INTEGRATION OF MACHINE LEARNING AND DEEP LEARNING METHODS FOR THE ENHANCEMENT OF RHEUMATOID ARTHRITIS DIAGNOSIS

KEMAL ÜRETEN

JANUARY 2024

ÇANKAYA UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DEPARTMENT OF COMPUTER ENGINEERING

Ph.D THESIS IN

COMPUTER SCIENCE AND ENGINEERING

INTEGRATION OF MACHINE LEARNING AND DEEP LEARNING METHODS FOR THE ENHANCEMENT OF RHEUMATOID ARTHRITIS DIAGNOSIS

KEMAL ÜRETEN

JANUARY 2024

ABSTRACT

INTEGRATION OF MACHINE LEARNING AND DEEP LEARNING METHODS FOR THE ENHANCEMENT OF RHEUMATOID ARTHRITIS DIAGNOSIS

ÜRETEN, Kemal

PhD in Computer Science and Engineering

Supervisor: Prof. Dr. H. Hakan MARAŞ

January 2024, 74 pages

In this study to classify normal and Rheumatoid arthritis hand radiographs, convolutional neural networks were used as object detectors and feature extractors. After pre-processing with YOLOv4, classification was performed with the pre-trained VGG-16 model. Accuracy, sensitivity, specificity, precision, F1 score, area under the curve (AUC), and Cohen's kappa performance results of 90.5%, 96.0%, 85.7%, 85.7%, 90.5%, 0.94, and 0.81 were obtained, respectively. Feature extraction was performed with VGG-16 model, and with this extracted features dataset, a 0.9% improvement in accuracy was achieved with the stacking method. Two novel contributions were made to the literature as a result of work performed in this thesis. First, segmental search property was added to the majority voting classifier, providing a 1% improvement in accuracy compared to VGG-16. Second, ANOVA and variance threshold methods were applied to the dataset for feature selection, and the machine learning classifier was inserted into the ANOVA algorithm to find the optimum number of features. The optimal feature set was searched iteratively. With this proposed ANOVA and variance threshold feature selection methods, a 2-5% improvement in performance metrics was achieved with Random Forest, Logistic Regression and Support Vector Machines algorithms. Training time was shortened with both proposed methods.

Keywords: Machine learning, deep learning, majority voting classifier, feature extraction, feature selection, ANOVA, variance threshold

ÖZET

ROMATOİD ARTRİT TANISININ İYİLEŞTİRİLMESİ İÇİN MAKİNE ÖĞRENMESİ VE DERİN ÖĞRENME YÖNTEMLERİNİN ENTEGRASYONU

ÜRETEN, Kemal

Bilgisayar Bilimleri ve Mühendisliği Doktora

Danışman: Prof. Dr. H. Hakan MARAŞ

Ocak 2024, 74 sayfa

Normal el radyografileri ve Romatoid artritli el radyografilerini sınıflandırmak için yapılan bu çalışmada CNN'ler nesne dedektörü ve özellik çıkarıcı olarak kullanıldı. YOLOv4 ile ön işlem sonrası önceden eğitilmiş VGG-16 modeli ile sınıflandırma yapıldı. Doğruluk, duyarlılık, özgüllük, kesinlik, F1 score, area under the curve (AUC) ve Cohen's kappa performans metriklerinde sırasıyla %90.5, %96.0, %85.7, %85.7, %90.5, 0.94 ve 0.81 sonuçları elde edildi. VGG-16 modeli ile özellik çıkarımı yapıldı ve bu veri seti ile topluluk öğrenme algoritmalarından stacking yöntemi ile doğruluk sonucunda % 0.9'luk bir iyileşme oldu. Performansı iyileştirmek için yapılan çalışmalar sonucunda literatüre iki yeni katkı sağlandı. Önce majority voting sınıflandırıcısına segmental arama özelliği eklendi ve önerilen bu majority voting modeliyle VGG-16'ya göre % 1 doğruluk artışı elde edildi. İkinci olarak verisetine filtre özellik seçim yöntemlerinden ANOVA ve varyans eşik yöntemleri birlikte uygulandı ve optimum özellik sayısını bulmak için makine öğrenmesi sınıflandırıcısı ANOVA algoritması içine yerleştirildi. Optimum özellik seti yinelemeli olarak arandı. Önerilen bu ANOVA ve varyans eşik yöntemiyle Rastgele Orman, Lojistik Regresyon ve Destek Vektör Makineleri algoritmaları ile performans metriklerinde %2-5 oranında iyileşme sağlandı. Önerilen her iki yöntemle de eğitim süresi kısaldı.

Anahtar kelimeler: Makine öğrenmesi, derin öğrenme, çoğunluk oylama sınıflandırıcısı, özellik çıkarma, özellik seçimi, ANOVA, varyans eşiği

ACKNOWLEDGEMENT

On the 101st anniversary of our republic, I would like to express my deepest respect and gratitude to Gazi Mustafa Kemal Atatürk.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

RA	:Rheumatoid arthritis
MRI	:Magnetic Resonance Imaging
CT	:Computed Tomography
CNN	:Convolutional Neural Networks
NLP	:Natural Language Processing
ReLU	:Rectified Linear Unit
ACO	:Ant Colony Optimization algorithm
PSO	:Particle Swarm Optimization algorithm
SVM	:Support Vector Machines
k-NN	:k-Nearest Neighbor
SVC	:Support Vector Classifier
LR	:Logistic Regression
RF	:Random Forest
ROI	:Regions of Interest
AUC	:Area under the curve
ANOVA	:Analysis of Variance
PCA	:Principal Component Analysis
XGBoost	:eXtreme Gradient Boosting
YOLO	:You Only Look Once
SSD	:Single Shot Detector
IRF	:iterative ReliefF
DT	:Decision Tree

CHAPTER I
INTRODUCTION

1.1 STUDY SUBJECT

Rheumatoid arthritis (RA) is a long-lasting, serious, autoinflammatory disease that especially affects the synovial joints, presenting with joint pain, swelling, tenderness, and limitation of movement. RA is 2-3 times more common in women. Environmental and genetic factors are responsible for the etiology of RA. The disease can lead to joint deformities and permanent disability in the late stages of the disease [1,2].

RA characteristically affects the joints symmetrically, usually and primarily affecting the small joints of the hands and feet. Metacarpophalangeal joints, proximal interphalangeal (PIF) joints of the hands, and wrists are primarily and frequently involved in the hands. Similarly, metatarsophalangeal joints and, (PIF) joints of the feet are frequently affected. Knee, shoulder, hip, and elbow joints may also be involved over time. Morning stiffness with pain, swelling, and tenderness in the involved joints is a significant complaint [3,4]. Early recognition and treatment of RA are important to prevent joint damage and disability. RA treatment includes anti-rheumatic drugs, physical therapy modalities, and lifestyle changes [5,6]. The patient's complaint, the style of the involved joints, blood laboratory tests, and imaging methods are used for diagnosis. Imaging techniques used in the evaluation of RA include radiographs, ultrasound imaging, and magnetic resonance imaging (MRI) [1].

X-rays are the most commonly used imaging method in the diagnosis and follow-up of RA due to their widespread availability and low cost. In the early stages of RA, periarticular soft tissue edema, juxta-articular osteopenia, and joint space narrowing are often seen on plain radiographs. As the disease progresses, marginal erosions of the bones around the joints and subluxations (partial dislocation) may appear in the late stages. With ultrasonography, synovial thickening, hyperemia, and fluid accumulation in the joint are detected in the early period [7]. With MRI, early

inflammatory changes, synovitis and bone marrow edema can be detected in the early period, cartilage and bone erosions, and tenosynovitis can be shown [8].

Imaging findings in RA may vary depending on the stage and severity of the disease. X-ray imaging has many advantages in the diagnosis and follow-up of RA;

1. Early detection of periarticular soft tissue edema, joint space narrowing, and marginal erosions by X-rays allows timely treatment to prevent further joint destruction.
2. X-rays are widely available and cost less than other advanced imaging modalities such as MRI or computed tomography (CT). This makes them more accessible for routine monitoring and follow-up of RA patients.
3. Plain radiographs provide useful information to determine treatment response, evaluate the course of the disease, and reveal the extent of joint involvement.
4. The initial X-ray taken at the onset of the disease can provide a basis for future comparisons and help determine disease progression or stability over time.
5. X-rays can be easily stored in Hospitals' Picture Archiving and Communication Systems (PACS) and thus easily shared between healthcare providers [1,4].

In the early stages of RA, joint damage and radiographic changes may not be revealed by X-rays. At this early stage, other imaging modalities such as ultrasound and MRI may be more sensitive in detecting early signs of inflammation. Therefore, a combination of clinical evaluation and laboratory tests as well as imaging techniques are often used for accurate diagnosis and monitoring of RA [1].

In RA, "baseline for future comparison" refers to the initial imaging study (e.g., X-ray) taken at diagnosis or the start of treatment. This initial image serves as a reference point against which subsequent imaging studies are compared over time.

1. Initial Imaging: When a patient is diagnosed with RA or begins treatment, X-rays of the affected joints are taken to assess the baseline condition of the joints and for future comparison.
2. Subsequent Follow-up and Comparison: New radiographs are taken while the patient is under treatment. The follow-up X-rays are then compared to the baseline X-ray. Changes in joint structures over time can be evaluated. Thus, the course of RA and the effectiveness of the treatment are monitored [6,9,10].

The quality of ultrasound images is affected by factors such as the skill and experience of the person performing the procedure, the type of ultrasound machine

used, patient characteristics (e.g., body composition), and the location of the joint being imaged. The advantages of X-rays over ultrasound are as follows;

1. The penetration of X-rays is good, which makes it possible to visualize deeper structures even in obese patients.
2. With X-rays, bone structures are better visualized, ideal for evaluating bone erosions and joint space narrowing.
3. X-rays can provide comprehensive imaging of multiple joints at the same time,
4. X-ray imaging protocols are well-structured and standardized. This standardization makes it easy to compare images taken at different times and from different patients. X-ray is considered the gold-standard imaging method in the diagnosis, follow-up, and monitoring of RA [4].

Artificial neural networks are a type of machine learning method inspired by the function and structure of the human brain. It consists of interconnected layers of "neurons" that process and transmit information. Neural networks have an input layer, a hidden layer, and an output layer [11,12]. There are many neurons in the hidden layer of Convolutional Neural Networks (CNN), which are designed with inspiration from the human visual cortex. A CNN hidden layer has multiple convolution layers, pooling layers, and finally fully connected layers. The task of the convolution layer is to find the local features of the input images at different locations and to generate feature maps. The pooling layer following the convolution layer acts as subsampling the feature maps of the previous convolution layer. In the CNN architecture, after the convolution layer, there is a non-linear activation function such as a rectified linear unit (ReLU) to generate the activation map or feature map. CNN is a deep learning network with a feed-forward multilayer hierarchical neural network structure. CNNs use a mathematical process called convolution, which allows features to be extracted from an image through filters called kernels [13,14].

CNNs are very successful at tasks such as image classification, feature extraction, object detection, and natural language processing (NLP). A large number of labeled data is needed for CNN training from scratch. If there are not enough labeled image data for training, the transfer learning method can be applied. Rather than starting a new model from scratch and training it on a large dataset, transfer learning allows to use of knowledge and representations learned from a previously trained model and applying it to a new task. Imagenet has more than 14 million

images belonging to more than 20.000 categories. A model trained on large-scale datasets such as ImageNet has learned general properties such as edges, textures, and shapes that can be applied to a variety of related tasks. By using pre-trained models as a starting point, it can save significant computational resources and training time. Nowadays, many pre-trained network models such as AlexNet [15], GoogLeNet [16], VGG [17], ResNet [18], EfficientNet [19], MobileNet [20], and DenseNet [21] can be used for transfer learning.

To perform image classification, features representing relevant data (feature extraction) must be given to the CNN. The convolution layer applies various filters to the images and performs feature extraction automatically. Figure 1 shows the CNN structure. Pre-trained networks can be used as feature extractors [22]. With this method, 512 features are obtained with the VGG-16 intermediate layer from images for image classification. Feature selection is the process of reducing the number of input variables, identifying relevant features, and removing redundant and irrelevant features while developing a model. This reduces the computational cost of the model, reduces the training time, and in some cases, improves its accuracy. Feature extraction and feature selection are current and popular research areas in the field of image classification. Feature selection methods are classified under three groups; filter, wrapped and embedded methods. Each method has advantages and disadvantages [23].

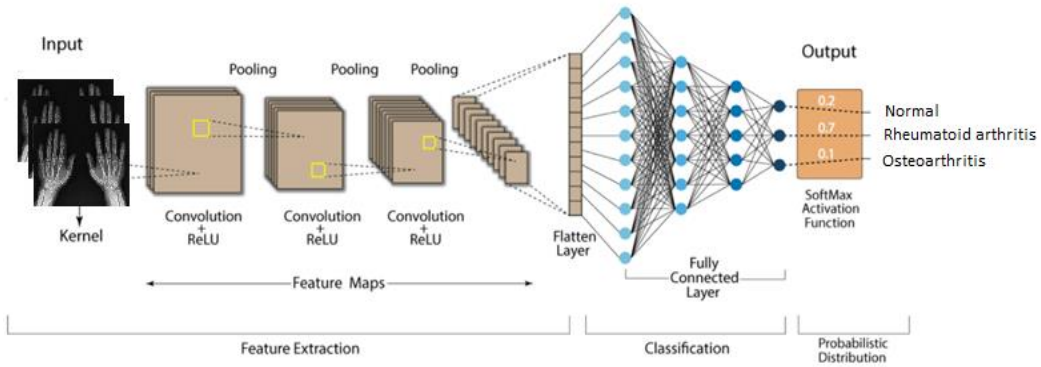


Figure 1: CNN structure illustration

1.2 OBJECTIVES

This study aimed to develop a machine learning model to detect RA from hand X-rays and to classify normal and RA hand radiographs. To improve the performance of the models, studies have been carried out with feature extraction,

feature selection methods, machine learning algorithms and ensemble learning methods. The YOLOv4 algorithm, which is a popular technique for object detection, was used to remove unnecessary areas on hand X-rays. After cropping the X-rays with the object detector obtained with the YOLOv4 algorithm, normal and RA hand X-rays were classified using the pre-trained VGG-16 model. Feature extraction was performed with the pre-trained VGG-16 network. The majority voting algorithm is a simple yet powerful technique in machine learning that has improved accuracy. It is compatible with various machine learning models and algorithms such as DT, RF, SVM, k-NN, and LR. Successful performance improvement results were achieved with majority voting in the medical area. In this study, an attempt was made to improve performance by adding segmental search property to the majority voting algorithm. ANOVA is a filter feature selection method and uses statistical methods. It is applied if the input variable is numeric and the output is categorical. With variance threshold, low-variance features that do not contribute to classification are eliminated. For this reason, ANOVA and variance threshold were used as filter feature selection methods. The performance of our proposed segmental search majority voting classifier and ANOVA+variance threshold feature selection method was evaluated in the classification of normal and RA hand X-rays. The performance of the developed models was evaluated with accuracy, sensitivity, specificity, precision, F1 score, AUC and Cohen's kappa performance metrics.

1.3 ORGANIZATION OF THE THESIS

The introduction chapter presents brief knowledge about RA and the purpose of the study. The background study chapter includes information about object detection, machine learning, ensemble learning, feature extraction, feature selection, dimensionality reduction, and a summary of feature selection studies on medical images. The materials and methods chapter provides information about the dataset, proposed models, and statistical analysis. In the results chapter, the performance metrics results, confusion matrices, and ROC curve images obtained in the study are given. In the discussion and conclusion chapter, the study results are summarized and the proposed methods are highlighted.

CHAPTER II
BACKGROUND STUDY

2.1 OBJECT DETECTION

CNNs are used in many tasks such as image classification, feature extraction, object detection, object tracking, and natural language processing. Some of the CNN structures used in object detection are Region Based CNN (R-CNN), Fast R-CNN, Faster R-CNN, Single Shot Detector (SSD) and You Only Look Once (YOLO) series algorithms [24–26].

Object detectors are divided into two classes: two-stage and single-stage object detectors. Object detector performance is measured by detection accuracy and inference time. In two-stage object detectors, the first stage creates Regions of Interest (ROI) and the second stage predicts bounding boxes to the proposed regions. The detection accuracy of two-stage object detectors is better than that of single-stage object detectors. Single-stage detectors have a faster inference time. These algorithms identify objects and their locations with the help of bounding boxes by looking at the image only once (YOLO). YOLO can identify multiple objects in images in real time. The whole image is sent to a single CNN. This CNN predicts multiple bounding boxes on the image with different probabilities. YOLO splits the input image into non-overlapping (S x S) grid cells. Each grid cell estimates the bounding box probability and conditional class probabilities of an object in the underlying grid. The confidence score indicates whether there are any objects in the bounding boxes. Intersection over union (IoU) is calculated with predicted and base true bounding boxes [24,27–29].

$$\text{intersection over union (IoU)} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{2.1}$$

$$\text{confidence score} = \Pr(\text{Object}) \text{ x } \text{IoU}_{\text{predicted}}^{\text{ground truth}} \tag{2.2}$$

YOLO is small in size and computational speed is fast. It turns the target detection problem into a regression problem. The grid where the center point of the object is located should find its class, height, and width of that object and draw a bounding box around that object. YOLO estimates more than one bounding box per grid cell, but the best bounding box is chosen with the help of an algorithm known as *non-maximum suppression* [30,31].

The network architecture typically follows the Darknet architecture, which consists of multiple convolution and pooling layers. The last layer predicts bounding boxes and class probabilities for each grid cell. YOLOv4 has demonstrated superior performance compared to previous iterations, surpassing other popular object detection algorithms in terms of both accuracy and speed [32]. YOLO achieves a high degree of localization accuracy and generalization across different object categories. The YOLOv4 algorithm demonstrates the progress achieved in the field of object detection and has received significant attention since its release [33].

2.2 MACHINE LEARNING

Machine learning is one of the subfields of artificial intelligence based on the principle that a system learns from existing data without requiring explicit programming. In machine learning, data is analyzed to make predictions [34–36]. Machine learning focuses on developing algorithms and models that can learn automatically from data, and make predictions or decisions. It has gained immense popularity and success in various domains, such as computer vision, NLP, recommendation systems, and healthcare. Machine learning algorithms learn from existing data and extract patterns, relationships, and insights to make accurate predictions or classifications on new, unseen data [35,37].

In machine learning, supervised learning, unsupervised learning, and reinforcement learning methods are used. In supervised learning, labeled data is used, it is aimed to create a model that best predicts the relationship between input and output, given a data sample and desired outputs. In unsupervised learning, unlabeled data is used to cluster the data and determine the relationships between the data. Frequently used and popular machine learning algorithms in classification and regression tasks are Logistic Regression (LR), k-NN algorithm, Decision Tree, SVM, RF, and Neural Networks algorithms [37,38].

2.2.1 Decision Trees

The decision tree (DT) algorithm is a popular and widely used supervised machine learning technique for classification and regression. DT creates a tree-like model that makes predictions or decisions by following a set of rules based on input properties. The DT model consists of roots, nodes, branches, and leaves. The DT algorithm aims to create homogeneous subsets concerning the target variable, choosing the best feature to split the data at each step using a measure of impurity or information gain. One commonly used measure is entropy, which measures impurity in a set of samples. Another metric is the Gini index, which measures the probability of misclassification of a randomly selected sample from the cluster. The purpose is to find the feature that maximizes the separation of classes or minimizes impurity in each partition. Once a feature is selected for splitting, the DT algorithm creates a node in the tree that represents that feature. Each branch emerging from the node corresponds to a possible value of the property. The algorithm then applies the same splitting process to subsets of data associated with each branch, creating child nodes (Figure 2) [39–43].

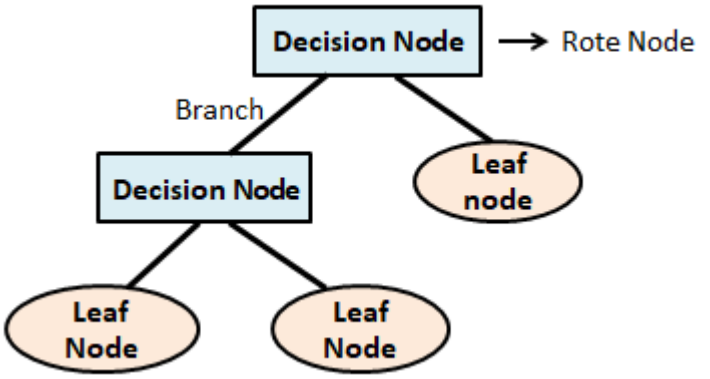


Figure 2: Decision Tree chart.

The DT can be applied to both numerical and categorical variables. However, DTs can be prone to overfitting, especially when the tree becomes too deep or the data contains noise or outliers. Techniques such as pruning, limiting the tree depth, or using ensemble methods like RFs can mitigate overfitting and improve generalization performance. There are many algorithms under the DT classifier. The most commonly used of these algorithms are ID3, C4.5, and CART algorithms. DT has been extensively studied and has inspired numerous extensions and variations in

the field of machine learning, making it a fundamental tool for data analysis and prediction [44].

2.2.2 k-Nearest Neighbor Algorithm

k-Nearest Neighbor (k-NN) algorithm is a classification algorithm. It works based on the principle that similar data points belong to the same class (Figure 3). The k-NN algorithm determines the class or value of an example by looking at its k in the training dataset. For classification, a value of k is chosen to determine the number of neighbors and a distance metric is chosen. The number of neighbors to consider to define the class of a data point is determined by k [45]. The distance between the new sample and all samples in the training dataset is calculated. The most common distance measure used is the Euclidean distance. The Manhattan distance and the Minkowski distance are other frequently used distance measurement methods. The distance metric depends on the nature of the data and the problem being solved. If the result is insufficient, the k value and distance measure can be changed to improve the model. The success of k-NN depends on three important factors; these are the k value, the distance metric used to determine the neighbors, and the sample size. Choosing large k values should be taken with care because this may cause overfitting [46–49].

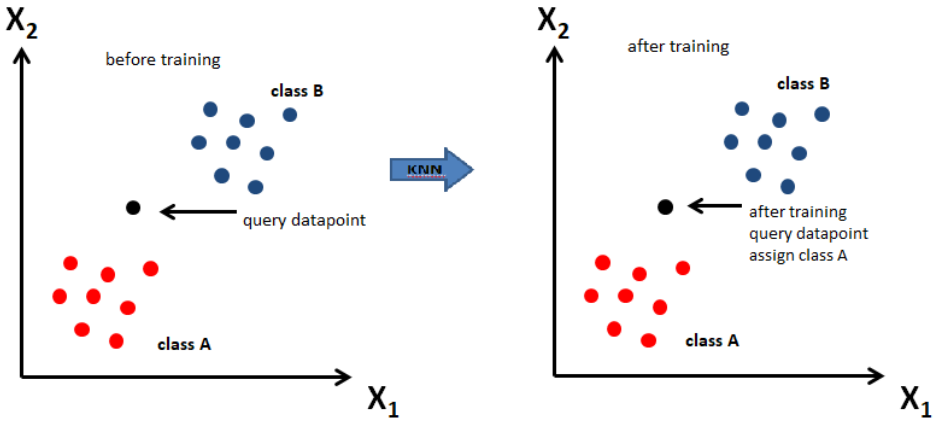


Figure 3: k-Nearest Neighbor algorithm diagram

2.2.3 Naive Bayes

Naive Bayes algorithm is a probabilistic classification algorithm based on the Bayes theorem. In the Naive Bayes algorithm, it is assumed that a feature belonging to a class in the data set is independent of other features. The probability of

observing a feature in a class is estimated by counting the occurrences of that feature within the samples of the given class and dividing it by the total number of samples in that class. This causes the algorithm to perform poorly. The Naive Bayes algorithm is fast and easy to implement compared to other algorithms and is more resistant to overfitting [50–52].

$$P(A|B) = \frac{\mathrm{P(B|A)} * P(A)}{P(B)} \tag{2.3}$$

P(A|B) means posterior probability, P(B|A) means likelihood, P(A) means prior probability, and P(B) means evidence.

2.2.4 Logistic Regression

Logistic Regression (LR) presents the relationship between multiple independent variables and a categorical dependent variable with a curve(Figure 4). Independent variables can be categorical variables or continuous variables, but the dependent variable must be categorical. LR usually uses the sigmoid function and finds the optimal parameters with this function. In LR, the relationship between features and the probability of belonging to a particular class is modeled using the sigmoid function. The logistic function maps any real-valued number to a value between 0 and 1. The performance of LR in large-scale applications is slow [53–55].

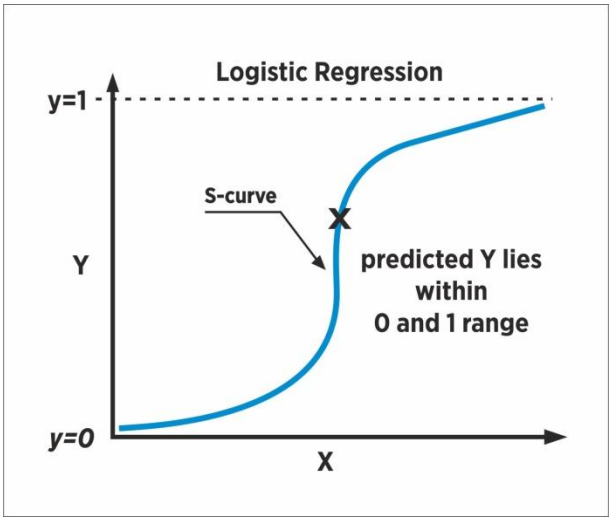


Figure 4: The Logistic Regression diagram.

2.2.5 Support Vector Machines

Support Vector Machines (SVM) are used for both classification and regression tasks. SVM is a supervised learning algorithm. It is particularly successful in solving binary classification problems, but can also be used in multi-class classification tasks. The goal of SVM is to find the optimal hyperplane that maximally separates the classes. A hyperplane is a decision boundary that separates examples into different classes based on their feature values. SVM attempts to find the hyperplane that not only separates classes but also preserves the maximum margin between the closest examples of each class. These samples closest to the decision boundary are called support vectors (Figure 5) [56–58].

SVM has many advantages such as the ability to handle high dimensional data, robustness against overfitting, and efficiency even with a small amount of training data. It is used in various fields such as image classification, text classification, bioinformatics and finance.

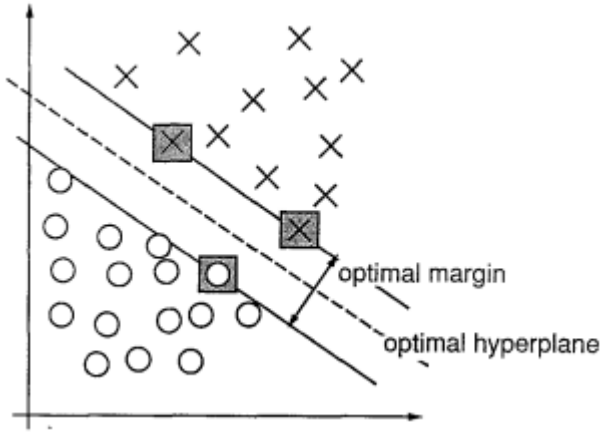


Figure 5: General illustration of SVM structure [56].

2.3 ENSEMBLE LEARNING

Ensemble learning is a machine learning technique that allows building a model with multiple learners instead of training the model with a single learner. With ensemble learning, it is aimed that the models make more accurate decisions together. Ensemble learning is a powerful technique to improving the performance and robustness of machine learning models by combining multiple individual models. The ensemble learning method is an algorithm that combines several base models to produce an optimal predictive model. By leveraging the diversity and collective intelligence of the ensemble, it can provide more accurate and reliable predictions for complex tasks. It helps to reduce overfitting. Ensemble learning can

be applied to both classification and regression tasks. Ensemble learning algorithms are classified under 3 groups; bagging, boosting, and stacking (Figure 6) [59–61].

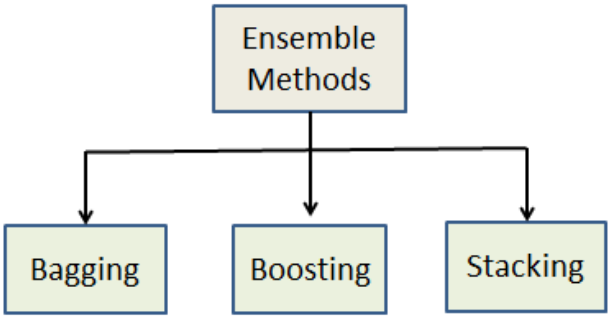


Figure 6: Ensemble learning methods.

2.3.1 Bagging

Bagging method, short for Bootstrap Aggregating, is an effective technique used to solve classification and regression problems in machine learning. Bagging is an ensemble learning algorithm that combines multiple base learners to create an accurate and robust model. The RF algorithm is an example of bagging. In the bagging technique, random samples are selected from the labeled training set, and the models are trained on each sub-dataset. Bagging aims to reduce the variance and improve the stability of the predictions by training each base learner on different subsets of the training data, generated through bootstrapping. The predictions of the base learners are combined to produce the final prediction by averaging for regression or voting for classification. Bagging is successful in reducing overfitting, increasing prediction accuracy, and improving the stability of the model [19,62–65].

2.3.2 Boosting

Boosting is an ensemble learning algorithm that combines multiple weak base learners to create an accurate and strong model. Unlike bagging, where the base learners are trained independently, boosting trains the base learners sequentially. Initially, all observations have equal weights. First, an equal weight value is assigned to each sample in the randomly selected small training data set. After the first model is trained, the weight value is increased for the samples that the model predicts incorrectly in the training dataset, while the weight values are decreased for the samples that it predicts correctly. In this way, multiple models are created, each correcting the errors of the previous model. The final model is the weighted average

of all models. The strengths of each algorithm are exploited and the weaknesses are compensated. Boosting helps to avoid the problem of underfitting by reducing the bias of the model. AdaBoost, Gradient Boosting, and XGBoost are examples of boosting algorithms [66,67].

2.3.3 Stacking

Stacking is a technique frequently used in the fields of ensemble learning and machine learning. This method is ideal for solving complex problems and allows different types of models to be combined to create customized solutions. Stacking produces a higher-performing prediction using predictions of various types of models as inputs for the meta-learner. Each model in the ensemble complements each other and the output of the next model is used as the input of the previous model. Thus, models can produce more accurate results by working in interaction with each other. Stacking ensemble learning is particularly useful when working with large datasets and can be used to reduce noise and get accurate results. This technique reduces the bias and variance of the model, helping to prevent overfitting and underfitting problems [68–70].

2.3.4 Majority Voting

The majority voting algorithm ensures accurate predictions as decisions are made through the collaboration of multiple models. Predictions from multiple models are combined. Different machine learning models such as DT, SVM, k-NN, RF or LR are trained on the same dataset, and the predictions of these models are combined. Each model votes on the outcome and the final prediction is determined by the majority decision. The majority voting algorithm is a simple but powerful technique in machine learning, used in many fields due to its improved accuracy. It is flexible and compatible with a variety of machine learning models and algorithms [71]. The majority voting algorithms can be used in the medical field, the finance industry, the field of image classification, NLP and the detection of fraud activities [72,73].

2.4 FEATURE EXTRACTION

Images consist of pixels, which are the smallest units representing color or intensity values. Features are distinctive patterns or characteristics derived from the pixel values in an image. Extracting and analyzing these features is a fundamental

step in computer vision tasks, enabling machines to understand and interpret visual information. By analyzing the patterns and relationships among pixels, we can extract useful information and derive higher-level features from images. Features can be local, describing specific regions or objects within an image, or global, representing the overall properties of the entire image. Feature extraction involves analyzing the pixel values in an image to identify meaningful patterns. Once features are extracted from an image, they can be further processed, combined, or used as input for machine learning models to perform tasks like image classification, object detection, or image retrieval. Feature extraction from images is to obtain the image data as quantitative data in accordance with the structure of machine learning algorithms. Feature extraction is the process of extracting features such as edges, shapes, textures, and colors, which are used to describe the content of the image. Feature extraction techniques can be categorized into two main types; handcrafted features and automatic features learned by deep learning models. There are many algorithms that perform handcrafted feature extraction, such as HOG, SURF, LBP, SIFT, and GLCM. [74]. The major difference between CNNs and traditional machine learning methods is that CNNs directly extract image features without the need for manual feature extraction. CNN is very successful in feature extraction. The main reason why CNN is considered the best feature extractor is that CNN can get more features than other methods, capture higher quality and powerful features, and improve accuracy in less time. In CNN, feature extraction is done through the convolution layer. CNNs can be used as feature extractors (Figure 7) [75,76].

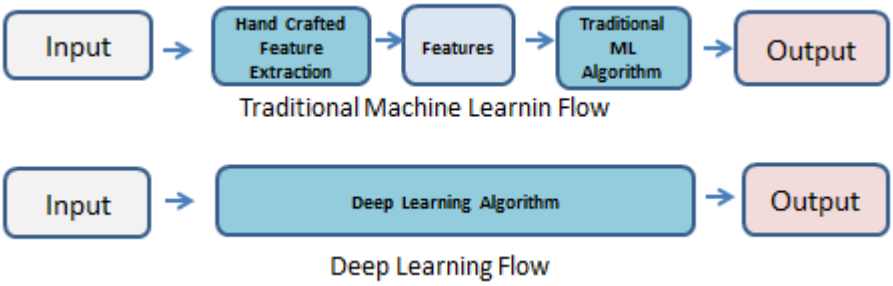


Figure 7: Traditional machine learning and deep learning methods flowchart

2.5 FEATURE SELECTION

Feature selection reduces the number of features in a dataset. Feature selection does not create new features, but rather aims to rank existing features in the dataset by importance and discard less important ones. Irrelevant features do not provide useful information. Redundant features are features that do not provide more information than the currently selected features. With feature selection, the size of the dataset is reduced, irrelevant or unnecessary features are removed, thus improving the performance of the learning algorithm. Feature selection helps reduce noise, overfitting, and bias in the model, leading to improved prediction accuracy. Reducing the number of features can reduce the computation time required to train and make predictions and make the learning process more efficient. Feature selection methods are grouped under 3 main groups: filter, wrapper, and embedded methods. The filter method uses statistical tools and thus provides fast results. Wrapper methods use machine learning algorithms and relate to the classifier at each stage. In embedded methods, machine learning algorithms and feature selection methods work together and also establish a relationship with the classifier at each stage (Figure 8) [23,77,78].

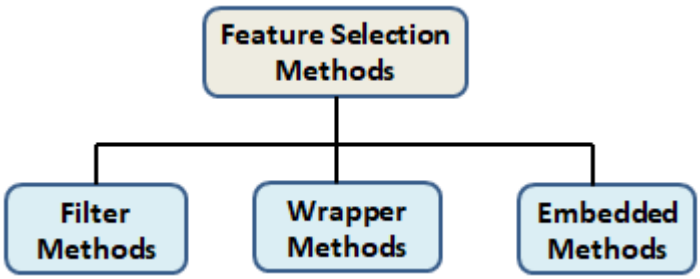


Figure 8: Feature selection methods diagram.

2.5.1 Feature Selection Techniques

Filter feature selection methods use statistical measures to rank features based on their correlation with the target variable. Filter method algorithms work fast, are less complex, and are more explainable than embedded and wrapped methods. Examples of filter methods include Pearson Correlation Coefficient, Analysis of Variance (ANOVA), information gain, mutual information and chi-square test [79,80].

Pearson's Correlation method measures the linear correlation between each feature and the target variable. It calculates the Pearson correlation coefficient, which ranges from -1 to 1. The chi-square test is used for feature selection when the target variable is categorical and the features are categorical [81].

ANOVA is a popular statistical method commonly used for feature selection in situations where the target variable is categorical and the features are continuous or categorical. ANOVA calculates two types of variability: between-group variability and within-group variability. The between-group variability measures how much the target variable varies between different groups. The within-group variability measures how much the target variable varies within each group. The F-statistic is calculated by dividing the between-group variability by the within-group variability.

$$F = \frac{Variance\ between\ samples}{Variance\ within\ samples} \tag{2.4}$$

It represents the ratio of the explained variation between groups to the unexplained variation within groups. The F-statistic is compared against a critical value or threshold to determine the significance of the feature. If the input data is numerical and the target variable is categorical, the two most frequently used feature selection methods for classification are the ANOVA f-test statistic and the mutual information statistic [81,82]. Feature selection is performed according to the input and output variable properties [83]. Figure 9 shows the feature selection methods choosing diagram.

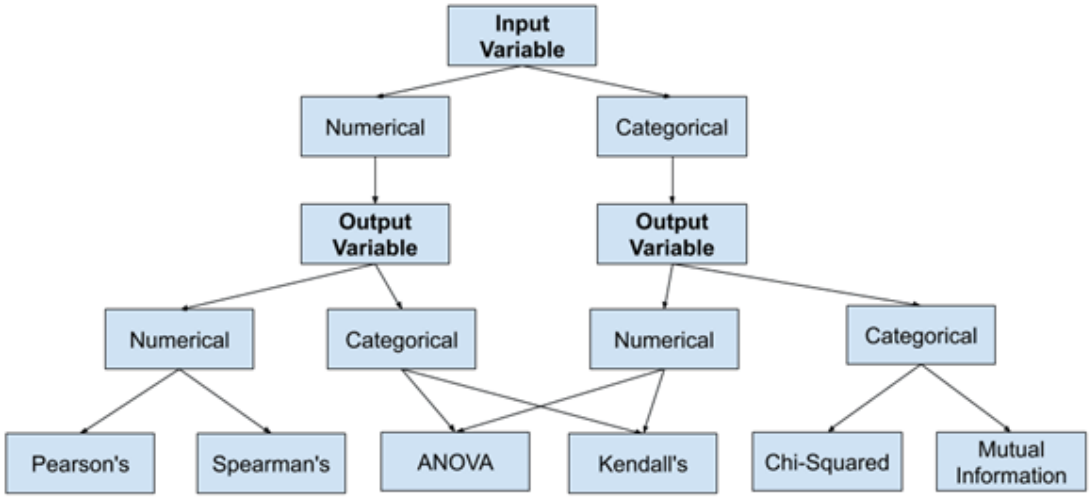


Figure 9: Choosing a feature selection method for machine learning [83].

Variance Threshold, another filter feature selection method, removes all features whose variance is below a certain threshold. It is based on the assumption that features that do not vary much within themselves have low predictive power. By default, it only removes features with zero variance [84].

Relief algorithm is an algorithm that uses the filter method in feature selection. The relief algorithm is designed for application to binary classification problems. It calculates a feature score for each feature and this score can then be applied to sort and select the features with the highest score for feature selection [85].

In wrapper methods, the feature selection process is embedded within the learning algorithm itself. In this method, the best subset creation and selection technique is more successful than filtering methods; however, it is slower and more computationally expensive. It involves evaluating different subsets of features by training and testing the model iteratively. Examples include recursive feature elimination, forward selection, and backward elimination. Wrapper methods can potentially lead to overfitting, especially when the dataset is small or noisy. Wrapper feature selection methods try to select the most valuable features by using various search algorithms on the features. Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO) algorithms can be given as examples of this method [86]. The Boruta feature selection method, built with the RF Classifier algorithm, is another wrapper feature selection group algorithm [87].

In embedded methods, both feature selection algorithms and classification algorithms are used together. They aim to find the best subset of features while

simultaneously optimizing the model's performance. Examples include Lasso regularization, decision tree-based feature importance, and regularization techniques such as Elastic Net. Embedded methods strike a balance between the efficiency of filter methods and the performance of wrapper methods. Embedded feature selection methods are structures with higher computational costs that include feature selection and classification algorithms at the same time. For example, machine learning algorithms such as Lasso and RF have feature selection algorithms suitable for their structure [23,88].

2.6 DIMENSIONALITY REDUCTION

In machine learning, high-dimensional data contains a large number of features or variables. As the number of features increases, the complexity of the model increases, which can lead to the emergence of overfitting. The Curse of Dimensionality is that the performance of the model decreases as the number of features increases. Dimensionality reduction can improve generalization performance by reducing the complexity of the model. The dimensionality reduction is the process of transforming high-dimensional data into a lower-dimensional field that still retains the essence of the original data. The dimensionality reduction process can be performed with Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Singular Value Decomposition (SVD) techniques as well as feature selection methods [80].

2.7 RELATED WORKS

Atallah R and Al-Mousa A, (2023) used the Cleveland dataset in their study for heart disease detection. They achieved 88% accuracy with the Stochastic Gradient Descent (SGD) classifier, 87% accuracy with the k-NN classifier, 87% accuracy with the RF classifier, and 87% accuracy with the LR classifier. The majority voting ensemble method accuracy result was 90% [89].

Karadeniz T et al, (2023) used Statlog, and Spectf datasets in their study for heart disease detection. They achieved better performance than ANN, SVM and Naive Bayes algorithms with two self-developed algorithms based on the majority voting algorithm. With different base estimators, they obtained 88% and 83% accuracy results with the Spectf dataset, and 88% and 87% accuracy results with the Statlog dataset [90].

Benyahia et al. (2022), in their study with ISIC 2019 and PH2 datasets for the classification of skin lesions, performed feature extraction with 17 pre-trained CNN networks, made classification with 24 machine learning algorithms and compared the performances of the models. In this study, feature extraction was made from intermediate layer activations. For the ISIC 2019 dataset, it was observed that DenseNet-201 together with k-NN and SVM performed better in terms of classification accuracy (92.34% vs 91.71%) [75].

Mehedi Masud et al. (2021) analyzed the chest radiographs and made a three-class classification to investigate the presence of pneumonia and its type (bacterial or viral). 70% of the dataset (4.974 samples) was used for training and 30% (2.132 samples) for testing. After using an augmentation technique, general characteristics as well as statistical properties of the lung X-ray images were extracted with a deep learning method. Then, both extracted feature groups were combined and feature selection was made by genetic algorithm. The final classification was performed using the RF classifier. This model can classify samples of the dataset with 86.30% accuracy and 86.03% F1 score, indicating the efficiency and reliability of the model [91].

Khanh Ho, T. K., and Gwak, J. (2019) studied a total of 112.120 chest radiographs in their study to classify 14 thorax pathologies. They used a classification approach that combines four hand-crafted features, HOG, SIFT, GIST, and LBP with deep features obtained with CNN. Training was performed with pre-trained DenseNet-121, 70% of the data was used for training, 10% for validation, and 20% for testing. With the feature integration approach, 84.62% classification accuracy was achieved, which is higher than the 80.97% accuracy of the DenseNet-121 model [74].

Shrivastava P et al. (2021), in their study to diagnose COVID-19, first combined chest CT and chest X-ray images of patients. The authors first performed classification with pre-trained ResNet-50, InceptionV4, and EfficientNetB0 networks on the combined dataset. These network performances were found to be compatible with each other. They obtained performance results of accuracy of 97.47%, sensitivity of 98.18%, specificity of 96.6%, and AUC of 95.36% with the Maximum Voting (ensemble learning) method. The Max Voting model achieved ~0.1% improvement in sensitivity, ~0.14% in specificity, and ~0.88% in accuracy compared to pre-trained individual models [92].

Sozan MA and Ramadhan JM (2022) classified knee osteoarthritis using knee X-ray images. Knee OA is classified into 5 groups according to the Kellgren-Lawrence grading system [93,94]. They used a pre-trained CNN model for feature extraction. They worked on the Osteoarthritis Initiative (OAI) dataset containing 9.786 knee X-ray images. Firstly, CLAHE was applied during preprocessing to make an accurate classification and improve the contrast of the radiographs. 80% of the dataset was used for training, 20% for testing, and 10% of the training dataset was used for validation. The Principal Component Analysis (PCA) algorithm was applied to obtain the most distinctive feature set. Then the SVM algorithm was used for classification. In addition to the Kellgren-Lawrence grading system, classification studies were carried out by grouping the data set in different ways. The accuracy was 62% with the 5-class classification, 87% with the 3-class classification, and 90.8% with the 2-class classification [95].

Hamid Nasiri H and Alavi SA. (2022), in their study to diagnose COVID-19, feature extraction from chest X-rays was performed with DenseNet169. Feature selection was performed by analysis of variance (ANOVA) to improve accuracy and reduce time and computations. ANOVA is a statistical approach that ranks features by calculating variances within and between dataset groups. The classification was performed with the XGBoost classifier. ANOVA selected 67 features out of 1664 for classification. This study achieved 98.72% accuracy in two-class classification (COVID-19, No Symptoms) and 92% accuracy in multi-class classification (COVID-19, Pneumonia, and No Symptoms). With their method, precision, recall and specificity rates of 99.21%, 93.33% and 100%, respectively, were obtained for two-class classification, and 94.07% precision, 88.46% recall and 100% specificity results were obtained for multi-class classification [82].

Prakash JA et al. (2023) used chest X-ray images in their study for pneumonia classification in children. They used the pre-trained Xception model for feature extraction. Extracted features were submitted to PCA for dimensionality reduction. In the first stage, the RF classifier, k-NN, LR, XGB classifier, SVM classifier, Nu-SVC, and MLP classifier are used. The second stage works on the LR algorithm. With the test of the model, 98.3% accuracy, 99.29% precision, 98.36% recall, 98.83% F1 score and 98.24% AUC score were obtained [96].

Ayaz M et al. (2021) used the Montgomery and Shenzhen datasets to diagnose tuberculosis (TB). In this study, handcrafted features extracted through the

Gabor filter and deep features extracted through 7 pre-trained deep learning models were combined in the ensemble learning method. The ensemble learning classifier was the Logistic Regression algorithm. The maximum accuracy achieved with the Montgomery dataset is 93.47% and AUC 0.97, and the maximum accuracy achieved for the Shenzhen dataset is 90.6% and AUC 0.94 [97].

Ureten K and Maraş HH. (2022) worked with 368 RA and 333 normal hand images in their study to diagnose RA from plain hand radiographs. During the preprocessing, the YOLOv4 algorithm was used to discard the parts of the image that were not required for training, such as the patient name, hospital name, and direction sign on the radiographs. 40 hand radiographs were used for YOLOv4 model training and 10 hand radiographs were used for validation. All X-ray images were cropped with this object detector obtained with YOLOv4. 70% of the dataset was used for training, 15% for validation, and 15% for testing. In this study, the pre-trained VGG-16 model was used for transfer learning and the results were successful. In the two-class classification for distinguishing normal and RA hand X-rays, 90.7% accuracy, 92.6% sensitivity, 88.7% specificity, 89.3% precision, and 0.97 AUC results were obtained. In the two-class classification for distinguishing normal and OA hand X-rays, 90.8% accuracy, 91.4% sensitivity, 90.2% specificity, 91.4% precision, and 0.96 AUC results were obtained. In the three-class classification for the distinguish of normal, RA, and OA hand radiographs, the accuracy was 80.6% [98].

In a recent study by Ma Y et al. (2023), hand radiography images of 7 hospitals were used, 9.714 hand radiographs were used for training the model, 250 hand radiographs were used for testing, transfer learning was applied with the EfficientNet-B0 CNN, and data augmentation was applied. Testing of the model yielded an AUC of 0.975 for arthritis versus non-arthritis. For RA versus non-RA hand radiographs, an AUC of 0.955 was obtained with the model. With this model, 0.806 Cohen's kappa and 87.2% accuracy were achieved in the test set for three-way classification (RA, OA, and Normal) [99].

Muzoğlu N et al. (2022) used CT images of 196 patients in their study to diagnose COVID-19. First, the pre-trained VGG-16 model using transfer learning was applied and 92.34% accuracy was achieved. Then, feature extraction was done from the fc8 layer of the pre-trained VGG-16 network, and feature selection was performed with the Boruta algorithm. The Boruta algorithm duplicates the existing features, and then creates a mixed version of the dataset using newly added features

called "shadow features". Unnecessary features are eliminated with the RF classifier. As a result of the optimization, it has been determined that 473 of 1000 features are useful. These features were classified by SVM and LDA by selecting 200, 300, and 400 features according to the efficiency order of the 473 features obtained. The accuracy was increased to 97.02% after the study with the SVM classifier using the 300 most valuable features selected with the Boruta selection algorithm [87].

CHAPTER III
MATERIALS and METHODS

3.1 DATASET

In this retrospective study, plain hand X-rays of the patients who were examined in the rheumatology outpatient clinic of Kırıkkale University Faculty of Medicine between January 1, 2012, and March 1, 2021, were used. Ethics committee approval was obtained from the local ethics committee for the study. Hand X-rays of patients with suspected RA were taken in the postero-anterior position of both hands together. These hand X-rays were classified as normal hand radiography, and RA hand radiography by 2 Rheumatologists (Abdurrahman Tufan, Medical Faculty of Gazi University, Department of Rheumatology and Levent Kılıç, Medical Faculty of Hacettepe University, Department of Rheumatology) with over 10 years' experience, without being aware of each other. In case of disagreement between the two experts, X-ray was excluded from the study. Table 1 shows the properties of the study data. Figure 10 shows samples of normal hand and RA hand X-rays.

Table 1: Study data properties

	Female	Male	Total	Mean age
RA hand X-rays	249	119	368	49
Normal hand X-rays	225	108	333	45

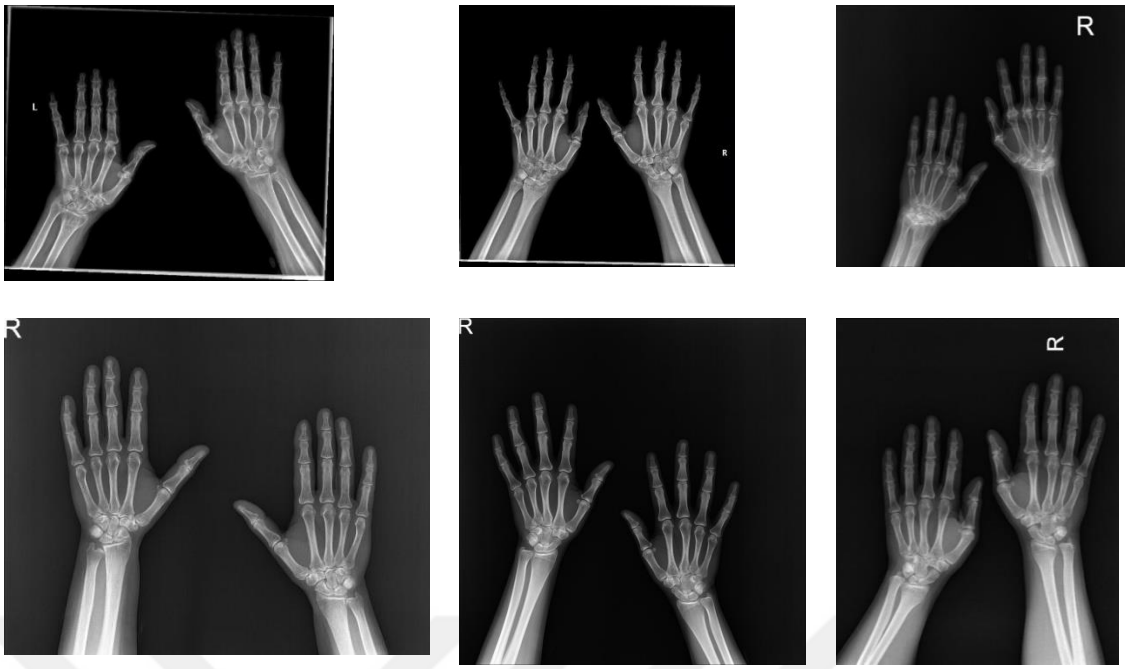


Figure 10: Samples of hand X-rays with RA (top), normal hand X-rays (bottom).

3.2 DATA PREPROCESSING

Some of the hand X-rays used in this study had the patient's name, date, some numbers, artifacts, and various directional signs and were of different sizes (width, height). Figure 11 shows noisy hand X-ray image samples.

Figure 11: Noisy hand X-ray image samples.

To eliminate these unnecessary, nuisance areas during preprocessing, both hand images had to be cropped from the entire image. The YOLOv4 algorithm was used as an object detector to automate the cropping task. 50 radiographs (40 X-rays for training, 10 X-rays for validation) were labeled in YOLO format on the online MakeSense.AI platform. YOLOv4 uses the pre-trained Darknet53 network. Table 2 shows the YOLOv4 hyperparameters used in this study.

Table 2: YOLOv4 hyperparameters.

Batch size	16
Subdivisions	8
Momentum	0.9
Learning rate	0.001
Iteration	2000

At the end of 2000 iterations, the object detector was obtained. With this object detector, all images in the data set were automatically cropped with the help of a non-maximum suppression algorithm. Figure 12 shows the YOLOv4 training chart. The classification task was performed with these cropped images. Figure 13 shows the hand radiography images with bounding boxes and a non-maximal suppression algorithm applied to this image.

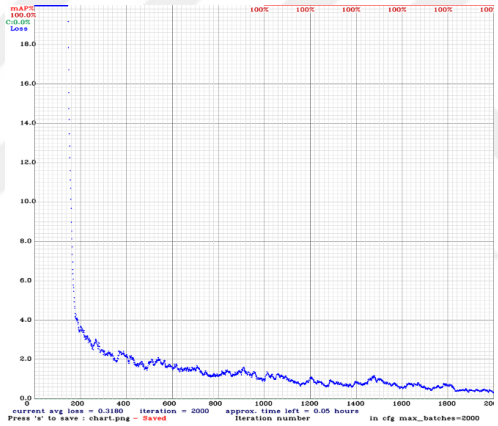


Figure 12: YOLOv4 training chart.

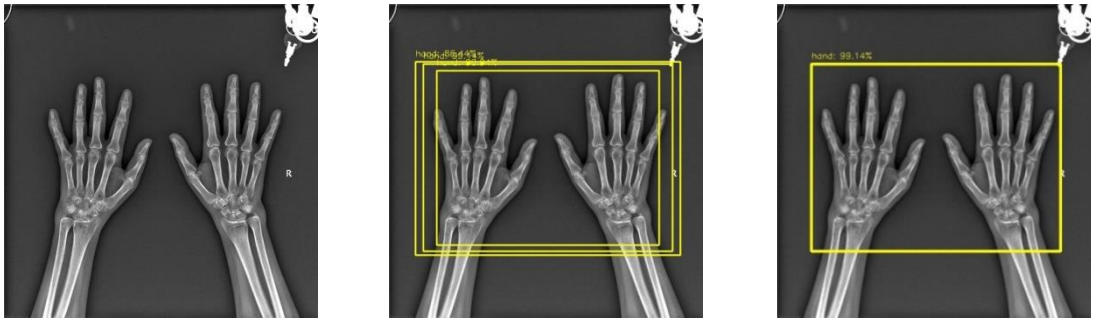


Figure 13: The bounding boxes obtained on the hand X-rays

The data set obtained by the cropping process was randomly split into training (85%) and test set (15%). The test set was not used during training and

validation, 15% of the training set was used for validation, validation set was used for parameter optimization. Table 3 shows this splitting.

Table 3: Numbers of training, validation and test images.

	Training	**Validation**	**Test**	**Total**
Rheumatoid Arthritis	266	46	56	368
Normal	240	42	51	333

3.3 IMPLEMENTATION OF THE PROPOSED MAJORITY VOTING

In the majority voting algorithm, different machine learning classifiers are trained on the same dataset, and the final prediction is made by majority decision. For each feature, a classification decision is made and the overall class label is determined by voting. For each feature, an optimal width of segments is found, and these segments are used to classify the input attribute. Assume that, as a toy example, the label sequence associated with an attribute is 0000111101011000. This sequence can be divided into segments such as 0000|1111|0101|1000 or 00001|11101|01100|0. In the first example, the segment size is 4, whereas in the second example segment size is 5. Each segment can be thought of as a decision maker and assigned to the segment the majority of its labels. For example, in the first segmentation 0000|1111|0101|1000, decisions from each segment are 0|1|0|0. The overall success of segmentation is measured by the total true decisions inferred from it. Here, in the first segmentation, the first segment and second segment have an accuracy of 1.0, the third segment has an accuracy of 0.5 (since two 1's are misclassified), and the last segment has an accuracy of 0.75 (only 1 is misclassified). The overall segmentation has success than in average (1.0 + 1.0 + 0.75 + 0.5)/4 = 0.8125. On the other hand, the second segmentation has a success rate (0.8 + 0.8 + 0.6 + 1.0)/4 = 0.8. Thus, it can be said that the first segmentation captures the classes more accurately.

The core idea is to find the optimal segmentation for each feature. min_width, max_width and step are the parameters to search for ideal segmentation. The process starts with segments of width min_width and goes until max_width by incrementing width in steps. The success of each segmentation is measured by the aforementioned method. This somehow corresponds to "blurring" the label sequence obtained by a feature. After determining for each feature the ideal segmentation, a sample's corresponding attribute is classified by the segment it falls. From each feature, a

classification decision is made and the overall class label is determined by the majority of the votes. Figure 14 shows some of the methods used in the prediction.

```
def __split(self, x, width):
  sl = []
  for i in range(0, len(x), width):
    bnd = min(len(x), i + width - 1)
    sl.append(x[i: bnd])
  return sl

def __binary_acc(self, x):
  acc_0 = float(len(x) - sum(x))/len(x)
  acc_1 = 1.0 - acc_0
  ret_val = (0, acc_0) if acc_0 > acc_1 else (1, acc_1)
  return ret_val

def __calc_max_acc_seg(self, x):
  max_acc = 0
  best_width = 0
  best_seg_labels = []
  for width in range(self.min_width, self.max_width + 1, self.step):
    sl = self.__split(x, width)
    s = 0
    seg_labels = []
    max_acc_0 = 0
    for seg in sl:
      label, acc = self.__binary_acc(seg)
      seg_labels.append(label)
      s += acc
    s /= len(sl)
    if s > max_acc:
      max_acc = s
      best_width = width
      best_seg_labels = seg_labels

  return (best_width, best_seg_labels, max_acc)
```

Figure 14: Proposed majority voting classifier source code example

3.4 FEATURE SELECTION PREPROCESSING

ANOVA is a statistical method used to analyze the differences between means, it is often used to evaluate the effect of independent variables on the dependent variable. ANOVA can be used for feature selection in datasets where the independent variables are numerical and the target variable is categorical. The SelectKBest function of the *scikit-learn* library and the F-statistics component (f_classif()) are used in this work. ANOVA basically evaluates the correlation between the characteristics of the data. Each of the features of the data is ranked according to the F-statistics component, and features with higher scores can be

selected as the optimal set of components. SelectKBest is used to select the features with the best variance, the f_classif() function is used as the scoring metric, and the *k* value represents the number of features. *k* is chosen by the analyst, and *k* number of features with the best variance are classified with a machine learning classifier. Figure 15 shows an example of ANOVA feature selection algorithm source codes.

```
import pandas as pd
import numpy as np

from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import f_classif
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt

import warnings
warnings.filterwarnings("ignore")

                                        # ANOVA feature selection
# load dataset
X_train = pd.read_csv("vgg_90/train.csv").values
y_train = pd.read_csv("vgg_90/train_labels.csv").values
X_test = pd.read_csv("vgg_90/test.csv").values
y_test = pd.read_csv("vgg_90/test_labels.csv").values


fvalue_Best = SelectKBest(f_classif, k= 50)      # k = number of columns
X_kbest = fvalue_Best.fit_transform(X_train, y_train)

fvalue_Best = SelectKBest(f_classif, k= 50)       # k = number of columns
X_ktest = fvalue_Best.fit_transform(X_test, y_test)
```

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score

fvalue_Best = SelectKBest(f_classif, k= 50)      # k = number of columns

fvalue_Best.fit(X_train, y_train)
X_kbest = fvalue_Best.transform(X_train)
clf = RandomForestClassifier()

clf.fit(X_kbest, y_train)
y_test_pred = clf.predict(X_ktest)

accuracy = accuracy_score(y_test, y_test_pred)
print('Accuracy: %.2f' % (accuracy*100))
```

```
Accuracy: 81.90
```

Figure 15: ANOVA feature selection algorithm source codes example.

ANOVA ranks features by calculating variances within and between groups. Finding the *k* parameter by trial and error is tedious and time-consuming. In this study, the machine learning classifier was inserted into the ANOVA algorithm to find the best *k* number, and the optimum number of features was searched iteratively. Alternatively, all features are first ranked from bigger to smaller according to the F1 score value. Various methods are applied to find the optimal set of features among

these candidate features. Figure 16 shows an example of ANOVA with *all* feature selection algorithm source codes. Figure 17 shows f-scores of these features.

```
import pandas as pd
import numpy as np
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import f_classif
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression,SGDClassifier,RidgeClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, roc_auc_score
from sklearn.metrics import cohen_kappa_score, matthews_corrcoef
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore")

# load data
X_train = pd.read_csv("vgg_90/train.csv").values
y_train = pd.read_csv("vgg_90/train_labels.csv").values
X_test = pd.read_csv("vgg_90/test.csv").values
y_test = pd.read_csv("vgg_90/test_labels.csv").values
```

```
# configure to select all features to obtain ANOVA f-scores
fs = SelectKBest(score_func = f_classif, k= 'all')
fs.fit(X_train, y_train)
X_train_fs = fs.transform(X_train)
X_test_fs = fs.transform(X_test)
```

```
#fs.scores_
```

Figure 16: ANOVA f-scores source codes example

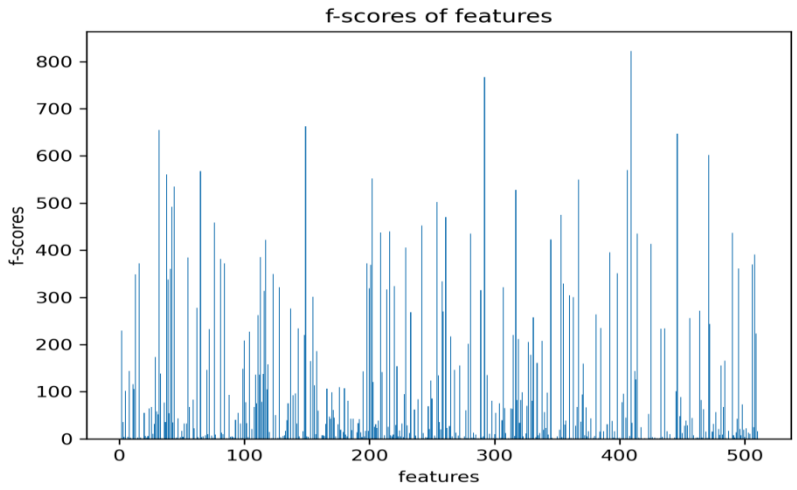
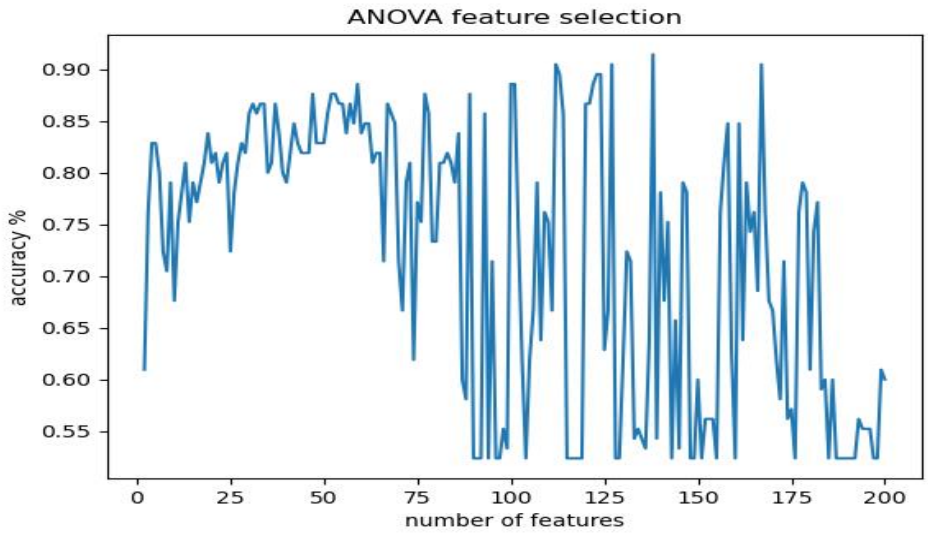


Figure 17: ANOVA f-scores of the features.

Each selected feature does not affect the accuracy positively, the accuracy may decrease with the added feature, and it is difficult to decide on the optimal number of features. Figure 18 shows how the accuracy changes with the number of features selected.



x axis shows the number of selected features, y axis shows accuracy

Figure 18: Accuracy changes with selected features

A large number of features are extracted from images, therefore, feature selection becomes more important when working with images. Feature selection reduces the risk of overfitting and improves performance by eliminating irrelevant features. Furthermore, training time and computational costs are reduced. Statistical tests are used to filter feature selection methods. The ANOVA F-value estimates the linearity between the input feature and the output feature. The variance threshold method eliminates features that are below a predetermined threshold value.

3.5 STUDY ENVIRONMENT

Google Colab environment was used to train the YOLOv4 algorithm. Similarly, the classification process using a pre-trained VGG-16 network was carried out on the Google Colab environment. Other preprocessing tasks, machine learning applications, and statistical studies were performed on a computer with Intel(R) Core(TM) i5-3230M CPU, 2.60 GHz speed, 8 GB RAM, and an internal graphics card. Python 3.9 programming language was used. Required libraries were implemented and run in Keras and Tensorflow environments. The *scikit-learn* library was used for statistical calculations. The steps followed during this study are given in flowchart (Figure 19).

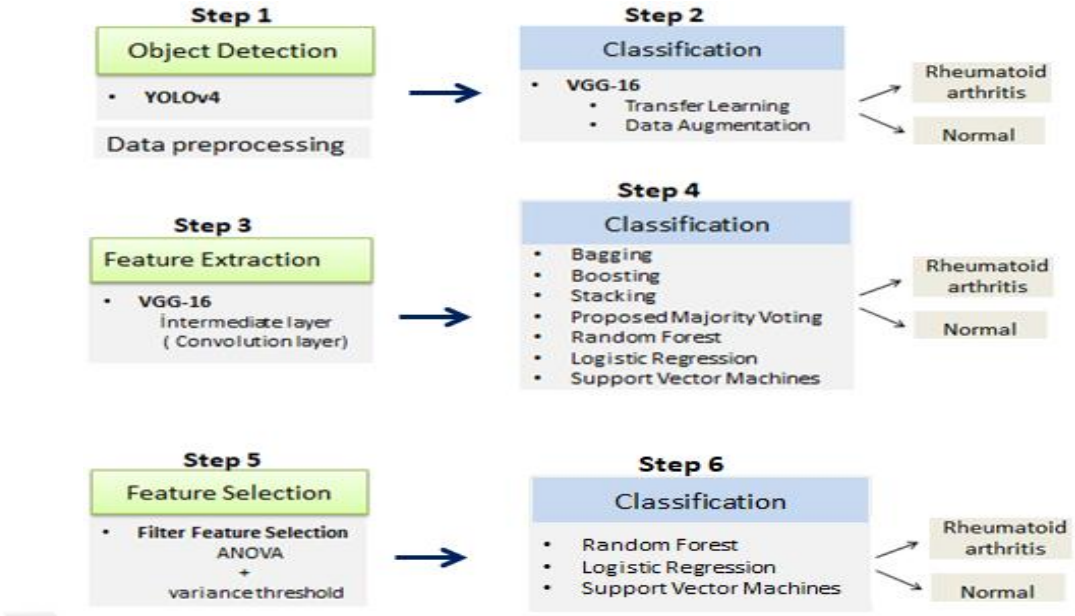


Figure 19: Steps followed throughout this study

3.6 TRANSFER LEARNING, DATA AUGMENTATION

If the number of data is insufficient for CNN training from scratch, transfer learning and data augmentation methods are applied to overcome this problem. We achieved successful results with VGG-16 in our previous studies [100–102]. For this reason, we performed transfer learning with VGG-16 in this study. VGG-16 is a pre-trained network that classifies 1000 different categories trained on the Imagenet dataset. ImageNet dataset images have RGB channels and 224*224 dimensions [17]. Hand X-rays used in this study have RGB channels. Therefore, the cropped images were resized to 224*224 for training with VGG-16, and the original classifier was replaced with a binary classifier. Table 4 shows the VGG-16 hyperparameters used in this study.

In this study, online data augmentation was applied to augment the number of training data, using the following parameters: rotation_range 25 degrees, zoom_range 0.1 pixels, width_shift_range 0.1 pixels, height_shift_range was set to 0.1 pixels. Horizontal_flip was applied. As a result of augmentation, 12.682 images were obtained.

Table 4: VGG-16 hyperparameters used in this study

optimizer	adam
batch_size	32
learning_rate	5e-5
epochs	25
loss	categorical_crossentropy

3.7 STATISTICAL ANALYSIS

The performance of a classification model can be evaluated using confusion matrices. A confusion matrix summarizes the performance of a machine learning model on test data and shows the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) produced by the model [103].

Table 5: Confusion matrix diagram

		Actual	
		Abnormal	Normal
Predicted	Abnormal	TP	FP
	Normal	FN	TN

true positives (TP), an instance that is actually abnormal and predicted as abnormal by the model

true negatives (TN), an instance that is actually normal and predicted as normal by the model

false positives (FP), an instance that is actually normal and predicted as abnormal by the model

false negatives (FN), an instance that is actually abnormal and predicted as normal by the model.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.5}$$

$$sensitivity\ (recall) = \frac{TP}{TP + FN} \tag{3.6}$$

$$specificity = \frac{TN}{TN + FP} \tag{3.7}$$

$$precision = \frac{TP}{TP + FP} \tag{3.8}$$

$$F1\ score = \frac{2 * (precision * recall)}{(precision + recall)} = \frac{2TP}{2TP + FP + FN} \tag{3.9}$$

$$kappa = \frac{Po - Pe}{1 - Pe} = \frac{2 * (TP * TN - FN * FP)}{(TP + FP) * (FP + TN) + (TP + FN) * (FN + TN)} \tag{3.10}$$

Po =observed agreement, *Pe* = expected agreement

A receiver operating characteristic curve (ROC curve) is a curve that shows the performance of a classification model across all classification thresholds. This curve shows the True Positive Rate (TPR) and the False Positive Rate (FPR);
TPR is synonymous with recall and sensitivity and is defined as:

$$TPR\ (sensitivity = recall) = \frac{TP}{TP + FN} \tag{3.11}$$

$$specificity = \frac{TN}{TN + FP} \tag{3.12}$$

FPR is defined as:

$$FPR\ (1 - spesificity) = \frac{FP}{FP + TN} \tag{3.13}$$

The ROC curve compares TPR and FPR at different classification thresholds. ROC is a probability curve and AUC represents the degree or measure of separability (Figure 20). AUC shows the ratio of the area under the curve to the total area. It tells how well the model can distinguish the classes. The higher the AUC, the better the model is at predicting class 0 as 0 and class 1 as 1. Similarly, the higher the AUC, the better the model is at distinguishing between X-rays with and without the disease [104].

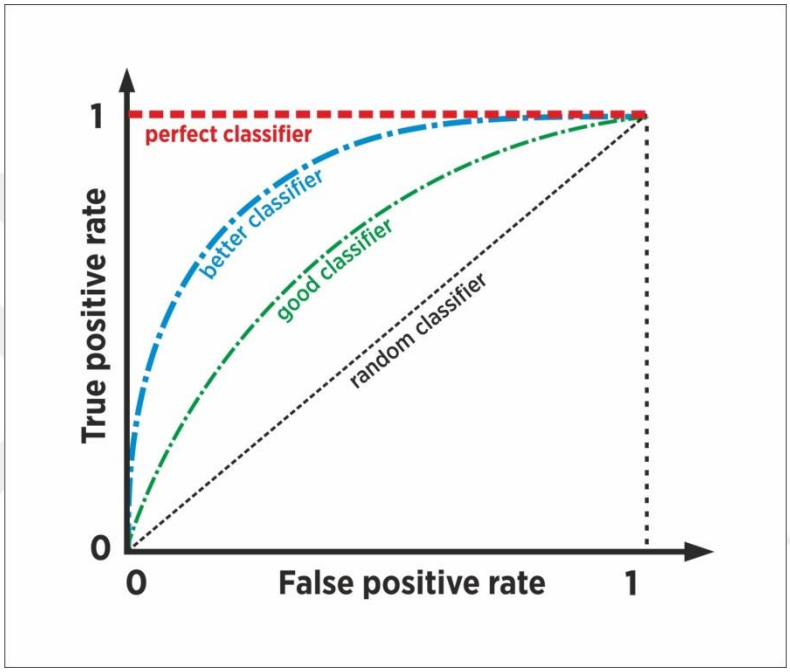


Figure 20: ROC curve diagram

CHAPTER IV
RESULTS

In this study data augmentation and transfer learning methods were applied with pre-trained VGG-16 network to classify normal and RA hand X-rays. VGG-16 model accuracy, sensitivity, specificity, precision, F1 score, area under the curve (AUC), and Cohen's kappa results are shown in Table 5. The confusion matrix and ROC curve obtained from the VGG-16 network are shown in Figure 21.

Table 6: Performance metric results obtained with the VGG-16 model.

	Accuracy	**Sensitivity**	**Specificity**	**Precision**	**F1 score**	**AUC**	**Cohen's kappa**
VGG-16	90.5	96.0	85.7	85.7	90.5	0.94	0.81

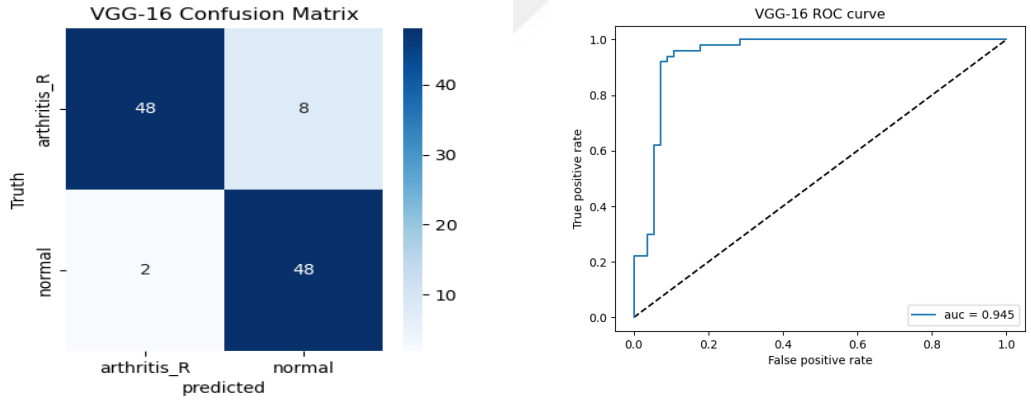


Figure 21: VGG-16 network confusion matrix and ROC curve.

VGG-16 model intermediate layer was used for feature extraction. Bagging (Random Forest Classifier), boosting (AdaBoost Classifier), and stacking (Random Forest, k-Nearest Neighbor, Logistic Regression Classifier) ensemble learning algorithms were applied to these extracted features. Table 6 shows the base estimators and hyperparameters of the ensemble learning models used in this study.

Table 7: The base estimators and hyperparameters of the ensemble learning models

Model	**base_estimator, classifier, parameters**
Bagging	base_estimator= RF classifier, n_estimators=200, random_state= 10 RandomForestClassifier(min_samples_leaf= 1, n_estimators=200, max_features = 2, max_depth = 100, bootstrap =True)
Boosting	AdaBoostClassifier(DecisionTreeClassifier(max_depth=1), n_estimators=200
Stacking	base_estimator: RandomForestClassifier(n_estimators=10, random_state=42), KNeighborsClassifier(n_neighbors=5) meta_estimator = LogisticRegression()

The bagging model elapsed time was 69.64 seconds, the boosting model elapsed time was 4.30 seconds, stacking model elapsed time was 0.64 seconds. Table 7 shows bagging, boosting and stacking performance metric results. Figures 22, 23 and 24 show the confusion matrices and ROC curves obtained by the bagging, boosting and stacking methods.

Table 8: Performance metric results of ensemble learning methods.

	Accuracy	**Sensitivity**	**Specificity**	**Precision**	**F1 score**	**AUC**	**Cohen's kappa**
Bagging	90.5	92.0	89.0	88.4	90.1	0.95	0.80
Boosting	90.5	88.0	92.7	91.6	89.7	0.95	0.80
Stacking	91.4	94.0	89.0	88.6	91.2	0.94	0.82
VGG-16	90.5	96.0	85.7	85.7	90.5	0.94	0.81

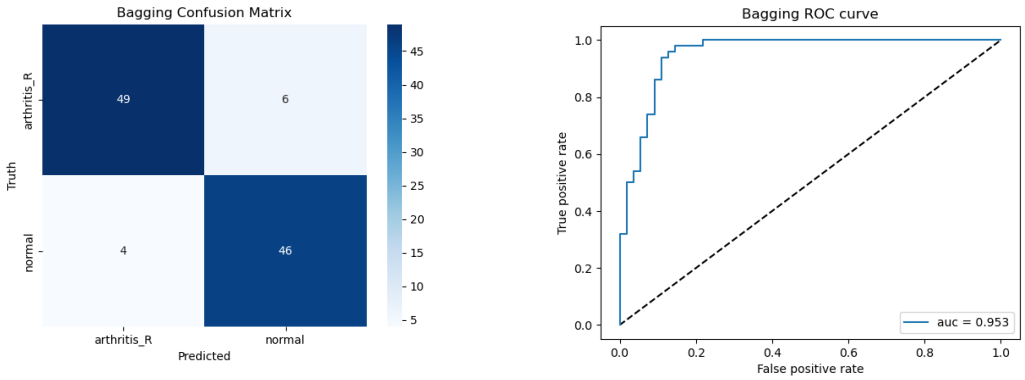


Figure 22: The confusion matrix and ROC curve obtained with bagging method.

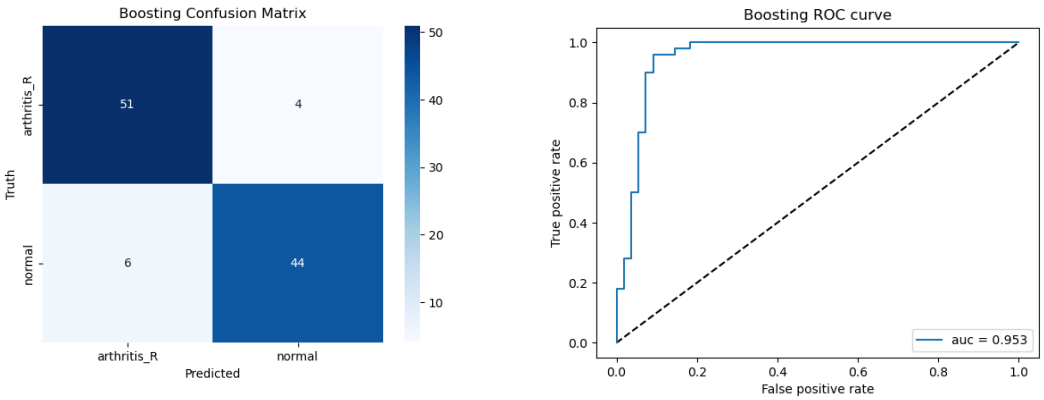


Figure 23: The confusion matrix and ROC curve obtained with boosting method.

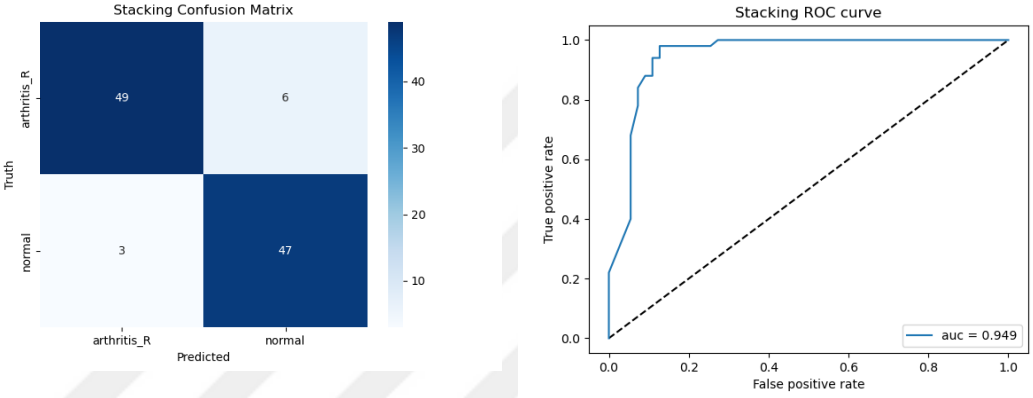


Figure 24: The confusion matrix and ROC curve obtained with stacking method.

In the implementation of the proposed majority voting algorithm, the base classifier is fed into a bagging scheme. Table 8 shows the proposed majority voting model's parameters. Table 9 shows the proposed majority voting model performance metrics, Figure 25 shows the confusion matrix and ROC curve. The proposed majority voting model elapsed time was 10.46 seconds. Figure 26 shows the accuracy scores of VGG-16 and ensemble learning models bar chart.

Table 9: Proposed majority voting model parameters

Model	parameters
Proposed Majority Voting	Base estimator: min_width = 10, max_width = 21, step = 1
	Bagging classifier: n_estimators =300, bootstrap = True, max_samples = 0.61, max_features = 0.03

Table 10: Performance metric results obtained by the proposed majority voting model.

	Accuracy	Sensitivity	Specificity	Precision	F1 score	AUC	Cohen's kappa
Voting	91.5	94.0	89.2	88.6	91.2	0.95	0.83
VGG-16	90.5	96.0	85.7	85.7	90.5	0.94	0.81

Voting: proposed majority voting model

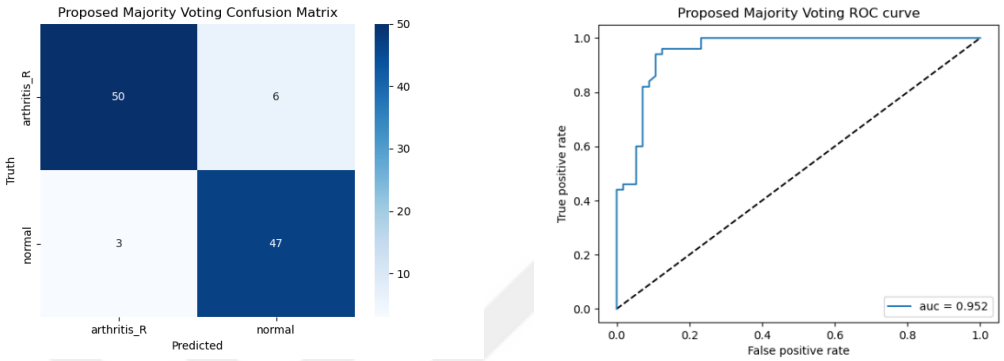


Figure 25: The confusion matrix and ROC curve obtained by proposed majority voting method.

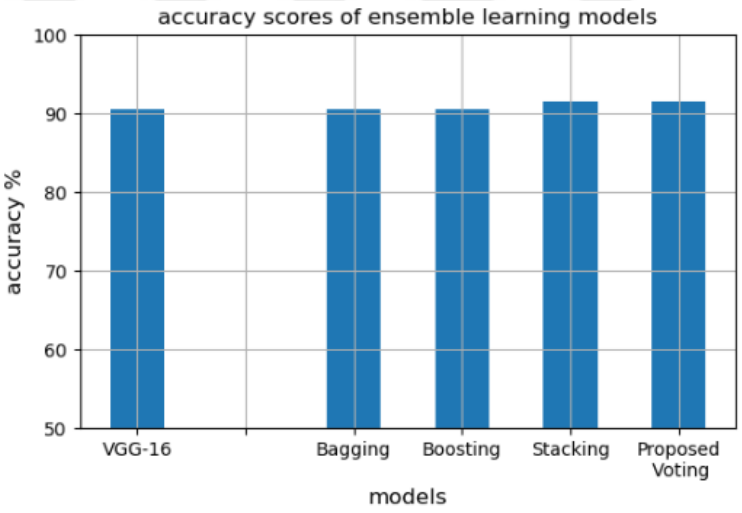


Figure 26: Accuracy scores of VGG-16 and ensemble learning models

The application of the proposed ANOVA feature selection method was carried out with the RF Classifier, LR classifier and SVM classifiers. Table 10 shows the hyperparameters of the machine learning classifiers used in this study.

Table 11: Machine learning classifiers hyperparameters used in this study

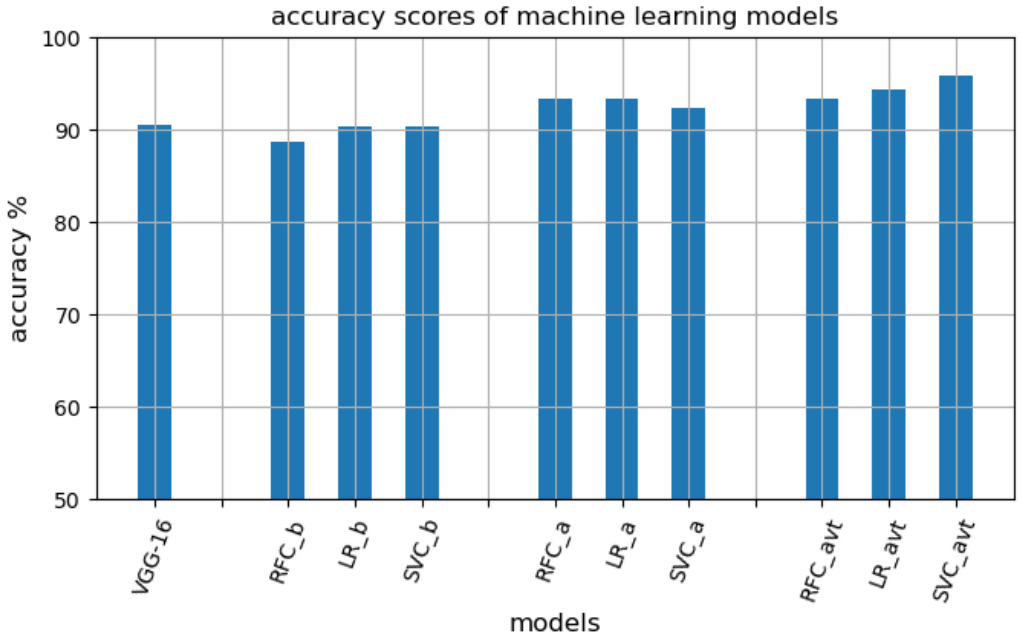
Kernel model	**Parameters**
LogisticRegression	solver='lbfgs', max_iter=200
RandomForestClassifier	n_jobs=-1, class_weight='balanced', max_depth= 5
SupportVectorClassifier	probability=True, default parameters

ANOVA and Variance threshold feature selection methods were applied to extracted features. Machine learning classifier inserted into ANOVA algorithm to find the optimal set of features. With the Variance threshold of 0, the number of features decreased from 512 to 441 and the performance of the ANOVA model did not change. With a threshold value of 0.1, the number of features decreased to 217, with a threshold value of 0.2, the number of features decreased to 147, and with a threshold value of 0.3, the number of features decreased to 101, and the performance of the model was not negatively affected. By applying the variance threshold and ANOVA, the number of features decreased from 512 to 101, the elapsed time decreased from 148.78 seconds to 22.62 seconds with the RF Classifier, from 24.70 seconds to 7.8 seconds with the LR classifier and from 43.97 seconds to 4.07 seconds with the SVM Classifier. Table 11 shows the performance metric results of machine learning algorithms. Figure 27 shows the accuracy scores of the VGG-16 model and machine learning models before and after feature selection methods.

Table 12: Performance metric results of machine learning algorithms before and after the proposed feature selection methods

	Accuracy	Sensitivity	Specificity	Precision	F1 score	AUC	Cohen's kappa
VGG-16	90.5	96.0	85.7	85.7	90.5	0.94	0.81
RFC	88.6	90.0	87.2	86.5	88.2	0.94	0.77
LR	90.4	90.0	90.9	90.0	90.0	0.94	0.80
SVC	90.4	98.0	83.6	84.4	90.7	0.94	0.81
After the proposed feature selection (ANOVA) method							
RFC	93.3	98.0	89.0	89.0	93.3	0.95	0.86
LR	93.3	96.0	90.9	90.5	93.2	0.94	0.86
SVC	92.3	98.0	87.2	87.5	92.4	0.95	0.84
After the proposed feature selection (ANOVA+variance threshold) method							
RFC	93.4	95.4	91.2	92.6	94.0	0.96	0.86
LR	94.3	96.9	91.2	92.7	94.8	0.97	0.88
SVC	95.9	98.4	92.9	94.2	96.2	0.97	0.91

RFC: RandomForestClassifier, LR: Logistic Regression, SVC: SupportVectorClassifier



RFC: RandomForestClassifier, LR: Logistic Regression, SVC: SupportVectorClassifier
'b' stands for before feature selection, 'a' stands for after feature selection ANOVA,
'avt' stands for after feature selection ANOVA + variance threshold

Figure 27: Accuracy scores of VGG-16 and machine learning models

To evaluate the performances of this proposed models on another dataset, proposed models were applied to the femoral neck fracture dataset. The femoral neck fracture dataset included 302 normal femur X-ray images and 296 femoral neck fracture images. First, training was carried out with pre-trained VGG-16 model, then feature extraction was performed from the the intermediate layer of the VGG-16 model, and feature selection was made with ANOVA and variance threshold. The proposed majority voting model and the proposed ANOVA feature selection model achieved improvement in all performance metrics in the femoral neck fracture dataset compared to the VGG-16 model. The performance results of the proposed models obtained with the femoral neck fracture dataset are shown in Table 12.

Table 13: Performance metrics obtained by applying the proposed models on the femoral neck fracture dataset.

	Accuracy	**Sensitivity**	**Specificity**	**Precision**	**F1 score**	**AUC**	**Cohen's kappa**
VGG-16	95.6	95.5	93.3	95.7	95.5	0.98	0.91
Voting	95.8	97.1	94.7	94.4	95.7	0.98	0.91
Before feature selection (ANOVA + variance threshold)							
RFC	92.2	93.4	90.0	91.4	92.4	0.98	0.84
LR	95.5	93.4	97.7	97.7	95.5	0.98	0.91
SVC	94.4	93.4	95.4	95.5	94.5	0.98	0.88
After feature selection (ANOVA + variance threshold)							
RFC	96.1	98.1	94.2	94.5	96.2	0.99	0.92
LR	97.1	94.5	100.0	100.0	95.1	0.99	0.94
SVC	99.0	100.0	98.0	98.1	99.0	0.99	0.98

VGG: VGG-16 network, Voting: Proposed majority voting, RFC: Random Forest Classifier, LR: Logistic Regression, SVC: Support Vector Classifier

CHAPTER V
DISCUSSION & CONCLUSION

5.1 DISCUSSION

In this study, the YOLOv4 algorithm was used as the object detector during preprocessing. All radiographs were cropped to include only hand images. Data augmentation and transfer learning were applied with the VGG-16 network, and plain hand X-rays were classified as normal or RA hand radiographs. Accuracy, sensitivity, specificity, precision, F1 score, AUC, and Cohen's kappa performance results of 90.5%, 96.0%, 85.7%, 85.7%, 90.5%, 0.94, and 0.81 were obtained, respectively. Feature extraction was performed from the pre-trained VGG-16 network intermediate layer and extracted features were classified by bagging, boosting, and stacking algorithms. The stacking method provided a 0.9% improvement in the accuracy metric. The proposed majority voting model that performs segmental search, outperformed the pre-trained VGG-16 model by 1% in accuracy. The proposed ANOVA+variance threshold method based on feature selection and finding the optimal number of features, improved the accuracy metric by 2-5% over the VGG-16 network with machine learning algorithms such as RF, LR and SVM. Improvements in other performance metrics and training time were achieved with both proposed techniques.

In this study, the performance of the models was measured using accuracy, sensitivity, specificity, precision, F1 score, AUC and Cohen's kappa metrics, which are frequently used metrics in the medical studies. A choice can be made between performance metrics in accordance with the purpose. It may be desirable to have higher sensitivity in one study and higher specificity in another, or vice versa [105]. With the proposed ANOVA method, feature selection can be made using other performance metrics as well as accuracy. In other words, the number of features that give the best results according to the preferred metric such as accuracy, sensitivity, specificity, precision and F1 score can be determined by the user.

Cohen's kappa is a statistical method that shows whether the agreement between two classifiers is due to chance. It is also used in unbalanced datasets. It takes values between -1 and +1. A Cohen's kappa value of 0 indicates that there is no agreement between the classifiers, and a value between 0.81 and 1 indicates that there is a very good agreement between the classifiers and that the result is not due to chance. In this study, Cohen's kappa score of proposed models was over 0.80, and in some models it was over 0.90, indicating that the agreement between the classifiers was perfect [106].

In recent years, many studies have been carried out using machine learning, deep learning and ensemble learning methods on both image processing and tabular data. To improve the performance of the models, studies are carried out with different feature extraction, feature selection and dimensionality reduction methods. Likewise, studies are being carried out to improve the performance of the algorithms [74,75,82,87,89,90,95–98,107].

Recently, some studies have been conducted with radiographs of hands with RA. In our previous study to classify normal, RA and osteoarthritis hand radiographs, a transfer learning method was applied using a pre-trained VGG-16 network. The dataset consisted of 368 RA hand X-rays and 333 normal hand X-rays. In that study, we obtained 90.7%, 92.6%, 88.7%, 89.3%, and 0.97 accuracy, sensitivity, specificity, precision, and AUC performance results, respectively in the two-class classification as RA and normal hand radiograph [98]. In their study with 9.714 hand radiographs, Ma Y et al. applied transfer learning with EfficientNet-B0 CNN and obtained an AUC result of 0.955 with the model for distinguishing RA and non-RA hand radiographs [99]. Wang HJ et al. used the YOLO algorithm in their study to calculate the Modified Total Sharp Score in RA hand finger joints. The average accuracy of joint classification was 0.88, while the accuracy of severe, mild, and healthy joint classifications reached 0.91, 0.79, and 0.9, respectively [108]. In these studies, classification was done using deep learning methods. In our study, classification was first performed with the pre-trained VGG-16 model, and then automatic feature extraction was performed with this model. In this way, performance was improved as a result of studies conducted with ensemble learning methods, machine learning methods and the two methods we proposed.

Some studies have been done recently to improve the performance of majority voting classifiers. Atallah R and Al-Mousa A, in their study with the

Cleveland data set, obtained better accuracy results with the majority voting method than other classifiers [89]. Karadeniz T and colleagues worked with Statlog and Spectf datasets and improved the performance of the majority voting classifier with the methods they proposed [90]. In their study with chest images, Shrivastava P and colleagues performed classification with pre-trained ResNet-50, InceptionV4, and EfficientNetB0 networks. With the max voting algorithm, they achieved ~0.1% improvement in sensitivity, ~0.14% improvement in specificity and ~0.88% improvement in accuracy compared to pre-trained models [92]. In this study, we achieved an improvement in the performance of the models by adding a segmental search property to the majority voting classifier, the improvement in accuracy was 1% compared to the pre-trained VGG-16 network.

ANOVA is a popular statistical method, if the input data is numeric and the target variable is categorical, the ANOVA f-test statistic is the appropriate feature selection method for classification. ANOVA ranks features by calculating variances within and between dataset groups. The ANOVA f-value estimates the linearity between the input feature and the output feature. In studies conducted with ANOVA, a two-stage method is applied; in the first step, the features are ranked according to the f-score, and in the second step, an appropriate number of features are selected from these features with various algorithms. In some studies, the number of features was determined by trial and error [82]. To find the optimal set of features among these candidate features, studies were carried out with algorithms such as genetic algorithm [109], amino acid pairs implemented classifier [107], particle swarm optimization [110], MapReduce-based ANOVA [111], Jaya-based forest optimization algorithm [112], and SVM based on normalized poly kernel algorithm [113]. In our proposed ANOVA method, the classifier is implemented into the ANOVA algorithm, and the feature set is tested iteratively to find the optimal number of features. In addition, variance threshold, another filter feature selection method, was applied along with ANOVA. By default, it only removes features with zero variance. With a variance threshold value of 0.3, the number of features decreased from 512 to 101, the elapsed time decreased from 148.78 seconds to 22.62 seconds with the RF Classifier, and from 5.07 seconds with the SVM Classifier. Compared to deep learning methods, 2-5% improvements were achieved in the model's performance metrics.

In this study, which was conducted to determine normal and RA hand X-rays, there were improvements in performance and elapsed times with the proposed majority voting method and proposed ANOVA+variance threshold feature selection method. Similar improvements were obtained in femoral neck fracture classification, which was done to evaluate the performance of the proposed models on another data set.

This study has some limitations. The first of these is the small number of data which might not capture the caracteristics of the wider patient population. Transfer learning with VGG-16, and data augmentation methods have been applied to overcome this problem. Secondly, the study was conducted with X-rays obtained from a single center. The performance of the models can be improved through multicenter studies and thus the proposed models can be used safely in daily practice.

5.2 CONCLUSION

In this study, which was conducted to classify normal and RA hand radiographs, feature extraction was performed using a pre-trained VGG-16 model. Extracted features were first trained using ensemble learning algorithms bagging, boosting and stacking. Studies were then carried out with the extracted features using two novel methods introduced to the literature, segmental search majority voting algorithm and ANOVA feature selection methods. Among the machine learning algorithms, RF, LR and SVM algorithms were used. Improvements in classification performance and training time were achieved with the majority voting classifier and ANOVA+variance threshold feature selection model proposed in this work. Then, the femur fracture dataset was classified with both proposed methods, and similar improvements were obtained showing that the methods can be generalized. Finding sufficient data in the medical field poses difficulties for reasons such as patient privacy, workload, and the rarity of some diseases. Therefore, to use existing data efficiently, it is necessary to continue improving classifiers, feature extraction methods, feature selection methods and other performance-improving studies.

REFERENCES

[1] Chen Hung Jen, Shuai Hong Han and Cheng Wen Huang (2023), "A Survey of Artificial Intelligence in Fashion", *IEEE Signal Processing Magazine*, Vol.40, Issue 3, pp. 64–73.

[2] Aletaha Daniel and Smolen Josef S. (2018), "Diagnosis and Management of Rheumatoid Arthritis: A Review", *JAMA - Journal of the American Medical Association*, Vol.320, Issue 13, pp. 1360–1372.

[3] Lee David M. and Weinblatt Michael E. (2001), "Rheumatoid arthritis", *The Lancet*, Vol.358, Issue 9285, pp. 903–911.

[4] McQueen Fiona M. (2013), "Imaging in early rheumatoid arthritis", *Best Practice and Research: Clinical Rheumatology*, Vol.27, Issue 4, pp.499–522.

[5] Smolen Josef S, Aletaha Daniel, Bijlsma Johannes W.J, Breedveld Ferdinand C, Boumpas Dimitrios, Burmester Gerd, Burmester Gerd, Combe Bernard, Cutolo Maurizio, de Wit Maarten, Dougados Maxime, Emery Paul, Gibofsky Alan, Gomez-Reino J.Jesus, Haraoui Boulos, Kalden Joachim, Keystone Edward C, Kvien Tore K, McInnes Iain, Martin-Mola Emilio, Montecucco Carlomaurizio, Schoels Monika, van der Heijde Desirée (2010), "Treating rheumatoid arthritis to target: Recommendations of an international task force", *Annals of the Rheumatic Diseases*, Vol.69, Issue 4, pp.631–637.

[6] Thabet Mohamed M, Huizinga Thomas W.J, Van der Heijde Désirée M and Van der Helm-van Mil Annette H.M. (2009), "The prognostic value of baseline erosions in undifferentiated arthritis", *Arthritis Research and Therapy*, Vol.11, Issue 5, pp.1–9.

[7] Jindal Gunjan, Bansal Saloni, Gupta Nishu, Singh Sanjay Kumar, Gahukar Shailesh, and Kumar Ashok (2021), "Comparison of Ultrasonography and X-Rays for the Diagnosis of Synovitis and Bony Erosions in Small Joints of Hands in Early Rheumatoid Arthritis: a Prospective Study", *Mædica - a Journal of Clinical Medicine*, Vol.16, Issue 1, pp.22–28.

[8] McQueen Fiona M., Stewart Neal, Crabbe Jeff, Robinson Elizabeth, Yeoman Sue, Tan Paul L.J., McLean Lachy (1999), "Magnetic resonance imaging of the wrist in early rheumatoid arthritis reveals progression of erosions despite clinical improvement", *Annals of the Rheumatic Diseases*, Vol.58, Issue 3, pp.156–163.

[9] Giovagnoni Andrea, Valeri Gianluca, Burroni Elisabetta and Amici Francesco (1998), "Rheumatoid arthritis: follow-up and response to treatment", *European Journal of Radiology*, Vol.27 (SUPPL. 1), pp.25–30.

[10] Rudwaleit M., Van Der Heijde D., Landewé R., Listing J., Akkoc N., Brandt J., Braun J., Chou C.T., Collantes-Estevez E., Dougados M., Huang F., Gu J., Khan M.A., Kirazli Y., Maksymowych W.P., Mielants H., Sørensen I. J., Ozgocmen S., Roussou E., Valle-Oñate R., U. Weber U., Wei J., Sieper J. (2009), "The development of Assessment of SpondyloArthritis international Society classification criteria for axial spondyloarthritis (part II): Validation and final selection", *Annals of the Rheumatic Diseases*, Vol.68, Issue 6, pp.777–783.

[11] Greenspan Hayit, Van Ginneken Bram, and Summers Ronald M. (2016), "Guest Editorial Deep Learning in Medical Imaging: Overview and Future Promise of an Exciting New Technique", *IEEE Transactions on Medical Imaging*, Vol.35, Issue 5, pp.1153–1159.

[12] Shen Dinggang, Wu Guorong, and Suk Heung Il (2017), "Deep Learning in Medical Image Analysis", Vol.19, pp.221–248, DOI:10.1146/Annurev-Bioeng-071516-044442.

[13] Fan Jiayi, Lee Janghyeon, and Lee Yongkeun (2021), "A Transfer Learning Architecture Based on a Support Vector Machine for Histopathology Image Classification", *Applied Sciences*, Vol.11, Issue 14, pp.6380.

[14] Yamashita Rikiya, Nishio Mizuho, Do Richard Kinh Gian, and Togashi Kaori (2018), "Convolutional neural networks: an overview and application in radiology", *Insights into Imaging*, Vol.9, Issue 4, pp.611–629.

[15] Krizhevsky Alex, Sutskever Ilya, and Hinton Geoffrey E. (2012), "ImageNet Classification with Deep Convolutional Neural Networks", *In, Advances in Neural Information Processing Systems*, Eds. F. Pereira and C.J. Burges and L. Bottou and K.Q. Weinberger, Curran Associates, Inc., New York.

[16] Szegedy Christian, Wei Liu, Yangqing Jia, Sermanet Pierre, Reed Scott, Anguelov Dragomir, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich (2015), "Going deeper with convolutions", *IEEE Conf. Comput. Vis. Pattern Recognit*., pp. 1–9, Boston, USA.

[17] Simonyan Karen and Zisserman Andrew (2014), "Very Deep Convolutional Networks for Large-Scale Image Recognition", *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp.1–14, San Diego, USA.

[18] He Kaiming, Zhang Xiangyu, Ren Shaoqing, and Sun Jian (2016), "Deep Residual Learning for Image Recognition", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, Las Vegas, USA.

[19] Tan Mingxing and Le Quoc V. (2019), "EfficientNet: Rethinking model scaling for convolutional neural networks", *36th International Conference on Machine Learning, ICML 2019*, pp.10691–10700, Long Beach, California.

[20] Howard Andrew G, Zhu Menglong, Chen Bo, Kalenichenko Dmitry, Wang Weijun, Weyand Tobias, Marco Andreetto, Hartwig Adam. (2017), "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications", *CoRR*, Vol. abs/1704.04861.

[21] Huang Gao, Liu Zhuang, Van Der Maaten Laurens, and Weinberger Kilian Q. (2017) "Densely Connected Convolutional Networks", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.4700-4708, Honolulu, USA.

[22] Kumar Gaurav and Bhatia Pradeep Kumar (2014), "A detailed review of feature extraction in image processing systems", *International Conference on Advanced Computing and Communication Technologies*, pp.5–12, Rohtak, India.

[23] Jović A, Brkić K, and Bogunović N. (2015), "A review of feature selection methods with applications", *38th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2015 - Proceedings*, pp.1200–1205, Opatija, Croatia.

[24] Aly Ghada Hamed, Marey Mohammed, El-Sayed Safaa Amin, and Tolba Mohamed Fahmy (2020), "YOLO Based Breast Masses Detection and Classification in Full-Field Digital Mammograms", *Computer Methods and Programs in Biomedicine,* Vol.200, pp.105823.

[25] Bochkovskiy Alexey, Wang Chien-Yao, and Liao Hong-Yuan Mark (2020), "YOLOv4: Optimal Speed and Accuracy of Object Detection", *CoRR,* Vol. abs/2004.10934, DOI:10.48550/arXiv.2004.10934.

[26] Cheng Richeng (2020), "A survey: Comparison between Convolutional Neural Network and YOLO in image identification", *Journal of Physics: Conference Series,* Vol. 1453, pp.12139.

[27] Diwan Tausif, Anirudh G, and Tembhurne Jitendra V. (2023), "Object detection using YOLO: challenges, architectural successors, datasets and applications", *Multimedia Tools and Applications,* Vol.82, Issue 6, pp.9243–9275.

[28] Redmon Joseph, Divvala Santosh, Girshick Ross, and Farhadi Ali (2016), "You Only Look Once: Unified, Real-Time Object Detection", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp.779-788, Las Vegas, USA.

[29] Terven Juan R and Cordova-Esparaza Diana M. (2023), "A Comprehensive Review of YOLO: From YOLOv1 and Beyond", *arXiv preprint arXiv:2304.00501*.

[30] Nie Yali, Sommella Paolo, O'Nils Mattias, Liguori Consolatina, and Lundgren Jan, (2019) "Automatic detection of melanoma with yolo deep convolutional neural networks", *E-Health and Bioengineering Conference (EHB) IEEE.* pp. 1-4. Romania.

[31] Neubeck Alexander and Van Gool Luc (2006), "Efficient non-maximum suppression", *In, Proceedings - International Conference on Pattern Recognition, IEEE,* Vol.3, pp.850–855.

[32] Jiang Peiyuan, Ergu Daji, Liu Fangyao, Cai Ying, and Ma Bo (2022), "A Review of Yolo Algorithm Developments", *Procedia Computer Science,* Vol.199, pp.1066–1073.

[33] Roy Arunabha M, Bose Rikhi, and Bhaduri Jayabrata (2022), "A fast accurate fine-grain object detection model based on YOLOv4 deep neural network", *Neural Computing and Applications,* Vol.34, Issue 5, pp.3895–3921.

[34] Carbonell Jaime G., Michalski Ryszard S., and Mitchell Tom M. (1983), "An Overview of Machine Learning", *In, Machine Learning*, pp.3–23, Springer, Berlin, Heidelberg.

[35] Liu Hongyu and Lang Bo (2019), "Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey", *Applied Sciences*, Vol.9, Issue 20, pp.4396.

[36] Mahesh Batta (2018), "Machine Learning Algorithms-A Review", *International Journal of Science and Research (IJSR)*, Vol.9, Issue 1, pp.381-386.

[37] Hutter Frank, Kotthoff Lars, and Vanschoren Joaquin (2019), *Automated Machine Learning: methods, systems, challenges*, Eds. Frank H., Lars Kotthoff, Joaquin Vanschoren, pp. 219, Springer.

[38] Jiang Tammy, Gradus Jaimie L, and Rosellini Anthony J. (2020), "Supervised Machine Learning: A Brief Primer", *Behavior Therapy*, Vol.51, Issue 5, pp.675–687.

[39] Breiman Leo (2001), "Random forests", *Machine Learning*, Vol.45, Issue 1, pp.5–32.

[40] Quinlan J.R. (1986), "Induction of decision trees", *Machine Learning*, Vol.1, Issue 1, pp.81–106.

[41] Costa Vinícius G. and Pedreira Carlos E. (2023) "Recent advances in decision trees: an updated survey", *Artificial Intelligence Review*, Vol.56, Issue 5, pp.4765-4800.

[42] Vanfretti Luigi and Arava V.S. Narasimha (2020), "Decision tree-based classification of multiple operating conditions for power system voltage stability assessment", *International Journal of Electrical Power & Energy Systems*, Vol.123, pp.106251.

[43] Xing Yanwei, Wang Jie, Zhao Zhihong, and Gao and Yonghong (2007), "Combination Data Mining Methods with New Medical Data to Predicting Outcome of Coronary Heart Disease", *International Conference on Convergence Information Technology (ICCIT 2007)*, pp. 868-872, Gwangju, Korea.

[44] Singh Sonia and Giri Manoj (2014), "Comparative Study Id3, Cart And C4.5 Decision Tree Algorithm: A Survey", *International Journal of Advanced Information Science and Technology (IJAIST)*, Vol.3, Issue 27. Issue *27*, pp.97-103.

[45] Bhatia Nitin and Vandana (2010), "Survey of Nearest Neighbor Techniques", *IJCSIS) International Journal of Computer Science and Information Security*, Vol.8, No. 2, pp. 302-305.

[46] Cover T.M. and Hart P.E. (1967), "Nearest Neighbor Pattern Classification", *IEEE Transactions on Information Theory*, Vol.13, Issue 1, pp.21–27.

[47] Chomboon Kittipong, Chujai Pasapichi, Teerarassammee Pongsakorn, Kerdprasop Kittisak, and Kerdprasop Nittaya (2015), "An Empirical Study of Distance Metrics for k-Nearest Neighbor Algorithm", *Proc. 2nd Int. Conf. Ind. Appl. Eng. 2015, The Institute of Industrial Applications Engineers*, pp. 280–285.

[48] Aydın Can (2018), "Makine Öğrenmesi Algoritmaları Kullanılarak İtfaiye İstasyonu İhtiyacının Sınıflandırılması", *Avrupa Bilim ve Teknoloji Dergisi*, Vol.14, pp.169–175.

[49] Goel Rati (2021), "Heart Disease Prediction Using Various Algorithms of Machine Learning", *Proceedings of the International Conference on Innovative Computing & Communication (ICICC-2021)*, Delhi, India.

[50] Blanquero Rafael, Carrizosa Emilio, Ramírez-Cobo Pepa, and Sillero-Denamiel M. Remedios (2021), "Variable selection for Naïve Bayes classification", *Computers & Operations Research*, Vol.1, Issue 135, pp.105456.

[51] Yao Jingxuan and Ye Yuntao (2020), "The effect of image recognition traffic prediction method under deep learning and naive Bayes algorithm on freeway traffic safety", *Image and Vision Computing*, Vol.1, Issue 103, pp.103971.

[52] Parthiban G, Rajesh A, Srivatsa S.K, and Professor Sr (2011), "Diagnosis of Heart Disease for Diabetic Patients using Naive Bayes Method", *International Journal of Computer Applications*, Vol.24, Issue 3, pp.975–8887.

[53] Kost Samuel, Rheinbach Oliver, and Schaeben Helmut (2021), "Using logistic regression model selection towards interpretable machine learning in mineral prospectivity modeling", *Geochemistry*, Vol.81, Issue 4, pp.125826.

[54] Boateng Ernest Yeboah, Abaye Daniel A, Boateng Ernest Yeboah, and Abaye Daniel A. (2019), "A Review of the Logistic Regression Model with Emphasis on Medical Research", *Journal of Data Analysis and Information Processing*, Vol.7 Issue 4, pp.190–207.

[55] Awoyemi John O, Adetunmbi Adebayo O, and Oluwadare Samuel A. (2017), "Credit card fraud detection using machine learning techniques: A comparative analysis", *IEEE International Conference on Computing, Networking and Informatics, ICCNI 2017*, pp.1–9, Lagos, Nigeria.

[56] Cortes Corinna, Vapnik Vladimir, and Saitta Lorenza (1995), "Support-vector networks", *Machine Learning*, Vol.20, Issue 3, pp.273–297.

[57] Mohan Lalit, Pant Janmejay, Suyal Priyanka, and Kumar Arvind (2020), "Support Vector Machine Accuracy Improvement with Classification", *12th International Conference on Computational Intelligence and Communication Networks*, pp.477–481, Bhimtal, India.

[58] Al-shargie Fares, Tang Tong Boon, Badruddin Nasreen, and Kiguchi Masashi (2018), "Towards multilevel mental stress assessment using SVM with ECOC: an EEG approach", *Medical and Biological Engineering and Computing*, Vol.56, Issue 1, pp.125–136.

[59] Sagi Omer and Rokach Lior (2018), "Ensemble learning: A survey", *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Vol.8, Issue 4, pp.1249.

[60] Dong Xibin, Yu Zhiwen, Cao Wenming, Shi Yifan, and Ma Qianli (2020), "A survey on ensemble learning", *Frontiers of Computer Science*, Vol.14, Issue 2, pp.241–258.

[61] Zhou Zhi-Hua (2021),"Ensemble Learning", *In,Machine Learning*, pp.181–210, Springer, Singapore.

[62] Ali Hanae Aoulad, Mohamed Chrayah, Abdelhamid Bouzidi, Ourdani Nabil, and Alami Taha El (2022), "A Comparative Evaluation use Bagging and Boosting Ensemble Classifiers", *International Conference on Intelligent Systems and Computer Vision*, pp.1–6, Fez, Morocco.

[63] Kabari Ledisi G. and Onwuka Ugochukwu C. (2019), "Comparison of Bagging and Voting Ensemble Machine Learning Algorithm as a Classifier", *International Journals of Advanced Research in Computer Science and Software Engineering*, Vol.9, Issue 3, pp.19–23.

[64] Mahum Rabbia, Rehman Saeed Ur, Meraj Talha, Rauf Hafiz Tayyab, Irtaza, Aun, El-Sherbeeny, Ahmed M, and Mohammed A. El-Meligy (2021), "A Novel Hybrid Approach Based on Deep CNN Features to Detect Knee Osteoarthritis", *Sensors*, Vol.21, Issue 18, pp.6189.

[65] Mohammed Ammar and Kora Rania (2022), “An effective ensemble deep learning framework for text classification”, J*ournal of King Saud University - Computer and Information Sciences*, Vol.34, Issue 10, pp.8825–8837.

[66] Ganaie M.A, Hu Minghui, Malik A.K., Tanveer M., and Suganthan P.N. (2022), “Ensemble deep learning: A review”, *Engineering Applications of Artificial Intelligence,* Vol.1, Issue 115, pp.105151.

[67] Azmi Syeda Sarah and Baliga Shwetha (2020), “An Overview of Boosting Decision Tree Algorithms utilizing AdaBoost and XGBoost Boosting strategies”, *International Research Journal of Engineering and Technology*,Vol.7, Issue 5, pp.6867-6870.

[68] Khraisat Ansam, Gondal Iqbal, Vamplew Peter, Kamruzzaman Joarder, and Alazab Ammar (2020), “Hybrid Intrusion Detection System Based on the Stacking Ensemble of C5 Decision Tree Classifier and One Class Support Vector Machine”, *Electronics*, Vol.9, Issue 1, pp.173.

[69] Fadli Vira Fitriza, Soesanti Indah, and Nugroho Hanung Adi (2022), “Performance Review of Ensemble Learning Method Use in COVID-19 Case Detection”, *IEEE International Conference of Computer Science and Information Technology (ICOSNIKOM),* pp. 1-7, Laguboti, Indonesia.

[70] Mahajan Palak, Uddin Shahadat, Hajati Farshid, and Moni Mohammad Ali (2023), “Ensemble Learning for Disease Prediction: A Review”, *Healthcare*, Vol.11, Issue 12, pp.1808.

[71] Sevakula Rahul Kumar and Verma Nishchal Kumar (2017), “Assessing Generalization Ability of Majority Vote Point Classifiers”, *IEEE Transactions on Neural Networks and Learning Systems*, Vol.28, Issue 12, pp.2985–2997.

[72] Alotaibi Bandar and Alotaibi Munif (2021), “Consensus and majority vote feature selection methods and a detection technique for web phishing”, *Journal of Ambient Intelligence and Humanized Computing*, Vol.12, Issue 1, pp.717–727.

[73] Karadeniz Talha, Tokdemir Gül, and Maraş Hadi Hakan (2021), “Ensemble Methods for Heart Disease Prediction”, *New Generation Computing*, Vol.39, Issue 3–4, pp.569–581.

[74] Ho Thi Kieu Khanh and Gwak Jeonghwan (2019), “Multiple Feature Integration for Classification of Thoracic Disease in Chest Radiography”, *Applied Sciences*, Vol.9, Issue 19, pp.4130.

[75] Benyahia Samia, Meftah Boudjelal, and Lézoray Olivier (2022), "Multi-features extraction based on deep learning for skin lesion classification", *Tissue and Cell*, Vol.74, pp.101701.

[76] Sungheetha Akey (2021), "Design an Early Detection and Classification for Diabetic Retinopathy by Deep Feature Extraction based Convolution Neural Network", *Journal of Trends in Computer Science and Smart Technology*, Vol.3, Issue 2, pp.81-94.

[77] Haq Nutan Farah, Onik Abdur Rahman, and Shah Faisal Muhammad (2015), "An ensemble framework of anomaly detection using hybridized feature selection approach (HFSA)", *2015 SAI Intelligent Systems Conference*, pp.989–995, London, UK.

[78] Cai Jie, Luo Jiawei, Wang Shulin, and Yang Sheng (2018), "Feature selection in machine learning:A new perspective", *Neurocomputing*, Vol.300, pp.70–79.

[79] Remeseiro Beatriz and Bolon-Canedo Veronica (2019), "A review of feature selection methods in medical applications", *Computers in Biology and Medicine*, Vol.112, pp.103375.

[80] Alhassan Afnan M. and Wan Zainon Wan Mohd Nazmee (2021), "Review of Feature Selection, Dimensionality Reduction and Classification for Chronic Disease Diagnosis*", IEEE Access*, Vol.9 pp.87310–87317.

[81] Sarker Iqbal H. (2021), "Machine Learning: Algorithms, Real-World Applications and Research Directions", *SN Computer Science*, Vol.2, Issue 3, pp.1–21.

[82] Nasiri Hamid and Alavi Seyed Ali (2022), "A Novel Framework Based on Deep Learning and ANOVA Feature Selection Method for Diagnosis of COVID-19 Cases from Chest X-Ray Images", *Computational Intelligence and Neuroscience*, Vol.2022, Article ID 4694567 , DOI:10.1155/2022/4694567.

[83] Brownlee Jason (2019), *How to Choose a Feature Selection Method For Machine Learning*, https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/, DoA.

[84] Al Fatih Abil Fida Muhammad, Ahmad Tohari, and Ntahobari Maurice (2021), "Variance Threshold as Early Screening to Boruta Feature Selection for Intrusion Detection System", *In, 2021 13th International Conference on Information and Communication Technology and System*, pp.46–50.

[85] Aslan Narin, Ozmen Koca Gonca, Kobat Mehmet Ali, and Dogan Sengul (2022), "Multi-classification deep CNN model for diagnosing COVID-19 using iterative neighborhood component analysis and iterative ReliefF feature selection techniques with X-ray images", *Chemometrics and Intelligent Laboratory Systems*, Vol.224, pp.104539.

[86] El Aboudi Naoual and Benhlima Laila (2016), "Review on wrapper feature selection approaches", *2016 Int. Conf. Eng. MIS*, pp. 1–5, Agadir, Morocco.

[87] Muzoğlu Nedim, Kaya Karaaslan Melike, Halefoğlu Ahmet Mesrur, Sıddık Bekir and Yarman Binboğa (2022), "Boruta Öznitelik Seçimi Algoritması ve Derin Öğrenme Yöntemleri Kullanılarak Covid-19 Hastalığının Prognozunun Tahmini", *Afyon Kocatepe Üniversitesi Fen Ve Mühendislik Bilimleri Dergisi*, Vol.22, Issue 3, pp.577–587.

[88] Neumann Julia, Schnörr Christoph, and Steidl Gabriele (2005), "Combined SVM-based feature selection and classification", *Machine Learning*, Vol.61 Issue 1–3, pp.129–150.

[89] Atallah Rahma and Al-Mousa Amjed (2019), "Heart Disease Detection Using Machine Learning Majority Voting Ensemble Method", *2nd International Conference on New Trends in Computing Sciences*, pp.1–6, Amman, Jordan.

[90] Karadeniz Talha, Maraş Hadi Hakan, Tokdemir Gül, and Ergezer Halit (2023), "Two Majority Voting Classifiers Applied to Heart Disease Prediction", *Applied Sciences*, Vol.13, Issue 6, pp.3767.

[91] Masud Mehedi, Bairagi Anupam Kumar, Nahid Abdullah Al, Sikder Niloy, Rubaiee Saeed, Ahmed Anas and Anand Divya (2021), "A Pneumonia Diagnosis Scheme Based on Hybrid Features Extracted from Chest Radiographs Using an Ensemble Learning Algorithm", *Journal of Healthcare Engineering*, Vol. 2021, Article ID 8862089, DOI:10.1155/2021/8862089.

[92] Shrivastava Priyanshi, Singh Apurva, Agarwal Saksham, Tekchandani Hitesh and Verma Shrish (2021), "Covid detection in CT and X-Ray images using Ensemble Learning", *In, 5th International Conference on Computing Methodologies and Communication, ICCMC 2021*, pp.1085–1090, Erode, India.

[93] KELLGREN J.H. and LAWRENCE J.S. (1957), "Radiological assessment of osteo-arthrosis", *Annals of the Rheumatic Diseases*, Vol.16, Issue 4, pp.494–502.

[94] Kohn Mark D., Sassoon Adam A., and Fernando Navin D. (2016), "Classifications in Brief: Kellgren-Lawrence Classification of Osteoarthritis", *Clinical Orthopaedics and Related Research*, Vol.474, Issue 8, pp.1886–1893.

[95] Ahmed Sozan Mohammed and Mstafa Ramadhan J. (2022), "Identifying Severity Grading of Knee Osteoarthritis from X-ray Images Using an Efficient Mixture of Deep Learning and Machine Learning Models", *Diagnostics*, Vol.12, Issue 12, pp.2939.

[96] Prakash J. Arun, Ravi Vinayakumar, Sowmya V., and Soman K.P. (2023), "Stacked ensemble learning based on deep convolutional neural networks for pediatric pneumonia diagnosis using chest X-ray images", *Neural Computing and Applications*, Vol.35, Issue 11, pp.8259–8279.

[97] Ayaz Muhammad, Shaukat Furqan, and Raja Gulistan (2021), "Ensemble learning based automatic detection of tuberculosis in chest X-ray images using hybrid feature descriptors", *Physical and Engineering Sciences in Medicine*, Vol.44, Issue 1, pp.183–194.

[98] Üreten Kemal and Maraş Hadi Hakan (2022), "Automated Classification of Rheumatoid Arthritis, Osteoarthritis, and Normal Hand Radiographs with Deep Learning Methods", *Journal of Digital Imaging*, Vol.35, Issue 2, pp.193–199.

[99] Ma Yuntong, Pan Ian, Kim Stanley Y., Wieschhoff Ged G., Andriole Katherine P., and Mandell Jacob C. (2023), "Deep learning discrimination of rheumatoid arthritis from osteoarthritis on hand radiography", *Skeletal Radiology*, Vol. 53, Issue 2, pp.377-383.

[100] Üreten Kemal, Sevinç Hüseyin Fatih, İğdeli Ufuk, Onay Aslıhan, and Maraş Yüksel (2022), "Use of deep learning methods for hand fracture detection from plain hand radiographs", *Turkish Journal of Trauma and Emergency Surgery*, Vol.28, Issue 2, pp.196–201.

[101] Üreten Kemal, Erbay Hasan, and Maraş Hadi Hakan (2020), "Detection of hand osteoarthritis from hand radiographs using convolutional neural networks with transfer learning", *Turkish Journal of Electrical Engineering & Computer Sciences*, Vol.28, Issue 5, pp.2968–2978.

[102] Üreten Kemal, Maraş Yüksel, Duran Semra, and Gök Kevser (2021), "Deep learning methods in the diagnosis of sacroiliitis from plain pelvic radiographs", *Modern Rheumatology*, Vol.33, Issue 1, pp.202-206.

[103] Grandini Margherita, Bagli Enrico, and Visani Giorgio (2020), “Metrics for Multi-Class Classification: an Overview”, *arXiv*, Vol. abs/ 2008.05756, DOI:10.48550/arXiv.2008.05756.

[104] Hajian-Tilaki Karimollah (2013), “Receiver Operating Characteristic (ROC) Curve Analysis for Medical Diagnostic Test Evaluation”, *Caspian Journal of Internal Medicine*, Vol.4, Issue 2, pp.627–35.

[105] Reitsma Johannes B., Glas Afina S.., Rutjes Anne W.S, Scholten Rob J.P.M., Bossuyt Patrick M., and Zwinderman Aeilko H. (2005), “Bivariate analysis of sensitivity and specificity produces informative summary measures in diagnostic reviews”, *Journal of Clinical Epidemiology*, Vol.58, Issue 10, pp.982–990.

[106] McHugh Mary L. (2012), “Interrater reliability: the kappa statistic”, *Biochemia Medica*, Vol.22, Issue 3, pp.276–282.

[107] Wang Yu Hao, Zhang Yu Fei, Zhang Ying, Gu Zhi Feng, Zhang Zhao Yue, Lin Hao, and Ke-Jun Deng (2022), "Identification of adaptor proteins using the ANOVA feature selection technique", *Methods*, Vol.208, pp.42-47.

[108] Wang Hao Jan, Su Chi Ping, Lai Chien Chih, Chen Wun Rong, Chen Chi, Ho Liang Ying, Chu Woei-Chyn, and Lien Chung-Yueh (2022), “Deep Learning-Based Computer-Aided Diagnosis of Rheumatoid Arthritis with Hand X-ray Images Conforming to Modified Total Sharp/van der Heijde Score”, *Biomedicines*, Vol.10, Issue 6, pp.1355.

[109] Lu Lei, Yan Jihong, and de Silva Clarence W .(2016), “Feature selection for ECG signal processing using improved genetic algorithm and empirical mode decomposition”, *Measurement*, Vol.94, pp.372–381.

[110] Mahapatra Satyajit and Sahu Sitanshu Sekhar (2022), “ANOVA-particle swarm optimization-based feature selection and gradient boosting machine classifier for improved protein–protein interaction prediction”, *Proteins: Structure, Function, and Bioinformatics*, Vol.90, Issue 2, pp.443–454.

[111] Kumar Mukesh, Rath Nitish Kumar, Swain Amitav, and Rath Santanu Kumar (2015), “Feature Selection and Classification of Microarray Data using MapReduce based ANOVA and K-Nearest Neighbor”, *Procedia Computer Science*, Vol.54, pp.301–310.

[112] Baliarsingh Santos Kumar, Vipsita Swati, and Dash Bodhisattva (2020), "A new optimal gene selection approach for cancer classification using enhanced Jaya-based forest optimization algorithm", *Neural Computing and Applications*, Vol.32, Issue 12, pp.8599–8616.

[113] Omer Nadir, Elssied Fadl, Ibrahim Othman, and Osman Ahmed Hamza (2014), "A Novel Feature Selection Based on One-Way ANOVA F-Test for E-Mail Spam Classification", *Research Journal of Applied Sciences, Engineering and Technology*, Vol.7, Issue 3, pp.625–638.

RELATED PUBLICATIONS

1. Duran S, **Üreten K,** Maraş Y, Maraş H. H, Gök K, Atalar E, & Çayhan V. Automatic detection of spina bifida occulta with deep learning methods from plain pelvic radiographs. Research on Biomedical Engineering. 2023;39(3), 655-661
2. Atalar E, **Üreten K,** Kanatlı U, Çiçeklidağ M, Kaya I, Vural A, & Maraş Y. (2023). The diagnosis of femoroacetabular impingement can be made on pelvis radiographs using deep learning methods. Joint Diseases and Related Surgery. 2023;34(2), 298
3. Atalar H, **Üreten K**, Tokdemir G, Tolunay T, Çiçeklidağ M, Atik OŞ. The Diagnosis of Developmental Dysplasia of the Hip From Hip Ultrasonography Images With Deep Learning Methods. Journal of Pediatric Orthopaedics. 2023 Feb 25;43(2):e132-7
4. **Üreten K**, Maraş Y, Duran S, Gök K. Deep Learning Methods in the Diagnosis of Sacroiliitis from Plain Pelvic Radiographs. Mod Rheumatol. 2023 Dec;33 (1), 202-206
5. **Üreten K**, Maraş HH. Automated Classification of Rheumatoid Arthritis, Osteoarthritis, and Normal Hand Radiographs with Deep Learning Methods. J Digit Imaging. 2022 Apr;35(2):193-199
6. Maraş Y, Tokdemir G, **Üreten K**, Atalar E, Duran S, Maraş H. Diagnosis of osteoarthritic changes, loss of cervical lordosis, and disc space narrowing on cervical radiographs with deep learning methods. Jt Dis Relat Surg. 2022;33(1):93-101
7. **Üreten K**, Sevinç HF, İğdeli U, Onay A, Maraş Y. Use of deep learning methods for hand fracture detection from plain hand radiographs. Ulus Travma Acil Cerrahi Derg. 2022 Jan;28(2):196-201
8. **Üreten K**, Arslan T, Gültekin KE, Demir AND, Özer HF, Bilgili Y. Detection of hip osteoarthritis by using plain pelvic radiographs with deep learning methods. Skeletal Radiol. 2020 Sep;49(9):1369-1374.
9. **Üreten K**, H Erbay H, HH Maraş HH. Detection of hand osteoarthritis from hand radiographs using convolutional neural networks with transfer learning. Turkish Journal of Electrical Engineering & Computer Sciences . 2020, Vol. 28 Issue 5, p2968-2978. 11p.
10. **Üreten K**, Erbay H, Maraş HH. Detection of rheumatoid arthritis from hand radiographs using a convolutional neural network. Clin Rheumatol. 2020 Apr;39(4):969-974. doi: 10.1007/s10067-019-04487-4. Epub 2019 Mar 8. PMID: 30850962.