross validation means 10 times of training for each model, 5 fold cross validation means 5 times of training for each model which halves the training time.

3
EARLY DIAGNOSIS OF DIABETES USING MACHINE LEARNING METHODS

Machine learning algorithms are developed to make accurate predictions using existing data. They are based on statistical and mathematical methods.

In this part of the study, classical machine learning methods are applied in order to predict if a subject is diabetic in the early stages. Since there are lots of machine learning methods, some of them are used to build models to see how they perform. Eight widely used machine learning techniques are performed on the Early Stage Diabetes Risk Prediction dataset and performance metrics are compared. Selected methods are XGBoost, decision tree, random forest, support vector machine, Naive Bayes, K nearest neighbor, artificial neural network and convolutional neural network. Obtained results are evaluated in terms of accuracy, recall, precision and f score using cross validation schema. Applied methodology is illustrated in Figure 3.1 and details of each method are explained in the following subsections.

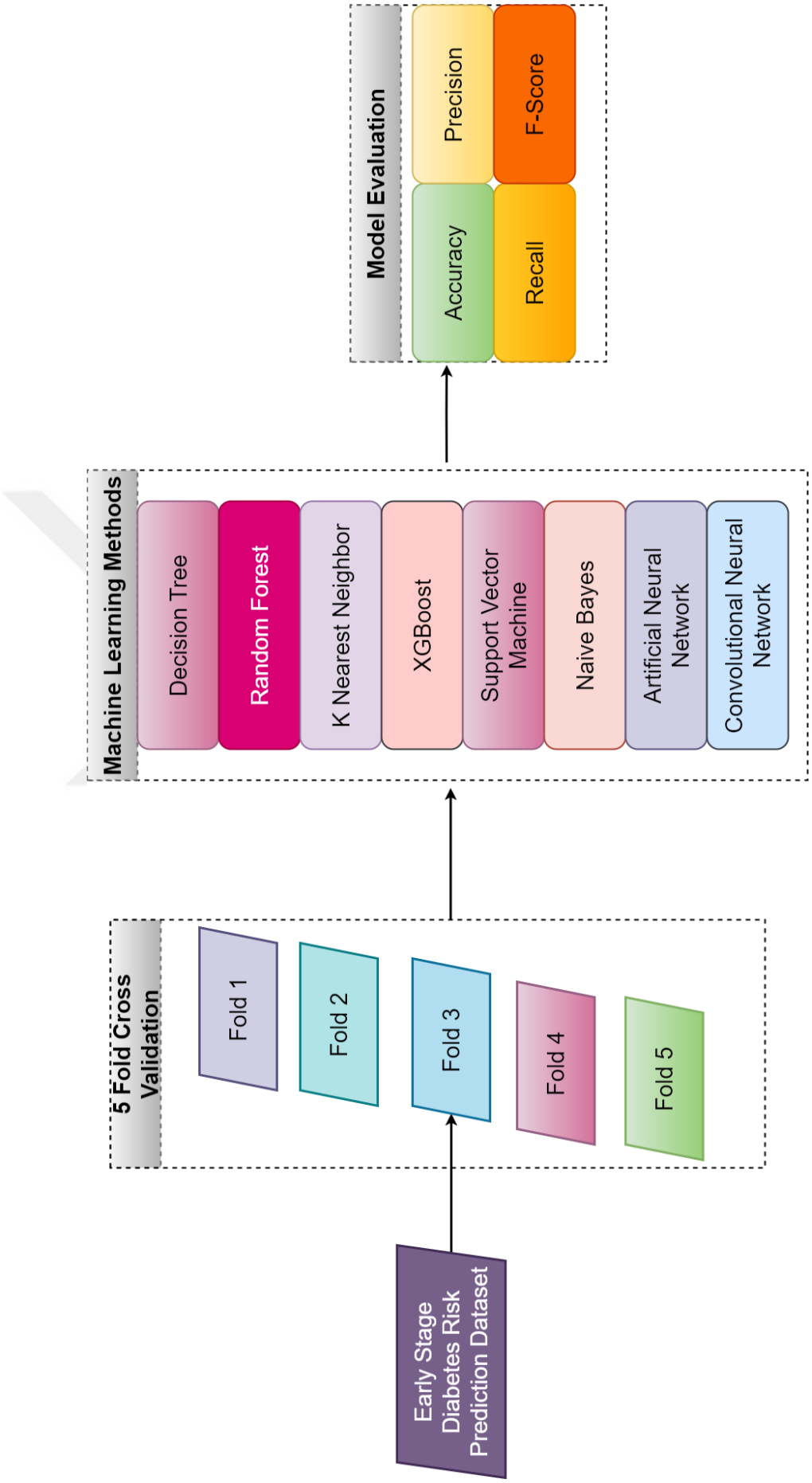


Figure 3.1 Flowchart of the diabetes prediction stage

3.1 Decision Tree

Decision tree is a structure in which the model is formed by branching like a tree at each decision. Since every step of the decision making is easy to understand, it is advantageous to understand how the model make predictions, so especially people who are not interested in mathematics prefer to use decision tree approach.

Considering training data, a tree is built by asking questions at each node. Tree nodes represent features, branches represent the direction must be taken after nodes and leaves represent the outcomes. New instances may be processed by traversing from root to leaves by choosing appropriate branches.

In addition to be able to understand how the model decides, it is also possible to see feature importances. By calculating mean decrease in impurity for all nodes of the tree, feature importances may be calculated. The most elective and informative feature takes place at the root node. Feature importances may be seen in Figure 3.2.

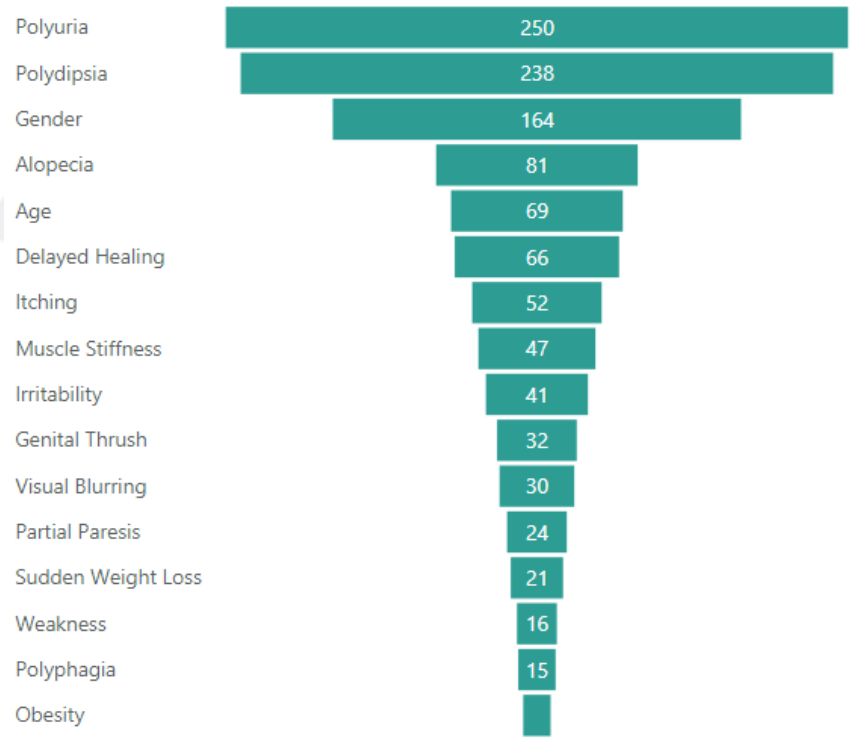


Figure 3.2 Calculated feature importances using decision tree method for early stage diabetes risk prediction dataset

In this study, tree have been created using Gini Information Gain applying two-level pruning. Maximum depth is set to 8 which is half the number of features. Polyuria is the root node of the tree, the tree branches from root node according to the existence of polyuria symptom. For instance, if an instance has polyuria symptom, their age is

checked if it is more or less then 70. If the age of given instance is less than 70, it is labeled as diabetic. Created decision tree for the early stage diabetes risk prediction dataset may be seen in Figure 3.3.

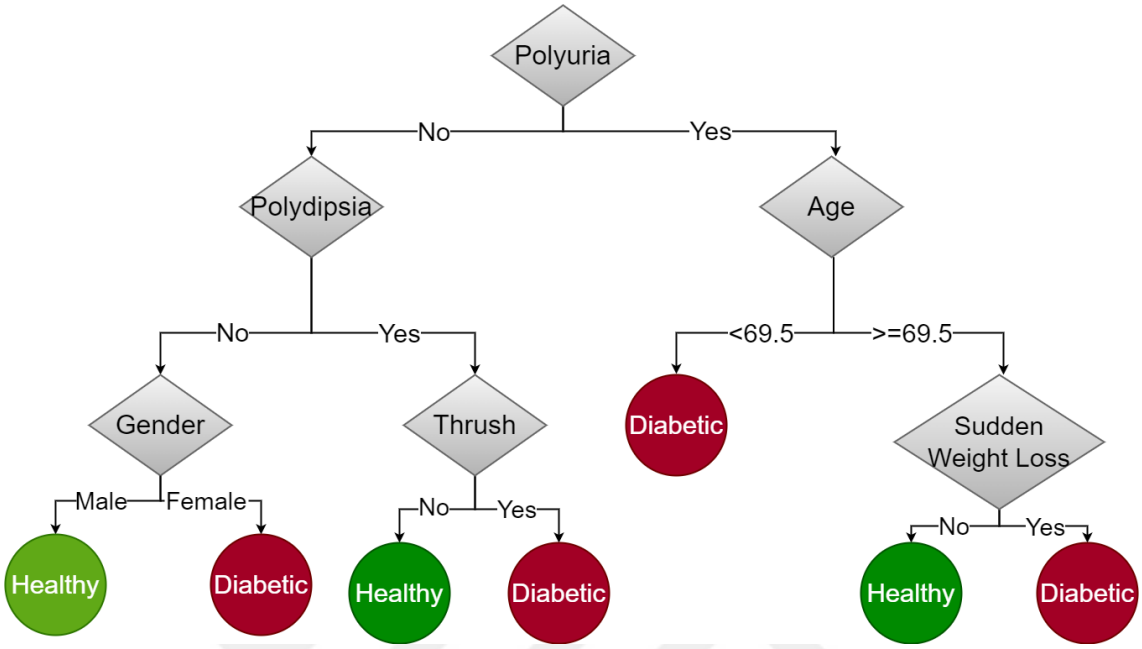


Figure 3.3 Created decision tree after pruning

3.2 Random Forest

Decision tree is sensitive to overfitting problem. An improper branching may lead to inaccurate predictions. Especially, if an improper branching applied at a node that is close to the root, it would affect all of the tree decisions.

Random forest is a method that focuses on to solve the overfitting issue. Former research [53, 54] that are performed on various datasets state that random forest method achieves better results than decision tree because of using ensemble learning idea. Lots of decision trees are built using diverse chunks of the original dataset. By doing so, different independent and individually successful decision makers are aimed to be built. Predictions of lots of randomly built decision trees are combined and a majority voting mechanism is applied to make predictions. Voting mechanism of random forest method is explained in Figure 3.4 [55].

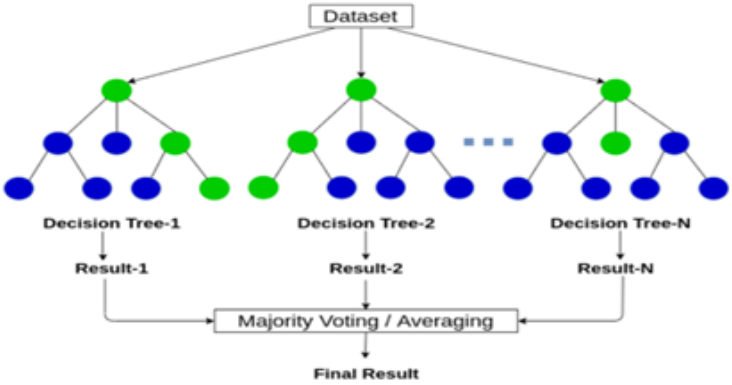


Figure 3.4 Voting mechanism of random forest [55]

In this study, 1000 base learners are used to build random forest. Gini information gain method is used and maximum depth is set to 8 which is half the number of features similar to single decision tree experments.

Similar to decision tree, random forest method allows seeing feature importances. Obtained feature importances are shown in Figure 3.5.

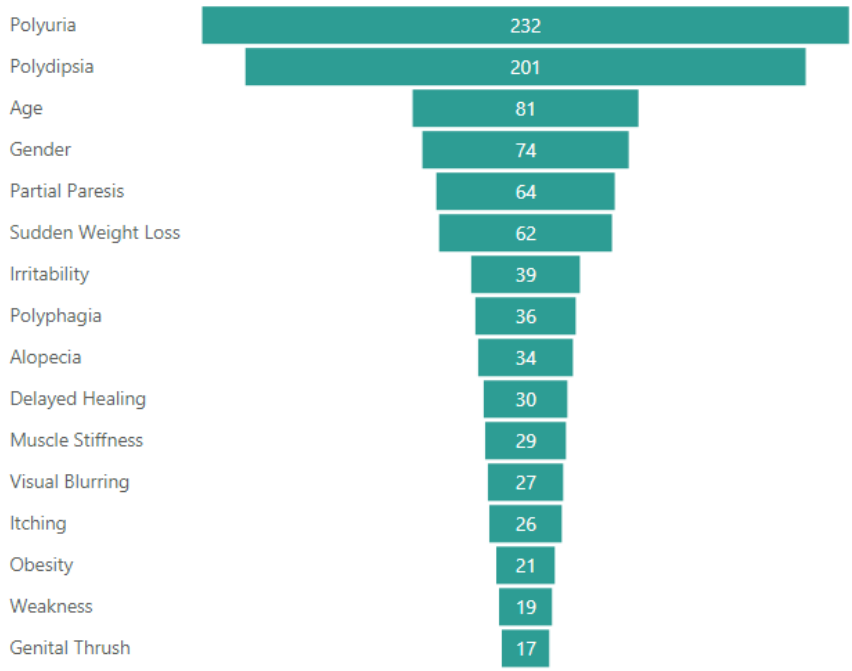


Figure 3.5 Calculated feature importances using random forest method for early stage diabetes risk prediction dataset

3.3 Support Vector Machine

Support vector machine (SVM) divides instances into logical groups using hyperplanes. Classification is done by mapping each data geometrically. In general principle a hyperplane in a feature space created by a kernel function k [56]. As the k increases, it may require a lot of training time and cost for large sized datasets, especially the ones with lots of attributes. Where d denotes the degree of the polynomial, SVM kernel equation is explained in the Equation 3.1

$$k(x_i, x_j) = (x_i * x_j + 1)^d \tag{3.1}$$

Working logic of SVM is shown in Figure 3.6 [57].

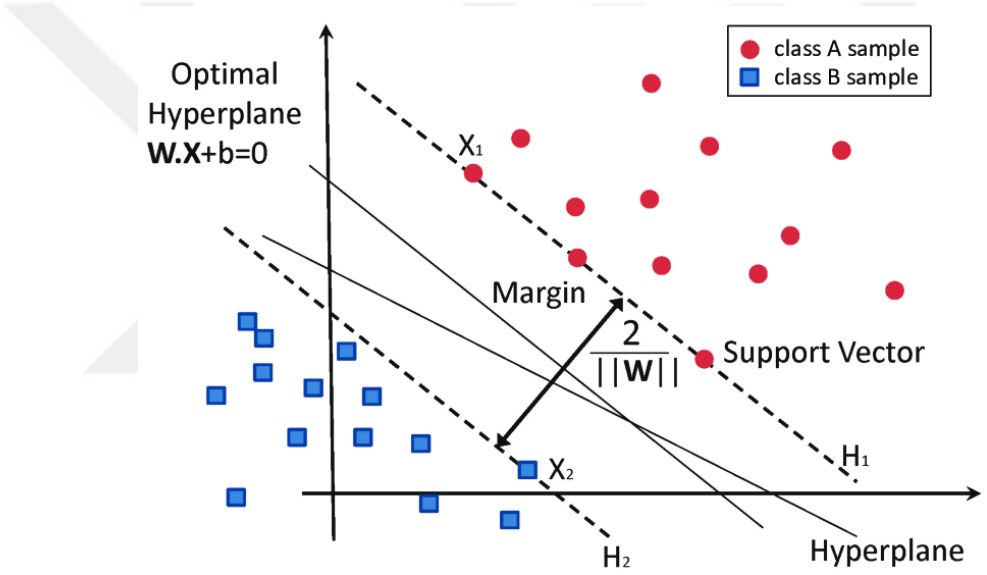


Figure 3.6 SVM working logic [57]

In this study, a third degree polynomial kernel and a regularization parameter as 0.1 is set.

3.4 XGBoost

Extreme gradient boosting (XGBoost) is a tree based ensemble machine learning methodology. New nodes are added parallelly to weak decision trees to make them more robust. Gradient of loss function is considered for this process. Majority voting is applied to make predictions. Working logic of XGBoost is shown in Figure 3.7 [58].

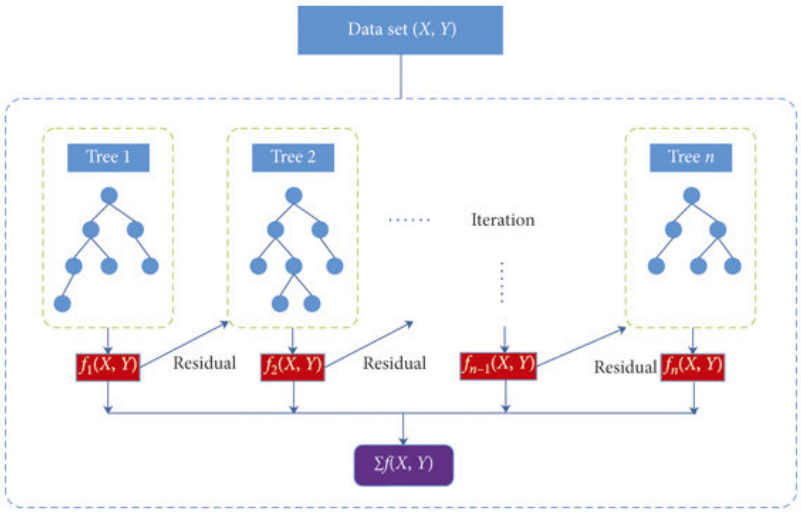


Figure 3.7 XBGoost working principle [58]

In this study, learning rate is set to 0.1 and maximum depth is set to 8 which is half the number of features.

According to the results of XGBoost classifier, age factor has a crucial importance in the classification. Then, polydipsia factor that indicates if the patient has the symptom of being excessively thirst is the second crucial parameter in the diagnosis of diabetes. The least important attributes were sudden weight loss and muscle stiffness symptoms of the patients as those can happen because of many other reasons such as excessive sportive activity or stress. Obtained feature importances using XGBoost classifier may be seen in Figure 3.8.

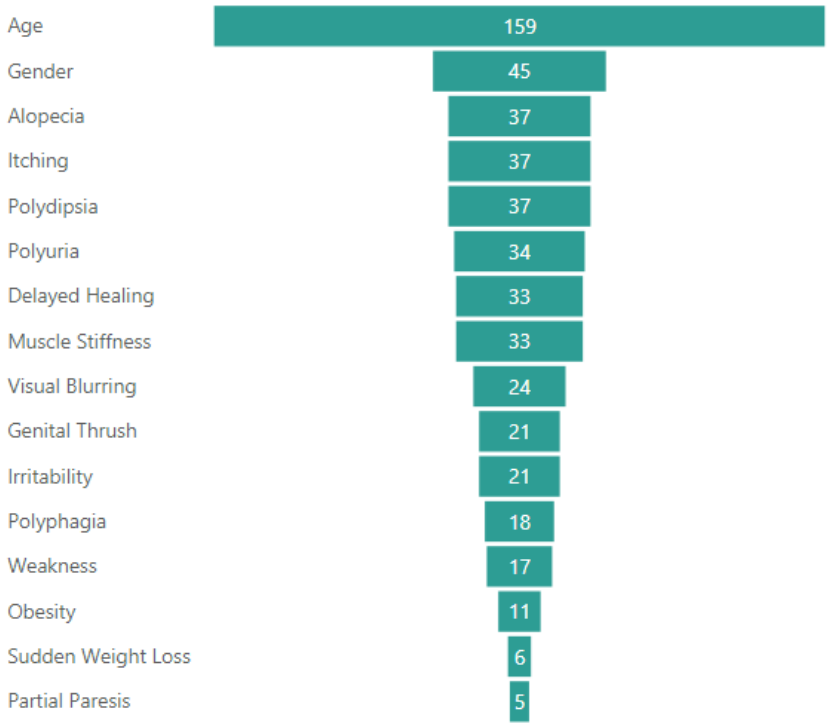


Figure 3.8 Calculated feature importances using XGBoost method for early stage diabetes risk prediction dataset

3.5 K- Nearest Neighbor

K nearest neighbor is a lazy learner algorithm which is a simple machine learning method. An assumption that instances of the same class will be closely positioned geometrically is made and test data are predicted according to this assumption. Labels of these closest n neighbors are considered and majority voting is applied to make predictions. K nearest neighbor algorithm is useful especially when train data is roughly divided into clusters and these clusters are discrete.

Neighbor considering step is shown in Figure 3.9 [59] . Considering 3 neighbors, given instance will be labeled as class B however, considering 6 classes given instance will be labeled as class A.

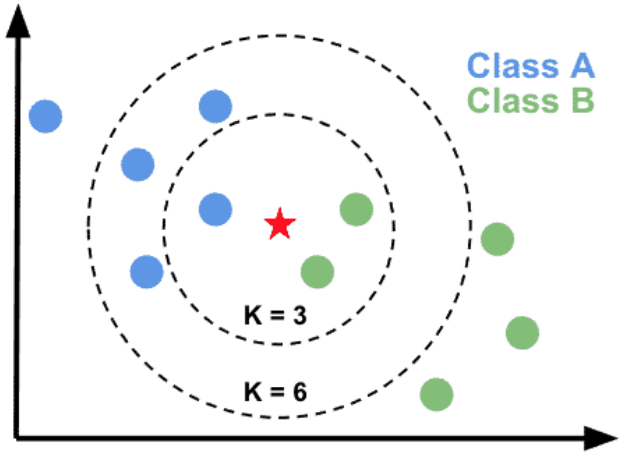


Figure 3.9 K nearest neighbor working principle [59]

In this study, k is selected as 7 which the class assignments performed by observing 7 nearest neighbours.

3.6 Gaussian Naive Bayes

Bayes theorem is a theorem that describes conditional probability of possible causes for a given outcome using probability of each cause and the conditional probability of the outcome of each cause. Let P(A) be the probability of A occuring, P(B) be the probability of B occuring, P(A|B) be the probability of A given B and P(B|A) be the probability of B given A, Bayes theorem is explained in Equation 3.2.

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)} \tag{3.2}$$

Naive Bayes is a method based on the Bayes theorem. An assumption that all features are independent are made so it does not produce good results when there are lots of features Gaussian Naive Bayes is a variant of the original Bayes method that uses Gauss normal distribution. An assumption that continuous values of classes are distributed according to a Gaussian distribution. Let x be the continuous variable and y be the associated class, Gaussian Naive Bayes theorem is explained in Equation 3.3.

$$P(x \mid y = c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} \left(-\frac{(x_i - \mu_c)^2}{2\sigma_c^2}\right) \tag{3.3}$$

3.7 Artificial Neural Network

Artificial neural networks (ANN) aim to imitate the mechanism of solving problems of the human brain. A network that includes nodes and connections to make predictions is created.

For weight initialization, random weight initialization is applied to each connection and weights are updated by calculating the loss according to the train data. To predict outcomes for a problem that has n possible outcomes, n nodes are set to the end of the network so that each one will return the possibility of current instance be belonged to that class. The input would be loaded into the input layer, which would then distribute it to the hidden layers. The input is typically in the form of a multidimensional vector. The learning process is when the hidden layers consider decisions from the previous layer and determine if a stochastic change inside itself worsens or improves the output [60].

Where n denotes the number of hidden neurons, f indicates the activation function of the output layers, and b_k refers the bias of neuron k of the output layer, w_{ki} signifies the weight connecting neuron k of the output layer and neuron j of the hidden layer. h_i denotes the output from the hidden layer neuron. The model's expected result y is shown in Equation 3.4 [61].

$$y = f(\sum_{i=1}^{n} w_{ki} \times h_i + b_k) \tag{3.4}$$

In this part of the study, a neural network which has 500 neurons and a sigmoid activation function is built. Sigmoid activation function is the part where output is determined as 0 or 1. Sigmoid function returns an output between the range of 0 to 1. Then the output is rounded to 0 or 1 to return a prediction. There are different activation functions too, but sigmoid is the most appropriate one for binary classification. Equation of Sigmoid function is given in Equation 3.5.

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{3.5}$$

Sigmoid activation function is shown in Figure 3.10 [62].

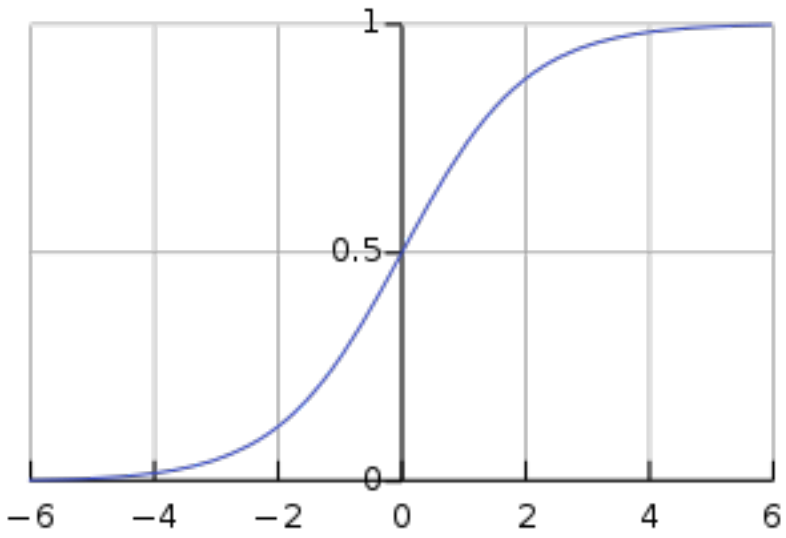


Figure 3.10 Sigmoid activation function [62]

3.8 Convolutional Neural Network

Convolutional neural network (CNN) is a deep learning methodology that is mainly used for image data. It filters the input data before feeding it into the neurons. It is helpful to reduce input size by eliminating uninformative features. Convolution, pooling and flatten layers are applied respectively, then output of these three layers are fed into the neural layer. It can be applied to numerical data, too. While 2 dimensional CNN architecture are used for images, 1 dimensional CNN may be used for numerical datasets.

The fact that CNNs are largely employed in the field of pattern recognition within images is the sole significant distinction between CNNs and conventional ANNs. This enables us to incorporate image-specific elements into the architecture, improving the network's suitability for image-focused tasks and lowering the number of parameters needed to create the model [60].

In this part of the study, 1 dimensional CNN which has 2 convolutional layers is applied over the diabetes dataset. Convolution layers have 8 and 4 filters. A dropout layer was added to the output of second convolutional layer for avoiding overfitting, then pooling and flattening layers were located at the end of the CNN structure. Flattened data is fed into the neural network which has 500 neurons and a sigmoid activation function.

$$\Delta l = h\Delta\theta \tag{3.6}$$

$$\frac{\Delta l}{\Delta t} = h\frac{\Delta \theta}{\Delta t} \tag{3.7}$$

$$\lim_{\Delta t \to 0} \frac{\Delta l}{\Delta t} = \lim_{\Delta t \to 0} h\frac{\Delta \theta}{\Delta t} \tag{3.8}$$

$$\lim_{\Delta t \to 0} \frac{\Delta l}{\Delta t} = \frac{dI}{dt} \tag{3.9}$$

Convolutional neural network consists of layers such as convolution, pooling, flattening and dense layers. These layers are used for;

- Convolutional layer is the layer where data is filtered to get the details that are helpful to find patterns. An example of convolution operation is shown in Figure 3.11 [63].

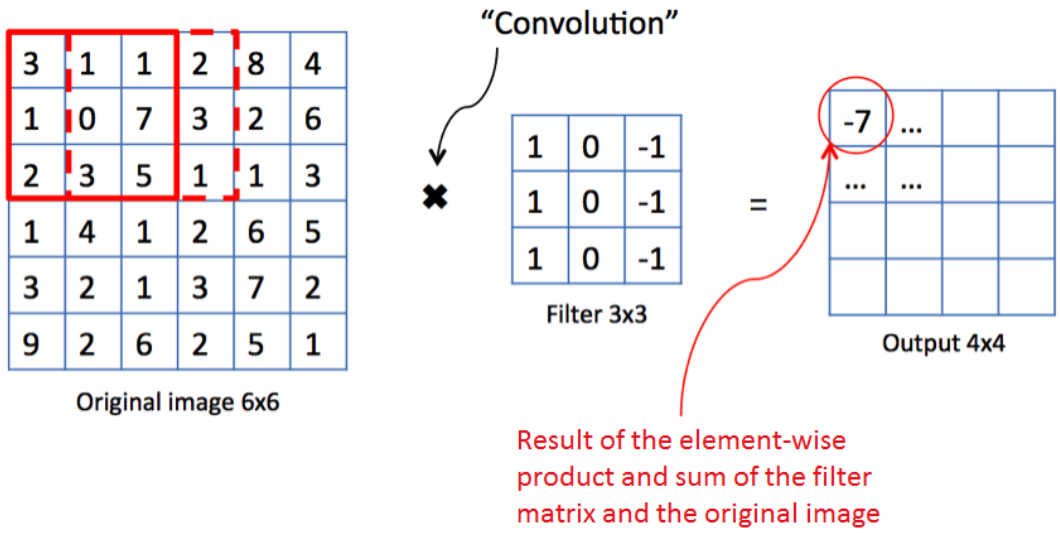


Figure 3.11 An example of the convolution operation [63]

- Pooling layer is the layer where unnecessary elements are removed. It is based on the assumption that neighbor elements are alike. It can be applied in different ways such as by keeping maximum valued element at that neighbor or getting average of the neigbors. An example of maximum pooling operation is shown in Figure 3.12 [64].

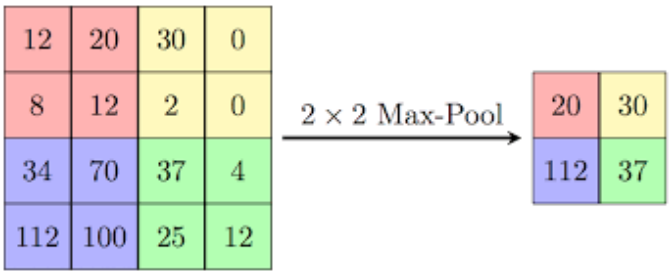


Figure 3.12 An example of the maximum pooling operation [64]

- Flattening layer is the layer where processed matrices are converted to a vector that is appropriate for feding to dense layer. An example of maximum pooling operation is shown in Figure 3.13 [65].

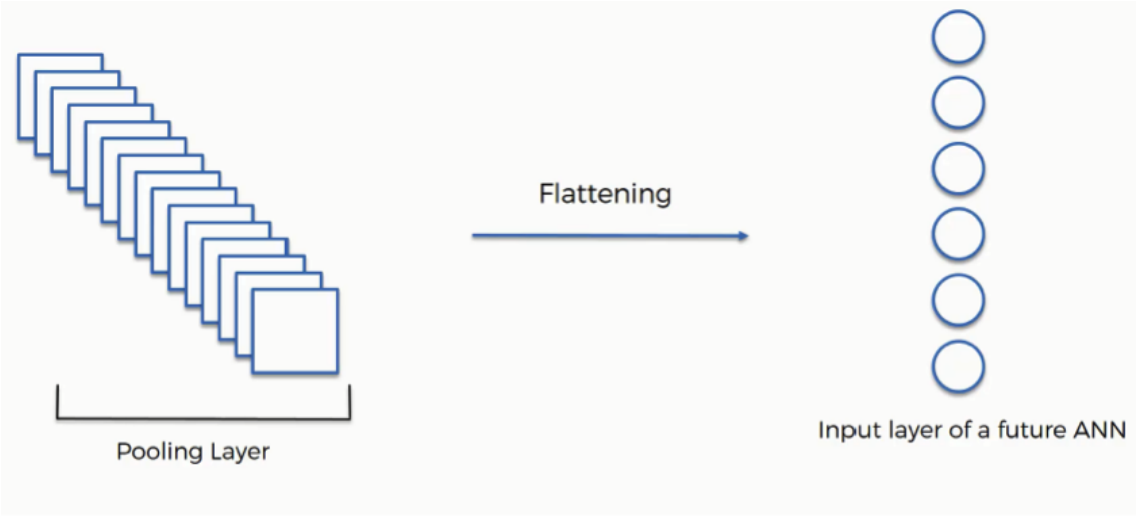


Figure 3.13 An example of the flattening operation [65]

- Dense layer is the layer that includes neurons and mathematical calculations take place at this step. Output of this layer is the model prediction. An example of dense layer architecture is shown in Figure 3.14 [66].

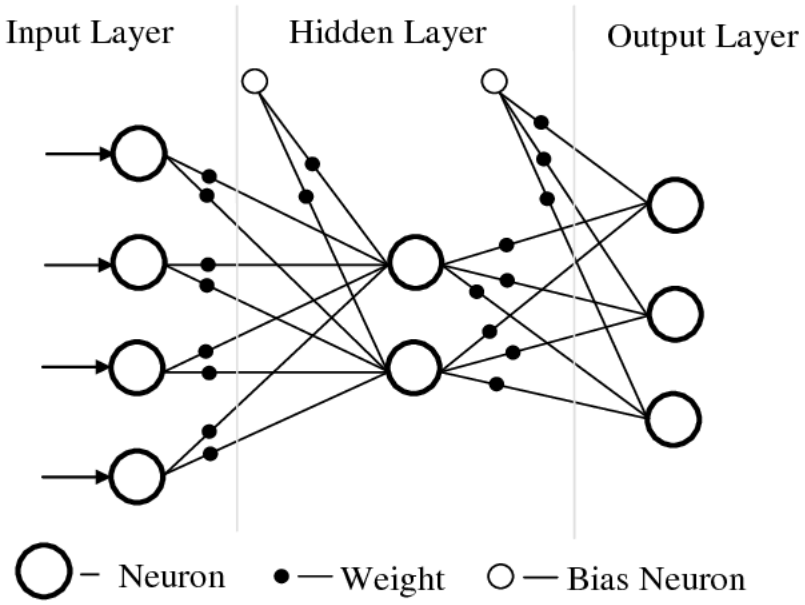


Figure 3.14 An example of dense layer [66]

An example architecture of the full convolutional neural network architecture which includes convolution, pooling, flattening and dense layers is shown in Figure 3.15 [67].

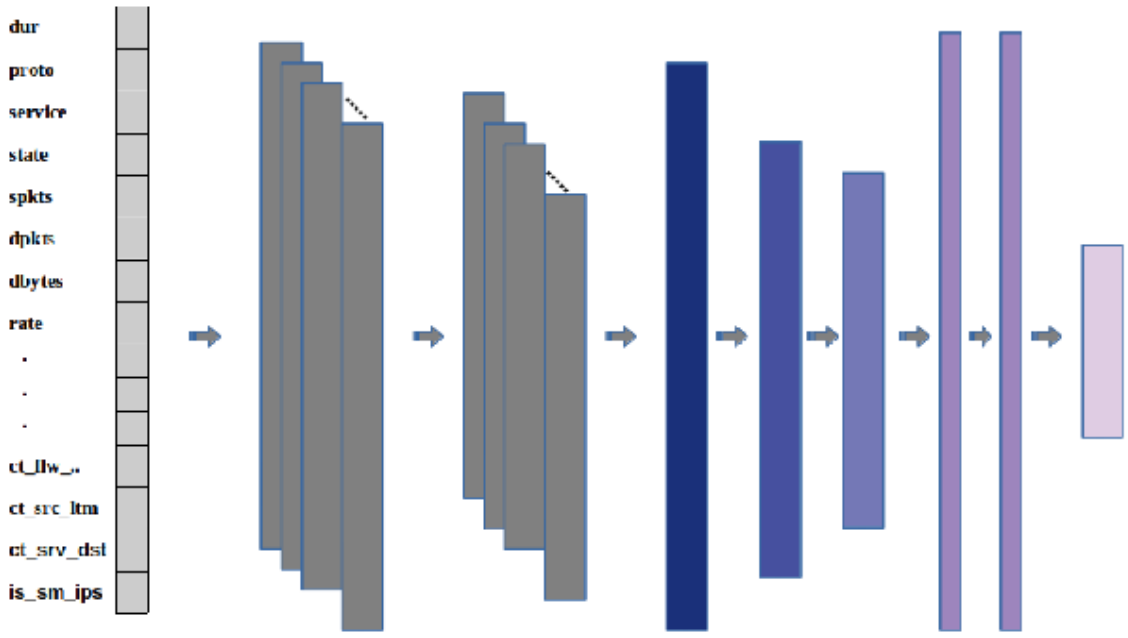


Figure 3.15 Convolutional neural network architecture [67]

Parameters of each model used in the experiment for the determination of diabetic status are summarized in Table 3.1.

Table 3.1 Optimized model parameters

Method	**Parameters**
Decision Tree	Gini information gain, 2-level pruning, max depth:8
Random Forest	1000 estimators, max depth: 8
Support Vector Machine	3rd degree polynomial kernel, 0.1 regularization factor
XGBoost	Learning rate: 0.1 max depth: 8
K-Nearest Neighbor	7 neighbors
Gaussian Naive Bayes	Default values are applied
Artificial Neural Network	500 neurons, sigmoid activation
Convolutional Neural Network	2 convolutional layers, 500 neurons, sigmoid activation

4
PREDICTING THE SEVERITY LEVEL OF DIABETIC RETINOPATHY USING ENSEMBLE OF DEEP NETWORKS

In this part of the study, ensemble approaches of the utilized five deep learning models has been proposed for the classification of the severity of DR in APTOS 2019 dataset. Several de-noising processes have been applied over the images before the training phase. VGG16, VGG19, EfficientNetB3, EfficientNetB7 and EfficientNetv2S deep neural network architectures were selected as the base learners of the proposed ensemble approach for decision level fusion step. 1D-CNN is designed to use in the stacking generalization phase. The flowchart of the presented study is summarized in Figure 4.1 and details will be given in the following sections.

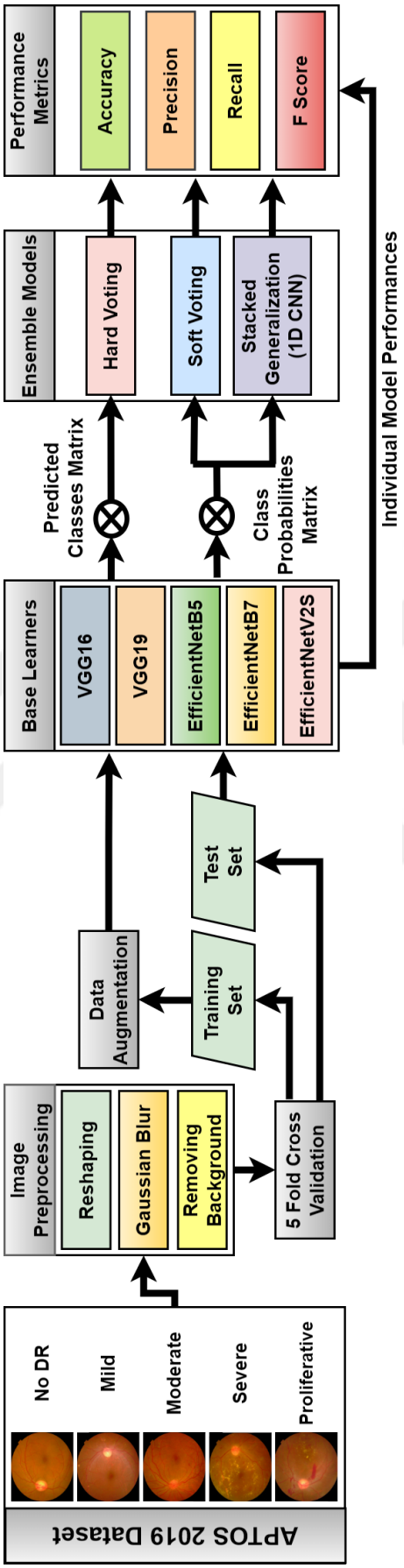


Figure 4.1 Flowchart of the proposed approach

4.1 Base Learners: Deep Learning Models

Machine learning is a subbranch of artificial intelligence that aims to use data in order to acquire predictive models for complex problems [68]. Deep learning is a subbranch of machine learning that is based on the idea of using neural networks. As distinct from classical neural networks, deep networks learn the relation among variables, also the knowledge that governs the relation as well as the knowledge that makes sense of the relation [69]. Therefore, deep learning methods result in much more effective and accurate results in complex problems such as diagnosing DR using existing retinal images.

In this work, deep neural networks were utilized as base learners of the proposed ensemble architecture for predicting the presence and severity level of DR over retinal images from one of the most well-known datasets namely APTOS 2019. Five deep learning models namely VGG16, VGG19, EfficientNetB5, EfficientNetB7 and EfficientNetV2S were selected. Transfer learning approach is applied using these 5 models where the models were pretrained by a huge dataset, Imagenet [70]. Some of the dense and dropout layers were modified in order to get better results for APTOS 2019 data. In the training phase, Adam optimizer with a learning rate of 0.0001 and batch size of 32 were fixed. Epoch number was set to 25 with an early stopping functionality.

EfficientNet is a CNN architecture that uniformly scales all dimensions of depth, width and resolution using a compound coefficient. Unlike classical practice that randomly scales these factors, the scaling method of EfficientNet uniformly scales network dimensions with a set of fixed scaling coefficients [71]. Figure 4.2 shows general architecture of EfficientNet models. EfficientNet architecture is especially applied, since former study [71] states this architecture may give better results than others. Measured performances are compared in Figure 4.3 [71].

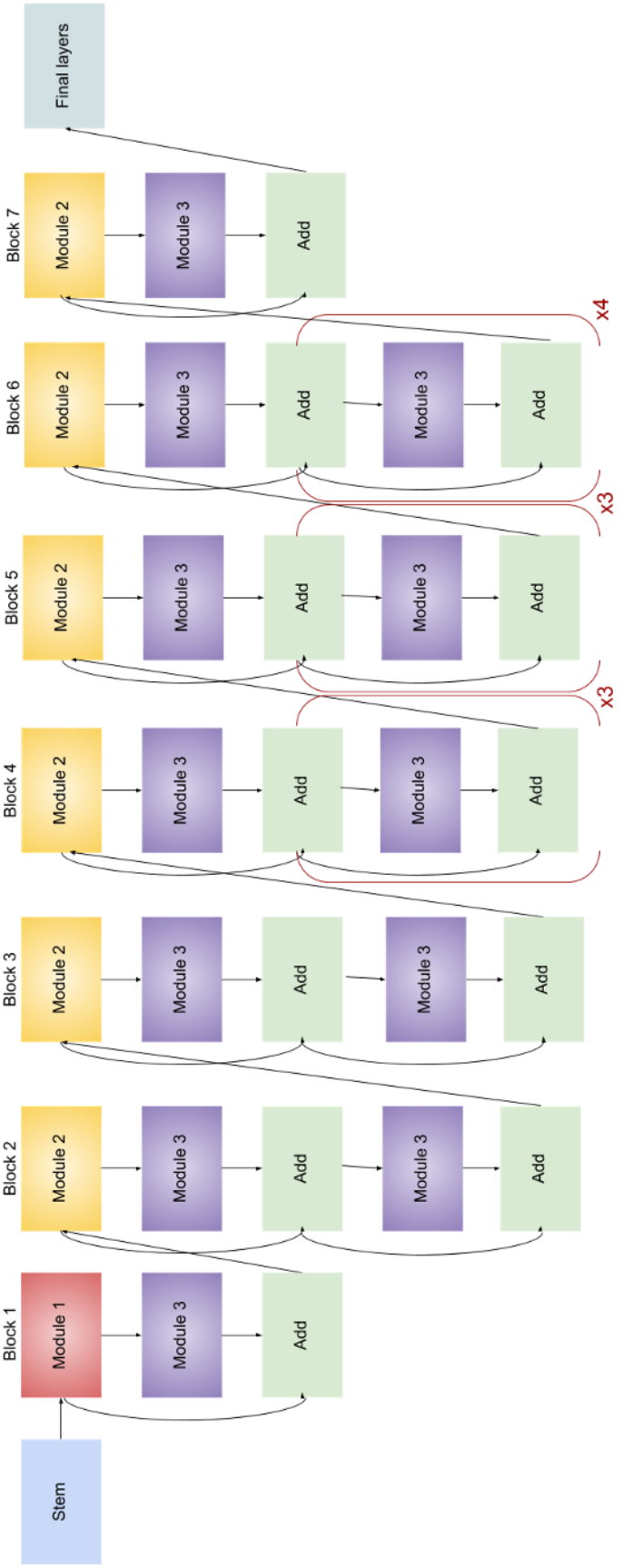


Figure 4.2 General architecture of the EfficientNet models [71]

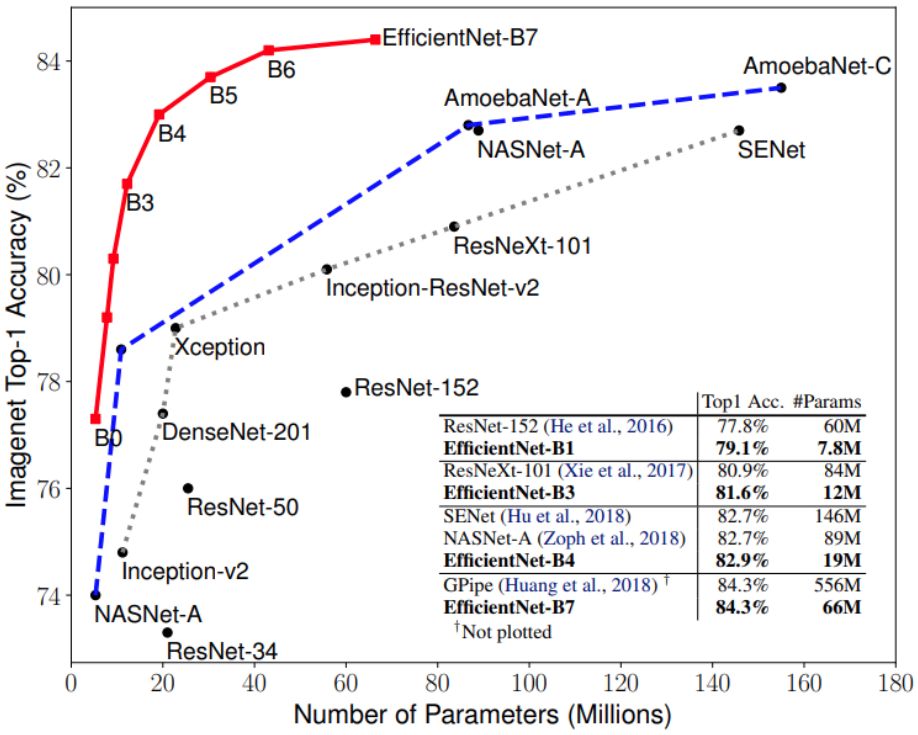


Figure 4.3 Performance of EfficientNets compared to other models [71]

VGG16 and VGG19 are well-known architectures. At some datasets, they provide better results than other architectures. Figure 4.4 [72] shows model performances over a health-related dataset.

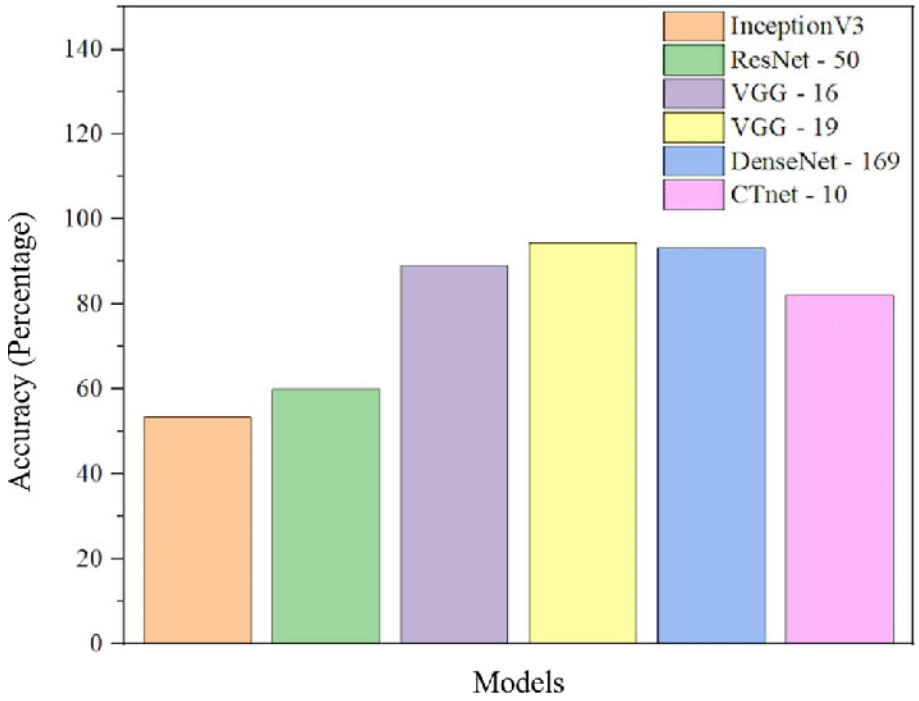


Figure 4.4 VGG16/19 model performances compared to other models [72]

VGG16 architecture is shown in Figure 4.5 [73] and VGG19 architecture is shown in Figure 4.6 [74] .



Figure 4.5 VGG16 layers [73]

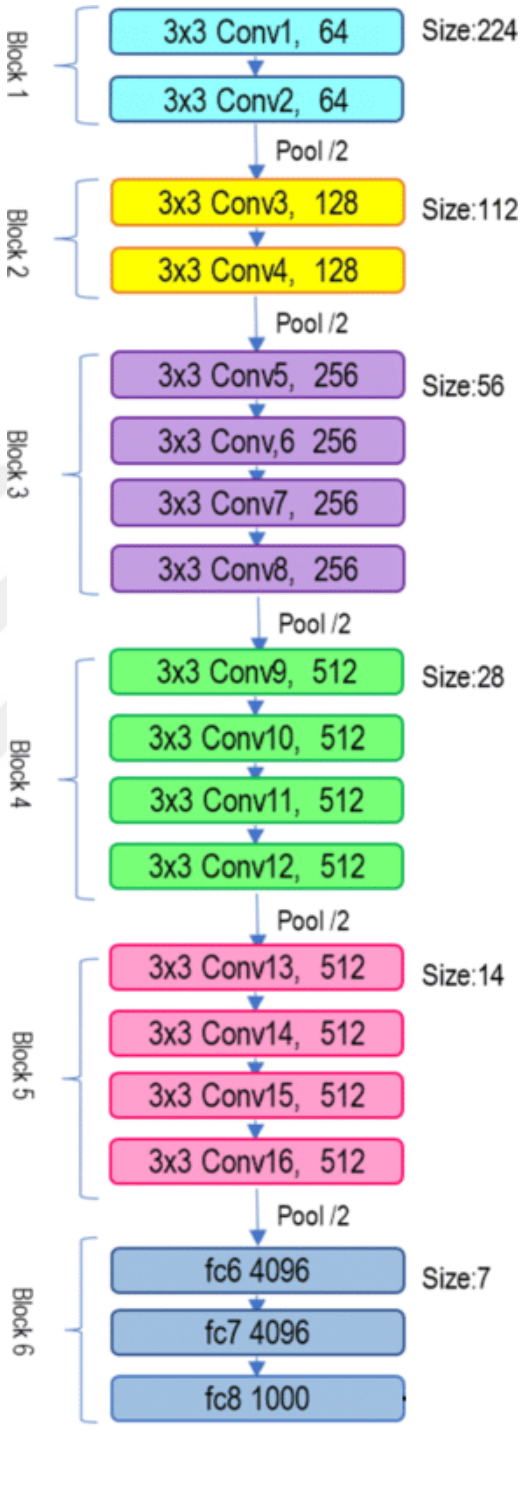


Figure 4.6 VGG19 layers [74]

In order to obtain the highest base learner classification accuracy, fine tuning process for hyper-parameters was performed as preliminary study. For each base learner,

different combinations of hyper-parameters were investigated such as number of neurons, learning rate, dropout rate, activation functions and optimization algorithm. In addition to hyper-parameter tuning, early stopping was accepted as the training termination criteria to prevent overfitting of the models. In the determination of the optimum learning rates, learning rate scheduling, which reduces the learning rate according to a pre-defined schedule was applied. Learning rate was not changed during the first 10 epochs, to let the optimizer take large steps before it reaches to a close point to the global optimum. After that, at the end of every 10 epochs, learning rate was reduced by the factor of 0.1. By doing so, it is aimed to let optimizer check a closer solution using given learning rate for 10 epochs. The parameters resulted in the best performance for each base learner were presented in Table 4.1.

Table 4.1 Training parameters of the models

Parameter	Configuration				
	VGG16	VGG19	EfficientNet B5	EfficientNet B7	EfficientNet V2
Batch Size	32				
Activation Function	ReLU				
Optimizer	Adam				
Loss Function	Sparse Categorical Crossentropy				
Learning Rate	0.0001				
Beta 1	0.9				
Beta 2	0.999				

Since class distribution of the dataset is highly imbalanced, class weight balancing method was applied. While training the model, a weight value which is inverse proportioned by the number of instances of each class is assigned to the classes as a coefficient. It changes the penalty of misclassification of each class, considering misclassification of a rare class as a worse mistake. Although model tends to produce more common class as output, applying class weight based penalty reduces this tendency.

4.2 Ensemble Models

Ensemble learning is the idea of developing multiple individual learners and making predictions using decisions of each base learner. It depends on the assumption that base learners will make wrong predictions for some data, but a mistake of a base learner can be tolerated by the other base learners by voting mechanism. This is also

known as decision level fusion of the deep neural networks. This approach is beneficial by producing both independent and individually accurate decision makers. If the base learners are not independent enough, they will mispredict the same samples, so voting mechanism will not help to tolerate those mispredictions. This mechanism is shown in Figure 4.7. It is shown that ensemble accuracy may be much more high than base learners if there are enough diversity.

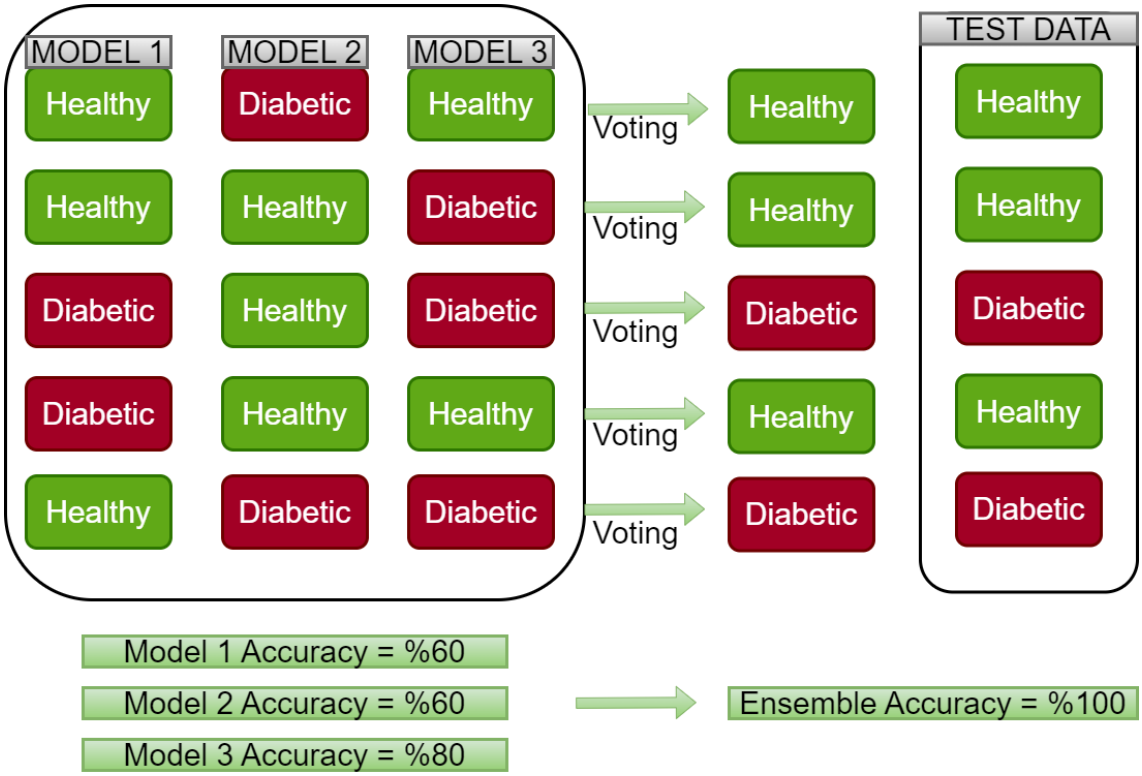


Figure 4.7 Importance of independent base learners to tolerate mispredictions

Also, if base learners are not successful enough to make accurate decisions, their predictions will not be useful. Decision level fusion approach is shown in Figure 4.8 [75].

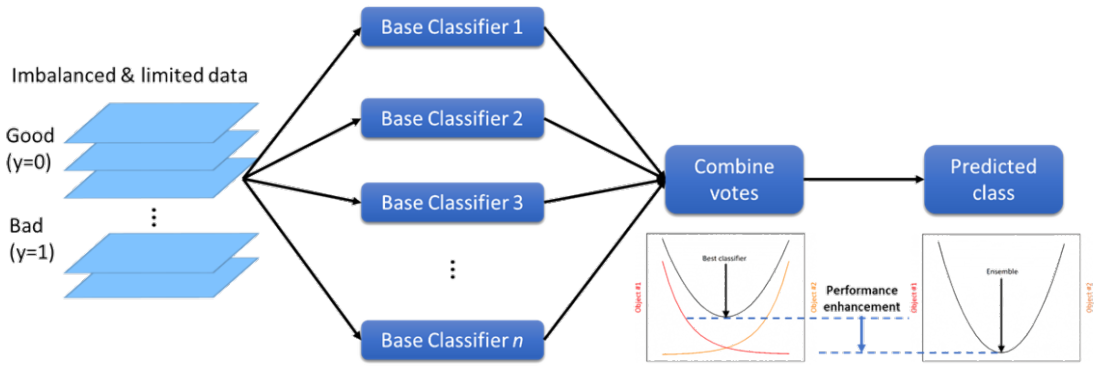


Figure 4.8 Ensemble learning approach [75]

In this work, three decision level ensemble architectures over the individual learners are applied to increase accuracy of predictions. 3 ensemble architectures are developed using the trained individual learners, to build more accurate models. These are hard voting, soft voting and stacked generalization architectures.

4.2.1 Hard Voting

Hard voting is an ensemble idea that combines predictions of multiple base learners applying voting approach. In this work, five base learners that are build using deep learning approach were trained with the same data and the predicted result of each base learner for the same test images were gathered. Then, applying voting approach, majority of the predictions is assigned as the output class of the given test image in hard voting based ensemble approach. The implementation of the hard voting is shown in Figure 4.9 over a retinal image.

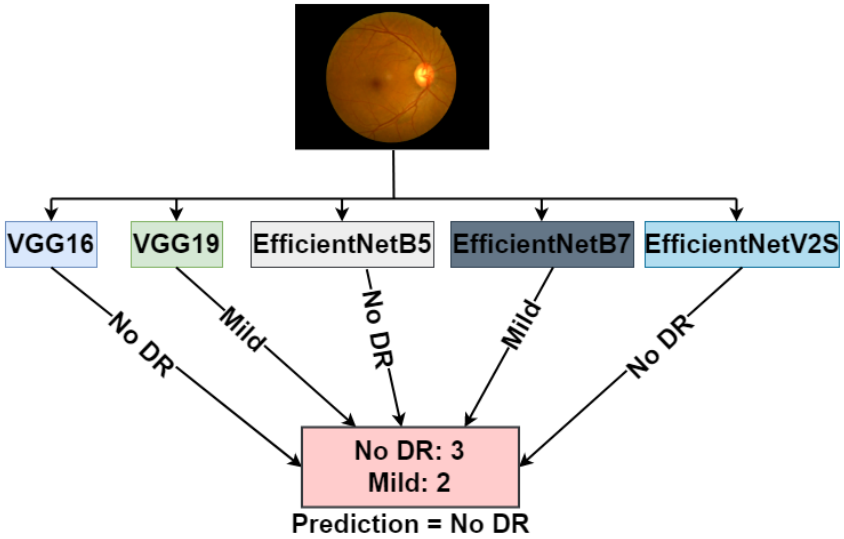


Figure 4.9 The implementation of hard voting method over an example retinal image

4.2.2 Soft Voting

Similar to hard voting approach, soft voting is also a decision level fusion method that uses the output of multiple base learners. Unlike hard voting, class probabilities of each base learner are considered to make the final prediction. Class-wise probabilities that are produced by 5 base learners are summed for each image, output of the ensemble is decided using the majority of the sum of total probabilities. Base models are trained by the same train images and output classification probabilities of these models obtained. The highest probability ratio defines the final prediction of the ensemble. The implementation of the soft voting methodology is shown in Figure 4.10 over a retinal image.

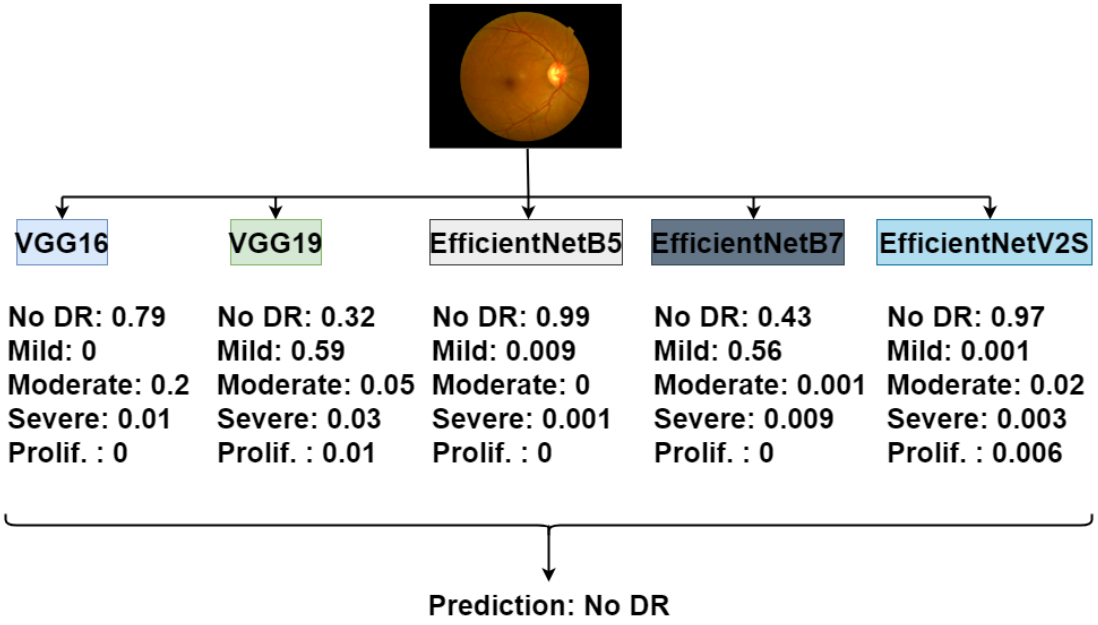


Figure 4.10 The implementation of soft voting method over an example retinal image

4.2.3 Stacked Generalization

Similar to hard and soft voting approaches, stacked generalization is another ensemble learning technique that uses the predictions of multiple base learners to make a final prediction. Predicted class probabilities of each base learner are gathered for training data then a new model is trained using these predictions. By doing so, it is aimed to gather insights from the predictions of base learners and build a more robust model to make final prediction [76]. Former research [77] state using stacking idea may lead to better results than classical methods. This may be achieved by using predicted classes or prediction probabilities of base models. Existing research states that using base learner prediction probabilities may produce more accurate results [78]. Using this information, a stacked generalization based ensemble that uses prediction probabilities of base learner models is implemented for predicting the severity level of diabetic retinopathy. Prediction probabilities of five base learners are fed to meta-learner 1D-CNN model. Since former studies [79–81] state that 1D-CNN models give more accurate results than classical machine learning algorithms, we used 1D-CNN architecture to fuse the decisions of each base learner. The flowchart of stacked generalization idea is shown in Figure 4.11.

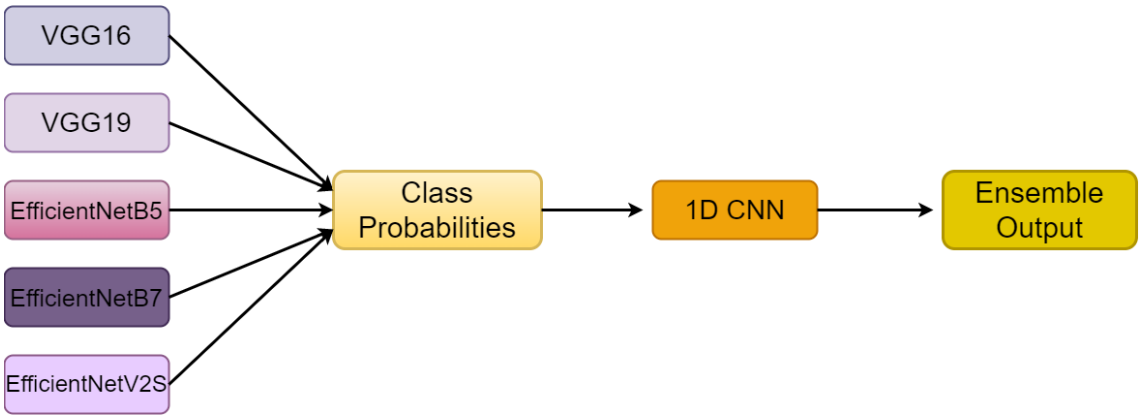


Figure 4.11 Stacked generalization flowchart

The applied stacked generalization process over an example image is shown in Figure 4.12.

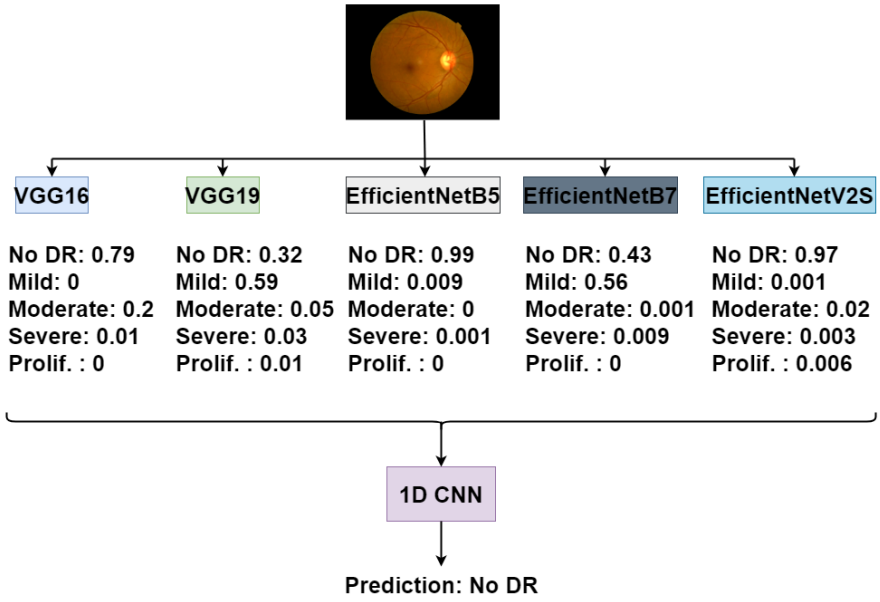


Figure 4.12 The implementation of stacked generalization method over an example retinal image

4.3 Performance Metrics

For evaluating model performances, accuracy, recall, precision and f1-score metrics are used in the presented work. Accuracy is the ratio of correctly predicted instances among all dataset. Precision shows how many samples labeled as positive is actually positive. Recall is the metric to show how many positive samples are labeled as positive. F1-score calculates harmonic mean of precision and recall scores to show if precision and recall are both high enough. The equations of the performed metrics

are given below where TP, TN, FP and FN indicate the true positive, true negative, false positive and false negative rates, respectively.

$$Accuracy = \frac{TP}{TP + TN + FP + FN} \tag{4.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.2}$$

$$Recall = \frac{TP}{TP + FN} \tag{4.3}$$

$$F1 - Score = 2 \ x \ \frac{Precision * Recall}{Precision + Recall} \tag{4.4}$$

On the other hand, confusion matrix is a commonly used way to show results of classifiers. Structure of confusion matrix is shown in Figure 4.13.

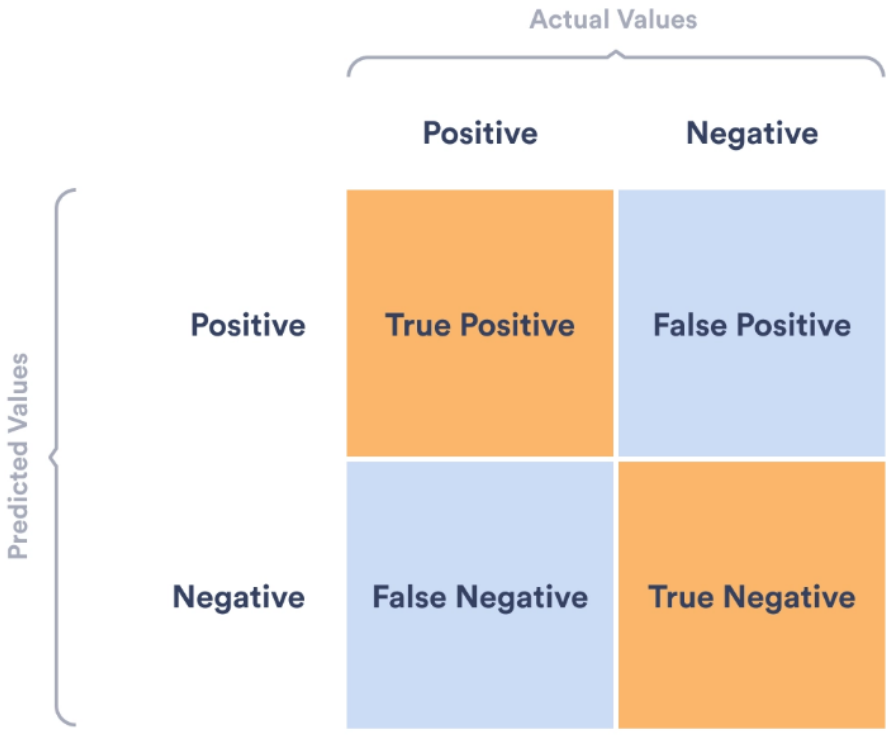


Figure 4.13 Structure of the confusion matrix

Considering the confusion matrix, performance metrics like accuracy, precision, recall

and f-score can be easily interpreted. Also, behavior of the model can be understood checking which classes it tends to misclassify.

5
RESULTS AND DISCUSSIONS

In this chapter, programming environment is detailed then obtained results for the diagnosis of diabetes and severity level prediction of diabetic retinopathy are explained in the following sections.

5.1 Programming Environment

This study is carried out using Python [82] programming language. Python is a programming language that is especially suitable for implementing data science concepts such as data analysing, data processing, scientific computing, matrix operations, image processing, machine learning and deep learning. Besides being very adventegeous for data science applications, Python became one of the most popular programming languages which is shown in Figure 5.1 [83].

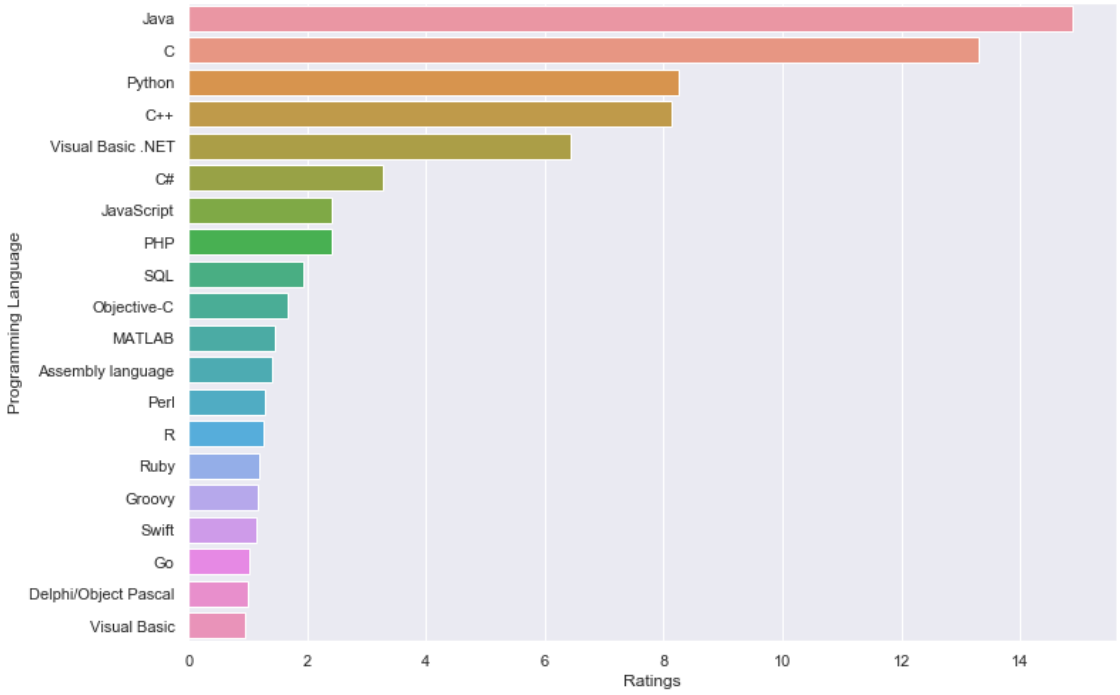


Figure 5.1 Popularities of programming languages [83]

Python has a lot of libraries available for applying data science techniques, some of them are listed below.

- Numpy [84]: Numpy includes plenty of mathematical functions which is optimized in terms of performance. It can be interpreted as well-optimized C code. It is understandable and easy to use.
- Pandas[85]: Pandas is a data manipulation library which makes it much easier to clean and analyze data. It represents data as an understandable way.
- Scikit Learn [86]: Scikit Learn is a machine learning library which is used for predictive data analysis. It has lots of functions that implements classical machine learning methods. It also provides functions to validata data such as train-test splitting, k fold validating, performance metrics measuring.
- OpenCV [87]: OpenCV provides a real time infrastructure for computer vision applications. It is developed using C++ language but it has wrappers for other programming languages such as Python, Ruby, Perl etc.
- Matplotlib [88]: Matplotlib is a data visualization library which helps to see data in many ways.
- Tensorflow [89]: Tensorflow is a deep learning library which makes it possible to build and deploy deep learning applications easily. It is developed by Google Brain team [90] in order to use internal Google research. It has an architecture which is flexible so that it provides easy deployment across a lot of platforms (CPUs, GPUs, TPUs), and a lot of devices such as desktops, clusters of servers and mobile devices.
- Keras[91]: Keras is a deep learning library and neural network Application Programming Interface (API) that can work on Tensorflow. It has plenty of methods for artificial neural network approaches such as Convolutional Neural Network and Recurrent Neural Network. It has built-in pretrained Convolutional Neural Network models such as VGG16, VGG19 and EfficientNet. It also provides performance measurement functions.

  There are also another built-in deep learning libraries but Tensorflow and Keras are mostly used ones. Library using statistics are shown in Figure 5.2 [92]. So in this work, these 2 libraries are used to implement models.

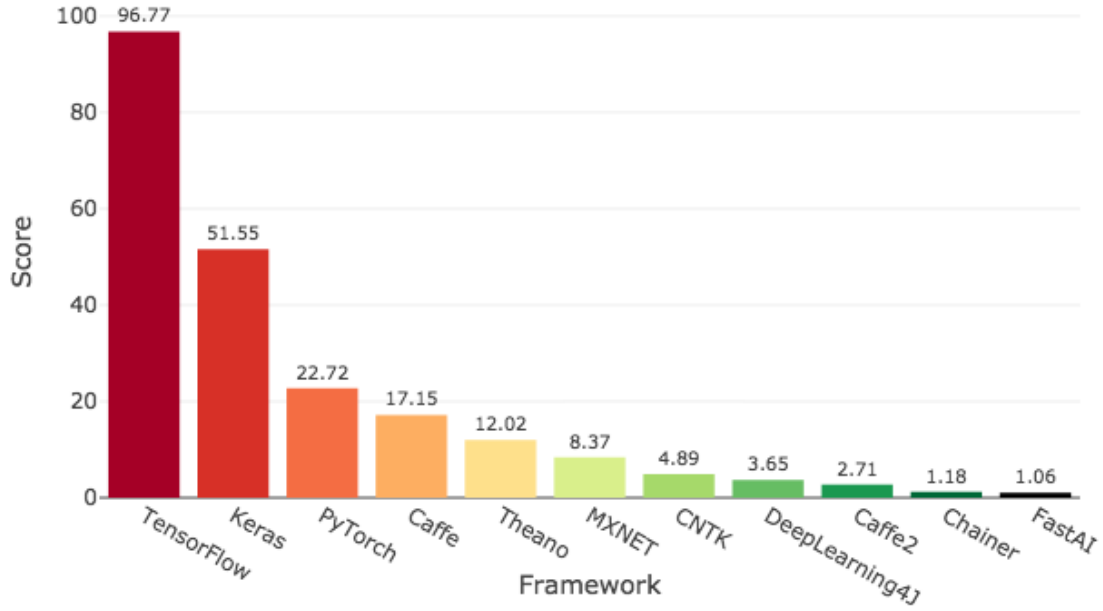


Figure 5.2 Deep learning libraries according to how often they are used [92]

5.2 Early Stage Diabetes Risk Prediction Results

Model performances over the diabetes risk prediction dataset are evaluated in terms of accuracy, recall, precision and f-score are presented in Table 5.1. Results are validated by using a 5-fold cross validation schema in which training and testing sets were arranged as 416 and 104 samples for each fold. General performance scores listed in Table 5.1 were calculated by averaging the fold results. According to the results, the highest classification accuracy is achieved by 1D CNN. An accuracy of 99.04% is obtained for the diagnosis.

Method	Accuracy	Precision	Recall	F-Score
K-Nearest Neighbor	89.42	88.92	90.34	89.09
Decision Tree	95.58	95.17	95.58	95.34
Random Forest	97.69	97.69	97.53	97.57
Support Vector Machine	94.62	94.18	94.51	94.31
Naive Bayes	88.85	88.30	88.11	88.16
XGBoost	97.89	97.80	97.77	97.76
Artificial Neural Network	92.31	91.82	92.40	92.07
Convolutional Neural Network	99.04	100.00	98.63	99.31

Table 5.1 Model performances over early stage diabetes risk prediction dataset

In terms of precision, CNN achieved 100% performance. This shows that when the model labels a sample as patient, the probability of them being not patient is very low. Since this is a health related problem, recall is also a crucial metric. Labeling a healthy person as patient can be tolerated but labeling a diabetic patient as healthy may cause serious health issues. In terms of recall, all of the built models provided satisfactory results. The highest recall score is measured as 98.63% and it is measured when using the designed 1D CNN over the early stage diabetes risk prediction dataset. Also, the F-score metric helps us to balance recall and preciison scores. The highest F-score is 99.31% and it is also measured using 1D CNN. Considering these results, CNN model is the most well-performed model to diagnose diabetes.

While the proposed CNN architecture performed the best classification score, the lowest model performance is seen while using Naive Bayes method. All evaluated metrics are measured below 90%. Since the attributes of the dataset are not independent, assumption of independent attributes may lead to the observed performance. Built Naive Bayes model may label given samples wrongly.

In Table 5.2, the results of the proposed approach were compared with state-of-the-art studies that have dealt with the same data set. Considering these results, it can be said that applying 1D-CNN approach results in more accurate predictions.

Publication	Methodology	Accuracy %
Oladimeji et. al	Random Forest	98.31
Le et. al	Multilayer Perceptron	97
Sadhu and Jadli	Random Forest	98.08
Oleiwi et al.	Radial Basis Function Network	98.8
Bilgin	K Nearest Neighbor	98.08
Nahzat and Yağanoğlu	Random Forest	88.31
Proposed Framework	1D-CNN	99.04

Table 5.2 Performance comparision with state of art studies

5.3 Diabetic Retinopathy Severity Level Predicting Results

The experiments were performed using Python via Google Colab. In order to speed up the process, Nvidia Tesla T4 (16GB) was used as the GPU in the training phase of the models. In addition to the GPU module, the test platform had 12 GB of RAM and an Intel(R) Xeon(R) CPU with a processing speed of 2.2 GHz.

Since the images are collected under different conditions, several image pre-processing methods as removing the background, blurring, resizing, and data augmentation were applied to the images for sustaining a standard input to models. Five deep learning models namely VGG16, VGG19, EfficientNetB5, EfficientNetB7, and EfficienNetV2 were selected as base learners. In the training phase of each model, the identical augmented version of dataset was used as training set.

In the validation of the performances for the base and ensemble models, K-Fold Cross validation schema was utilized. K value is defined as 5 which refers the training and test set of data was split into 80% and 20%, respectively. According to fold validation schema, this arrangement was realized five times with differently created 20% test set for each fold. At the end of the tests, model performances are validated according to the scores obtained from the classification results indicated in confusion matrix of each folds.

Confusion matrices of the base learners and ensemble models are shown in the following figures below.

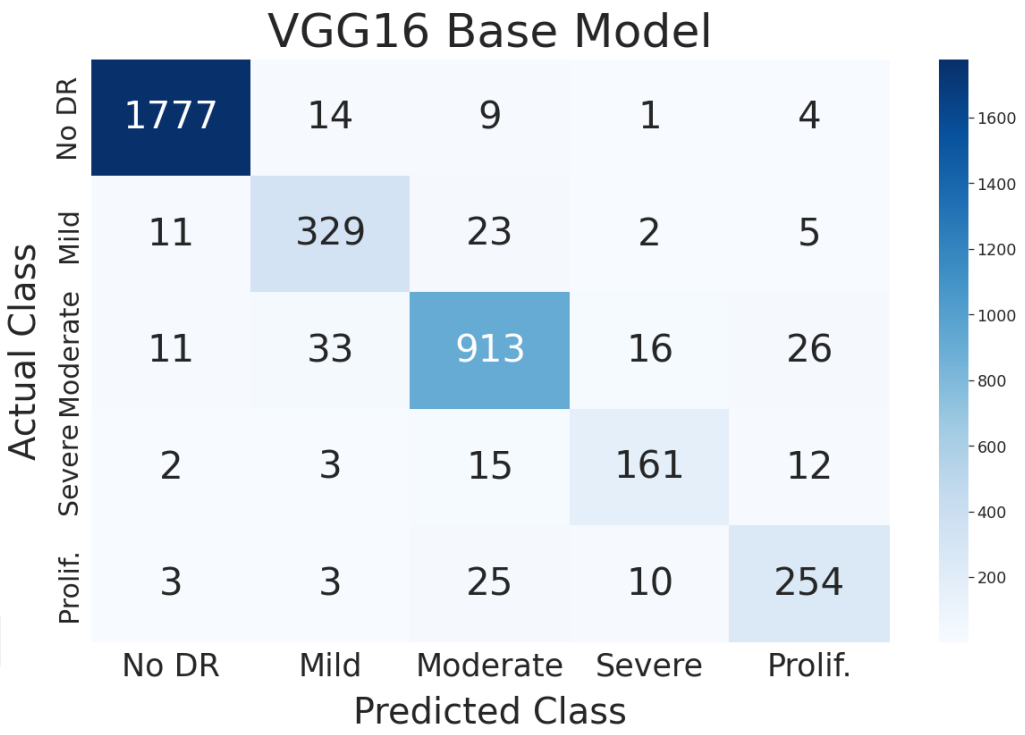


Figure 5.3 Confusion matrix of the VGG16 base model

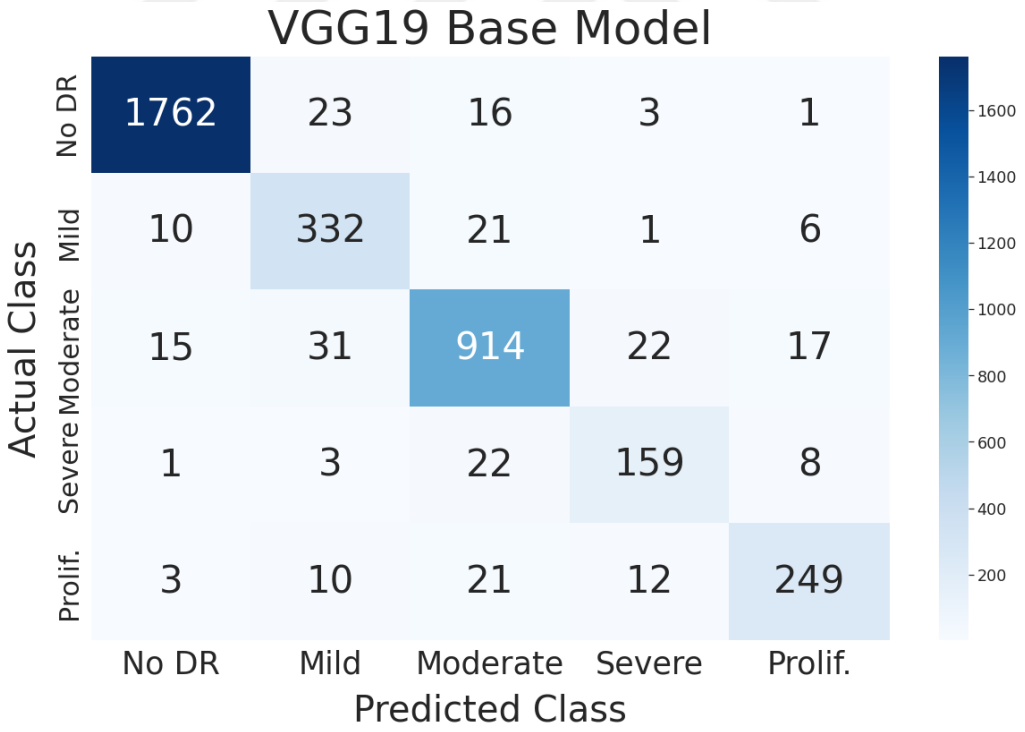


Figure 5.4 Confusion matrix of the VGG19 base model

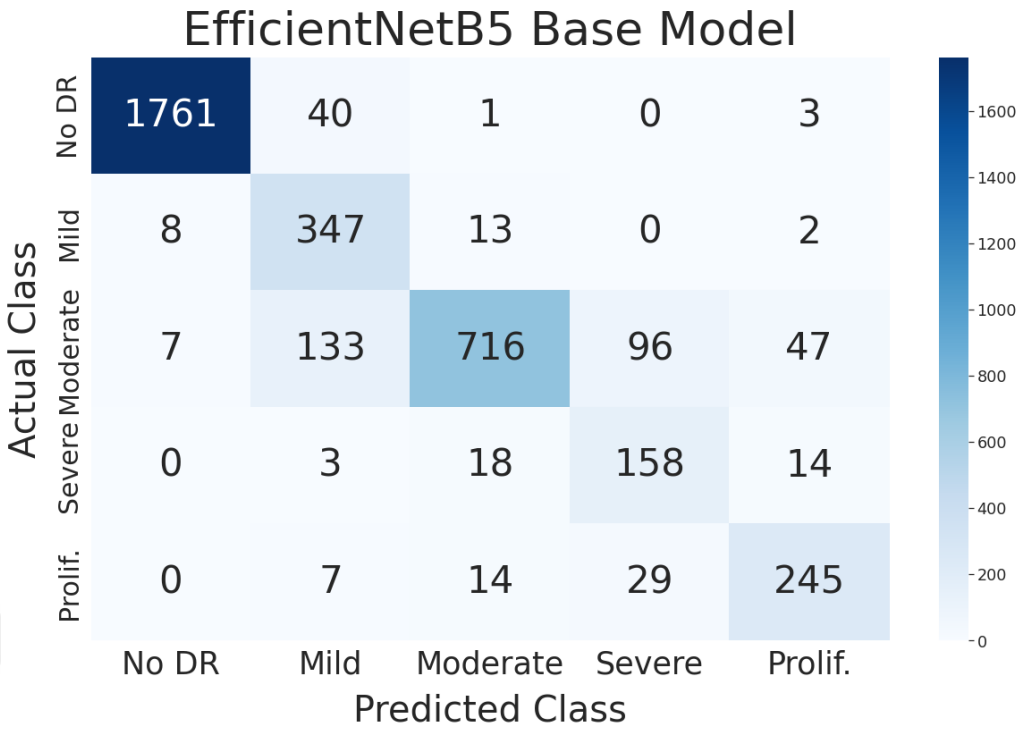


Figure 5.5 Confusion matrix of the EfficientNetB5 base model

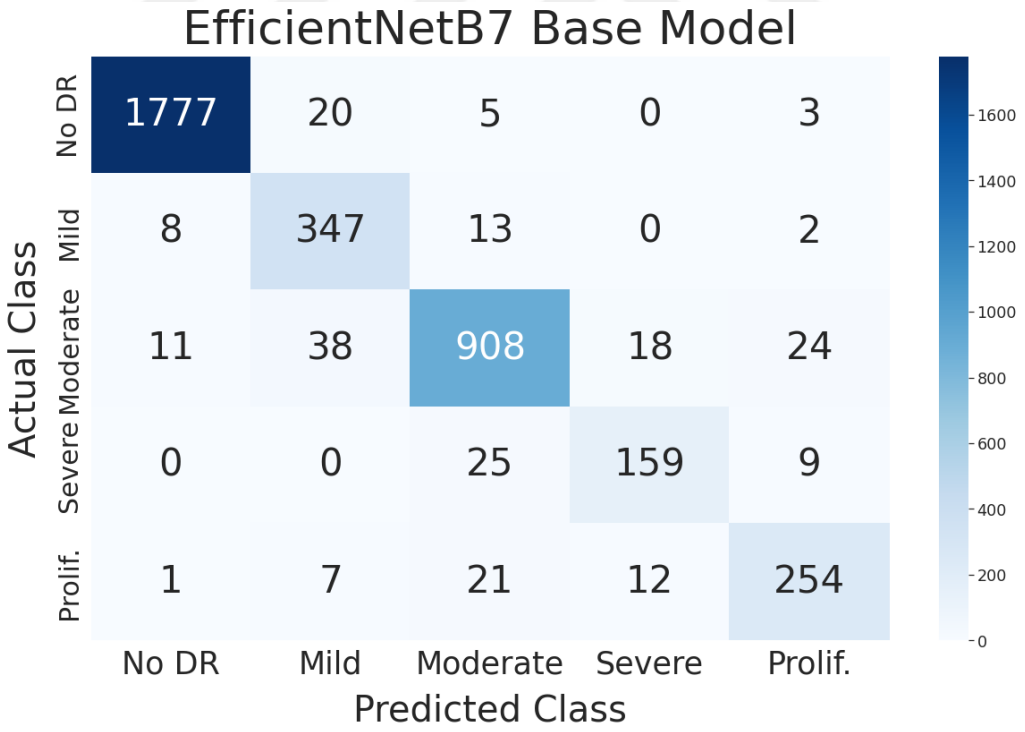


Figure 5.6 Confusion matrix of the EfficientNetB7 base model

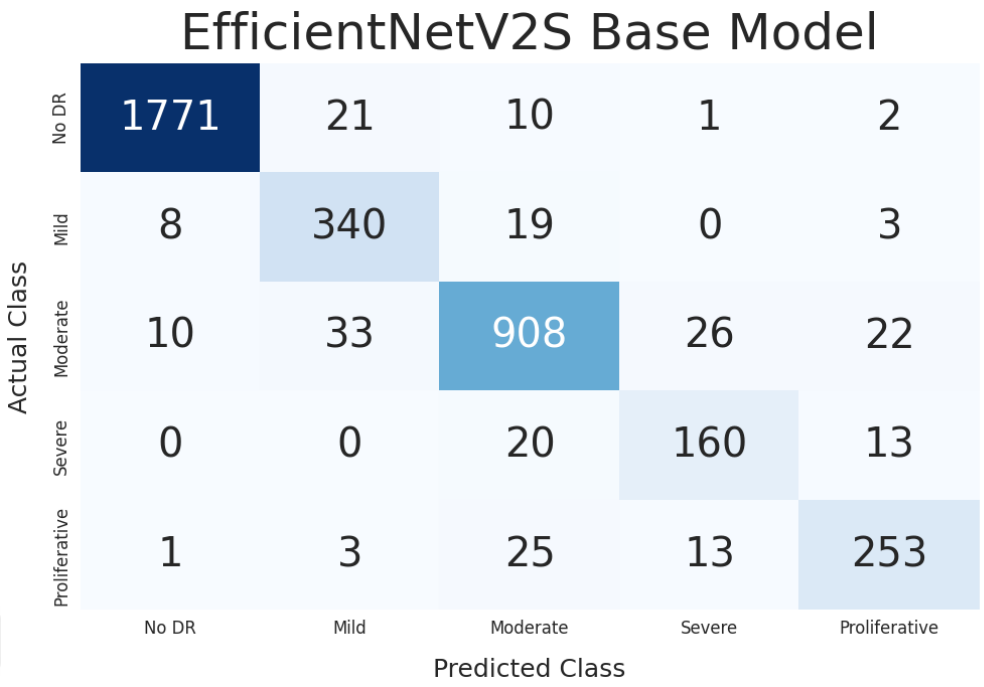


Figure 5.7 Confusion matrix of the EfficientNetV2S base model

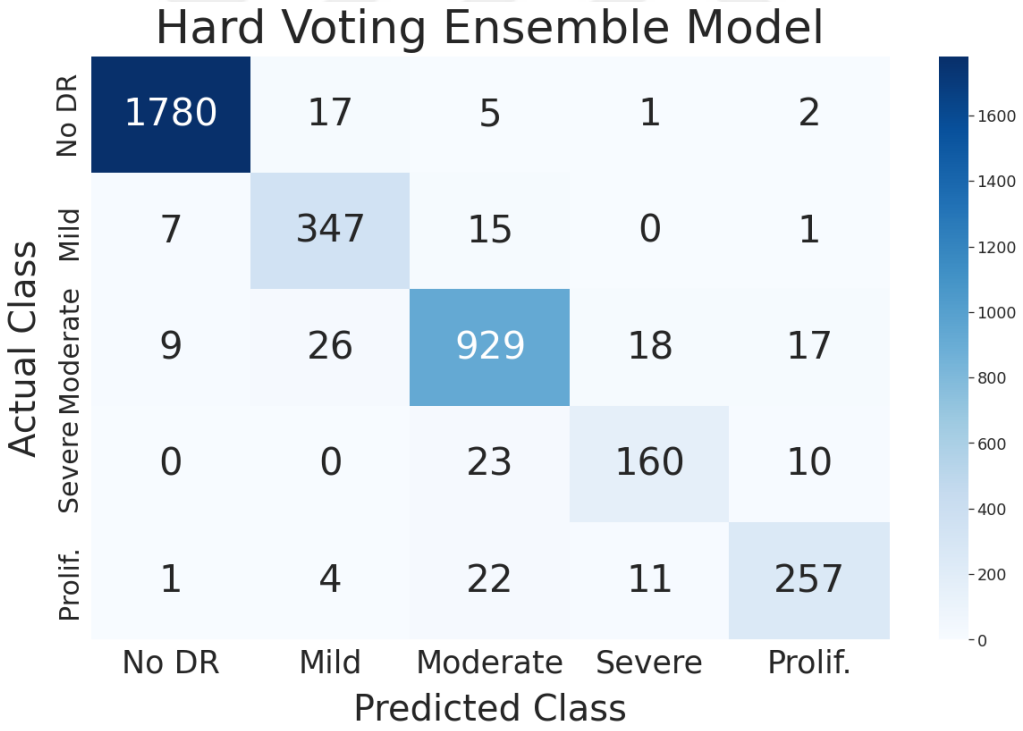


Figure 5.8 Confusion matrix of the hard voting ensemble model

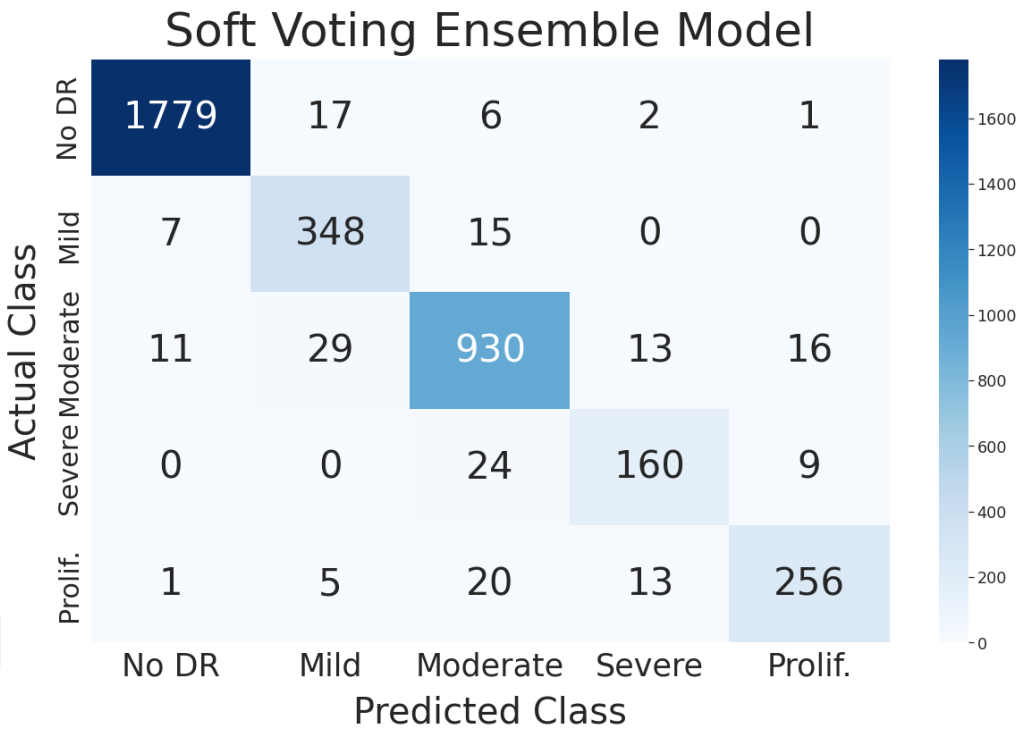


Figure 5.9 Confusion matrix of the soft voting ensemble model

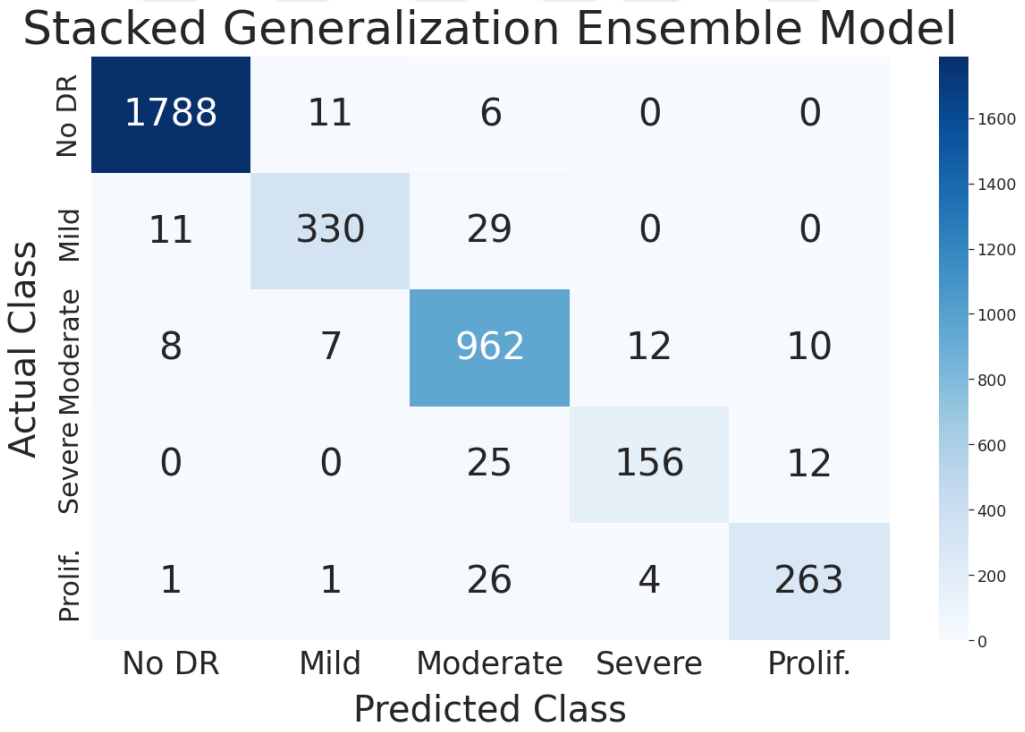


Figure 5.10 Confusion matrix of the stacked generalization ensemble model

The Accuracy, Precision, Recall, and F1 Scores derived from the confusion matrices are summarized in Table 5.3.

Table 5.3 The averaged performance scores of individual models and ensemble approaches

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
VGG16	93.78	89.29	89.66	89.47
VGG19	93.28	89.57	89.13	88.49
EfficientNetB5	88.12	78.64	85.87	81.97
EfficientNetB7	94.07	89.53	90.32	89.92
EfficientNetV2S	93.71	88.68	88.91	89.29
Hard Voting	94.84	90.87	**91.08**	90.97
Soft Voting	94.84	91.10	**91.08**	91.09
Stacked Generalization	**95.55**	**93.64**	90.90	**92.25**

According to Table 5.3, VGG16 and VGG19 models resulted in similar classification accuracy. F1 score, which uses precision and recall together, is much more informative for the comparison of methods resulted in very similar accuracies such as in VGG16 and VGG19 models. In this regard, VGG16, which is smaller and faster network than VGG19, is much more effective to use in the classification of DR according to almost 1% more F1 Score.

There are different sub-versions designed in terms of width, depth, resolution and compound aspects for the EfficientNet baseline architecture. In this study, two extended versions of original EfficientNet baseline, namely B5 and B7, and EfficientNet V2s, which is a custom design over the baseline architecture for shorten the training time, have been tested. According to the averaged results of 5-fold validation schema, EfficientNet B5 model has the lowest performances in terms of all metrics. Accuracy, precision, recall scores were measured as 88.12%, 78.64%, and 85.87%, respectively. On the other hand, EfficientNet B7, which is deeper and wider network than other EfficientNet versions, resulted in the highest scores compare to all individual networks. Accuracy, precision, and recall were measured as 94.07%, 89.53%, and 90.32%, respectively. F1 score, which is calculated over precision and recall, is also the highest score as 89.92% in the evaluation of individual model performances.

In this study, three ensemble approach was also implemented over the individual models. In the hard voting approach, only the class assignment results of each model were combined and the final decision made upon majority idea of model-based assignments. In the soft voting approach, the class assignment probabilities were combined by summing up, then the final classification made upon the maximum probability of class assignments. Hard and soft voting approaches are also known as

decision level fusion. In another ensemble approach namely stacked generalization, 1D CNN architecture based classification process was perform over the output of soft voting approach. The main idea of ensemble approaches is to increase the classification performance using the contribution of each base models.

According to the obtained results listed in Table 5.3, all three ensemble approaches were resulted in higher performance scores for all metrics compare to individual network performances. Hard and soft voting achieved 94.84% same classification accuracy. But soft voting is more effective in the determination of class labels because of obtained precision score as 91.10%. The maximum classification accuracy as 95.55% was achieved by the stacked generalization idea. Additionally, due to the highest precision score, the maximum f1 score was also obtained by the stacked generalization ensemble idea of models (92.25%).

Since the ensemble idea of models give better results, class-wise metrics of three utilized ensemble approaches are given in Table 5.4 for more detailed analysis. Three ensemble approaches resulted in almost similar performance scores for No DR and Moderate classes as around 98% and 93%, respectively. Stacked generalization idea is slightly better than hard and soft voting ensemble approaches in the classification of mild and proliferative classes. The most difference has been measured in the classification of severe class between stacked generalization idea and hard/soft voting. In terms of precision score, which indicates the correct assignments over actual labels, the performance of stacked generalization was measured almost 5% more than hard/soft voting approaches. However, recall scores of hard and soft voting was obtained almost 2% more compare to stacked generalization. As a result, stacked generalization has 2% and 1.5% more F1 scores compare to and hard and soft voting approaches.

Table 5.4 Class-wise performance scores of ensemble approaches

Ensemble Method	Accuracy	Label	Precision	Recall	F1-Score
Hard Voting	94.84	No DR	99.05	98.61	98.83
		Mild	88.07	93.78	90.84
		Moderate	93.46	92.99	93.22
		Severe	84.21	82.90	83.55
		Proliferative	89.55	87.12	88.31
		Average	90.87	91.08	90.97
Soft Voting	94.84	No DR	98.94	98.56	98.75
		Mild	87.22	94.05	90.50
		Moderate	93.47	93.09	93.28
		Severe	85.10	82.90	83.99
		Proliferative	90.78	86.78	88.73
		Average	91.10	91.08	91.09
Stacked Generalization	95.55	No DR	98.89	99.06	98.98
		Mild	94.56	89.19	91.79
		Moderate	91.79	96.29	93.99
		Severe	90.70	80.83	85.48
		Proliferative	92.28	89.15	90.69
		Average	93.64	90.90	92.25

In this study, the performance evaluation of five deep learning models and three ensemble approaches over the classification of diabetic retinopathy has been presented. The performances were measured in terms of accuracy, precision, recall, and F1 score metrics. In the experiments, one of the well-known dataset namely APTOS 2019 was used. In determining the models to be used in the experiments, VGG16 and VGG19 were selected for the evaluation of the performances for the classical convolutional neural network architectures. Two official enhancement (B3 and B7) and one custom design (v2s) models based on EfficientNet architecture were chosen to evaluate more complex architectures over APTOS 2019 dataset. Addition to individual model performance evaluations, three ensemble idea as hard voting, soft voting and stacked generalization were tested.

VGG16 and VGG19 models have similar and deep architectures but VGG19 has more convolutional layers. While VGG16 has 138.4M parameters, VGG19 consists of 143.7M parameters. According to the experimental results, both models have similar performance over APTOS 2019 dataset, which indicates that the dataset size should

be more in order to train the deeper models more effectively such as VGG19.

Other three models were selected from the variations of EfficientNet architecture. B5 and B7 versions of EfficientNet has 30.6M and 66.7M parameters, respectively. EfficientNet V2S, which is a custom design based on EfficientNet backbone in order to speed up the training phase, consists of 21.6M parameters [91]. Although EfficientNet models have less parameters than VGG16 and VGG19, the most accurate classification was performed by EfficientNet B7. On the other hand, the lowest score was obtained for EfficientNet B5 which has at least 10% difference in terms of precision score when compared to other models. In this respect, EfficientNet B5 has the maximum class assignment errors over the actual classes.

Experimental results indicates that using ensemble methods increase the classification performances. The weakest base learner has an individual classification accuracy of 88% (EfficientNet B5), but the performance of ensemble approaches are higher than any individual performances. In this regard, multi model implementation in the decision step tolerates the incorrect classification outputs of base models.

Since soft voting uses the class probabilities of base learners, it is especially useful in case of the output probabilities of a model are close for example 49% for class 1 and 51% for class 2. Another model which can sharply separates these 2 classes may tolerate the instability of the first one. Although, obtained results of this study state that soft voting and hard voting ensembles achieved similar performances. This indicates that the class distinctions of base learners are stable and consistent. Therefore, using the class assignments or probabilities did not differ the ensemble results.

Stacked generalization is expected to give better results than soft voting, since it uses the class probabilities to train another model. This model may be able to determine tendencies of each base learner, looking at their outputs and the actual class label. By considering these tendencies, it can achieve more accurate results. According to the experimental results of this study, stacked generalization model has the highest accuracy, precision and f1-score values. On the other hand, diagnosing the patients as healthy is a significant mistake. From this perspective, recall is a crucial metric because it is the ability of a model to diagnose patients. Both hard voting and soft voting models have 91% recall values. Also stacked generalization has a recall of 90.9% which is very close to other ensembles.

In Table 5.5, the results of the proposed approach were compared with state-of-the-art studies that have dealt with the same data set. According to the literature comparison table, the implementation of three ensemble idea of the deep networks, which is soft

voting, hard voting and stacked generalization approach, are resulted with an overall contribution by improving the results of the existing studies. Proposed ensemble idea achieved better results in almost every metric. In terms of accuracy and recall, Rahman and Dola obtained better results, however proposed stacked generalization framework performs better predictions in terms of precision. Also, Sikder et al. obtained higher precision (94.34%) than the proposed methods. On the other hand, they reported lower accuracy and recall results. Since this is a health problem, recall is a crucial metric which means being able to determine if DR exists.

Table 5.5 Performance comparison of proposed framework and present studies

Dataset	Paper	Methodology	Accuracy %	Precision %	Recall %
APTOS 2019 (3662 images from 5 categories)	Rahman and Dola [23]	DenseNet169 Transfer Learning	96.54	90.79	96.23
	Sugeno et al. [24]	EfficientNet-B3	84.00	-	-
	Sheikh and Qidwai [22]	DenseNet121	90.5	87.00	90.00
	Bodapati et al. [27]	Multi-Model Fusion on Pre-trained VGG16 and Inception Models	82.54	82.00	83.00
	Gangwar and Ravi [25]	Pre-trained Inception-ResNet-v2 Model that is Customized by Adding Convolutional Layers on top of it	82.18	-	-
	Islam et al. [26]	Supervised Contrastive Learning Using Margin Hyperparameter Tuning for Optimization	84.36	70.51	73.84
	Bodapati et al. [34]	Analyzing Feature Maps of Pre-trained VGG16 Model	84.31	-	-
	Shaik and Cherukuri [28]	Hinge Attention Network	85.54	85.91	85.54
	Shaik and Cherukuri [29]	Training a Neural SVM Using the Lesion-Specific Features that are Extracted by Using Xception Architecture	84.31	75.86	66.16
	Sikder et al. [33]	Training a tuned XGBoost model using GLCM and histogram features	94.20	94.34	92.68
	Sikder et al. [32]	Using Histogram Features of Retinal Images on an Ensemble of Decision Trees that is built by using Bagging Method	91.07	90.40	89.54
	Proposed Framework	Hard Voting Ensemble	94.84	92.01	90.87
	Proposed Framework	Soft Voting Ensemble	94.84	91.07	91.10
	Proposed Framework	Stacked Generalization Ensemble	95.55	90.91	93.64

Hereby, in this research, machine learning and deep learning approaches were applied for early diagnosis of diabetes and determining the degree of diabetic retinopathy disease. Deep learning models trained for the diabetic retinopathy problem were

supported by ensemble learning algorithms, resulting in an increase in the success rate. Objectivity of the study is applying 1 dimensional CNN over early stage diabetes risk prediction dataset and applying soft voting and stacked generalization ensemble ideas over APTOS 2019 dataset. Obtaining satisfactory accuracies are aimed by applying these previously unapplied approaches.

The best result for early stage diagnosis of diabetes was obtained with 1 Dimensional Convolutional neural network model and the accuracy rate is 99.04

For the classification of the degree of diabetic retinopathy disease, 5 different deep learning models were trained. Since retinal images were collected under different conditions, image pre-processing methods such as removing uninformative background, blurring, resizing and data augmentation were applied. Then, hyperparameter tuning was performed to train the base models more effectively. Different combinations of number of neurons, learning rate, dropout rate, activation functions and optimization algorithm were investigated to find the best parameters. Additionally, early stopping and learning rate scheduling were applied. Base models of the proposed ensemble approaches were determined as VGG16, VGG19, EfficientNetB5, EfficientNetB7 and EfficienNetV2S. Base model outputs were combined with 3 different ensemble methods: hard voting, soft voting and stacked generalization. Using Ensemble learning methods, success was achieved above the base learner performances, and the highest accuracy value was 95.55%, and this value was obtained with the stacked generalization model. Also, hard voting and soft voting models have 94.84% of accuracy. Although ensemble methods provide better results, they raise computational cost and extend training and testing time.

Compared to the previous studies, the proposed approach achieved higher performance. Since the results are effective, creating an application to predict severity of the diabetic retinopathy is aimed to be developed in the future work. Additionally, different networks such as NasNet, DenseNet, ResNet etc. will be utilized in the stacked generalization ensemble approach to increase the performance in the further studies. The increase in the computational cost and time may be decreased using 3 base models instead of 5. Obtained accuracy can be increased by combining the existing data sets published on retinopathy and training a model on these data. In addition, accurate results can be obtained by applying decision level fusion approaches applied in this study on other existing data sets. For stacked generalization method, classical machine learning methods such as support vector machine may be used for the decision fusion, instead of 1 dimensional convolutional neural network. Also, generative models, data upsampling and downsampling methods may be applied to solve class imbalance problem in the diabetic retinopathy dataset.

REFERENCES

[1] *Diabetes*. [Online]. Available: https://www.who.int/health-topics/diabetes#tab=tab_1 (visited on 2022).

[2] M. I. Harris, R. Klein, T. A. Welborn, M. W. Knuiman, "Onset of niddm occurs at least 4-7 yr before clinical diagnosis," *Diabetes Care*, vol. 15(7), pp. 815–819, 1992.

[3] A. Ramachandran, S. Chamukuttan, "Early diagnosis and prevention of diabetes in developing countries," vol. 9(3), pp. 193–201, 2008.

[4] *Diabetes: A national plan for action*. [Online]. Available: https://aspe.hhs.gov/sites/default/files/migrated_legacy_files//40656/NDAP04.pdf (visited on 2022).

[5] *Diabetic retinopathy: Causes, symptoms, treatment*, 2022. [Online]. Available: https://www.aao.org/eye-health/diseases/what-is-diabetic-retinopathy.

[6] *Diabetic retinopathy*, 2022. [Online]. Available: https://www.nei.nih.gov/learn-about-eye-health/eye-conditions-and-diseases/diabetic-retinopathy.

[7] *Diabetic retinopathy*, 2022. [Online]. Available: https://www.nhs.uk/conditions/diabetic-retinopathy/.

[8] *What is machine learning?* [Online]. Available: https://www.ibm.com/topics/machine-learning (visited on 2022).

[9] *Deep learning*, 2022. [Online]. Available: https://www.ibm.com/cloud/learn/deep-learning.

[10] *Early stage diabetes risk prediction dataset*. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Early+stage+diabetes%20+risk+prediction+dataset. (visited on 2022).

[11] O. Oladimeji, A. Oladimeji, O. Oladimeji, "Classification models for likelihood prediction of diabetes at early stage using feature selection," *Applied Computing and Informatics*, vol. 17, 2021.

[12] T. M. Le, T. M. Vo, T. N. Pham, S. V. T. Dao, "A novel wrapper–based feature selection for early diabetes prediction enhanced with a metaheuristic," *IEEE Access*, vol. 9, pp. 7869–7884, 2021.

[13] A. Sadhu, A. Jadli, "Early-stage diabetes risk prediction: A comparative analysis of classification algorithms," *International Advanced Research Journal in Science, Engineering and Technology*, vol. 8, pp. 193–201, 2021.

[14] A. K. Oleiwi, L. Shi, Y. Tao, L. Wei, "A comparative analysis and risk prediction of diabetes at early stage using machine learning approach," *International Journal of Future Generation Communication and Networking*, vol. 13, pp. 4151–4163, 2020.

[15] G. Bilgin, "Makine öğrenmesi algoritmaları kullanarak erken dönemde diyabet hastalığı riskinin araştırılmas," *Zeki Sistemler Teori ve Uygulamaları Dergisi*, vol. 4, pp. 55–64, 2021.

[16] İ. Özer, "Uzun kısa dönem bellek ağlarını kullanarak erken aşama diyabet tahmini," *Mühendislik Bilimleri ve Araştırmaları Dergisi*, vol. 2, pp. 50–57, 2020.

[17] *Aptos 2019 blindness detection*. [Online]. Available: `https://www.kaggle.com/competitions/aptos2019-blindness-detection/data` (visited on 2022).

[18] *Diabetic retinopathy detection (eyepacs)*. [Online]. Available: `https://www.kaggle.com/competitions/diabetic-retinopathy-detection/data` (visited on 2022).

[19] *Indian diabetic retinopathy image dataset (idrid)*. [Online]. Available: `https://ieee-dataport.org/open-access/indian-diabetic-retinopathy-image-dataset-idrid` (visited on 2022).

[20] *Messidor dataset*. [Online]. Available: `https://www.adcis.net/en/third-party/messidor/` (visited on 2022).

[21] *Messidor-2 dataset*. [Online]. Available: `https://www.adcis.net/en/third-party/messidor2/` (visited on 2022).

[22] S. Sheikh, U. Qidwai, "Smartphone-based diabetic retinopathy severity classification using convolution neural networks," *Intelligent Systems and Applications*, pp. 469–481, 2020.

[23] M. T. Rahman, A. Dola, "Automated grading of diabetic retinopathy using densenet-169 architecture," pp. 1–4, 2021.

[24] A. Sugeno, Y. Ishikawa, T. Ohshima, R. Muramatsu, "Simple methods for the lesion detection and severity grading of diabetic retinopathy by image processing and transfer learning," *Computers in Biology and Medicine*, vol. 137, p. 104 795, 2021.

[25] A. K. Gangwar, V. Ravi, "Diabetic retinopathy detection using transfer learning and deep learning," *Advances in Intelligent Systems and Computing*, vol. 1176, pp. 679–689, 2020.

[26] M. R. Islam *et al.*, "Applying supervised contrastive learning for the detection of diabetic retinopathy and its severity levels from fundus images," *Computers in Biology and Medicine*, vol. 146, p. 105 602, 2022.

[27] J. D. Bodapati, N. S. Shaik, V. Naralasetti, "Deep convolution feature aggregation: An application to diabetic retinopathy severity level prediction," *Signal, Image and Video Processing*, vol. 15, pp. 923–930, 2021.

[28] N. S. Shaik, T. K. Cherukuri, "Hinge attention network: A joint model for diabetic retinopathy severity grading," *Applied Intelligence*, vol. 52, 2022.

[29] N. S. Shaik, T. K. Cherukuri, "Lesion-aware attention with neural support vector machine for retinopathy diagnosis," *Machine Vision and Applications*, vol. 32, p. 126, 2021.

[30] S. Cahoon, M. Shaban, A. Switala, A. Mahmoud, A. El-Baz, "Diabetic retinopathy screening using a two-stage deep convolutional neural network trained on an extremely un-balanced dataset," *IEEE SoutheastCon 2022*, pp. 250–254, 2022.

[31] S. S. Karki, P. Kulkarni, "Diabetic retinopathy classification using a combination of efficientnets," *2021 International Conference on Emerging Smart Computing and Informatics (ESCI)*, vol. 392, pp. 68–72, 2021.

[32] N. Sikder, M. S. Chowdhury, A. S. M. Arif, A. Nahid, "Early blindness detection based on retinal images using ensemble learning," *2019 22nd International Conference on Computer and Information Technology (ICCIT)*, pp. 1–6, 2019.

[33] N. Sikder, M. Masud, A. K. Bairagi, A. S. M. Arif, A. A. Nahid, H. A. Alhumyani, "Severity classification of diabetic retinopathy using an ensemble learning algorithm through analyzing retinal images," *Phys. Rev E.*, vol. 13, p. 670, 2021.

[34] J. D. Bodapati, N. S. Shaik, V. Naralasetti, "Composite deep neural network with gated-attention mechanism for diabetic retinopathy severity classification," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 9825–9839, 2021.

[35] C. Matsoukas, J. F. Haslum, M. Söderberg, K. Smith, "Is it time to replace cnns with transformers for medical images?" *arXiv*, 2021, arXiv: 2108.09038.

[36] G. Çinarer, K. Kiliç, "Diabetic retinopathy detection with deep transfer learning methods," vol. 392, 2021.

[37] S. Sanjana, S. S. Nazmus, M. Farzana, "Automated diabetic retinopathy detection using transfer learning models," 2021.

[38] C. Zhang, T. Lei, P. Chen, "Diabetic retinopathy grading by a source-free transfer learning approach," *Biomedical Signal Processing and Control*, 2022.

[39] R. Adriman, K. Muchtar, N. Maulina, "Performance evaluation of binary classification of diabetic retinopathy through deep learning techniques using texture feature," *Procedia Computer Science*, vol. 179, pp. 88–94, 2020.

[40] Y. Li, Z. Song, S. Kang, S. Jung, W. Kang, "Semi-supervised auto-encoder graph network for diabetic retinopathy grading," *IEEE Access*, vol. 9, pp. 140 759–140 767, 2021.

[41] M. M. F. Islam, R. Ferdousi, S. Rahman, H. Y. Bushra, "Likelihood prediction of diabetes at early stage using data mining techniques," *Computer Vision and Machine Intelligence in Medical Image Analysis*, pp. 113–125, 2022.

[42] *What is polydipsia?* [Online]. Available: `https://www.verywellhealth.com/polydipsia-4783881` (visited on 2022).

[43] *Could rapid weight loss mean diabetes?* [Online]. Available: `https://www.verywellhealth.com/rapid-weight-loss-5101064` (visited on 2022).

[44] *Causes of polyphagia*. [Online]. Available: `https://www.verywellhealth.com/polyphagia-5114624` (visited on 2022).

[45] *Thrush*. [Online]. Available: https://www.diabetes.co.uk/diabetes-complications/diabetes-and-yeast-infections.html (visited on 2022).

[46] *What is blurred vision?* [Online]. Available: https://www.verywellhealth.com/blurred-vision-5114184 (visited on 2022).

[47] *What is paresis?* [Online]. Available: https://www.verywellhealth.com/paresis-5184820 (visited on 2022).

[48] *What causes muscle rigidity?* [Online]. Available: https://www.healthline.com/health/muscle-rigidity (visited on 2022).

[49] *Does diabetes cause hair loss?* [Online]. Available: https://www.healthline.com/health/does-diabetes-cause-hair-loss (visited on 2022).

[50] *How to split data into three sets (train, validation, and test) and why?* [Online]. Available: https://towardsdatascience.com/how-to-split-data-into-three-sets-train-validation-and-test-and-why-e50d22d3e54c (visited on 2022).

[51] V. Garcia, J. S. Sanchez, A. I. Marques, "Synergetic application of multi-criteria decision-making models to credit granting decision problems," vol. 59, 2019.

[52] S. A. Yadav, S. Shukla, *Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification*, 2016.

[53] T. R. Prajwala, "A comparative study on decision tree and random forest using r tool," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4(1), pp. 196–199, 2015.

[54] J. Ali, R. Khan, N. Ahmad, "Random forests and decision trees," *International Journal of Computer Science Issues(IJCSI)*, vol. 9, 2012.

[55] *Random forests understanding*. [Online]. Available: https://ai-pool.com/a/s/random-forests-understanding (visited on 2022).

[56] G. Wahba, *Splines for observational data*. SIAM, 1990.

[57] E. Garcia-Gonzalo, P. J. G. Fernandez-Muniz Z.and Nieto, A. B. Sanchez, M. Menendez, "Hard-rock stability analysis for span design in entry-type excavations with learning classifiers," *Journal of Artificial Intelligence Research*, vol. 9, p. 531, 2016.

[58] Y. Han *et al.*, "Coupling a bat algorithm with xgboost to estimate reference evapotranspiration in the arid and semiarid regions of china," *Advances in Meteorology*, vol. 2019(9), pp. 1–16, 2019.

[59] *K-nearest neighbors (knn) in python*, 2022. [Online]. Available: https://www.jcchouinard.com/k-nearest-neighbors/.

[60] K. O'Shea, R. Nash, "An introduction to convolutional neural networks," *ArXiv e-prints*, 2015.

[61] A. I. A. Osman, A. N. Ahmed, M. F. Chow, Y. F. Huang, A. El-Shafie, "Extreme gradient boosting (xgboost) model to predict the groundwater levels in selangor malaysia," *Ain Shams Engineering Journal*, vol. 12, pp. 1545–1556, 2021.

[62] *Sigmoid function*. [Online]. Available: https://en.wikipedia.org/wiki/Sigmoid_function (visited on 2022).

[63] *Deeplearning series: Convolutional neural networks*. [Online]. Available: https://medium.com/machine-learning-bites/deeplearning-series-convolutional-neural-networks-a9c2f2ee1524 (visited on 2022).

[64] *File:maxpoolsample2.png*. [Online]. Available: https://computersciencewiki.org/index.php/File:MaxpoolSample2.png (visited on 2022).

[65] *Convolutional neural networks (cnn): Step 3 - flattening*. [Online]. Available: https://www.superdatascience.com/blogs/convolutional-neural-networks-cnn-step-3-flattening (visited on 2022).

[66] P. Kulkarni, S. Londhe, M. Deo, "Artificial neural networks for construction management: A review," *Soft Computing in Civil Engineering*, vol. 1(2), pp. 70–88, 2017.

[67] A. Meliboev, J. Alikhanov, W. Kim, "1d cnn based network intrusion detection with normalization on imbalanced data," in *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, 2020, pp. 218–224.

[68] *What is machine learning*. [Online]. Available: https://www.ibm.com/cloud/learn/machine-learning (visited on 2022).

[69] W. J. Zhang, G. Yang, Y. Lin, C. Ji, M. M. Gupta, *On definition of deep learning*, 2018.

[70] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[71] V. Tan, Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," *International Conference on Machine Learning*, 2019.

[72] V. Shah, R. Keniya, A. Shridharani, M. Punjabi, J. Shah, N. Mehendale, "Diagnosis of covid-19 using ct scan images and deep learning techniques," *Emergency Radiology*, vol. 28, 2021.

[73] *Vgg16 architecture*. [Online]. Available: https://iq.opengenus.org/vgg16/ (visited on 2022).

[74] A. Khattar, S. M. K. Quadri, "Generalization of convolutional network to domain adaptation network for classification of disaster images on twitter," *Multimedia Tools and Applications*, vol. 81, pp. 30 437–30 464, 4 2022.

[75] P. Zuvela, M. Lovric, A. Yousefian-Jazi, J. J. Liu, "Ensemble learning approaches to data imbalance and competing objectives in design of an industrial machine vision system," vol. 59, pp. 4636–4645, 2020.

[76] D. H. Wolpert, "Stacked generalization," *Neural networks: the official journal of the International Neural Network Society*, vol. 5(2), pp. 241–259, 1992.

[77] S. Wang, J. Yang, K. Chou, "Using stacked generalization to predict membrane protein types based on pseudo-amino acid composition," *Journal of Theoretical Biology*, vol. 242, pp. 941–946, 2006.

[78] K. M. Ting, I. Witten, "Issues in stacked generalization," *Journal of Artificial Intelligence Research*, vol. 10(1), pp. 271–289, 1999.

[79] M. Hassan, S. Ullah, J. Khan, K. Khurshid, "Comparative analysis of svm, ann and cnn for classifying vegetation specie using hyperspectral thermal infrared data," Jun. 2019.

[80] Y. Muchao, X. Wang, W. Zhou, L. Xiang, W. Juan, Z. Zhongyuan, "Research on transformer condition recognition based on acoustic signal and one-dimensional convolutional neural networks," *Journal of Physics Conference Series*, vol. 2005(1), 2021.

[81] Ö. N. Ergün, H. O. İlhan, "Early stage diabetes prediction using machine learning methods," *Avrupa Bilim ve Teknoloji Dergisi (EJOSAT)*, vol. 29, 2021.

[82] *Python*. [Online]. Available: https://www.python.org/ (visited on 2022).

[83] *Visualize programming language popularity using tiobeindexpy*. [Online]. Available: https://towardsdatascience.com/visualize-programming-language-popularity-using-tiobeindexpy-f82c5a96400d (visited on 2022).

[84] *Numpy*. [Online]. Available: https://numpy.org/ (visited on 2022).

[85] *Pandas*. [Online]. Available: https://pandas.pydata.org/ (visited on 2022).

[86] *Scikit-learn*. [Online]. Available: https://scikit-learn.org/stable/ (visited on 2022).

[87] *Opencv*. [Online]. Available: https://opencv.org/ (visited on 2022).

[88] *Matplotlib: Visualization with python*. [Online]. Available: https://matplotlib.org/ (visited on 2022).

[89] *Tensorflow*. [Online]. Available: https://www.tensorflow.org/ (visited on 2022).

[90] *Brain team*. [Online]. Available: https://research.google/teams/brain/ (visited on 2022).

[91] *Keras*. [Online]. Available: https://keras.io/ (visited on 2022).

[92] *Deep learning framework power scores 2018*. [Online]. Available: https://towardsdatascience.com/deep-learning-framework-power-scores-2018-23607ddf297a (visited on 2022).

PUBLICATIONS FROM THE THESIS

Papers

1. Ö. N. Ergün, H. O. İlhan, "Early Stage Diabetes Prediction Using Machine Learning Methods", *Avrupa Bilim ve Teknoloji Dergisi (EJOSAT)*, vol. 29, pp. 52-57, 2021.