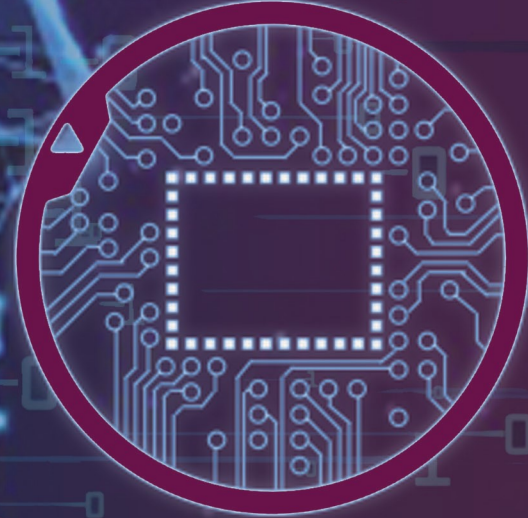




VOLUME - 1  
ISSUE - 1  
2021

# JOURNAL OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE



**İZMİR KÂTİP ÇELEBİ UNIVERSITY**

Artificial Intelligence and Data Science  
Research and Application Center



# JAIDA

[HTTPS://JAIDA.IKCU.EDU.TR/](https://jaida.ikcu.edu.tr/)



**Privilege Owner**

Prof. Dr. Saffet Köse, Rector (İzmir Katip Çelebi University)

**Editor-in-Chief**

Prof. Dr. Ayşegül Alaybeyoğlu (İzmir Katip Çelebi University)

**Associate Editors**

Assist. Prof. Dr. Osman Gökçalp (İzmir Katip Çelebi University)

Assist. Prof. Dr. Serpil Yılmaz (İzmir Katip Çelebi University)

**Managing Editor**

Assist. Prof. Dr. Serpil Yılmaz (İzmir Katip Çelebi University)

**Editorial Board**

Assoc. Prof. Dr. Merih Palandöken (İzmir Katip Çelebi University)

Assoc. Prof. Dr. Aytuğ Onan (İzmir Katip Çelebi University)

Assoc. Prof. Dr. Mustafa Agah Tekindal (İzmir Katip Çelebi University)

Assist. Prof. Dr. Osman Gökçalp (İzmir Katip Çelebi University)

Assist. Prof. Dr. Serpil Yılmaz (İzmir Katip Çelebi University)

Assist. Prof. Dr. Esra Aycan Beyazıt (İzmir Katip Çelebi University)

Assist. Prof. Dr. Onan Güren (İzmir Katip Çelebi University)

Assist. Prof. Dr. Ümit Aydoğan (İzmir Katip Çelebi University)

**International Advisory Board**

Prof. Dr. Adnan Kaya (İzmir Katip Çelebi University)

Prof. Dr. Abd Samad Hasan Basari (Universiti Tun Hussein Onn Malaysia)

Prof. Dr. Filiz Güneş (Yıldız Teknik University)

Prof. Dr. Nejat Yumuşak (Sakarya University)

Prof. Dr. Chirag Paunwala (Sarvajanik College of Engineering and Technology)

Prof. Dr. Narendra C. Chauhan (A D Patel Institute of Technology)

Prof. Dr. Saurabh Shah (GSFC University)

Assoc. Prof. Dr. Amit Thakkar (Charusat University)

Assoc. Prof. Dr. Cheng Jin (Beijing Institute of Technology)

Assoc. Prof. Dr. Mustafa Emiroğlu (Tepecik Training and Research Hospital)

Assoc. Prof. Dr. Ali Turgut (Tepecik Training and Research Hospital)

Assoc. Prof. Dr. Mohd Sanusi Azmi (Universiti Teknikal Malaysia Melaka)

Assoc. Prof. Dr. Peyman Mahouti (İstanbul University- Cerrahpaşa)

Assoc. Prof. Dr. Mehmet Ali Belen (İskenderun Technical University)

Assoc. Prof. Dr. Ferhan Elmalı (İzmir Katip Çelebi University)

Assoc. Prof. Dr. Sharnil Pandya (Symbiosis International University)

Dr. Zihao Chen (Harbin Institute of Technology)

Dr. Kadriye Filiz Balbal (Ministry of Education)

**Aim & Scope**

The Journal of Artificial Intelligence and Data Science (JAIDA) is an international, scientific, peer-reviewed, and open-access e-journal. It is published twice a year and accepts only manuscripts written in English. The aim of JAIDA is to bring together interdisciplinary research in the fields of artificial intelligence and data science. Both fundamental and applied research are welcome. Besides regular papers, this journal also accepts research field review articles.

**Contact**

Web site: <https://jaida.ikcu.edu.tr/>

E-mail: [ikcujaida@gmail.com](mailto:ikcujaida@gmail.com)

Phone: +90 (232) 329 35 35/3731/ 3808/3819

Fax: +90 (232) 325 33 60

Mailing address: İzmir Katip Çelebi Üniversitesi, Yapay Zeka ve Veri Bilimi Uygulama ve Araştırma Merkezi, Balatçık Kampüsü, Çiğli Ana Yerleşkesi, 35620, İzmir

Phone: +90 (232) 329 35 35 / 3731 / 3808 / 3819

Fax: +90 (232) 325 33 60



**TÜRKİYE CUMHURİYETİMİZİN KURUCUSU**  
**BİRİNCİ CUMHURBAŞKANIMIZ**  
**MUSTAFA KEMAL ATATÜRK**



**“Hayatta en hakiki mürşit ilimdir.”**

**“Our true mentor in life is science.”**

**ONİKİNCİ CUMHURBAŞKANIMIZ  
RECEP TAYYİP ERDOĞAN**



**“Yapay zeka çağına geçiş noktasında Türkiye lider  
ülkelerden biri olacak.”**

**“Turkey will be one of the leading countries in the  
transition to the age of artificial intelligence.”**

**SANAYİ VE TEKNOLOJİ BAKANIMIZ**  
**MUSTAFA VARANK**



**“Yapay zeka yarışında biz de varız.”**  
**“We are in the artificial intelligence race.”**

**İZMİR KÂTİP ÇELEBİ ÜNİVERSİTESİ  
REKTÖRÜMÜZ**

**PROF.DR. SAFFET KÖSE**

**(Dergi Sahibi)**

**(Privilege Owner)**



**“Günümüz dünyasına rengini veren dijital teknolojilerin odağındaki ana unsur yapay zekadır.”**

**“Artificial intelligence is the main element in the focus of digital technologies that color today's world.”**

## ÖNSÖZ

Yapay Zeka ve Veri Bilimi alanındaki teknolojik ve bilimsel gelişmeler; Yapay Zekanın endüstri, sağlık, otomotiv, ekonomi, eğitim gibi bir çok farklı alanda uygulanmasına imkan sağlamıştır. Ülkemiz Ulusal Yapay Zeka Stratejisinde; yeni bir çağın eşiğine geldiği, yapay zekayla üretim süreçleri, meslekler, gündelik yaşam ve kurumsal yapıların yeni bir dönüşüm sürecine girdiği vurgulanarak, Yapay Zekanın öneminden bahsedilmiştir.

Sayın Cumhurbaşkanımızın da belirttiği gibi ülkemiz adına insan odaklı yeni bir atılım yapmanın zamanının geldiğine inanıyoruz. Yapay zeka çağına geçiş noktasında Türkiye'nin lider ülkelerden biri olması motivasyonu ile üniversitemizde yapay zeka teknolojilerinin kullanıldığı projeler gerçekleştirmekte, kongreler ve bilimsel etkinlikler düzenlemekteyiz.

Günümüz dünyasına rengini veren dijital teknolojilerin odağındaki ana unsurun yapay zeka teknolojilerinin olduğu düşüncesi ile yola çıkarak hazırlamış olduğumuz Yapay Zekâ ve Veri Bilimi Dergisinin, Ülkemiz Ulusal Yapay Zeka Stratejisinde belirtilen “Dijital Türkiye” vizyonu ve “Milli Teknoloji Hamlesi” kalkınma hedefleri doğrultusunda katkı sağlayacağı inancındayız.

Dergimizin hazırlanmasında emeği geçen üniversitemiz Yapay Zekâ ve Veri Bilimi Uygulama ve Araştırma Merkez Müdürü, Baş Editör Prof. Dr. Ayşegül ALAYBEYOĞLU'na, Editör ve Danışma kurulu üyelerine, akademik çalışmalarını ile sağladıkları destek için tüm yazarlara, hakem olarak görev alan değerli bilim insanlarına teşekkür eder, dergimizin ilk sayısının ülkemize hayırlı olmasını dilerim.

**Prof. Dr. Saffet KÖSE, Rektör**

**Dergi Sahibi**

## **PREFACE**

**Technological and scientific developments in Artificial Intelligence and Data Science enabled the application of Artificial Intelligence in many different fields such as industry, health, automotive, economy and education. In our country's National Artificial Intelligence Strategy; the importance of Artificial Intelligence was mentioned by emphasizing the transformation process of production processes, occupations, daily life and corporate structures with artificial intelligence.**

**As stated by our President, we believe that the time has come to make a new human-oriented breakthrough on behalf of our country. With the motivation of Turkey being one of the leading countries at the point of transition to the age of artificial intelligence, we realize projects in which artificial intelligence technologies are used, and organize congresses and scientific events at our university.**

**We have prepared the Journal of Artificial Intelligence and Data Science with the idea that the main element in the focus of digital technologies that color today's world is artificial intelligence technologies, and we believe that our journal will contribute to the development goals of the "Digital Turkey" vision and "National Technology Move" stated in the National Artificial Intelligence Strategy of our country.**

**I would like to thank Prof. Dr. Ayşegül ALAYBEYOĞLU, the Director of Artificial Intelligence and Data Science Application and Research Center of our university. I would also like to thank to Editor and Advisory Board members, to all authors for their supports with their academic studies and to reviewers for their contributions to the preparation of our journal. I wish the first issue of our journal to be beneficial for our country.**

**Prof. Dr. Saffet KÖSE, Rector**

**Privilege Owner**



## BAŞ EDITÖR'DEN



Değerli Araştırmacılar ve Dergi Okuyucuları;

İzmir Kâtip Çelebi Üniversitesi Yapay Zekâ ve Veri Bilimi Uygulama ve Araştırma Merkezi olarak Rektörümüz Prof. Dr. Saffet Köse sahipliğinde Yapay Zekâ ve Veri Bilimi Dergisinin ilk sayısını sizlerle buluşturmanın gururunu yaşamaktayız.

Son yıllarda yapay zekâ ve veri bilimi alanındaki gelişmeler; tüm ülkeleri yapay zekanın potansiyel kazanımlarına yönelik ciddi adımlar atmaya yönlendirmiştir. Ülkemizde de bakanlıklar düzeyinde, kamu kurum ve kuruluşlarında yapay zeka ile ilgili kurumsal yapılanmaların oluşturulduğu görülmektedir. Biz de İzmir Kâtip Çelebi Üniversitesi Yapay Zekâ ve Veri Bilimi Uygulama ve Araştırma Merkezi olarak dergi, kongre, eğitim, bilimsel etkinlikler ve proje faaliyetleri gerçekleştirerek bu gelişim ve değişim sürecine katkı sağlamayı hedeflemekteyiz. Bu amaçla; Sanayi ve Teknoloji Bakanlığı ve Cumhurbaşkanlığı Dijital Dönüşüm Ofisi Başkanlığı ile iletişimlerimiz devam etmektedir.

Farklı üniversitelerden, bilimsel disiplinlerden ve alanlardan değerli araştırmacıların hazırlamış oldukları 15 adet İngilizce araştırma makalesi bu sayı kapsamında sunulmaktadır. JAIDA çok titiz bir hakem sürecine sahip çevrimiçi ve açık erişimli bir dergidir.

Yapay Zeka ve Veri Bilimi alanlarında geliştirilen yeni yöntemleri ve bu teknolojilerin bir çok farklı alanda uygulama örneklerini içeren uluslararası nitelikteki bilimsel araştırma makalelerini sunan dergimizin önemli bir boşluğu dolduracağı ve bilim dünyasına katkılarda bulunacağı inancındayız. Siz değerli araştırmacılarımızın destekleri ile kaliteyi daha da arttırarak en kısa sürede ulusal ve uluslararası indekslerde taranan bir dergi olmayı hedeflemekteyiz.

Dergimizin yayın hayatına başlamasında büyük desteklerini gördüğümüz başta Rektörümüz Sayın Prof. Dr. Saffet KÖSE olmak üzere; dergimize olan destekleri için tüm yazarlara, dergimizin yayına hazırlanmasında heyecanla çalışan ve çok büyük emek harcayan Baş Editör Yardımcıları Dr. Serpil YILMAZ'a, Dr. Osman GÖKALP'e, Editör ve Danışma kurulu üyelerimize, hakem olarak görev alan tüm değerli bilim insanlarına en derin şükranlarımı sunarım.

Saygılarımla,

Prof. Dr. Ayşegül ALAYBEYOĞLU

Baş Editör

## Letter from the Editor-in-Chief



Dear Researchers and Readers of the Journal,

As İzmir Katip Çelebi University Artificial Intelligence and Data Science Application and Research Center, we are proud to present you the first issue of the Journal of Artificial Intelligence and Data Science (JAIDA), hosted by our Rector Prof. Dr. Saffet Köse.

In recent years, developments in artificial intelligence and data science have led all countries to take serious steps towards the potential gains of artificial intelligence. In our country, institutional structures related to the artificial intelligence have been established at the level of ministries, public institutions and organizations. As İzmir Katip Çelebi University Artificial Intelligence and Data Science Application and Research Center, we aim to contribute to this development and change by carrying out journals, congresses, education, scientific events and project activities. With this motivation, our communications with the Ministry of Industry and Technology and the Head of the Digital Transformation Office of the Presidency of Republic Turkey will continue.

15 research articles prepared by valuable researchers from the different universities, scientific disciplines and fields are presented within the scope of this issue. JAIDA is an online and open access journal which has a rigorous peer review process.

We believe that our journal, which presents international scientific research articles containing new methods developed in the fields of Artificial Intelligence and Data Science and the application examples of these technologies in many different fields, will fill an important gap and contribute to the scientific world. We aim to become a journal that is scanned in national and international indexes as soon as possible by further raising the quality with the support of you, our dear researchers.

I would like to express my deepest gratitude to our Rector, Prof. Dr. Saffet KÖSE, who supported the publication of our journal; to all the authors for their supports to our journal; to our Associate Editors Dr. Serpil YILMAZ and Dr. Osman GÖKALP, who worked enthusiastically and put great efforts into the preparation of our journal; to our Editorial and Advisory Board members, and all esteemed scientists who served as reviewers.

Best Regards,

Prof. Dr. Ayşegül ALAYBEYOĞLU

Editor-in-Chief

# CONTENTS

Turkish Sentiment Analysis Using Machine Learning Methods: Application on Online Food Order Site Reviews .....	1
<b>Özlem AKTAŞ, Berkay COŞKUNER, İlker SONER</b>	
Identification of Breast Cancer Metastasis Using Boosting Algorithms on Cytopathologic Data .....	11
<b>Safak KAYIKCI</b>	
Fabric Defect Classification Using Combination of Deep Learning and Machine Learning .....	22
<b>Fatma Günseli YAŞAR ÇIKLAÇANDIR, Semih UTKU, Hakan ÖZDEMİR</b>	
A Face Authentication System Using Landmark Detection .....	28
<b>Velican ERCAN, M. Erdal ÖZBEK</b>	
A Novel Approach for Detecting Defective Expressions in Turkish .....	35
<b>Atilla SUNCAK, Özlem AKTAŞ</b>	
Development of Android-based Internet of Things Application for Data Tracking in Smart Marble Factories .....	41
<b>Cağlar GURKAN</b>	
A Flower Status Tracker and Self Irrigation System (FloTIS) .....	45
<b>Rumeysa KESKİN, Furkan GÜNEY, M. Erdal ÖZBEK</b>	
Turkish Character Usage in Text Classification .....	53
<b>Ali Aycan KOLUKISA</b>	
Social Distancing Automation Software Based on Cloud .....	59
<b>Ali Şahin DEMİR, Sevcan EMEK</b>	
Estimation of Scattering Parameters of U-Slotted Rectangular RFID Patch Antenna with Machine Learning Models .....	63
<b>Ismail AKDAG</b>	
Optimization of Process Parameters for Green Composites in Abrasive Water Jet Machining Process Using Neuro-Regression Analysis .....	71
<b>Serap TANRIVERDİ, Levent AYDIN</b>	
Modeling and Design Optimization to Determine the Mechanical Properties of a Recent Composite ..	80
<b>Naciye Burcu KARTAL</b>	
Modeling and optimum design for wire electrical discharge machining of $\gamma$ titanium aluminide alloy ..	89
<b>Ömer Faruk BÜYÜKYAVUZ</b>	
A Novel, Nelder-Mead Optimization Approach, based on Neuro-regression modeling for the Energy Efficiency Parameters of End Milling Process .....	96
<b>Şeyda İŞBİLİR</b>	
Electron Beam Welding (EBW) of Aerospace Alloy (Inconel 825): Optimization and Modeling of Weld Bead Area .....	106
<b>Gamze ÖZAKINCI, Levent AYDIN</b>	

# Turkish Sentiment Analysis Using Machine Learning Methods: Application on Online Food Order Site Reviews

Özlem AKTAŞ<sup>1</sup>, Berkay COŞKUNER<sup>1,\*</sup>, İlker SONER<sup>1</sup>

<sup>1</sup> Dokuz Eylül University, Faculty of Engineering, Department of Computer Engineering, Turkey

## Abstract

Satisfaction measurement, which emerges in every sector today, is a very important factor for many companies. In this study, it is aimed to reach the highest accuracy rate with various machine learning algorithms by using the data on Yemek Sepeti and variations of this data. The accuracy values of each algorithm were calculated together with the various natural language processing methods used. While calculating these accuracy values, the parameters of the algorithms used were tried to be optimized. The models trained in this study on labeled data can be used on unlabeled data and can give companies an idea in measuring customer satisfaction. It was observed that 3 different natural language processing methods applied resulted in approximately 5% accuracy increase in most of the developed models.

**Keywords:** *Machine learning; natural language processing; sentiment analysis; yemek sepeti.*

## 1. Introduction

Sentiment analysis is the process of analyzing and labeling the emotion created by the being that can create emotion. Today, when sentiment analysis is mentioned, studies on human and Twitter data usually come to mind. In the coming years, we may need to do sentiment analysis even from the human-like interfaces of the machines. However, the issue we are focusing on right now is a part that is closely related to both the academic world and the business world. Customer satisfaction is always at the forefront. Although this satisfaction is tried to be measured by various methods (questionnaire etc.), it is insufficient after a while and needs to be automated. This study enables us to analyze real emotion among the complex writings of people in an automated way.

To summarize in one sentence, sentiment analysis is the process of classifying the emotion that a person reveals together through the communication that the person uses, as positive or negative.

Every day, more data is created than human processing. If all people in the world quit all their work and try to read the data produced in just one day, it will not be successful. Therefore, there is a need for a more efficient way of processing data. There are so much data that cannot be detected with the naked eye. When you look at a table or reviews for a restaurant or comments for a product, you have a general idea, but you will never see the final result because you cannot read all of the comments. At this point, models are developed for processing the data. The purpose of this study is to help adapt the sentiment analysis method to real life and to test its accuracy.

Sentiment analysis studies can be performed on various predetermined emotions. It can also be diversified with other emotions such as fear, anger, sadness. In this study, it was evaluated only as positive and negative. The developed model shows whether a given sentence is positive or negative. While making this classification, user comments on Yemeksepeti.com site were used as data. The data set was not used ready, it was created.

The size of the created data set is approximately 676 thousand. Since the data set is completely homogeneous, no model tends to be positive or negative, there are 338 thousand positive and 338 thousand negative comments. The dataset was created using the Selenium module in Python.

There are 3 different techniques used in the study.

- Lemmatization
- Word Correction
- Keyboard (Our Method)

The Keyboard method is either used inside the Word Correction method or not. It cannot be used separately. There are 6 different datasets created using variations of these methods. These datasets were created to determine the effectiveness of the methods. All of the data sets are differentiated as 90% train 10% test randomly and have not been subjected to any processing other than the above-mentioned processes. The process of converting words to lowercase is common to all, as it is an operation done at the very beginning. The process of removing nonsense words and stop words is a common operation, as is converting words to lowercase. 6 different data sets used in the study are:

\*Corresponding author e-mail address: [berkay.coskuner@ceng.deu.edu.tr](mailto:berkay.coskuner@ceng.deu.edu.tr)



- Word Correction + Lemmatization (Default Dataset)
- Word Correction
- Lemmatization
- Word Correction without Keyboard
- Word Correction without Keyboard + Lemmatization
- No Operation

The difference in accuracy between the default dataset and the no-operation dataset will be based to see the final results of the study.

The major contributions of the study to the literature are:

- A dataset of homogeneous comments with a label of a size not previously available in Turkish.
- The potential contribution of the keyboard method to sentiment analysis and similar studies.
- Effective use of Lemmatization and Word Correction methods and observing their effects on similar studies.

To summarize the study, it was aimed to classify the comments as positive or negative by using the food basket comments. Datasets created with variations of 3 different methods and different machine learning algorithms were compared.

## 2. Previous Works

Similar studies conducted before us did not have such a large data set and keyboard method. Some similar studies are given in this chapter.

In the study of Erşahin et al. [1], a hybrid dictionary-based algorithm is used together with machine learning algorithms. The algorithms have been tested separately in 3 different data sets and the differences between them have been observed. The size of these data sets is 2000, 11000, and 50000. SVM, J48 and NB algorithms are used. They obtained the best result with an accuracy of 91% using SVM + SentiTurkNet library. Compared to single methods, hybrid methods show an average of 7% higher accuracy values.

In another study of Emekli et al. [2], it is aimed to classify Turkish tweets sent to the leading GSM operators of Turkey as positive and negative. In this study, deep learning methods are suggested for the classification of tweets. The posted tweets were first prepared with Natural Language Processing methods, and then they were classified and compared with deep learning methods such as Convolutional Neural Networks, Recurrent Neural Networks, and Long Short-Term Memory models. According to the results of the deep learning models, Convolutional Neural Networks could not increase the performance up to 9 epoch values on small data sets. Long Short-Term Memory, on the other hand, achieved a more successful result in the large datasets compared to other methods with a 98.64% accuracy rate. Recurrent Neural Networks achieved higher performance in the smaller datasets with a 98.8% success rate. It has been observed that the smaller the data set, the faster the performance in the Recurrent Neural Networks is learning.

In the study of Yilmaz et al. [3], offensive language was detected on the OffensEval dataset. This dataset consists of 31756 tweets. While 6131 of them have offensive content, 25625 of them consist of non-aggressive content. Accordingly, an untagged collection of approximately 1 million tweets was prepared. Afterward, the effect of word representations obtained from the labeled data in the OffensEval dataset, and the word representations obtained from the large untagged corpus on the classification performance were compared. Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (BiLSTM) networks are used as machine learning models in the study. The classification performances of these deep neural networks are evaluated as accuracy, recall and precision, and F-score. While examining the effects of the extended corpus on deep learning models, it was determined that the success of the model was increased by using the expanded corpus. With this method, performance has been improved by approximately 40% to 47% compared to the F-score value, which is a measure of test accuracy. In addition, LSTM performed better than the deep learning models used. The performance values of the LSTM model were approximately 86% accuracy, 55% sensitivity, 68% precision, and 61% F-score, respectively. Performance values of the BiLSTM model were approximately 86% accuracy, 55% sensitivity, 66% precision, and 60% F-score, respectively.

A study of Baştürk [4] has been published on Kaggle with egebasturk1 account. A single-layer LSTM model was created with a total of 8570 Yemeksepeti comments. A wide variety of data preprocessing has been

performed on the data by using Zemberek library, which is widely used in the NLP field. As a word representation, vectors prepared previously with word2vec method were used. 84% accuracy was taken as a result of the study. The project owner mentioned the insufficiency of word2vec vectors as self-criticism. In our eyes, a good accuracy value was obtained in this study because the structure of the neural network is weak, word vectors are weak, but data preprocessing is strong.

In the study of Yelmen [5], sentiment analysis was performed using various methods. These methods are Neural Network, Support Vector Machine, and Centroid Based Algorithm. In the study, 2 different methods were used, namely genetic algorithm, to find attributes. Accuracy, F-Measure, Recall, Precision values were used as performance criteria. In the data pre-processing phase, the root-finding process was performed using the Zemberek library.

The study of Aytuğ [6] on Turkish Tweets is aimed to have information about global events and to help crisis situations. The dataset contained 5300 positive and 5300 negative tweets with a total of 10600. Some natural language processing methods have been used together with various machine learning algorithms. Instead of the commonly used word2vec algorithm, the N-Gram algorithm is used. Accuracy, F-Measure, and AUC values were used among performance metrics. The highest success rate was achieved with the Bayes algorithm in the data set using the features obtained by using 1-Gram and 2-Gram common.

Another Turkish Sentiment Analysis on Turkish tweets but this study was about a specific topic. Albayrak et al. [7] did this study to find out what people's comments are about 'paid military service' which was one of the most popular topics of the period. 12739 tweets about the topic were collected to apply sentiment analysis. The messages obtained from Twitter were compared as a word with the words in the SentiTurk data set. The polarity score obtained for each tweet was evaluated according to their positivity, negativity, and neutral status. The tweets got a score according to the number of positive and negative words in their messages. In other words, no machine learning method was used in this study. Sentiment analysis was attempted by comparing the existing library and tweets.

In this research of Alpkocak et al. [8], The outputs of different machine learning methods were compared. In this study, TREMO dataset was used. Sentiment analysis is considered as a text classification problem and a different approach than normal approaches has been tried to be used. A study was conducted on 6 different labeled emotions. These feelings are: happiness, fear, anger, sadness, disgust, and surprise, were used as emotion categories. In the study, root finding was done with F5, and TF-IDF was applied. As the performance metric, accuracy is used. The main purpose of the study was to develop a better model than the SVM method, one of the methods used in the past and considered a successful method. After experimenting with various layers and numbers of neurons, an artificial neural network was developed with accuracy (86%) 0.0045% greater than SVM.

The study of İlhan et al. [9] is another example of Sentiment Analysis on Turkish tweets and it focuses mainly on sentiment analysis of Twitter which is one of the popular social media sites used by numerous users where users publish a status update in the form of tweets. The dataset that Nagehan İlhan and Duygu Sağaltıcı used has 1,578,627 classified tweets. And the model was built for classifying tested tweets into positive and negative sentiment. To perform this analysis, an intelligent model has been created by using machine learning methods such as Naïve Bayes and Support Vector Machine and the compared results has been given. In conclusion, Unigram and Bigram Support Vector Machine give the best results with 64% accuracy. In the total dataset, Naive Bayes gave a 42% success rate, while VADER had a 27% success rate. When looking at the results, it is seen that the results are improved when Unigram and Bigram are used together with Support Vector Machine.

In the study Sarıman et al. [10] of collected the tweets that have been posted about coronavirus in Turkey since 11 March 2020, In this study using logistic regression, approximately 2 million tweets were used. The samples were collected on 5 different topics, but they were divided into 2 parts as positive and negative while classifying.

Since success rates were not at the desired level according to randomly generated training sets at first, training sets were determined according to positive and negative word groups. As a result of this change, AUC values increased from an average of 0.80 to an average of 0.95. And when the general results are examined, it is observed that people's comment on government's mask application is generally evaluated as positive, but other applications are generally evaluated as negative.

Offensive language was detected on the OffensEval dataset in the study of Yılmaz et al. [11]. This dataset consists of 31756 tweets. While 6131 of them have offensive content, 25625 of them consist of non-aggressive content. Accordingly, an untagged collection of approximately 1 million tweets was prepared. Afterward, the effect of word representations obtained from the labeled data in the OffensEval dataset, and the word

representations obtained from the large untagged corpus on the classification performance were compared. Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (BiLSTM) networks are used as machine learning models in the study. The classification performances of these deep neural networks are evaluated as accuracy, recall and precision, and F-score. While examining the effects of the extended corpus on deep learning models, it was determined that the success of the model was increased by using the expanded corpus. With this method, performance has been improved by approximately 40% to 47% compared to the F-score value, which is a measure of test accuracy. In addition, LSTM performed better than the deep learning models used. The performance values of the LSTM model were approximately 86% accuracy, 55% sensitivity, 68% precision, and 61% F-score, respectively. Performance values of the BiLSTM model were approximately 86% accuracy, 55% sensitivity, 66% precision, and 60% F-score, respectively.

In the study of Gezici et al. [12] movie reviews are used. In addition to supervised methods, a lexicon-based method was also used. In addition, the Turkish polarity lexicon called SentiTurk was also used.

The comments are taken from the movie site called the big screen. The best 2 results are from Naive Bayes and Support Vector Machine algorithms. They have found that working with large lexicons always gives better results. The dataset is homogeneous and includes 5300 positive and negative comments. When all features are used, approximately 75% success was achieved in both algorithms.

### 3. Methodologies

The first step is to prepare the raw data set. This unprocessed data set was obtained from Yemek Sepeti via Selenium module in Python. Selenium module is a module for getting information from websites using Python. By taking HTML/CSS codes and making them more suitable for the eye, it provides the desired data from websites.

After the raw form of the dataset is prepared, its variations should be created according to the methods. Word2vec [13] (Mikolov et al., 2013) models have been developed to represent words specifically for all datasets. General flow of the project is given in Figure 1.

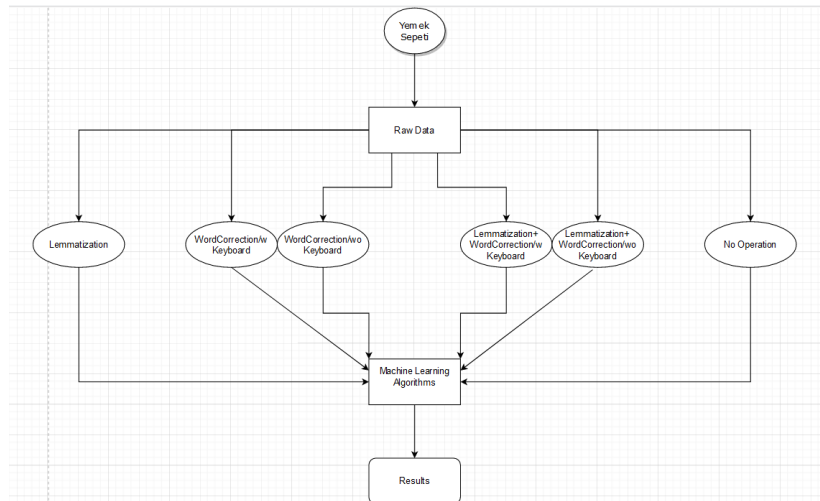


Figure 1. Flow of Study.

Lemmatization is the process of converting a word to its base form. It helps to provide an increase in accuracy by putting words that have the same meaning but are spelled differently in the data set into the same form. In Figure 2 and Figure 3 it can be observed that the lemmatization library works quite well and helps to obtain better and more accurate results with the studied datasets. The library used for lemmatization is called zemberek-python. The upperparts are the raw data, the lower parts are lemmatized using the zemberek-python library.

```

['Yazılan', 'notları', 'dikkate', 'almadığınız', 'için', 'size', 'şöyle', 'kötü', 'bir', 'puan', 'verelim']
['yazmak', 'not', 'dikkat', 'almamak', 'için', 'siz', 'şöyle', 'kötü', 'bir', 'puan', 'vermek']
  
```

Figure 2. Example Lemmatization 1.

```
['Tam', 'bir', 'buçuk', 'saatte', 'geldi', 'Defalarca', 'aramamıza', 'rağmen', 'telefonu', 'asla', 'açmadılar']
['tam', 'bir', 'buçuk', 'saat', 'gelmek', 'defalarca', 'aramak', 'rağmen', 'telefonu', 'asla', 'açmamak']
```

**Figure 3.** Example Lemmatization 2.

The library used for the word correction method is called trnlp. This library corrects Turkish misspelled and incorrect words. It also converts words written in English characters into Turkish characters. These two processes are important for the accuracy increase and for training models efficiently.

```
['Surekli', 'icerik', 'yanlis', 'geliyor']
['süreklî', 'içerik', 'yanlış', 'geliyor']
```

**Figure 4.** Example of Word Correction.

The working principle of the created keyboard method is based on a simple principle: the human factor. Most spelling mistakes made on phones or computers actually occur around the letter you want to type. The Keyboard method tries to calculate which of these probabilities is greater and returns you the best possible result. This method is used together with the Word Correction method. The Word Correction method suggests 10 words for a word correction. The Keyboard algorithm, on the other hand, calculates between the first 2 suggested words and the originally written word and tries to find out which of the 2 suggestions is more likely. Keyboard matrix created as seen in Figure 5.

```
a z s w q
b v g h n
c x d f v
ç ş l k m
d e r f c x s
e w s d r
f r t g v c d
g t y h b v f
ğ p ş i ü
h y u j n b g
ı u j k o
i ü ğ ş
j u ı k m n h
k ı o l ö m j
l o p ş ç ö k
```

**Figure 5.** Part of Keyboard Matrix.

As it can be seen in Fig.5, there is an alphabetical order in the first column. In the other columns, there are the letters around the current letter in the Turkish Q keyboard.

### 3.1. Parameter Tuning

The neural network model was developed using TensorFlow, and all other models were developed using the sklearn module. Each algorithm has its own parameters and the best parameters need to be found to make the algorithm work well. Some algorithms have a low number of parameters or the ratio of these parameters to the accuracy value, but there are algorithms that have the opposite. The neural network parameter optimization, which is the first of the methods used, was systematically performed manually on the default data set. The values seen in Table 1 include the results of the experiments carried out with a high number of neurons and hidden layers, without taking into account the runtimes.

**Table 1.** Neural Network Optimization 1.

Batch Size	Epoch	Structure	Neuron Type	Accuracy
1024	5	64-64-16-8-4-1	GRU	0,8389



2048	5	64-64-16-8-4-1	GRU	0,8257
512	5	64-64-16-8-4-1	GRU	0,8515
256	5	64-64-16-8-4-1	GRU	0,8606
128	5	64-64-16-8-4-1	GRU	0,8618
2048	5	128-64-16-8-4-1	GRU	0,8384
2048	5	512-64-16-8-4-1	GRU	0,8354
2048	1	64-64-16-8-4-1	GRU	0,8029
2048	3	64-64-16-8-4-1	GRU	0,8207
2048	7	64-64-16-8-4-1	GRU	0,8212
2048	5	512-128-16-8-4-1	GRU	0,8443
2048	10	64-64-16-8-4-1	GRU	0,8400
2048	12	64-64-16-8-4-1	GRU	0,8410
2048	5	512-256-128-16-8-4-1	GRU	0,8102
2048	17	64-64-16-8-4-1	GRU	0,8518
2048	20	64-64-16-8-4-1	GRU	0,8490
2048	5	64-64-64-64-64-64-1	GRU	0,8449
2048	5	64-64-16-8-4-1	CuDNNGRU	0,8317
2048	25	64-64-16-8-4-1	CuDNNGRU	0,8552
2048	5	64-64-16-8-4-1	CuDNNGRU	0,8358
2048	5	64-128-128-128-128-128-1	GRU	0,8555
2048	5	64-256-256-256-256-256-1	GRU	0,8600
2048	5	64-64-16-8-4-1	CuDNNLSTM	0,8143
2048	5	64-256-256-256-256-256-256-1	CuDNNGRU	0,8544
2048	5	64-64-16-8-4-1	Bidirectional CuDNNGRU	0,8546

As given in Table 2, experiments were performed with smaller numbers of neurons and hidden layers, and operating times were taken into account.

Here are the best values obtained after manual work.

- Batch Size = 32
- Layer Type = Bidirectional CuDNNGRU
- Epoch = 10
- Structure = 8-8-8-1

**Table 2.** *Neural Network Optimization 2.*

Batch Size	Epoch	Structure	Neuron Type	Runtime(s)	Accuracy
32	10	8-8-8-8-1	GRU	2500	0,8569
<b>32</b>	<b>10</b>	<b>8-8-8-8-1</b>	<b>Bidirectional CuDNNGRU</b>	<b>800</b>	<b>0,8612</b>
2048	10	8-8-8-8-1	Bidirectional CuDNNGRU	16	0,8492
2048	20	8-8-8-8-1	Bidirectional CuDNNGRU	16	0,8552
2048	40	8-8-8-8-1	Bidirectional CuDNNGRU	16	0,8578
8	5	8-8-8-8-1	Bidirectional CuDNNGRU	1250	0,8595
32	10	8-8-8-1	Bidirectional CuDNNGRU	215	0,8637
32	10	8-8-1	Bidirectional CuDNNGRU	130	0,8517
32	10	64-8-8-1	Bidirectional CuDNNGRU	215	0,8601
32	10	64-8-1	Bidirectional CuDNNGRU	135	0,8509
512	10	8-8-8-8-1	Bidirectional CuDNNGRU	27	0,8585
512	10	64-8-8-1	Bidirectional CuDNNGRU	13	0,8524
8	10	8-8-8-1	Bidirectional CuDNNGRU	714	0,8590
32	10	8-8-8-1	Bidirectional CuDNNLSTM	1065	0,8576
32	10	8-8-8-1	CuDNNGRU	440	0,8558
32	10	8-8-8-1	CuDNNLSTM	444	0,8499

As 2 experiments show, in Table 1 and Table 2, small-batch and optimum epoch number are effective in giving maximum performance. In addition, it is obvious that there is no proportionality between the complexity of the model and the accuracy value.

A systematic study could not be performed in the support vector machine and the k-nearest neighbor method due to the long run times. The best values from several random trials:

#### Support Vector Machine

- Kernel = Polynomial
- C = 0.1
- Gamma = 0.1

#### K-Nearest Neighbor

- K = 7
- Algorithm = Auto
- Weights = Uniform

The GridSearchCV function, which is a function of the sklearn module, was used in Naive Bayes and Linear Regression algorithms. The values are as follows:

#### Naïve Bayes

- var\_smoothing = 0.151

#### Linear Regression

- fit\_intercept = True
- normalize = True

The studies were continued with the same parameters in all data sets.

### 3.2. Evaluation Metrics

Accuracy, f-measure, precision, and recall were used as calculation metrics. In addition to these, the Mean Squared Error (MSE) metric is also used to measure performance between algorithms. However, for the simplicity of the tables, only the accuracy and MSE values are included. Since linear regression outputs are continuous data, methods such as MSE are used to measure the success of the algorithm. Therefore, there are no accuracy values for linear regression in the tables.

- True Positive (TP), prediction is positive while real value is positive.
- True Negative (TN), prediction is negative while real value is negative.
- False Positive (FP), prediction is negative while real value is positive.
- False Negative (FN), prediction is positive while real value is negative.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$Fm = 2 * \frac{(Precision*Recall)}{(Precision+Recall)} \quad (4)$$

MSE(Mean Squared Error) is the error metric that tells how close a regression line is to the predictions. It is both positive and usually greater than 0 because it is derived from the square of the Euclidean distance. It was used to measure the success of the regression model.

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - Y'_i)^2 \quad (5)$$

#### 4. Results

In this study, as seen in Table 9, there is an accuracy value difference of approximately 5% between the unapplied state of all methods and the applied state of all methods. Except for the k-nearest neighbor algorithm, there is a noticeable linear difference in the other algorithms. In the case of lemmatization, word correction, and keyboard methods applied separately, approximately 1% increase in accuracy value was observed for each. The results on the default data set can be seen in Table 3.

**Table 3.** Default Data Set Results.

Default Data Set	Accuracy	MSE	Runtime(s)
Neural Network	%86,37	0,136	2150
Support Vector Machine	%87,24	0,127	21977
K-Nearest Neighbor	%81,88	0,181	907
Naive Bayes	%83,55	0,164	175
Linear Regression	-	0,131	69

In the following tables, the results of the remaining 5 data sets can be observed.

**Table 4.** *Word Correction Data Set Results.*

Word Correction Data Set	Accuracy	MSE	Runtime(s)
Neural Network	%83,9	0,135	2152
K-Nearest Neighbor	%83,2	0,133	903
Naive Bayes	%82,05	0,131	177
Linear Regression	-	0,134	72

**Table 5.** *Lemmatization Data Set Results.*

Lemmatization Data Set	Accuracy	MSE	Runtime(s)
Neural Network	%84,06	0,138	2137
K-Nearest Neighbor	%83,8	0,133	885
Naive Bayes	%82,14	0,142	165
Linear Regression	-	0,132	71

**Table 6.** *Word Correction without Keyboard Data Set Results.*

Word Correction without Keyboard Data Set	Accuracy	MSE	Runtime(s)
Neural Network	%82,66	0,141	2002
K-Nearest Neighbor	%84,1	0,134	1003
Naive Bayes	%81,16	0,142	189
Linear Regression	-	0,140	71

**Table 7.** *Word Correction without Keyboard + Lemmatization Data Set Results.*

Word Correction without Keyboard + Lemmatization Data Set	Accuracy	MSE	Runtime(s)
Neural Network	%84,94	0,140	1987
K-Nearest Neighbor	%84,9	0,130	1132
Naive Bayes	%82,98	0,135	167
Linear Regression	-	0,131	70

**Table 8.** *No Operation Data Set Results.*

No Operation Data Set	Accuracy	MSE	Runtime(s)
Neural Network	%81,15	0,142	2003
K-Nearest Neighbor	%83,2	0,131	912
Naive Bayes	%80,34	0,135	154
Linear Regression	-	0,143	55

The table containing all the most recent results of the study is shown below. It has been observed that there is an increase of 1% for each of the methods applied separately, and 5% when applied together.

**Table 9.** *Final Results Accuracy/MSE.*

Dataset/Algorithm	Neural Network	Naïve Bayes	K-Nearest Neighbor	Linear Regression
Default	%86,37/0,136	%83,55/0,164	%81,88/0,181	-/0,131
WordCorrector	%83,9/0,135	%82,05/0,131	%83,2/0,133	-/0,134
Lemmatizing	%84,06/0,138	%82,14/0,142	%83,8/0,133	-/0,132
WordCorrector w/o Keyboard	%82,66/0,141	%81,16/0,142	%84,1/0,134	-/0,140
WordCorrector w/o Keyboard + Lemmatizing	%84,94/0,140	%82,98/0,135	%84,9/0,130	-/0,131
No Operation	%81,15/0,142	%80,34/0,135	%83,2/0,131	-/0,143



## 5. Conclusion

In this study, various machine learning methods were tested and tried to be improved using approximately 676 thousand comments. The adaptation of NLP applications to Turkish and their effects on accuracy values were observed. As seen in previous studies, it is a fact that the Support Vector Machine algorithm performs very well in NLP applications in any way. However, if there is a time constraint, it may be more useful to use other simple but effective algorithms. The main focus of the study was Artificial Neural Networks. The parameters were systematically optimized without using GridSearchCV. The accuracy value obtained was close to the Support Vector Machine algorithm. It is possible to say that the 2 algorithms go head-to-head, but when the subject is considered as a hyperparameter, it seems that artificial neural networks can be developed further. Likewise, there are parameters that can be optimized in the Support Vector Machine algorithm. Much better accuracy can be achieved when a little more time and computational power is added to the study. This study, unlike other studies, showed the effect of the Keyboard method in the field of Turkish NLP. In addition, a large data set and different machine learning algorithms have revealed the effect of different natural language processing methods.

Since such a large Turkish data set was not encountered in literature research, the data set was created from scratch. The generated dataset is published in Kaggle [14] to be used for everyone works and interested in this area. It is hoped that this data set and the results of the studies will contribute to future Turkish studies.

## Declaration of Interest

As authors, we declare that we have no conflict of interest with anyone related to our work.

## References

- [1] B. Erşahin, Ö. Aktaş, D. Kilinc, and M. Erşahin, "A hybrid sentiment analysis method for Turkish", *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 27, no. 3, pp. 1780-1793, 2019.
- [2] B. Emekli and İ.H. Selvi, "GSM Operatörlerine Yönelik Atılan Türkçe Tweetlerin Derin Öğrenme Yöntemleriyle Duygu Analizi", 4. Uluslararası Marmara Fen Bilimleri Kongresi, 2019.
- [3] Ş. Yılmaz, İ. Özer, and H. Gökçen, "Türkçe Metinlerde Derin Öğrenme Yöntemleri Kullanılarak Duygu Analizi", in *International Symposium of Scientific Research and Innovative Studies*, vol. 22, pp. 25, 2021.
- [4] E. Baştürk, "Yemeksepeti Sentiment Analysis". Kaggle [Online]. Available: <https://www.kaggle.com/egebasturk1/yemeksepeti-sentiment-analysis>. [Accessed Oct. 2020].
- [5] İ. Yelmen. "Doğal Dil İşleme Yöntemleriyle Türkçe Sosyal Medya Verileri Üzerinde Duygu Analizi", Doctoral dissertation, İstanbul Aydın Üniversitesi Fen Bilimleri Enstitüsü, 2016.
- [6] O. Aytuğ, "Twitter Mesajları Üzerinde Makine Öğrenmesi Yöntemlerine Dayalı Duygu Analizi". *Yönetim Bilişim Sistemleri Dergisi*, vol. 3, no 2, 1-14, 2017.
- [7] M. Albayrak, K. Topal, and V. Altıntaş, "Sosyal Medya Üzerinde Veri Analizi: Twitter". *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, vol. 22 (Kayfor 15 Özel Sayısı), 2017.
- [8] A. Alpkoçak, M.A. Tocoglu, A. Çelikten, and İ. Aygün, "Türkçe metinlerde duygu analizi için farklı makine öğrenmesi yöntemlerinin karşılaştırılması". *Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi*, vol. 21, no. 63, 719-725, 2019.
- [9] N. İlhan and D. Sağlaticı, "Twitter'da Duygu Analizi". *Harran Üniversitesi Mühendislik Dergisi*, vol. 5, no. 2, 146-156, 2020.
- [10] G. Sarıman and E. Mutaf, "COVID-19 Sürecinde Twitter Mesajlarının Duygu Analizi", *Euroasia Journal of Mathematics, Engineering, Natural & Medical Sciences*, vol 7, no. 10, pp. 137-148, 2020.
- [11] Yılmaz, Ş. Ş., Özer, İ., & Gökçen, H, "Türkçe Metinlerde Derin Öğrenme Yöntemleri Kullanılarak Duygu Analizi". In *International Symposium of Scientific Research and Innovative Studies*, vol. 22, pp. 25, 2021.
- [12] G. Gezici and B. Yanıkoğlu, "Sentiment analysis in Turkish. In *Turkish natural language processing*", Springer, Cham, pp. 255-271, 2018.
- [13] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space", *arXiv preprint arXiv:1301.3781*, 2013.
- [14] B. Coskuner, *Yemek Sepeti Comments*, 2021. [Online] Available: <https://www.kaggle.com/berkaycokuner/yemek-sepeti-comments>.

# Identification of Breast Cancer Metastasis Using Boosting Algorithms on Cytopathologic Data

Safak KAYIKCI \*

Bolu Abant İzzet Baysal University, Faculty of Engineering, Department of Computer Engineering, Turkey

## Abstract

Breast cancer is the second most common cancer among women after lung cancer. Early diagnosis of cancer can positively affect the recovery process from disease. Several machine learning-based approaches have been studied for cancer detection on histopathological images. In this study, identification of cancer type has been made using Gradient Boosting Machine (GBM), eXtreme Gradient Boost (XGBoost), and Light Gradient Boosting Machine (LightGBM) algorithms. The performances of these techniques have been measured on the Breast Cancer Wisconsin (Diagnostic) dataset. According to the results obtained, Gradient Boosting Machine (GBM) got the highest accuracy rate with 97.02% success. Although there is no pathological prior knowledge about the disease, high success has been achieved in diagnosing with the deep learning architectures used.

**Keywords:** *Breast cancer; eXtreme gradient boost; gradient boosting machine; light gradient boosting machine.*

## 1. Introduction

Breast cancer is a widespread kind of cancer that is a dangerous type that affects women the most. The World Health Organization (WHO) reported that approximately two million women die from this disease cancer every year. Approximately 627,000 women died of breast cancer in 2018 [1]. Therefore, early detection studies are critical.

Histopathological images of breast cancer are automatically categorized as benign or malignant cancer with the help of computer-aided diagnosis systems, and early treatment must be initiated as soon as possible. While detection of breast cancer, mammography can be applied by techniques like Magnetic Resonance Imaging (MRI), ultrasound, tomography, and a biopsy of the breast tissue are required for the definitive diagnosis [2][3]. By examining histological samples of cells or tissues taken from the body using the biopsy method under a microscope, pathologists try to make a definite diagnosis about the change in the breast [4]. They try to classify as benign, malignant, or normal according to diagnosis results. These samples are analyzed with different microscopic magnification rates. Since the examination of histopathological images is a troublesome duration, evaluation of these images with computer-aided methods will make a serious contribution to the correct diagnosis. The experience and attention of pathologists during the analysis of these images are essential for the correct diagnosis and diagnosis. Computer-based systems can minimize the wrong diagnoses decisions by being affected by negative factors such as fatigue and distraction that may occur in the daily lives of pathologists. Thus, it can enable experts to focalize on crucial cases to diagnose [5].

Many important studies have been carried out in this field using digital image processing techniques. Automatic classification of benign and malignant breast cancer types has become especially important. Various shallow machine learning algorithms such as Artificial Neural Networks (ANN), Random Forest (RF), Support Vector Machines SVM and Principal Component Analysis (PCA) have been used primarily for the diagnosis of breast cancer.

Spanhol et al. published an open-source BreakHis data set containing 7,909 images obtained from 82 patients [6]. The authors proposed LeNet and AlexNet models for the classification of breast cancer images. In this study, the success of the AlexNet model in classifying histopathological images was reported to be better than the LeNet model. The dimensions of the images submitted to AlexNet are defined as  $32 \times 32$  or  $64 \times 64$  pixels. Simple fusion rules such as maximum, product, and some are used in the proposed models for comparison. The authors achieved the highest classification success of  $85.6 \pm 4.8$  with the AlexNet model they proposed for dual breast cancer image classification. In 2017, Han et al. used the Class Structure-based Deep Convolutional Neural Network (CSDCNN) model [7]. Dimensions of the images given as an introduction to the CSDCNN model have been resized to be  $256 \times 256$ . Filters of  $3 \times 3$ ,  $5 \times 5$ , or  $7 \times 7$  are applied in the convolution layers. Two different ways have been tried in the training process. First, training of model on the BreakHis data set was made from scratch, but when it was seen that the poor result was obtained, the weights of the model trained on ImageNet with the transfer learning method were used as the second way. The proposed model has reached the highest performance of  $96.9 \pm 1.9$  in the binary classification of images. In 2018, Wang et al. used an ensemble

\*Corresponding author e-mail address: [safak.kayikci@ibu.edu.tr](mailto:safak.kayikci@ibu.edu.tr)

algorithm based on a support vector machine (SVM) on images at different magnifications [8]. The recommended Weighted Area Under the Receiver Operating Characteristic Curve Ensemble (WAUCE) model decreased the variance by %97.89, whereas accuracy is increased by %33.34. Wang et al. won first place in the competition in Symposium on Biomedical Imaging (ISBI) challenge [9]. They achieved 0.925 area under the curve (AUC) for the goal of image categorization and a 0.7051 score for evaluating biopsy images. Bejnordi et al. applied machine learning solutions to get exceptional achievement than 11 participant pathologists in a general pathology simulation demo [10]. Yala et al. proposed several models like risk factor-based logistic regression model (RF-LR), deep learning model (image only DL), and a hybrid DL model that used both traditional risk factors and mammograms [11]. In another study, Khan et al. proposed a fully connected dense architecture using pre-trained convolutional neural network frameworks for cancer type categorization with average pooling [12]. Filipczuk et al. achieved a performance of %98 by applying four distinct classifiers trained with a 25-dimensional feature space to distinguish 737 breast cancer images as benign or malignant [13]. In 2016, Albarqouni et al. present an innovative conception as learning from crowds which uses data aggregation straightly in training operation of the convolutional neural networks by added crowdsourcing layer (AggNet) [14]. In a study done in 2019, three different convolutional neural networks (CNNs) of Inception V3, Inception-ResNet V2, and ResNet-101 architectures were successfully achieved by Zhou et al. [15]. Terasa et al. made their categorization at the patient level using a model similar to AlexNet for classifying breast cancer images [16]. In this study, the maximum fusion method using different fusion techniques with an average recognition accuracy of %90 and %85.6 has been reported. In another study, %83.25 classification success was achieved at the patient level using CNN and multitasking CNN (MultiTask Cascaded Convolutional Networks, MTCNN) models [17]. In 2018, Alom et al. performed dual and multiple classifications at the patient and image level by the Inception Recurrent Residual Convolutional Neural Network (IRRCNN) model [18]. The proposed model has an architecture consisting of the combination of Inception-v4, ResNet, and Recurrent Convolutional Neural Network models. 128, 256, 512, and 1024 feature maps were used in the blocks used for the construction of the model, respectively. The model includes approximately 9.3 million parameters. Images submitted to the model were considered as randomly cropped or non-overlapping patches. With the model used in this study, an average performance of %97 was achieved [19].

## 2. Dataset

The Breast Cancer (Wisconsin) Diagnosis dataset [20] contains the diagnosis and a set of 30 features defining the specifics of the cell nucleus that exist in the digital appearance of a fine needle aspirate (FNA) of a breast mass. FNA is a distinguishing process utilize to examine lumps or tumors. In this approach, a slim (23–25 scale) perforated needle is injected into the subject for exemplification under a microscope (biopsy). FNA operations are harmless and riskless light surgical protocols.

There are 569 samples, of which 212 of them are malignant, and 357 of them are benign. The percent of the malignant tumor is %37.3. The percent is curiously big. The dataset does not specify a general medical anatomization distribution. Instead, ten substantial features are measured for every cell: radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension.

Pearson Correlation or Pearson Moment Correlation (PPMC), or bivariate correlation, is the standard measure of correlation in statistics. It represents the linear conjunction among two samples. As the correlation coefficient approaches 1, it implies that for each positive increment in a parameter, there will be a positive increment in the other related parameter at a fixed rate. Zero means there is no positive or negative increase with any increase. These two are not related. The correlation coefficient -1 implies that for each positive increment in a parameter, there will be a positive decrement at a specified rate in the other variable. The correlation coefficient in absolute value gives us the strength of the relationship. The larger the number, the stronger the partnership. With Pearson correlation, only a linear relationship between two continuous variables can be tested (A relationship is linear only when a change in one variable is associated with a proportional change in the other variable). Pearson correlation The highest correlations are between perimeter\_mean and radius\_worst, area\_worst and radius\_worst, perimeter\_worst and radius\_worst, perimeter\_mean, area\_worst, area\_mean, radius\_mean, texture\_mean, and texture\_worst. Pearson Correlation of features is shown in Figure 1.

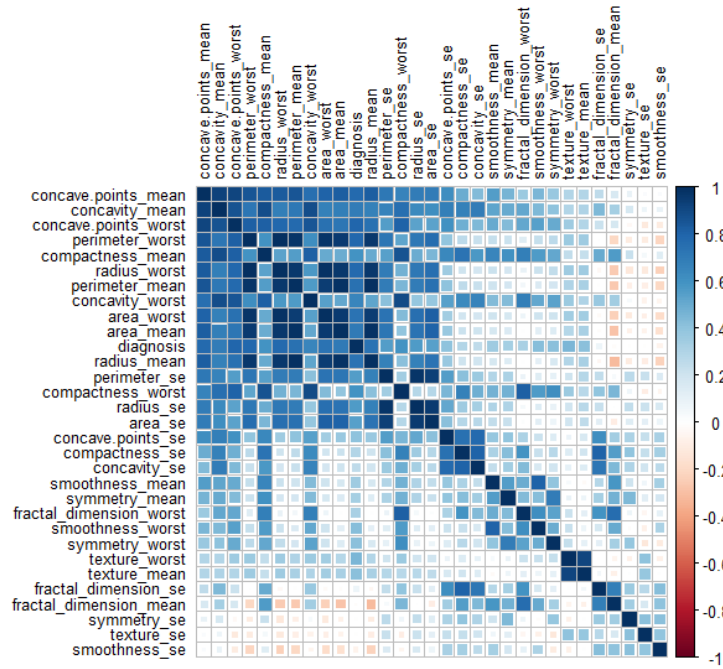


Figure 1. Pearson correlation of features.

These highly correlated features are represented in Figure 2 with the usage of boxplot2g, showing the scatter plot (in the two dimensions given by the selected features) for the clustered data (grouped by diagnosis), over which are superposed the elliptical-shaped boxes in an equivalent (but still enhanced) way a boxplot will visualize the same information for a single dimension. It is observed that some of the correlated pairs are showing a good separation as well between data with diagnosis B and data with diagnosis M.

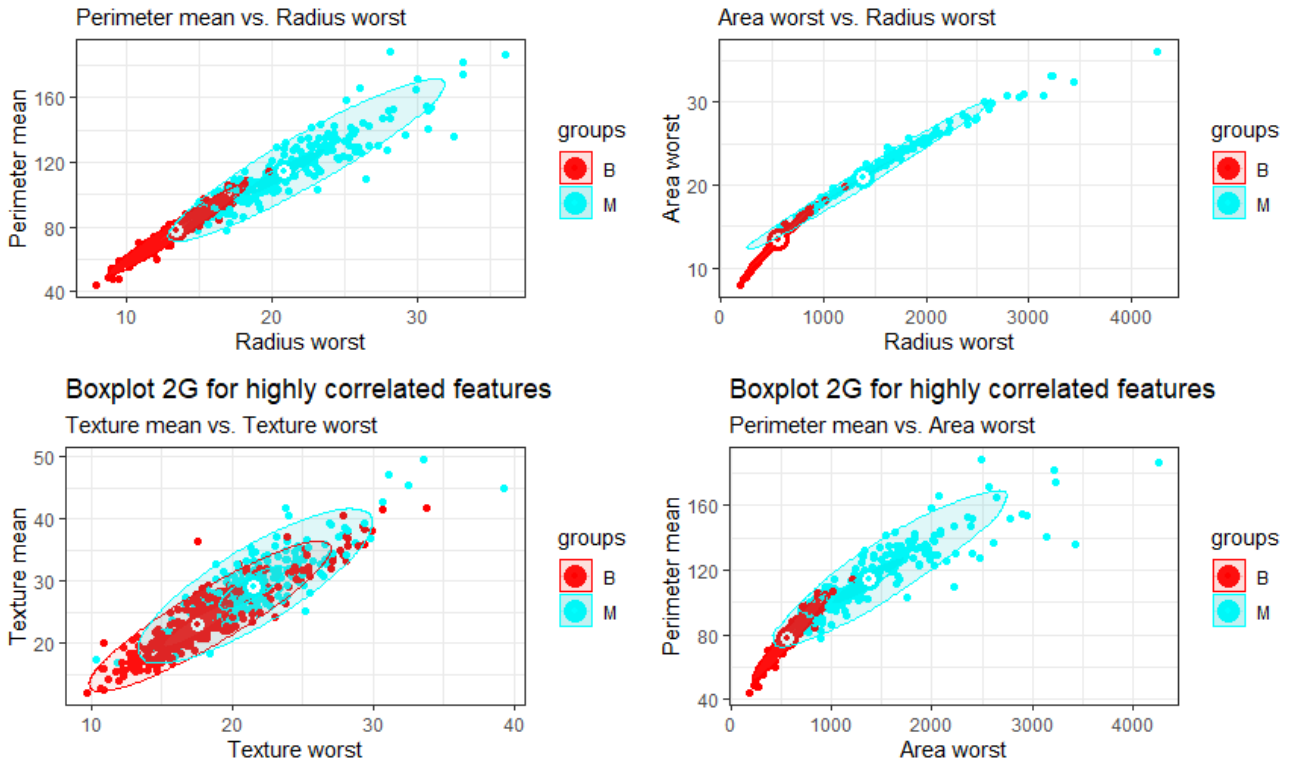
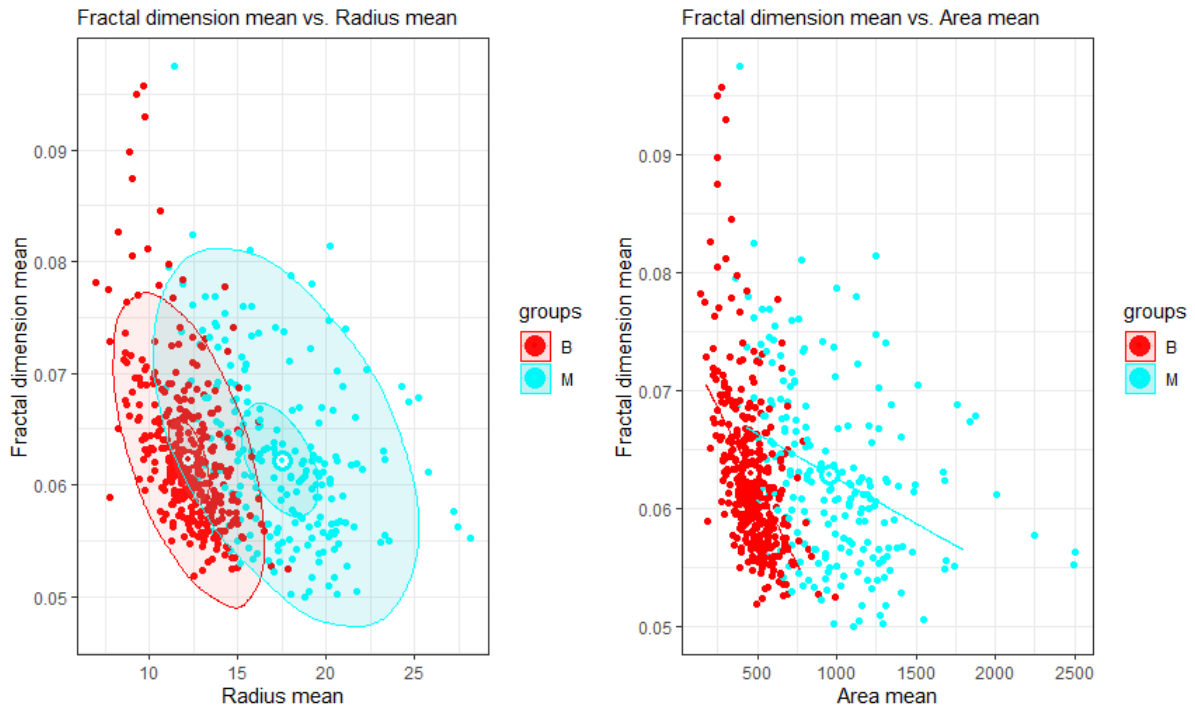


Figure 2. Highly correlated pairs.



Boxplots for some inverse correlated and low correlated pairs of features are shown in Figure 3 and Figure 4. It is observed that low correlated features that have in the same time a considerable overlap for the two 'M' and 'B' groups (ex: 'fractal\_dimension\_worst' and 'area\_se') as well as low correlated features that have in the same time a good selectivity for 'M' and 'B' groups (ex: 'perimeter\_worst' and 'fractal\_dimension\_se').



Figure

3. Inverse correlated pairs.

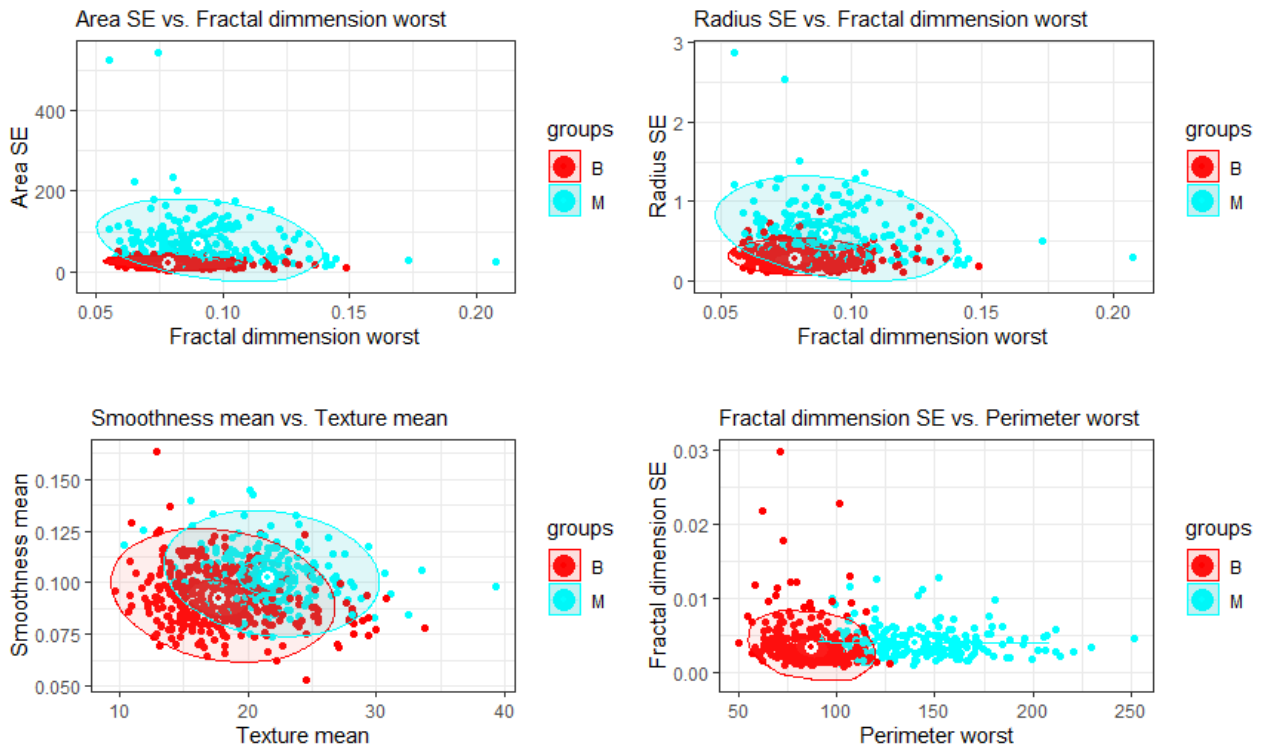


Figure 4. Low correlated pairs.

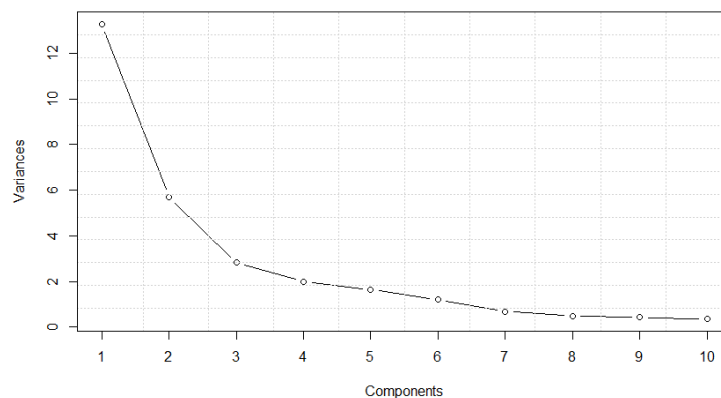
### 3. Method

#### 3.1. Principal Components Analysis (PCA) transform

Principal component analysis (PCA) is a numerical method that converts mutually related variables group into another group of variables that are not mutually related. These new variables, made up of eigenvectors, represent the original data differently. It is the mere of true eigenvector-based multivariate analysis. Most of the time, this work is seen as disclosing the inner composition of the data by revealing the variances in the data in the best way. If a multivariable dataset is considered as a coordinate system with one variable per axis, the principal component analysis provides the user with a low-dimensional shadow picture that contains the most informative view of the data at hand. PCA moves data to a new coordinate system with a linear orthogonal transformation. Thus, the largest variance obtained from the initial data is the first coordinate and is considered the first fundamental component. It can be defined as the oldest statistical tool used to analyze multivariate databases and reduce large data to lower dimensions.

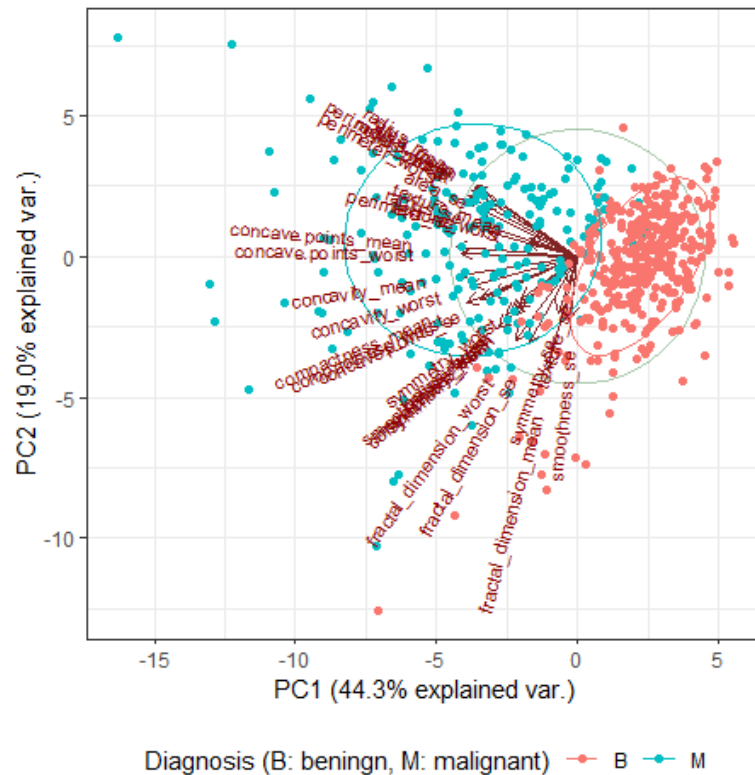
PCA was first proposed by Pearson in 1901 [21], and the development of the theory was made by Hotelling in 1933 [22]. It can be used to show the relationship, similarity, or differences of information in a multidimensional database. The biggest advantage of PCA is that once the model of the information in the database is defined, the data size is reduced and compressed, and data loss is minimized. This feature of PCA is also used in compressing pictures. In this way, the same picture can be obtained again without losing much information. Out of these specialties, PCA also has other advantages such as low sensitivity to noise, reduced memory and capacity needs, and more efficient operation in small-sized spaces. It is one of the most preferred methods in the field due to its ease of application based on Karhunen-Loeve (K-L) or Hotelling Transformation. In addition, PCA is a linear method that can be used to reduce redundancies based on the least mean square error rate with the linear transformation of the coordinates of the data.

In this study, the data is projected in the plane of the two principal components. The direction of the features is represented in the same plane. Two ellipses are showing the 0.68 probability boundary for the distribution of the two groups of diagnosis, B (benign) and M (malignant). A circle superposed over the scatter plot data helps evaluate the relative ratio between the features in the most important principal components plane. The attributes with the most magnitudes or aligned with the governing principle component have the maximum variance.



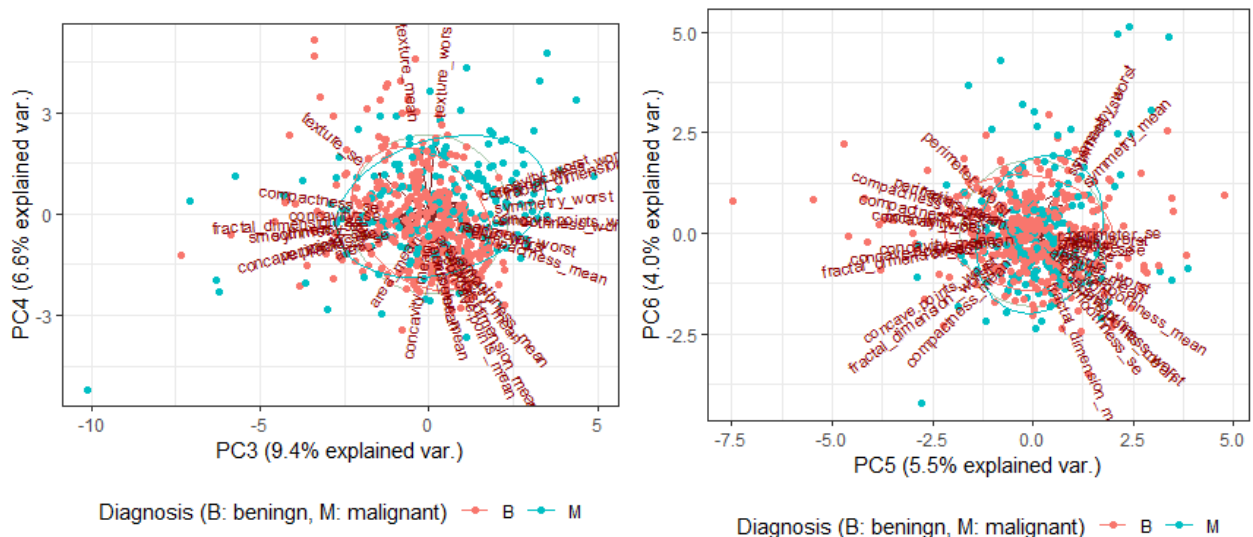
**Figure 5.** Principle component analysis weights.

The first two Principal Components explain together a %63.3 from the variance as shown in Figure 6.



**Figure 6.** Data distribution in the plan of PC1 and PC2.

The projection of the data in the {PC3, PC4} and {PC5, PC6} principal components planes is shown in Figure 7. Principal components PC3-PC6 are explaining together %25.5 variations. It is also observed that not only there is no significant alignment of a certain feature with one of the PC3:PC6 principal components but also in the planes {PC3, PC4} and {PC5, PC6} the B and M points are not separated in distinct clusters like it is the case in the {PC1, PC2} plane.



**Figure 7.** Data distribution in the plan of PC3-PC4-PC5 and PC6.

### 3.2. Gradient Boosting Machine (GBM)

Gradient Boosting Machine (GBM) was presented by Jerome Friedman in 2001 [23]. Other than this name, GBM is also referred to as MART (Multiple Additive Regression Trees) or GBRT (Gradient Boosting Regression Trees) in the literature. It constructs a forward stage-wise additive model by implementing gradient descent in function space. Cross-validation with five folds is used in the study. To find the best number of trees to use for the prediction for the test data, ‘gbm.perf’ function is used [24]. Mathematical implementation of boosting in gbm is explained below [25]:

Initialize  $\hat{f}(x)$  to be a constant with sampling rate  $p$ ,  $\hat{f}(x) = \arg \min_p \sum_{i=1}^N \Psi(y_i, p)$  (1)  
 For  $t$  in  $1, \dots, T$  number of iterations do

1. Compute the negative gradient as the working response

$$z_i = - \frac{\delta}{\delta f(x_i)} \Psi(y_i, f(x_i)) \mid_{f(x_i) = \hat{f}(x_i)} \quad (2)$$

2. Randomly select  $p \times N$  cases from the dataset.
3. Fit a regression tree with the depth of each tree  $K$  terminal nodes,  $g(x) = E(z/x)$ . This tree is fit using only those randomly selected observations.
4. Compute the optimal terminal node predictions,  $\rho_1, \dots, \rho_k$ , as

$$p_k = \arg \min_p \sum_{x_i \in S_k} \Psi(y_i, \hat{f}(x_i) + p) \quad (3)$$

where  $S_k$  is the set of  $x$ s that define terminal node  $k$ . Again this step uses only the randomly selected observations.

5. Update  $\hat{f}(x)$  with learning rate  $\lambda$  as

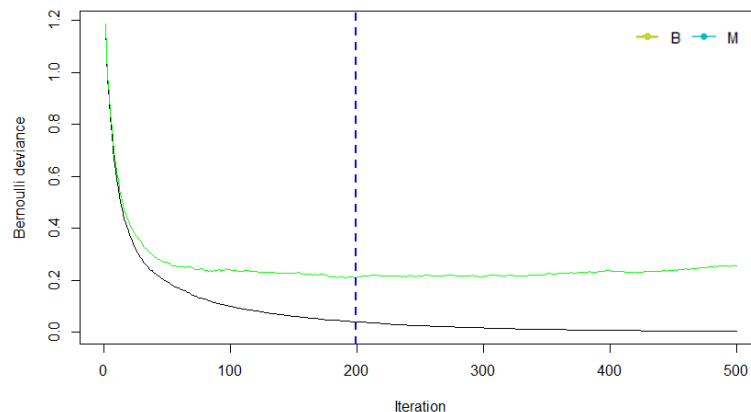
$$\hat{f}(x) \leftarrow \hat{f}(x) + \lambda p_{k(x)} \quad (4)$$

where  $k(x)$  indicates the index of the terminal node into which an observation with features  $x$  would fall. This function is used because it returns the optimal number of trees for prediction.

GBM parameters used in the study are given in Table 1 and Figure 8.

**Table 1.** Gradient Boosting Machine parameters.

distribution	bernoulli
n.trees	500
shrinkage	0.1
n.minobsinnode	15
cv.folds	5
n.cores	1



**Figure 8.** The number of trees in GBM.

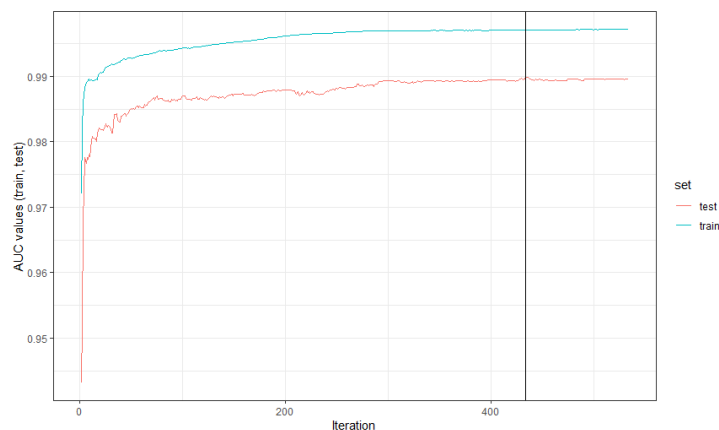
### 3.3. eXtreme Gradient Boost (XGBoost)

Extreme Gradient Boosting (XGBoost) is a machine learning algorithm that has become increasingly popular in both the data science and remote sensing field with its high classification performance and works based on gradient boosted decision trees [26]. XGBoost, one of the new generation community learning algorithms, increases the general accuracy (performance) of the model by preventing the problem of overfitting during the training process of the algorithm. The main reason underlying the success of this method is the purpose function it uses in the learning process. The objective function consists of the loss/loss function and the term regularization. The loss/loss function calculates the difference of each predicted value made by the model from its (predicted class) real value. The term regularization, on the other hand, controls the complexity of the model, and this eliminates the overfitting problem in the model [27][28]. The parameters determined within the scope of the study are given in Table 2.

**Table 2.** XGBoost parameters.

objective	binary:logistic
eval_metric	AUC
eta	0.012
subsample	0.8
max_depth	8
colsample_bytree	0.9
min_child_weight	5
ifold	5
nrounds	5000
nthread	1
early_stopping_round	100

The most suitable parameter values (for the number of leaves and learning rate parameters) were found by the grid search algorithm. The most suitable parameters were searched between [100-1000] for the number of leaves and [0.1-1] for the learning rate. The grid search algorithm trains all combinations for the parameter pair (number of leaves and learning rate parameters) and calculates the model accuracy with the cross-validation technique. The parameter pair that gives the highest model accuracy is accepted as the most suitable parameter value. The binary logistic objective function is used for the process. The evaluation metric is chosen as AUC (area under the curve). Initial start is given by  $n = 0.012$ ,  $subsample=0.8$ ,  $max\_depth=8$ ,  $colsample\_bytree=0.9$  and  $min\_child\_weight=5$ . Model is trained using cross-validation with five folds. The number of rounds equal to 5000, with early stopping criteria for 100 steps, are used. Also, the frequency of printing partial is set for results every 100 steps. The area under curve values in Figure 9.



**Figure 9.** The area under curve values for XGBoost.

### 3.4. Light Gradient Boosting Machines (LightGBM)

Light gradient enhancement machines (LightGBM) are one of the next-generation community learning algorithms based on decision trees operating under the framework of gradient enhancement [29]. Developed by Microsoft in 2017, LightGBM has attracted the attention of researchers in both data science and remote sensing, especially with its outstanding achievements in machine learning competitions in recent years. As the method's

name suggests, it takes the prefix "Light" because it is an algorithm with high processing speed. A feature that distinguishes this method from other gradient enhancement algorithms is the growth strategy used in the training of decision trees. While LightGBM uses the vertical growth strategy (leaf growth), other gradient enhancement methods use the level-wise growth strategy. Another important detail that makes LightGBM exclusive and unique is the two new algorithms it contains and enables it to increase processing speed. These algorithms are gradient-based one-sided sampling (Gradient-based One-Side Sampling, GOSS) and exclusive feature bundling (EFB) methods[30]. It uses the subsampled dataset that it generates from the data instead of using the entire data with gradient-based one-sided sampling. It also reduces transaction complexity by converting sparse features to more frequent/dense features with exclusive feature support. The parameters used in the classification process are shown in Table3. The selection of the parameters has been chosen in consideration of the suggestions on the "parameter setting" page on the main page of LightGBM. Light GBM grows trees vertically while another algorithm grows trees horizontally, meaning that Light GBM grows tree leaf-wise while another algorithm grows level-wise. LightGBM parameters are given in Table 3.

**Table 3.** *LightGBM parameters.*

params	lightGBM.grid
learning_rate	AUC
num_leaves	0.012
num_threads	0.8
nrounds	8
early_stopping_rounds	0.9
eval_freq	5
eval	
ifold	5
stratified	5000

#### 4. Results

The confusion matrix table was used for performance measurements in this study, as is done in most studies. The data set is randomly divided into train (%70 of data = 398) and test parts (%30 of data = 171) at random. The train set is used for finding the optimal model parameters by cross-validation whereas the test set is used only to measure the performance of the model. Accuracy success is measured by the percentage of correctly classified data specimens. Table 4 shows the confusion matrix values for algorithms with indicating true and predicted values.

**Table 4.** *Confusion matrix values.*

Malignant	GBM = 1 XGBoost = 4 LightGBM = 1	GBM = 62 XGBoost = 60 LightGBM = 56
Benign	GBM = 105 XGBoost = 102 LightGBM = 111	GBM = 3 XGBoost = 5 LightGBM = 3
	Benign	Malignant

Receiver Operating Characteristic (ROC) is the ratio of True Positive Rate (TPR) and False Positive Rate (FPR) that is a metric by which we understand whether the models established to solve classification problems are working well. The Roc Curve (Curve) is the curve showing the values that FPR will take if TPR increases, ie 1 convergence. Here, it is desired that the TPR converges to 1, but it is desired that the FPR remain low in addition to the situation where this TPR converges to 1. Area Under the Curve(AUC) is the area under the ROC curves. As this area approaches 1, the performance of the model increases. As shown in Table 5, GBM provided the most successful solution with 0.972. LightGBM had 0.970 and XGBoost had 0.963 AUC values.



**Table 5.** Area under curve (AUC) values.

algorithm	value
Gradient Boosting Machine (GBM)	0.972
eXtreme Gradient Boost (XGBoost)	0.963
Light Gradient Boosting Machine (LightGBM)	0.970

## 5. Conclusion

In this study, the evaluation of pathology images with deep learning architectures is presented. Gradient Boosting Machine (GBM), eXtreme Gradient Boost (XGBoost), and Light Gradient Boosting Machine (LightGBM) have been used because of their minimum classification error. With these architectures, it is possible to detect cancer cells without manually extracting any feature. A success rate of  $97 \pm 1$  was determined. It has been observed that each deep learning model gives different success results. The highest success is obtained from GBM architecture. The hybrid boosting models can improve the classification performance, but training times and degrees of complexity will increase. Microscopic images of breast tissue must be interpreted in the diagnosis of breast cancer among women. A digital medical photography technique is used by doctors to detect breast cancer. However, for accurate detection, it should be sufficient in the field and spend more time. Computer-aided systems recommended assisting experts are extremely important. Various techniques have been developed for these systems to detect cancer and accurately display cancer cells on the monitor. Advances in digital imaging techniques have automated the diagnostic methods recommended in the pathology workflow. This situation accelerates the diagnosis of the disease. The superior success of deep learning on classification and feature extraction has also shown itself in this area. Comparison of classification methods for diagnosis has been proposed, which can be used in addition to the success of breast cancer diagnosis and the competence of the histopathologist. Continuation of such studies is of great importance and essential in the medical field.

## Declaration of Interest

The authors declare that there is no conflict of interest.

## References

- [1] N. Yassin, S. Omran, M. E. Houbay, and H. Allam, "Machine learning techniques for breast cancer computer aided diagnosis using different image modalities," *A Systematic Review. Comput Meth Prog Bio*, vol. 156, no. 1, pp. 25-45, 2018.
- [2] M. Allinen, R. Beroukhi, L. Cai, C. Brennan, J. Lahti-Domenici, H. Huang, and K. Polyak, "Molecular characterization of the tumor microenvironment in breast cancer," *Cancer Cell*, vol. 6, no. 1, pp. 17-32, 2004.
- [3] S. H. Jafari, Z. Saadatpour, A. Salmaninejad, F. Momeni, M. Mokhtari, J. S. Nahand, and M. Kianmehr, "Breast cancer diagnosis: imaging techniques and biochemical markers," *Journal of Cellular Physiology*, vol. 233, no. 7, pp. 5200-5213, 2018.
- [4] N. Antropova, B. Q. Huynh, and M. L. Giger, "A deep feature fusion methodology for breast cancer diagnosis demonstrated on three imaging modality datasets," *Medical physics*, vol. 44, no. 10, pp. 5162-5171, 2017.
- [5] W. Yue, Z. Wang, H. Chen, A. Payne, and X. Liu, "Machine learning with applications in breast cancer diagnosis and prognosis" *Designs*, vol. 2, no. 2, pp. 13, 2018.
- [6] F. Spanhol, E. Oliveira, C. Petitjean, and L. Heutte, "Breast cancer histopathological image classification using convolutional neural networks," *International Joint Conference on Neural Networks (IJCNN)*, vol. 32, no. 4, pp. 2560-2567, 2016.
- [7] Z. Han, B. Wei, Y. Zheng, Y. Yin, K. Li, and S. Li, "Breast cancer multi-classification from histopathological images with structured deep learning model," *Scientific Reports*, vol. 7, no. 1, pp. 4172-4182, 2017.
- [8] H. Wang, B. Zheng, S. W. Yoon, and H. S. Ko, "A support vector machine-based ensemble algorithm for breast cancer diagnosis," *European Journal of Operational Research*, vol. 267, no. 2, pp. 687-699, 2018.
- [9] D. Wang, A. Khosla, R. Gargeya, H. Irshad, and A. H. Beck, "Deep learning for identifying metastatic breast cancer," *arXiv preprint arXiv:1606.05718*, 2016.
- [10] B. E. Bejnordi, M. Veta, P. J. V. Diest, B. V. Ginneken, N. Karssemeijer, G. Litjens, and CAMELYON16 Consortium, "Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer," *Jama*, vol. 318, no. 22, pp. 2199-2210, 2017.
- [11] A. Yala, C. Lehman, T. Schuster, T. Portnoi, and R. Barzilay, "A deep learning mammography-based model for improved breast cancer risk prediction," *Radiology*, vol. 292, no. 1, pp. 60-66, 2019.
- [12] S. Khan, N. Islam, Z. Jan, I. U. Din, and J. J. C. Rodrigues, "A novel deep learning based framework for the detection and classification of breast cancer using transfer learning," *Pattern Recognition Letters*, vol. 125, pp. 1-6, 2019.
- [13] P. Filipczuk, T. Fevens, A. Krzyzak, and R. Monczak, "Computer-Aided Breast Cancer Diagnosis Based on the Analysis of Cytological Images of Fine Needle Biopsies," *IEEE Trans. Med. Imaging*, vol. 32, no. 12, pp. 2169-2178, 2013.
- [14] S. Albarqouni, C. Baur, F. Achilles, V. Belagiannis, S. Demirci, and N. Navab, "Aggnet: deep learning from crowds for mitosis

- detection in breast cancer histology images,” *IEEE transactions on medical imaging*, vol. 35 no. 5, pp. 1313-1321, 2016.
- [15] L. Q. Zhou, X. L. Wu, S. Y. Huang, G. G. Wu, H. R. Ye, Q. Wei, and C. F. Dietrich, “Lymph node metastasis prediction from primary breast cancer US images using deep learning,” *Radiology*, vol. 294 no. 1, pp. 19-28, 2020.
- [16] T. Araujo, G. Aresta, F. Castro, J. Rouco, P. Aguiar, C. Eloy, A. Polónia, and A. Campilho, “Classification of breast cancer histology images using convolutional neural networks,” *Plos One*, vol. 12 no. 6, e0177544, 2017.
- [17] N. Bayramoglu, J. Kannala, and J. Heikkila, “Deep learning for magnification independent breast cancer histopathology image classification,” *23rd International Conference on Pattern Recognition (ICPR) ’2016* pp. 2440-2445.
- [18] Z. Alom, C. Yakopcic, M. Taha, and K. Asari, “Breast Cancer Classification from Histopathological Images with Inception Recurrent Residual Convolutional Neural Network,” *J Digit Imaging*, vol. 45, no. 3, pp. 1-13, 2019.
- [19] M. Veta, P. Pluim, J. Van, and A. Viergever, “Breast cancer histopathology image analysis: A review,” *IEEE T BIO-MED ENG*, vol. 61, no. 5, pp. 1400-1411, 2014.
- [20] D. Dua, and C. Graff, “UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>],” Irvine, CA: University of California, School of Information and Computer Science, 2019.
- [21] Karl Pearson F.R.S. “LIII. On lines and planes of closest fit to systems of points in space” *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, <https://doi.org/10.1080/14786440109462720>, 1901.
- [22] H. Hotelling, “Analysis of a complex of statistical variables into principal components,” *Journal of Educational Psychology*, vol. 24, no. 6, pp. 417, 1993.
- [23] J. H. Friedman, “Greedy function approximation: a gradient boosting machine,” *Annals of Statistics*, pp. 1189–1232, 2001.
- [24] B. Greenwell, B. Boehmke, J. Cunningham, G. B. M. Developers, M. B. Greenwell, 2019. Package ‘gbm’.
- [25] G. Ridgeway, “Generalized Boosted Models: A guide to the gbm package,” *Update*, vol. 1, no. 1, 2007.
- [26] T. Chen, and C. Guestrin, “Xgboost: a scalable tree boosting system,” In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining ’07,2016* pp. 785-794.
- [27] A. M. Abdi, “Land cover and land use classification performance of machine learning algorithms in a boreal landscape using Sentinel-2 data,” *GIScience and Remote Sensing*, vol. 57, no. 1, pp. 1-20, 2020. doi: 10.1080/15481603.2019.1650447
- [28] L. Rumora, M. Miler, and D. Medak, “Impact of various atmospheric corrections on sentinel-2 land cover classification accuracy using machine learning classifiers,” *ISPRS International Journal of Geo-Information*, vol. 9, no. 4, pp. 277, doi: 10.3390/ijgi9040277.
- [29] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, ... T.-Y. Liu, “Lightgbm: a highly efficient gradient boosting decision tree,” *Advances in Neural Information Processing Systems*, vol. 30, pp. 3146–3154, 2017.
- [30] R. Wang, Y. Liu, X. Ye, Q. Tang, J. Gou, M. Huang, and Y. Wen, “Power system transient stability assessment based on bayesian optimized LightGBM,” In *2019 IEEE 3rd Conference on Energy Internet and Energy System Integration (EI2) ’11,2019*, pp. 263-268.

# Fabric Defect Classification Using Combination of Deep Learning and Machine Learning

Fatma Günseli YAŞAR ÇIKLAÇANDIR <sup>1,\*</sup>, Semih UTKU <sup>2</sup>, Hakan ÖZDEMİR <sup>3</sup>

<sup>1</sup> İzmir Katip Çelebi University, Faculty of Engineering and Architecture, Computer Engineering Department, Turkey

<sup>2</sup> Dokuz Eylül University, Faculty of Engineering, Computer Engineering Department, Turkey

<sup>3</sup> Dokuz Eylül University, Faculty of Engineering, Textile Engineering Department, Turkey

## Abstract

Automatic systems can be used in many areas, such as the production stage in factories, country defense, and traffic control. They provide the opportunity to reach results faster with higher success rates thanks to human-computer vision cooperation. In this study, it is aimed to develop an intelligent system that automatically detects and classifies defects in fabrics. Thanks to the developed system, the cause of the malfunction is eliminated, and the recurrence of the malfunction is prevented. Using deep learning methods in fabric defect classification studies has a disadvantage compared to other methods. Multiple layers in deep learning cause a time-consuming process. Therefore, a combination of Deep Learning and Support Vector Machines (SVM) has been used in this study. The success of the provided system has been compared with other deep learning algorithms in terms of time and accuracy.

**Keywords:** *Convolutional neural network; fabric defect classification; machine learning.*

## 1. Introduction

Fabrics, which differ according to their usage areas, constitute one of the most basic needs of people. Textile is becoming increasingly popular sector with the increase in online shopping. High customer satisfaction, especially due to scoring in online shopping, causes more demand for the product sold. When the customer is satisfied with the product, it provides more profit to the businesses by not returning the product they bought. Customer satisfaction depends on fabric quality, sewing quality, and whether the fabric is defective or not. Since fabric defects affect the gain, it is necessary to find the defects on the fabric produced and take the necessary measures. There are many types of fabric defects. ISO standards [1] indicate that there are 130 different fabric defects. There are some reasons for the occurrence of these defects like raw materials, machines, or humans [2]. It is important to identify and correct the cause of any defect so that the defect does not happen again. Many studies have been carried out to automate the fabric control traditionally done by human power. While some of the studies distinguish fabrics as defected or non-defected, other studies also classify fabric defects detected. It is meaningless not to classify the defect on the fabric in terms of making necessary corrections.

Zhu et al. [3] optimize DenseNet, which is a CNN (Convolutional Neural Network) algorithm. They combine the new method with new hardware for fabric defect detection. Huang et al. [4] perform segmentation and detection based on neural networks. Huang determined the location of the fault as well as the fault detection. Karlekar et al. [5] use wavelet decomposition and different preprocessing operations to obtain segmented defects. Chang et al. [6] develop a new method for patterned fabrics. Fabrics are divided into lattices, including periodic patterns. Then, the lattice containing the defect is detected. Wei et al. [7] make a combination of compressive sampling theorem with CNN. This new method is more effective compared to traditional methods and performs well in small data sets. Studies on the Tilda data set, also used in our study, were examined in detail. Başbüyük et al. [8] have achieved 97% success by applying particle filtering in some parts of the Tilda dataset (c1-r1, c1-r3). Sezer et al. [9] apply independent component analysis (ICA) for defect detection. They use the parts of c1-r1, c1-r3, c2-r2, c2-r3 of the Tilda dataset. Bissi et al. [10] use Gabor filter bank and principal component analysis (PCA) and test the performance using the parts of c1-r1 and c1-r3 of Tilda. This study, with more than 98% success, is more effective than the study of Başbüyük et al. [8]. Jing et al. [11] combine the Gabor filter and genetic algorithm in their study. When the performance of Local Homogeneity Analysis (LHA) is compared with Wavelet transform and Gabor Transform, it is seen that LHA gives the highest accuracy rate (96.40%) in the study of Kure et al. [12]. After partitioning the images into blocks, feature vectors extracted from each block are used in a regression-based method named PG-LSR in the study of Cao et al. [13]. Liu et al. [14] use ELM (Extreme Learning Machine) method after extracting the features from segmented defects in fabrics. The accuracy of the system they provided is 94.5%. Jing et al. [15] use a convolutional neural network (CNN) after division the images into patches. Tilda is one of the databases used to test the proposed method. In this database, 97.48% classification

\*Corresponding author e-mail address: [fatmagunseli.yasar@ikc.edu.tr](mailto:fatmagunseli.yasar@ikc.edu.tr)

accuracy rate has been achieved with Tilda dataset. Jeyeraj et al. [16] use a transfer learning-based CNN algorithm called AlexNet. They obtain a high accuracy rate (96.55%). Cuifang et al. [17] extract features using pyramid histogram of oriented gradients (PHOG) and perform classification using support vector machine (SVM) in their study. The performance of PHOG is superior to the performances of scale-invariant feature transform (SIFT) and histogram of oriented gradients (HOG). It is seen that machine learning-based methods are used in most of the studies using the Tilda dataset. There are very few studies in the literature that test the performance of deep learning algorithms on the Tilda dataset.

The common feature of the algorithms used in this study is that the input images to be classified must have dimensions of  $224 \times 224 \times 3$ . Therefore, the images in the dataset were pre-processed. In the pre-processing step, the performances of different color maps used to make the images three-dimensional were also examined. The performances of ResNet18 and GoogLeNet have been investigated using the three-dimensional version of the Tilda dataset. Then, the used deep learning algorithms are combined with SVM to decrease the response times of deep learning algorithms.

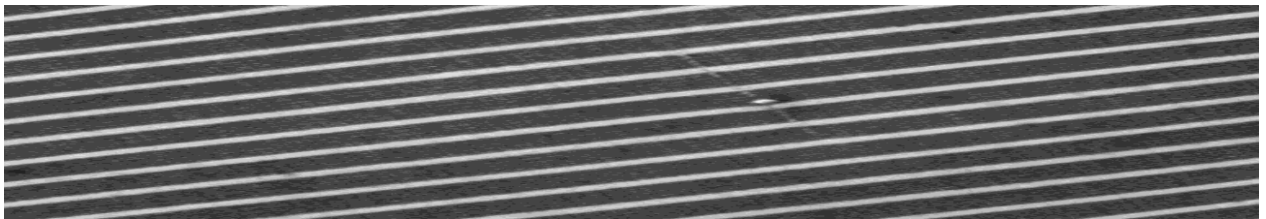
## 2. Methodology

### 2.1. Dataset

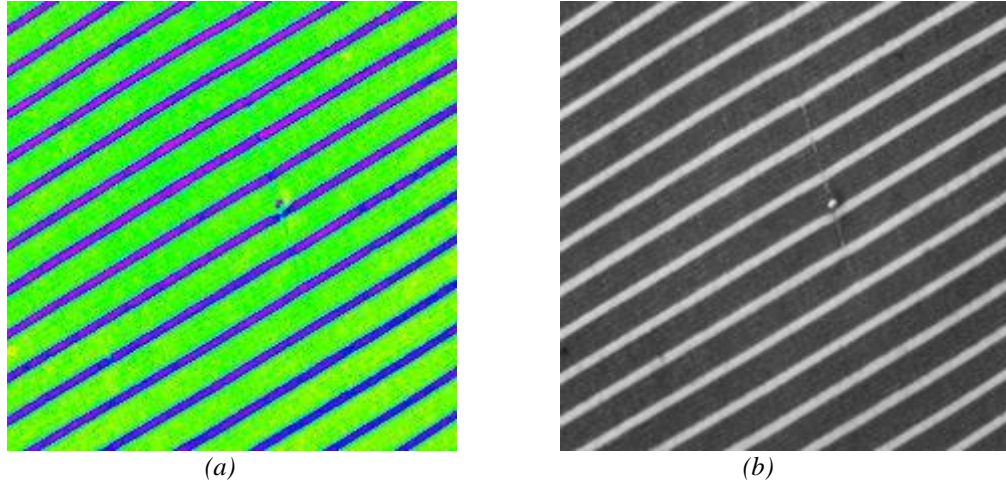
Tilda data has 12 different groups. Not all of these groups are available for use. Only eight groups (c1-r1, c1-r3, c2-r2, c2-r3, c3-r1, c3-r3, c4-r1, c4-r3) can be accessed. The groups of c1-r1, c1-r3, c2-r2, c2-r3 (groups with the names beginning with c1 and c2) consist of un-patterned fabric samples, while the other four groups (groups with the names beginning with c3 and c4) consist of patterned fabric samples. There are eight different classes (e0, e1, e2, e3, e4, e5, e6, e7) in each group. Each group contains a total of 400 images, with 50 images of each class. Table 1 contains un-defective fabric samples belonging to e0 classes of groups for each group.

### 2.2. Preprocessing Operations

Tilda dataset has images of sizes  $768 \times 512$ , and all of them are gray levels (Figure 1). Deep learning algorithms to be used are trained with RGB images. It is not known whether these images in the Tilda dataset are gray level to reduce the space needed on the Ram or Hard disk, but they must be RGB to be processed in deep learning. Therefore, the images are converted into three dimensions using two different color maps (HSV and gray). Images ready to be used in deep learning algorithms are  $224 \times 224 \times 3$  in size (Figure 2). Converting the images from one dimensional to three dimensional will result in loss of information. Lack of knowledge in the processes will affect the performance of all methods. For this reason, performance comparisons between the methods were made in our study. Without the lack of knowledge, the success of the methods will be higher than expected.



**Figure 1.** Image before preprocessing (c3-r3)



**Figure 2.** Image after preprocessing (a) using hsv colormap (b) using a gray colormap

### 2.3 Deep learning & Machine Learning

Deep learning is an improved version of artificial neural networks. Many studies are using deep learning in the literature. Deep learning is used in various fields such as image processing, video processing, signal processing, object recognition, defense industry, and robotics. Deep learning uses many layers of nonlinear processing units for feature extraction and conversion. The output from each previous layer is used by each subsequent layer as input.

There are many different types of deep learning architectures established by increasing the number of layers in artificial neural networks. CNN is one of these architectures. It is used in image processing. Two of the most widely used CNN models, which are ResNet and GoogLeNet, have been compared in this study. The structure of the ResNet, which is the first network with 34 layers, consists of residual blocks. It is a model designed deeper than all architectural structures. The GoogLeNet model, which consists of the combination of Inception modules, has a complex structure.

A large number of layers in CNN algorithms causes a waste of time. In this study, a new method has been developed to eliminate this disadvantage. This study consists of two basic parts: feature extraction and classification. The features extracted by CNN have been inserted into the Support Vector Machines (SVM) classifier. Thanks to the SVM classifier, it is found whether there is a defect in the fabric. If there is a defect in the fabric, it is found out what this type of defect is.

### 3. Results

In the experimental phase of this study, the methods have been compared in terms of accuracy and execution time. They have been tested in a personal computer (Intel (R) Core (TM) i7- 6700HQ CPU @2.60 GHz). Table 1 and Table 3 contain the results obtained using color maps of HSV and gray, respectively. In Table 1, the accuracy rate of the ResNet18 algorithm is up to 87.5% (for the group of c3-r1). The classification accuracy of GoogLeNet is a maximum of 81.67% (for the group of c2-r2). The maximum accuracy rates from combinations with SVM are close to the maximum accuracy rate of ResNet18 (SVM & ResNet18, SVM & GoogLeNet). Looking at the average accuracy rates of the methods, it is seen that ResNet18 is again the most successful method (76.98%). Considering the accuracy rates of the groups, it is seen that c1-r1 is classified with the highest performance (78.54%). In Table 3, the maximum accuracy rate is obtained when using the ResNet18 algorithm (84.17%). C1-r1 is the set with the highest accuracy rate for all methods.

On the other hand, ResNet18 is the method with the highest accuracy rate. However, it is observed that there are problems in the classification of the c4-r1 and c4-r3. All methods have very low accuracy rates for these groups. In Table 1 and Table 3, it is seen that close results are obtained that support each other when two different color maps are used. Here, it can be concluded that when using any of the color maps of HSV and gray, similar results are obtained unless different processes such as filtering and morphological processing are applied.

**Table 1.** Accuracy rates for hsv colormap

	c1		c2		c3		c4		Average
	r1	r3	r2	r3	r1	r3	r1	r3	
<b>ResNet18</b>	85.83%	81.67%	76.67%	81.67%	87.5%	75%	61.67%	65.8%	76.98%
<b>GoogLeNet</b>	63.33%	72.5%	81.67%	71.67%	83.33%	71.67%	46.67%	53.3%	68.02%
<b>SVM &amp; ResNet18</b>	86.67%	76.67%	77.5%	69.17%	78.33%	67.50%	60.83%	55%	71.46%
<b>SVM &amp; GoogLeNet</b>	85%	69.17%	68.33%	70%	72.5%	69.17%	52.5%	55%	67.71%
<b>Average</b>	78.54%	73.96%	76.04%	73.13%	77.71%	70.84%	55.42%	57.28%	

Table 2 and Table 4 contain time comparisons for the methods. The completion times of ResNet18 and GoogLeNet are given in minutes, and the completion times of SVM-based methods are given in seconds. Deep learning-based methods give results in a longer time (28.13 minutes). On the other hand, classification with SVM gives results in a very short time (about 30 seconds). As it can be understood from here, deep learning-based methods respond about sixty times longer than SVM-based methods. This method will be better than deep learning-based methods in terms of time and performance when the performance of SVM-based Resnet18 is improved.

**Table 2.** Time comparison for hsv colormap

	c1		c2		c3		c4		Average
	r1	r3	r2	r3	r1	r3	r1	r3	
<b>ResNet18</b>	29.15 min	30.92 min	29.35 min	28.27 min	29.42 min	28.92 min	31.33 min	30.0 min	29.36 min
<b>GoogLeNet</b>	27.03 min	26.85 min	29.22 min	27.42 min	32.42 min	31.18 min	29.37 min	28.93 min	
<b>SVM &amp; ResNet18</b>	27 sec	26 sec	28 sec	28 sec	28 sec	27 sec	27 sec	29 sec	30 sec
<b>SVM &amp; GoogLeNet</b>	32 sec	32 sec	32 sec	31 sec	32 sec	33 sec	32 sec	32 sec	

**Table 3.** Accuracy rates for gray colormap

	c1		c2		c3		c4		Average
	r1	r3	r2	r3	r1	r3	r1	r3	
<b>ResNet18</b>	84.17%	80.0%	82.5%	80%	80%	76.7%	62.5%	60.0%	75.73%
<b>GoogLeNet</b>	80.83%	72.5%	77.5%	70.00%	75.83%	65%	49.17%	41.67%	66.56%
<b>SVM &amp; ResNet18</b>	77.50%	74.17%	70.00%	68.33%	74.17%	65.83%	65.83%	55.00%	68.85%
<b>SVM &amp; GoogLeNet</b>	74.17%	63.33%	62.50%	74.17%	70.83%	58.33%	50.00%	47.50%	62.60%
<b>Average</b>	79.17%	72.5%	73.13%	73.13%	75.21%	66.47%	56.88%	51.04%	



**Table 4.** Time comparison for gray colormap

	c1		c2		c3		c4		Average
	r1	r3	r2	r3	r1	r3	r1	r3	
<b>ResNet18</b>	26.62 min	27.65 min	27.18 min	27.4 min	26.43 min	26.72 min	27.32 min	27.32 min	26.89 min
<b>GoogLeNet</b>	26.03 min	29.27 min	26.23 min	26.25 min	26.88 min	26.55 min	26.43 min	25.87 min	
<b>SVM &amp; ResNet18</b>	29 sec	28 sec	23 sec	28 sec	27 sec	27 sec	28 sec	27 sec	29 sec
<b>SVM &amp; GoogLeNet</b>	30 sec	31 sec	33 sec	31 sec	31 sec	30 sec	30 sec	31 sec	

Some parts of the Tilda dataset consist of non-patterned fabrics, and the other part of it consists of patterned fabric samples. Table 5 shows the performances of the methods according to the un-patterned/patterned fabrics. While all methods are more successful in classifying defects in non-patterned fabrics (75.29%), they show less success in classifying patterned fabric defects (64.19%).

**Table 5.** Average accuracy rates for un-patterned/patterned fabrics

Method	Un-Patterned Fabrics	Patterned Fabrics
<b>ResNet18</b>	81.56%	71.15%
<b>GoogLeNet</b>	73.75%	60.83%
<b>SVM &amp; ResNet18</b>	75.00%	65.31%
<b>SVM &amp; GoogLeNet</b>	70.83%	59.48%
<b>Average</b>	75.29%	64.19%

#### 4. Conclusion

Deep learning is one of the most popular concepts in recent years. The loss of time caused by deep learning algorithms has been eliminated with the system proposed in this study. What distinguishes deep learning from artificial neural networks is the high number of layers. This feature provides higher success when using deep learning. However, the disadvantage of deep learning is that it is a time-consuming process. Deep learning may not be preferred in areas where speed is important, as its algorithms take a long time to produce results. In this study, the combination of deep learning with SVM has been carried out to eliminate this disadvantage of deep learning. After extracting the features using deep learning methods, the classification has been performed in the SVM algorithm.

When looking at the performance comparisons of the methods, it is seen that SVM-based methods give results about 60 times shorter than deep learning-based methods. Looking at the accuracy rates, it is seen that ResNet18 is the most successful method in classification. The second most successful method after ResNet18 is the combination of SVM and ResNet18. The time advantage of the SVM & ResNet18 combination avoids the disadvantage of having low classification success.

The failure of the methods in the groups of c4-r1 and c4-r3 where there are fabric images with mixed patterns indicates that these methods should take extra precautions for such fabrics. The process of detecting defects on patterned fabrics requires different pre-processing operations than detecting non-patterned fabric defects. The results obtained in this study also reveal the necessity of this.

Tilda dataset, a data set used in previous studies, has been used in this study. Only a portion of this dataset is public. For this reason, tests have been carried out on the public part. Additionally, this data set is one dimensional. Data loss occurs during the conversion of images from one dimensional to three dimensional. For this reason, the system performance seems to be lower than the performances of the studies in the literature. The success of this preliminary study will be increased by adding innovations to the preprocessing step.

#### Declaration of Interest

The authors declare that there is no conflict of interest.

#### References

- [1] ISO (1990) "Woven Fabrics – Description of defects – Vocabulary," ISO 8498: 1990 (E/F).
- [2] B. Barış, and H. Z. Özek, "Dokuma Kumaş Hatalarının Sistematik Sınıflandırılması Üzerine Bir Çalışma," *Tekstil ve Mühendis*, vol. 26, no. 114, pp.156-167, 2019.

- [3] Z. Zhu, G. Han, G. Jia, and L. Shu, "Modified densenet for automatic fabric defect detection with edge computing for minimizing latency," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9623-9636, 2020.
- [4] Y. Huang, J. Jing, and Z. Wang, "Fabric Defect Segmentation Method Based on Deep Learning," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1-15, 2021.
- [5] V. V. Karlekar, M. S. Biradar, and K. B. Bhangale, "Fabric defect detection using wavelet filter," In *2015 International Conference on Computing Communication Control and Automation*, pp. 712-715, 2015.
- [6] X. Chang, C. Gu, J. Liang, and X. Xu, "Fabric defect detection based on pattern template correction," *Mathematical Problems in Engineering*, 2018.
- [7] B. Wei, K. Hao, X. S. Tang, and Y. Ding, "A new method using the convolutional neural network with compressive sensing for fabric defect classification based on small sample sizes," *Textile Research Journal*, vol. 89, no. 17, pp. 3539-3555, 2019.
- [8] K. Basibuyuk, K. Coban, and A. Ertuzun, "Model based defect detection problem: Particle filter approach," In *2008 3rd International Symposium on Communications, Control and Signal Processing*, pp. 348-351, 2008.
- [9] O. G. Sezer, A. Erçil, and A. Ertuzun, "Using perceptual relation of regularity and anisotropy in the texture with independent component model for defect detection," *Pattern Recognition*, vol. 40, no. 1, pp. 121-133, 2007.
- [10] L. Bissi, G. Baruffa, P. Placidi, E. Ricci, A. Scorzoni, and P. Valigi, "Automated defect detection in uniform and structured fabrics using gabor filters and PCA," *Journal of Visual Communication and Image Representation*, vol. 24, no. 7, pp. 838-845, 2013.
- [11] J. Jing, P. Yang, P. Li, and X. Kang, "Supervised defect detection on textile fabrics via optimal gabor filter," *Journal of Industrial Textiles*, vol. 44, no. 1, pp. 40-57, 2014.
- [12] N. Kure, M. S. Biradar, and K. B. Bhangale, "Local neighborhood analysis for fabric defect detection," In *2017 International Conference on Information, Communication, Instrumentation and Control*, pp. 1-5, 2017.
- [13] J. Cao, J. Zhang, Z. Wen, N. Wang, and X. Liu, "Fabric defect inspection using prior knowledge guided least squares regression," *Multimedia Tools and Applications*, vol. 76, no. 3, pp. 4141-4157, 2017.
- [14] L. Liu, J. Zhang, X. Fu, L. Liu, and Q. Huang, "Unsupervised segmentation and elm for fabric defect image classification," *Multimedia Tools and Applications*, vol. 78, no. 9, pp. 12421-12449, 2019.
- [15] J.F. Jing, H. Ma, and H. H. Zhang, "Automatic fabric defect detection using a deep convolutional neural network," *Coloration Technology*, vol. 135, no. 3, pp. 213-223, 2019.
- [16] P. R. Jeyaraj, and E.R.S. Nadar, "Computer vision for automatic detection and classification of fabric defect employing deep learning algorithm," *International Journal of Clothing Science and Technology*, 2019.
- [17] Z. Cuifang, C. Yu, and M. Jiacheng, "Fabric defect detection algorithm based on PHOG and SVM," *Indian Journal of Fibre & Textile Research (IJFTR)*, vol. 45, no. 1, pp. 123-126, 2020.

# A Face Authentication System Using Landmark Detection

Velican ERCAN, M. Erdal ÖZBEK\*

İzmir Kâtip Çelebi University, Faculty of Engineering and Architecture,  
Department of Electrical and Electronics Engineering, İzmir, Turkey

## Abstract

Biometric data is the key for many security applications. Authentication relies on the individual's measurable biometric properties collected as features. In this study, a face authentication system is built to be used in opening the entrance door accessing to the apartments and housing estates. The proposed system consists of three stages. In the first stage, landmarks on the face are captured using a deep neural network. Then six selected features from the landmarks are extracted and traditional machine learning algorithms are used to authenticate users. In the last stage, a user interface is built. Face recognition tests achieved an accuracy rate of 89.79%.

**Keywords:** *Biometrics, face authentication, facial recognition, landmark detection.*

## 1. Introduction

The physiological or behavioral characteristics of an individual is referred as biometrics [1]. They are preferred in security systems or applications, due to their advantages over traditional systems in authentication. They are unique and inseparable for an individual, thus they cannot be lost nor forgotten. Therefore, it is the effective authentication method that allows authenticating the user directly, rather than her/his smartphone, smart card, or any secret she/he knows. Face is one of the physiological characteristics which is now frequently used in smartphone unlock screens [2].

The first facial recognition system was designed in the 1960s under the leadership of Woody Bledsoe. The system, which could identify forty people per hour at that time [3], can now recognize people in times measured in milliseconds. A facial recognition system is a technology capable of matching a human face from a digital image or a video frame against a database of faces, which works by pinpointing and measuring facial features from a given image or video [4]. In the early 1970s, facial recognition was considered as a two-dimensional pattern recognition problem [5, 6]. A recent review of the technology discussing the future development direction can be found in [7].

Basically, there are three steps in the face recognition process. These are face detection, extraction of attributes, and recognition. Face detection is defined as the process of extracting face/faces from images or frames. It is difficult to determine the detection methods precisely because most of the algorithms are designed to work with other methods, e.g., object detection, in order to improve the performance computed based on accuracy and/or speed [8]. Generally, detection methods can be divided into two categories: knowledge-based and image-based [9]. Knowledge-based methods are rule-based methods. These methods use information about face features, skin colors or template matching. Some rules are easy to guess, e.g., a face generally has two symmetrical eyes, and the eyes are darker than the cheeks. The image-based method uses a training-learning approach to make comparisons between face and non-face images. These kinds of methods have to be trained with large number of face and non-face images to increase the accuracy of this system. There are various face detection algorithms in the literature using Eigenface, AdaBoost, Neural Networks, and Support Vector Machines (SVMs) [10-13].

Recognition consists of preprocessing the face image, vectorizing the image matrices, creating a database of images, and finally classifying the image. Because face recognition is a pattern recognition problem, a learning method should be utilized to compare faces. The most common methods for two-dimensional recognition are Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), Gabor wavelet, and spectral feature analysis [14, 15]. The two-dimensional methods generate features for each individual and then classify them. It is also important to generate abstract feature vectors in order to reduce the size of the input images. Lighting is another important aspect where image enhancement techniques are used to suppress the bad lighting conditions by a logarithm transformation and normalization [9]. The illumination difference between the left and right sides of the face is eliminated by using the mirror of the average illuminated part [16]. Promising results using deep learning approaches on different conditions such as lower and upper face occlusions, misalignment, varying head posture angles have been reported [17-19].

The aim of this study is to strengthen the security in the entrance of the apartments with a biometric system. Biometrics are basically measurable biological traces of a person. By considering the biometric data as the basis

\*Corresponding author e-mail address: [merdal.ozbek@ikcu.edu.tr](mailto:merdal.ozbek@ikcu.edu.tr)

of the face recognition system, landmarks are extracted using deep neural networks. Authentication is based on the extracted features from the landmarks where machine learning algorithms are used to establish identification. A simple user interface is also generated for ease of common usage.

The remainder of the paper is organized as follows: Section 2 explains the system architecture and implementation of the proposed face recognition system including the landmark extraction and user interface. Section 3 reports the accuracy performance of the system. Section 4 summarizes and concludes the paper.

## 2. Methodology

This section presents the information about the proposed system, determining landmarks, extraction of features from landmarks, and the implementation of the graphical user interface (GUI).

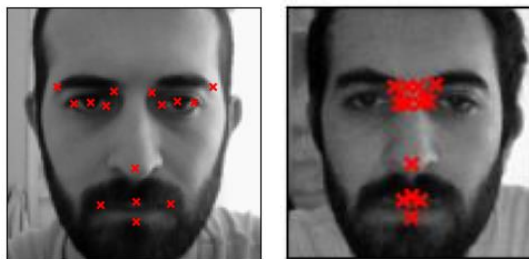
### 2.1. Landmark detection

The semantic meanings of facial contours and facial components are delivered via facial landmarks positioned around those contours and components, as well as the subject's face [20]. Facial landmark detection has attracted a lot of interest in the field of computer vision as a crucial step for a variety of face-related applications [21]. It can be basically stated as predicting the coordinates of the center of eyes, nose tip, eyebrows, and mouth for a given face. The most important step of the facial recognition system is the determination of the key points on the face.

Improvements in landmark detection have been made recently thanks to the use of convolutional neural networks (CNNs). Because of its fault tolerance and self-learning capabilities, this type of feedforward neural network with shared weights and local connections has been widely used in the field of image processing. Therefore, in this study, landmark detection is applied by deep CNN models.

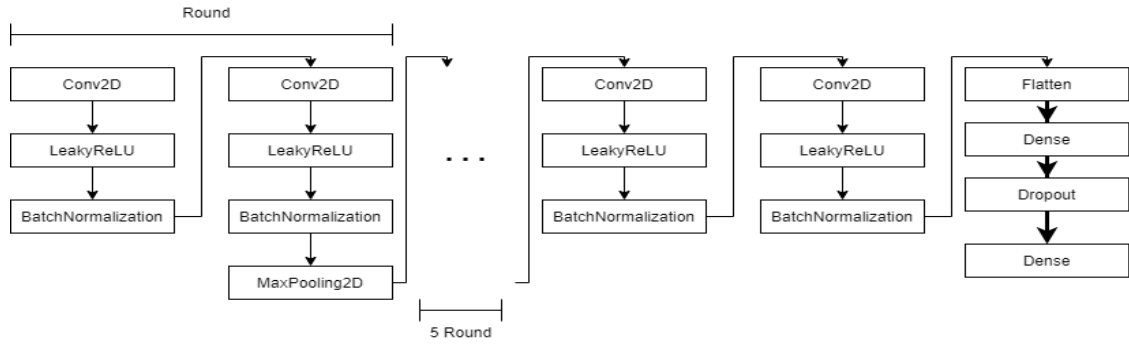
In literature, an efficient CNN model approach for facial key points detection has been offered [22]. In this approach, they augmented the image data with horizontally flipping, then vertically stacked. The image is pre-processed with the pixels by normalizing in the range of  $[0,1]$ , and then the training targets are zero centered to take the values in the range of  $[-1,1]$ . They have created fifteen different so-called NaimishNet models. Each of the fifteen different models has been filtered out the non-missing values. They have shown that the predicted locations are very close to the actual locations.

The facial key points detection model of this study is based on the NaimishNet model. The dataset used in the training of the landmark model belongs to Dr. Yoshua Bengio [23]. The dataset is augmented in order to artificially increase the number of training samples. However, it is found that in NaimishNet model, insufficient light affects the performance as shown in Figure 1.



**Figure 1.** Images with normal (left) and low light (right) conditions with detected key points.

Therefore, we propose a 45-layer CNN model instead of 25-layer, and train models with artificially manipulated brightness values in order to increase accuracy in any poor light conditions while augmenting data. The proposed CNN architecture is shown in Figure 2.



**Figure 2.** CNN architecture.

The proposed architecture is different from the NaimishNet architecture. This model includes additional LeakyReLU and BatchNormalization layers after each of the Conv2D layers. The number of total model parameters is 7,264,318. In order to avoid overfitting, the number of epochs for the training of the neural network was determined as 100, where the validation and training error values were started to increase.

**2.2. Feature extraction**

The landmark model returns an array of coordinates which are then converted to distances including the golden ratio [24], which were also selected in previous studies [25, 26]. The features are extracted based on the key points by proportioning the calculated distances to each other. In this study, two different approaches are considered for feature extraction, with twenty features and six features, respectively. The first approach with twenty features was found too complex for standard machine learning algorithms. Especially if the number of data (image dataset in this study) is not sufficient, the algorithm may not be able to learn and may memorize the data what is known as “overfitting” [27]. In cases of overfitting, very high accuracy values may be observed when training, however very low scores will most likely occur when testing. Besides, similar features are highly correlated. For example, the eye center distance and mouth midpoints are reflected in five features, while nose to eye points are reflected in three features. Therefore, it affects three or five features for each unit pixel estimation that will be found faulty. This may end up with inconsistent results, leading the trained model to make inaccurate predictions.

The features are reduced to six in order to avoid overfitting and to satisfy that the landmark model predicts with the higher accuracy. For example, the deviation in the pupil is more than that in the eyebrow ends. These ratios given in Table 1 are also expected to be less than one, eliminating the necessity of normalizing the data.

**Table 1.** Extracted features.

Feature name	Feature
f1	Eyebrow Mid to Nose / Eyebrow Left Out to Mouth Left
f2	Nose to Mouth Right / Eyebrow Distance
f3	Mouth Width / Eyebrow Mid to Nose
f4	Eyebrow Distance / Eyebrow Right In to Mouth Right
f5	Nose to Mouth Left / Nose tip to Eyebrow Mid
f6	Right Eyebrow / Mouth Width

**2.3. Proposed face recognition system**

After the extraction of six features from the ratios of the landmarks, the collected biometric data is used to perform the recognition step. The traditional machine learning algorithms are available to be used but SVM is found to have the best performance.

The basis of the SVM algorithm used for recognition in this study is based on finding the optimal hyperplane of the separable two classes with the maximum margin [28]. The goal of classifying is to find a line that divides the input space into distinct areas. Assuming the independent and identically distributed input-output training data pair  $(x, y)$  follows an unknown probability distribution,

$$(x_1, y_1), \dots, (x_n, y_n) \in R^N, y_i = \{-1, +1\} \tag{1}$$

If the training data can be separated by a hyperplane, the margin is the shortest distance between a sample and the decision hyperplane. A kernel function is used to translate the input data to a high-dimensional feature space, and then a linear classifier is created in this space to solve the following optimization problem

$$\min_{w,b} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i \tag{2}$$

where  $w$  is the weight vector, and  $b$  is a scalar (bias), subject to,

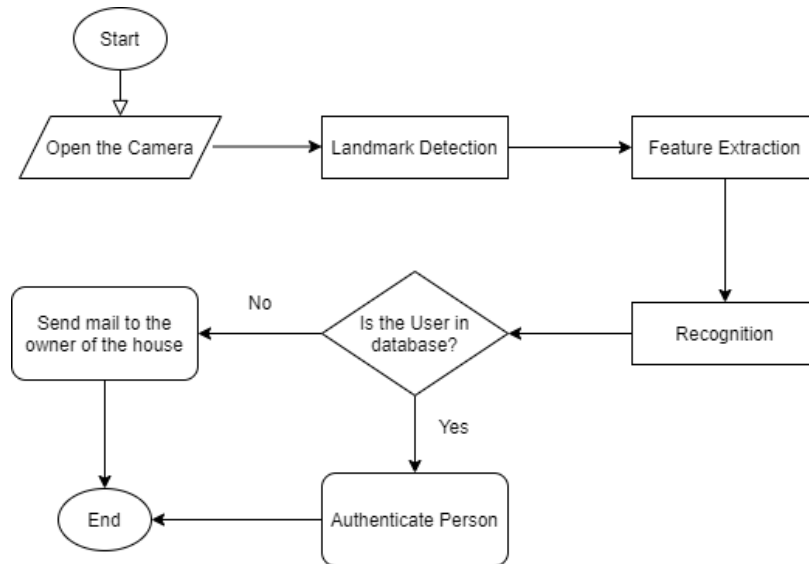
$$y_i(\varphi(x_i)w + b) \geq 1 - \xi_i, \quad \xi \geq 0, \quad i = 1, \dots, n, \quad C > 0 \tag{3}$$

where  $\varphi$  is a function mapping the input data to feature space,  $\xi_i$  is slack variable and  $C$  is the trade-off parameter between the error and margin. Then, the discriminant function can be used to predict the label of unknown data using a kernel function  $k(x_i, x)$  by

$$f(x) = \text{sign}(\sum_{i=1}^n \alpha_i y_i k(x_i, x) + b) \tag{4}$$

where  $\alpha_i$  is the Lagrange multiplier.

As a summary, we propose a system consisting of three essential stages, including deep CNN for landmark extraction, six features derived from the landmarks, and machine learning algorithms, in particular SVM, for recognition and authentication. Figure 3 illustrates the proposed system architecture.



**Figure 3.** Proposed system architecture.

## 2.4. Graphical user interface

A graphical user interface is built to merge all the processes for simplicity and ease of use. For this purpose, PyQt5, a set of Python bindings is considered. It consists of over thirty-five extension modules [29] that allow Python to be used as an alternative to C++ for application development on all supported platforms, including iOS and Android. The interface consists of three main pages. The first page allows the user to create a dataset. The camera is deployed in the middle captures images and a new folder is created with a hundred cropped gray-level face photos that are taken. In the second page, the model can be trained by selecting a machine learning algorithm with the users added to the database, later to be used in the authentication. The classification accuracy, confusion matrix, and classification report can be observed in the lower right corner. If the model is trained as desired without overfitting or underfitting, the model can be saved as shown in Figure 4.



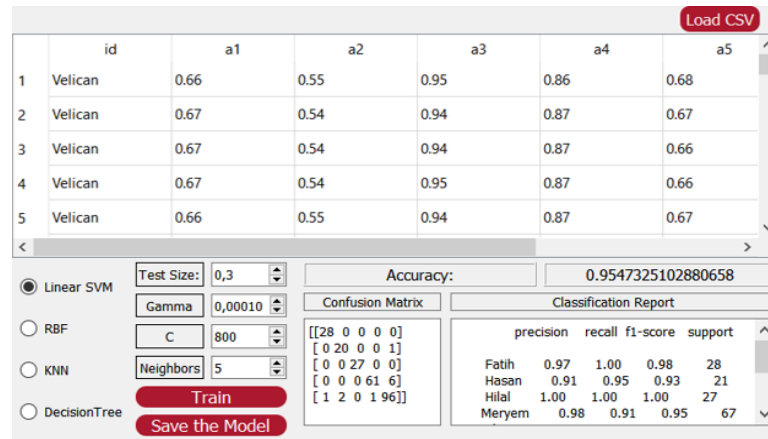


Figure 4. Training page.

The third page is designed for authentication as an automatic door lock. As shown in Figure 5, there is a live camera view on the left, and there is a keypad and a menu on the right, where the trained machine learning algorithm can be selected. If the user is known and recognized as a valid user, the text “Welcome User” will be displayed. If the user is not in the database and is accepted as an intruder, the text “Access Denied” will appear in the text box and the photo of the person trying to log in is sent to the e-mail address of the host, with the time and date information included.

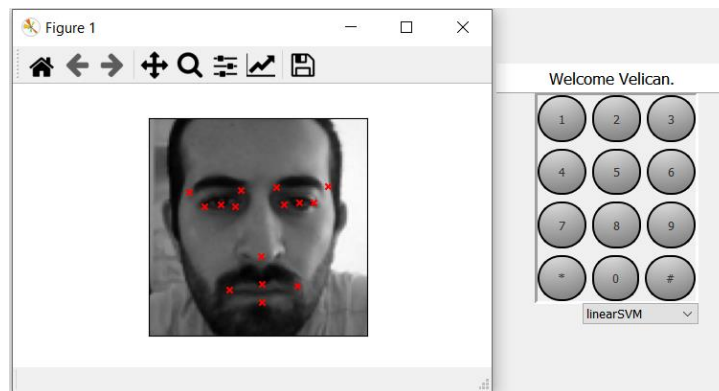
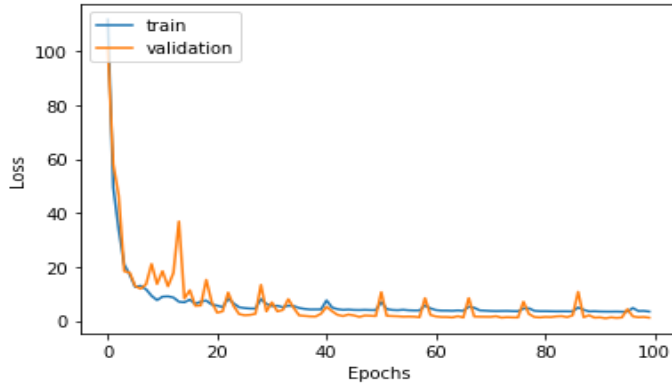


Figure 5. Authentication page.

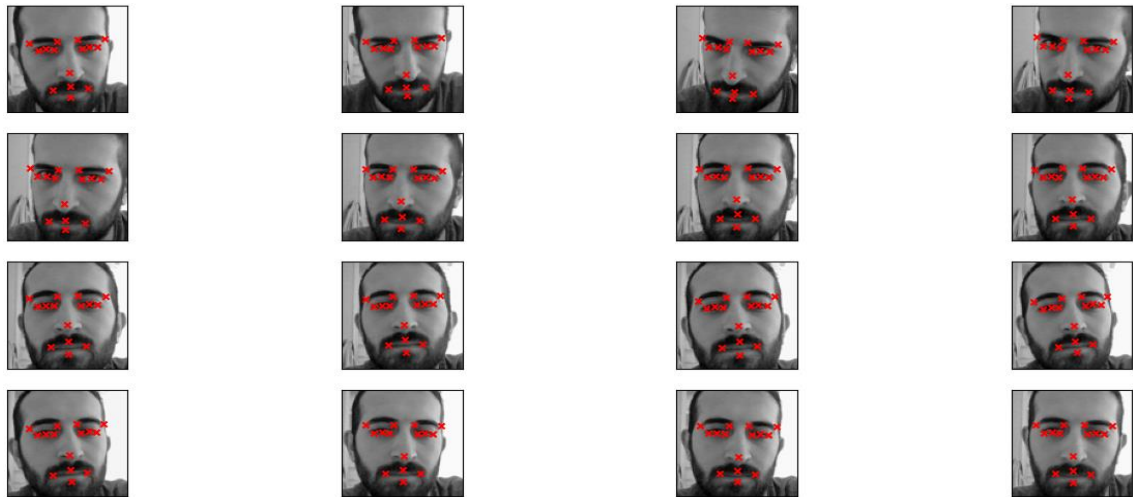
### 3. Results

The deep CNN experiments were made with NVIDIA Tesla K80 Graphics Processing Unit on the Google Colaboratory cloud system. The deep CNN model used in landmark detection was trained by monitoring the mean absolute error and accuracy values. Loss functions are used to determine how well an algorithm matches the data it is trained on. The difference between the anticipated and actual values is used to calculate the loss. The loss function will generate a very big number if the projected values are distant from the actual values. The basic goal of a learning model is to change the weight vector values using different optimization approaches, such as backpropagation in neural networks, to minimize the loss function's value with regard to the model's parameters. At the end of the 100 epoch, the best loss values in the training set were 2.779 while 1.1952 in the validation set, respectively. Figure 6 shows the training and validation loss variations, and it is seen that 100 epochs were enough for the models to converge.



**Figure 6.** Loss for training and validation.

In the tests, the key points from wide angles were detected with high accuracy even with the moving head and obtained consistent results. This model, which is quite ahead of other models, can make consistent predictions in changing conditions. The importance of consistency of results is vital because the landmarks that do not change in the same pose would be helpful in the face recognition part. The test results of the trained landmark model are shown in Figure 7.



**Figure 7.** Landmark model test results.

The proposed system is developed and tested in Python Software, on an eight-core Intel i5 2.50 GHz computer on a database consisting of images of five people and forty-nine test photos with a resolution of 200x200 pixels. All the test photos were taken at a distance of fifty-five cm from the camera and in the same pose. Five of the results from forty-nine test photos independent of the training data are wrongly classified. Therefore, the recognition rate is obtained as 89.79% with a linear SVM kernel.

**4. Conclusions**

Artificial intelligence, machine learning, and deep learning started to change our lives day by day and shape our future. This new ecosystem touches every aspect of our lives, disrupts our habits, and makes our work easier. Facial recognition systems are becoming more important as they become a part of biometric systems, where a physical need such as a key is ended, time is saved and security is increased by providing instant information to the host.

The main purpose of this work is to build and implement a facial recognition system to be used in automatic door unlocking. Face detection and recognition procedures were used to determine the landmarks of the face by the deep learning model. The biometric data was obtained by using the distances and ratios between the landmarks, and the authentication was carried out by the recognition process performed by the machine learning model. As facial expression, imaging condition, and poor illumination can affect the proposed system, augmentation of data to combat these conditions were proposed. The realization of the recognition was processed in a very short time and the rapid provision of security constitute the advantages of the system. On the other hand, misrecognition of

the face due to different pose angles was found as the major disadvantage. The solution was to take an image at the same distance and at the same angle. In this way, the system has achieved an accuracy of 89.79%.

In the future this study might be extended to be used in banks, in ATM's, for identification of users, or in factories for workers' entry pass as of today most of the working places still use signature or card based systems. Moreover, this system can be set out to be used in classrooms for attendance control with crowd analysis.

### Declaration of Interest

The authors declare that there is no conflict of interest.

### References

- [1] A. K. Jain, A. Ross, and S. Pankanti, "Biometrics: A tool for information security," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 125-143, June 2006.
- [2] R. Spolaor, Q. Li, M. Monaro, M. Conti, L. Gamberini, and G. Sartori, "Biometric authentication methods on smartphones: A survey," *PsychNology Journal*, vol. 14, no. 2-3, pp. 87-98, 2016.
- [3] S. B. Thorat, S. K. Nayak, and J. P. Dandale, "Facial recognition technology: An analysis with scope in India," arXiv preprint, arXiv:1005.4263, 2010.
- [4] C. A. Hansen, *Face Recognition*, Institute for Computer Science University of Tromso, Norway, 2009.
- [5] A. K. Jain and S. Z. Li, *Handbook of Face Recognition*, vol. 1, New York: Springer, 2011.
- [6] Wikipedia, *The Free Encyclopedia*, [online] Available at: [https://en.wikipedia.org/wiki/Facial\\_recognition\\_system](https://en.wikipedia.org/wiki/Facial_recognition_system), [Accessed: June 21, 2021].
- [7] L. Li, X. Mu, S. Li, and H. Peng, "A review of face recognition technology," *IEEE Access*, vol. 8, pp. 139110-139120, 2020.
- [8] C. Li, R. Wang, J. Li, and L. Fei, "Face detection based on YOLOv3," *Recent Trends in Intelligent Computing, Communication and Devices*, vol. 1006, pp. 277-284, 2020.
- [9] C. Gürel, *Development of a Face Recognition System*, Master of Science Thesis, Atılım University, 2011.
- [10] M. Turk, M and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71-86, 1991.
- [11] R. E. Schapire, *Explaining Adaboost*, In: *Empirical Inference*, Springer, Berlin, Heidelberg, pp. 37-52, 2013.
- [12] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23-38, 1998.
- [13] W. S. Noble, "What is a support vector machine?," *Nature Biotechnology*, vol. 24, no. 12, pp. 1565-1567, 2006.
- [14] C. Liu and H. Wechsler, "Independent component analysis of Gabor features for face recognition," *IEEE Transactions on Neural Networks*, vol. 14, pp. 919-928, 2003.
- [15] A. H. Boualleg, C. Bencheriet, and H. Tebbikh, "Automatic face recognition using neural network-PCA," In *Proc. 2nd Information and Communication Technologies (ICTTA)*, Damascus, Syria, 2006, pp. 1920-1925.
- [16] Y. Song, Y. Kim, U. Chang, and H. B. Kwon, "Face recognition robust to left-right shadows facial symmetry," *Pattern Recognition*, vol. 39, pp. 1542-1545, 2006.
- [17] P. S. Prasad, R. Pathak, V. K. Gunjan, and H. V. R. Rao, "Deep learning based representation for face recognition," *ICCCE 2019, Lecture Notes in Electrical Engineering*, vol. 570, Springer, Singapore, 2020, pp. 419-424.
- [18] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," In X. Xie, M. W. Jones, and G. K. L. Tam, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, 2015, pp. 41.1-41.12.
- [19] X. Wu, R. He, Z. Sun, and T. Tan, "A light CNN for deep face representation with noisy labels," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2884-2896, 2018.
- [20] W. J. Baddar, J. Son, D. H. Kim, S. T. Kim, and Y. M. Ro, "A deep facial landmarks detection with facial contour and facial components constraint," In *IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3209-3213.
- [21] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof, "Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization," In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 2144-2151.
- [22] N. Agarwal, A. Krohn-Grimberghe, and R. Vyas, "Facial key points detection using deep convolutional neural network-NaimishNet," arXiv preprint, arXiv:1710.00977, 2017.
- [23] Y. Bengio, "Facial keypoints detection," [Online] Available at: <https://www.kaggle.com/c/facial-keypoints-detection>, [Accessed: August 21, 2021].
- [24] E. P. Prokopoulos, I. M. Vlastos, V. A. Picavet, T. G. Nolst, R. Thomas, C. Cingi, and P. W. Hellings, "The golden ratio in facial symmetry," *Rhinology*, vol. 51, no. 1, pp. 18-21, 2013.
- [25] P. S. Gaikwad, V. B. Kulkarni, "Face recognition using golden ratio for door access control system," In: Merchant S. N., Warhade K., Adhikari D. (eds), *Advances in Signal and Data Processing, Lecture Notes in Electrical Engineering*, vol. 703, Springer, Singapore, 2021, pp. 209-231.
- [26] R. Feng and B. Prabhakaran, "A novel method for post-surgery face recognition using sum of facial parts recognition," *IEEE Winter Conference on Applications of Computer Vision*, 2014, pp. 1082-1089.
- [27] T. Dietterich, "Overfitting and undercomputing in machine learning," *ACM Computing Surveys*, vol. 27, no. 3, pp. 326-327, 1995.
- [28] N. Cristianini and J. S. Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press, 2000.
- [29] Python bindings for the Qt cross platform application toolkit, [online] <https://pypi.org/project/PyQt5/>, [Accessed: June 27, 2021].

# A Novel Approach for Detecting Defective Expressions in Turkish

Atilla SUNCAK <sup>1\*</sup>, Özlem AKTAŞ <sup>2</sup>

<sup>1</sup>Dokuz Eylul University, The Graduate School of Natural and Applied Sciences, Department of Computer Engineering, İzmir 35390, Turkey

<sup>2</sup>Dokuz Eylul University, Faculty of Engineering, Department of Computer Engineering, İzmir 35390, Turkey

## Abstract

The use of machine learning has been increasing rapidly in recent years by being more efficient in comparison to rule-based techniques. However, NLP (Natural Language Processing) operations generally require language specific solutions, especially semantic problems. Therefore, deep learning techniques are the best approach for detecting ambiguities in Turkish sentences as they do not need rule-based code implementations. Embedding word vectors are the vectorial visualizations of texts and are beneficial to analyze the word relationships in terms of semantics. In this study, CNN (Convolutional Neural Network) model is proposed to detect defective semantic expressions in Turkish sentences, and the accuracy results of the model are decided to be analyzed. This study makes a crucial contribution for Turkish in terms of semantic analysis and for further related performances.

**Keywords:** *Conv1D, defective expressions, deep learning, NLP, semantic ambiguities, Turkish*

## 1. Introduction

Language is the most important tool in terms of narrating our senses and thoughts to others. However, the sentences we are forming during narrating those compositions, it is crucial that those sentences must be open and clear, understandable, have no unnecessary components, away from conflicts, and compatible with the grammar issues in terms of semantics. When a sentence has one or some of the characteristics aforementioned above, then that sentence has semantic ambiguity, which is called 'defective expression' that leads to communication problems.

In the Turkish language, defective expressions are separated into two parts grammatically which are semantic and morphologic defective expressions. Defective expressions are generally caused by missing elements of the sentences, such as subjects and objects. What is more, they may be occurred due to misuse of some suffixes or conjunction words. In this study, we mainly focused on defective semantic expressions. In the Turkish language, there are seven different types of semantic ambiguities named as follows:

- Use of the unnecessary word
- Use of contrast words in terms of meaning
- Use of the wrong word in terms of meaning
- Use of the word in the wrong place
- Use of the wrong idiom in terms of meaning
- Ambiguity in meaning
- Erroneous in word order and logic

As defined above, semantic ambiguities are accurate or correctly formed sentences in terms of grammar. However, they generally occur due to misuse of suffixes, conflicted words, or using unnecessary additional words. This study suggests an approach to detect those kinds of defective expressions using deep learning techniques.

First of all, we created a dataset that consists of 9700 Turkish sentences that are tagged as positive (not-ambiguous) and negative (ambiguous), explained in Chapter 3 in details. However, that amount of data is inadequate for the model to train. Therefore the dataset was augmented up to almost 30,000 sentences by using the Turkish Synonym Dictionary [1]. After preprocessing the data, we created a corpus embedding Turkish word vectors from this dataset using the word2vec technique [2]. This is because recurrent neural network (RNN) models perform more accurately with word vectors than the sentences themselves. As for the code implementation, the python programming language is used with the other essential libraries such as Keras, Tensorflow, etc.

Convolutional neural networks (CNN) utilize layers with convolving filters applied to local features [3]. Despite being invented for computer vision in the first place, CNN models have been proven to be effective for NLP operations and have accomplished crucial results in semantic parsing [4], search query retrieval [5], sentence modeling [6], and other traditional NLP tasks [7].

\*Corresponding author e-mail address: [atillasuncak@kastamonu.edu.tr](mailto:atillasuncak@kastamonu.edu.tr)

In this study, we will discuss 1 – dimension form of the CNN (Conv1D) approach to detect semantic ambiguities because text data are the word sequences that only have 1-dimension. As for the flow of the algorithm, each sentence of the dataset is transformed into vectors using the embedding word corpus. After that, the data divided into two parts as training set and test set with their labels. Finally, training set will be used to train the model, and a test set will validate the model to measure the accuracy.

Consequently, the main goal is to combine the semantic knowledge of Turkish with artificial intelligence techniques. Thus, we will be one-step closer to create Turkish WordNet by the contribution of this study.

## 2. Previous Studies

In the study of Ferrari & Esuli [8], the language-specific ambiguities in requirements elicitation have been analyzed. Two different approaches are proposed, which are Language Model Generation and Cross-Domain Term Selection. The main idea is to detect the terms that occur in all domains or specific ones. Ambiguous words in seven different elicitation scenario within five domains of interest have been ranked by ambiguity scores using word embeddings that measure the differences of use of a word and estimate its potential ambiguity. In the evaluation phase, the ambiguity rankings are compared. Even though there are some acceptable accuracy results in a few elicitation, such as 81% or 88%, the approach was not successful enough in terms of performance for several elicitation.

The review study of Bano [9] focuses on ambiguities in the documents of requirement engineering. A mapping approach has been applied that focuses on NLP techniques for ambiguity detection. 174 literature reviews published during 1995 – 2015 have been resulted in the systematic search, and 28 of them have been selected to be analyzed in terms of ambiguities and detection techniques. The result shows that 81% of the techniques detect ambiguities such as Alpino Tools, Wordnet, LOLITA (Large-scale Object-based Linguistic Interactor Translator Analyser), Knowledge Graph and etc. However, the study also addresses a lack of NLP tools and techniques for addressing the ambiguities in requirements.

The empirical study of Hoceini, et al. [10] proposes a technique for disambiguation in no-vowel-Arabic text data. The proposed method is a combination of decision theory and MCDA (Multiple Criteria Decision-Aid) in order to develop a coherent system that integrates contextual data analysis into decision making in case of ambiguity. The used techniques are Probabilistic Hidden Markov Model, N-grams, rule-based linguistic constraints, etc. The main idea is the multi-scenario classification of ambiguity cases in the texts and determine the best performance to reduce the candidate scenarios.

## 3. Methodology

Deep learning is a machine learning technique that helps us train the artificial intelligence model, which will estimate the result with the input data. This technique requires no rule-based parameter of configurations in correspondent to the semantic issues, which means there is no need to implement the code for grammar rules. This state of art technique opened a new vantage point for the researchers, especially in NLP field as the language itself is a living existence, and the semantics of the words and phrases change from generation to generation inevitably. In the following sections, the dataset and corpus formations, Conv1D approach, and the algorithm flow will be explained in detail.

### 3.1. Dataset and Corpus Preparation

#### 3.1.1. Dataset and corpus preparation

Dataset or input data is the most important component for training the deep learning model. Even if the well configured learning model were created, it would never have the optimum accuracy due to the inadequacy of the input data because the model is as successful as its input data which train it.

This study requires a comprehensive dataset to be used for training the model, and we needed thousands of ambiguous and not-ambiguous sentences. However, there exists no dataset study previously collected for this purpose. Thus we had to individually collect all the sentences and mark them whether they have ambiguities or not. First of all, almost 50 different sources from several websites of schools, courses, and even the official exam center of Turkey (OSYM) have been investigated, and as a result of almost 3-month research, 9700 sentences, almost half of which have ambiguities, have been collected. That amount of sentences for an RNN model is insufficient in general. Therefore a data augmentation operation was held by using the Turkish Synonym Dictionary. As a result, the dataset has been augmented from 9700 sentences to 30,000 sentences. Finally,

preprocess operations of the dataset have been performed, such as correction of misspelled words or omitting stop-words and punctuation marks, etc.

Bu davranışımı tehdit olarak algıladığını belirtiyorsun.   POSITIVE
Yaptıklarınla herkesi şaşırtmaya devam ediyorsun.   POSITIVE
Bu sözlerinle beni sinirlendirmek için çalışıyorsun.   NEGATIVE
Sorduğun sorularla konuyu başka bir yere çekmeye çalışıyorsun.   POSITIVE
Evin, bin bir çeşit meyve ağacı ve sebze yetiştiren bir bahçesi var.   NEGATIVE
Diplomalarını alacak öğrenciler salona sırayla giriş yaptılar.   NEGATIVE
Müjdeyi vermek için mutfağa, annesinin yanına heyecanla koştu.   POSITIVE
Konuşmasına başlamadan önce dinleyicilere şöyle bir baktı.   POSITIVE
Eski öğrencilerin de katıldığı büyük bir toplantı düzenlediler.   POSITIVE

**Figure 1.** Samples of input data.

### 3.1.2. Word embeddings as corpus

Word2Vec is a technique that reveals the relationship between words in a sentence by transferring each word into a fixed-size numeric-value vectors called word embeddings. It helps the model to measure the distance relations of the words in the analyzed texts in terms of semantics, and those measured values can be easily visualized. With the help of this technique, we can develop a recommendation system by finding the nearest words to the reference word.

In this study, we performed word2vec approach with 200-dimensions using CBOW (Continuous Bag-of-Words) technique to generate a corpus. The window size, which addresses the context, is specified as five words, which means that while a word's vector is calculated, the surrounding five words before and after have been determined as context. This corpus is used to generate the embedding matrix from the dataset in order to train the models.

### 3.2. Conv1D Approach

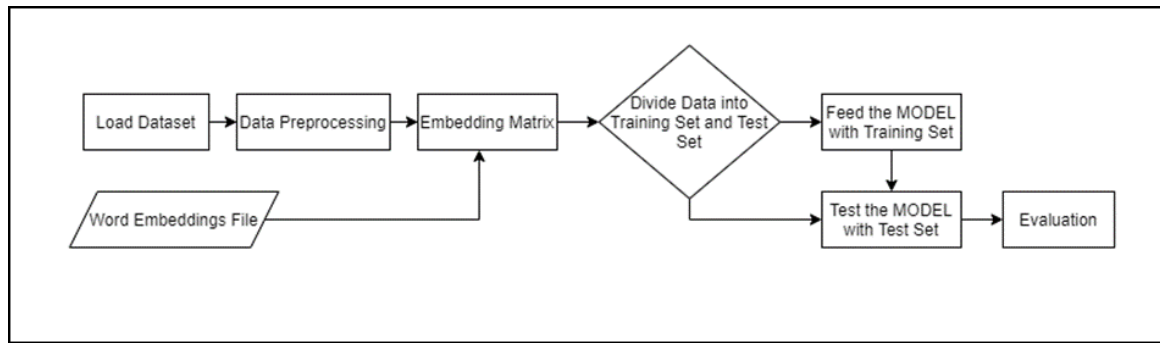
Convolutional Neural Network is generally used in computer vision operations, especially in image classifications where data is processed in the form of 2-dimensions [11]. Text data, on the other hand, have only 1-dimension in the form of the word sequence. Thus Conv1D is implemented due to the suitability for the characteristics of the text data [1] [12]. The reason why Conv1D approach has been chosen is because the model has great performance and success in NLP tasks such as text classifications and sentiment analysis [13] [14], text categorizations [15], and many others [16].

In the experimental phase of this research, several number of convolution layers with the 'relu (Rectified Linear Unit)' activation function, variations of filter values, and kernel sizes have been analyzed and tested. MaxPooling1D is the pooling layer that performs pooling operations. In order to avoid overfitting, the dropout layer, which is adjusted with the value of '0.5', is applied. The Flatten layer is used to make the multidimensional output linear to pass it onto the dense layer, the model functioning layer as a classifier with 'softmax activation' function.

### 3.3. Model Implementation

This study is a text classification approach in terms of semantics that detects Defective Expressions in Turkish sentences. The classification results in two classes; 'POSITIVE,' which means the sentence has no defective expression, and 'NEGATIVE' that is the sentence that has defective expression. The workflow that we implemented starts with the data preprocess, then creating the embedding matrix by using the embedding corpus, and finally train and test operations as seen on Figure 2.





**Figure 2.** The flow diagram of the learning model.

First of all, the input data are preprocessed to be prepared for the classification. Thus the output data will have a new form which makes the dataset understandable for the classification model as input. After that, in order to prepare the Embedding Layer of the model, and embedding matrix is created from each sentence in the dataset by using embedding vector corpus.

In model determination, we implemented the Conv1D model based on the text classification example from Keras Documentation as model baseline [17] [2]. Then we performed some changes on the layers, such as adding Embedding Layer as the first layer that will provide the weights on the model. After that, we performed several value trials on the hyper-parameters of the model, explained in chapter4 in details, in order to determine the best performance.

## 4. Results

### 4.1. Model Preparation

The sentences used as input data in this research to train and test the model have been gathered together from different sources through manual investigation because there has been no such dataset available in the literature. As the result of this investigation, there have been 9710 Turkish sentences, almost half of them have defective expressions. As this number of sentences is insufficient for model training, we performed data augmentation using the Turkish Synonym dictionary. Thus we finally acquired 29756 sentences with labels, which are then split into training for 20829 sentences and testing for 8927 sentences. Before they are processed by the model, we performed data preprocess operations which include punctuation and stop-word omitting, lexical sentence correction and etc.

The empirical model preparation consists of adjusting several values on hyper-parameter such as number of filters, kernel sizes, and pooling sizes. In order to get better performance, we try to focus on testing the hyper-parameters in range of slight increase and decrease of the base values with the fixed parameters such as learning rate of '0,001', dropout layer of '0.5', batch size of '85' and the number of epoch '50 times' using 'MSE (Mean Squared Error)' loss function, 'Softmax' activation function and 'Adam' optimizer. The aforementioned empirical trials have all been performed to determine which scenario resulted the best performance in terms of model accuracy and validation loss.

### 4.2. Evaluation and Discussion

The experimental results of hyper-parameter tunings are listed in Table 1. This table also shows the best performances according to the number of convolutional layers of the model. Experiment 17 (Exp17) is resulted as the best performance that uses three convolutional layers and the same number of kernel sizes and pool sizes.

The next best scenarios after Exp17 are Exp3 that has one convolutional layer, and Exp11 that has two convolutional layers respectively. The first three best experiments are all have the kernel size value(s) of 128. When analyzing the results in terms of the number of convolutional layers; the one-layer and two-layer results show that the model accuracy gets generally higher when increasing the number of filters of the layers. On the other hand, increase in the number of filters did not increase the model accuracy when analyzing the three-layer experiments. As seen in Exp18 and Exp19, even though the same patterns of both kernel sizes and pool sizes are applied with Exp17, their model accuracies performed a clear decrease. In conclusion, the results show that the used patterns of kernel sizes and pool sizes in the best performance of three-layer convolutional model have the optimal values and this model can be used as a solution approach for this research.

**Table 1.** *Experimental results of the model.*

Experiment (E)	Conv1D filters	Kernel sizes	Pooling sizes	Validation accuracy	Validation loss.
Exp1	64	3	2	0.8248	0.1283
Exp2	64	3	3	0.8167	0.1327
<b>Exp3</b>	<b>128</b>	<b>3</b>	<b>2</b>	<b>0.8432</b>	<b>0.1207</b>
Exp4	128	3	3	0.8345	0.1213
Exp5	64, 64	3, 2	2, 2	0.8198	0.1294
Exp6	64, 64	3, 3	2, 2	0.8018	0.1407
Exp7	64, 64	3, 3	2, 3	0.7918	0.1451
Exp8	64, 64	3, 3	3, 2	0.8177	0.1339
Exp9	64, 128	3, 2	2, 2	0.8100	0.1346
Exp10	64, 128	3, 3	3, 2	0.8276	0.1245
<b>Exp11</b>	<b>128, 128</b>	<b>3, 2</b>	<b>2, 2</b>	<b>0.8374</b>	<b>0.1228</b>
Exp12	128, 128	3, 3	3, 2	0.8247	0.1251
Exp13	64, 64, 64	3, 3, 2	3, 2, 2	0.8054	0.1365
Exp14	64, 64, 64	2, 3, 2	3, 2, 2	0.7912	0.1438
Exp15	64, 64, 128	3, 3, 2	3, 2, 2	0.8142	0.1323
Exp16	64, 128, 128	3, 3, 2	3, 2, 2	0.8244	0.1237
<b>Exp17</b>	<b>128, 128, 128</b>	<b>3, 3, 2</b>	<b>3, 2, 2</b>	<b>0.8433</b>	<b>0.1217</b>
Exp18	64, 128, 256	3, 3, 2	3, 2, 2	0.8125	0.1322
Exp19	256, 256, 256	3, 3, 2	3, 2, 2	0.8325	0.1165

## 5. Conclusion

In the light of the results of this research, it is clearly interpreted that Conv1D approach with word embeddings is compatible to use as a model in detecting defective expressions in Turkish sentences. The best performance in terms of model accuracy is adjusted after several experimental tests. This research also showed that an increase in filters generally results in better performance when adjusting the compatible values for kernel size and pool size.

It is also predicted that the tested values of hyper-parameters are in a specific range, thus making further experimental trials by using a wider range of values with this model may result in potential better performances. Yet this study is a huge contribution to Turkish semantic NLP and a source for other researchers who studies in this field.

## Declaration of Interest

As authors, we declare that we have no conflict of interest with anyone related to our work.

## References

- [1] Ö. Aktaş, Ç.C. Birant, B. Aksu and Y. Çebi, "Automated synonym dictionary generation tool for turkish (ASDICT)", Bilig, vol. 65, pp. 47-68, March 2013.
- [2] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," ICLR, 2013.
- [3] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," In Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.
- [4] W. T. Yih, X. He, and C. Meek, "Semantic parsing for single-relation question answering," In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, vol. 2, pp. 643-648, June 2014.
- [5] Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil, "Learning semantic representations using convolutional neural networks for web search," In Proceedings of the 23rd international conference on world wide web, 2014.
- [6] K. Nal, G. Edward, and B. Phil, "A convolutional neural network for modelling sentences," Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, vol. 1, pp. 655-665, 2014.
- [7] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," Journal of Machine Learning Research, vol. 12, pp. 2493-2537, 2011.
- [8] F. Alessio, and A. Esuli, "An NLP approach for cross-domain ambiguity detection in requirements engineering." Automated Software Engineering, pp. 559-598, 2019.

- [9] M. Bano, "Addressing the challenges of requirements ambiguity: a review of empirical literature," IEEE Fifth International Workshop on Empirical Requirements Engineering (EmpiRE), pp. 21-24, 2015.
- [10] Y. Hoceini, M. A. Cheragui, and M. Abbas, "Towards a new approach for disambiguation in NLP by multiple criterion decision-aid," Prague Bull. Math. Linguistics, pp. 19-32, 2011.
- [11] M. Hussain, J.J. Bird, and D.R. Faria. "A study on cnn transfer learning for image classification," 18th Annual UK Workshop on Computational Intelligence, UKCI, 2018.
- [12] D. S. Dewantara, I. Budi, and M. O. Ibrohim, "3218IR at semEval-2020 task 11: conv1D and word embedding in propaganda span identification at news articles," In Proceedings of the Fourteenth Workshop on Semantic Evaluation, pp. 1716-1721, 2020.
- [13] K. Yoon, "Convolutional neural networks for sentence classification," Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, '08, 2014
  
- [14] K. Radhika, K. R. Bindu, and P. Latha, "A text classification model using convolution neural network and recurrent neural network," International Journal of Pure and Applied Mathematics, '01, 2018, pp. 1549-1554.
- [15] H. Mark, L. Irene, K. Spyro, and S. Toyotaro, "Medical text classification using convolutional neural networks," Studies in Health Technology and Informatics, '04, 2017.
- [16] M. Kyoung Hyun, P. Jaesun, J. Myeongjun, and K. Pilsung, "Text classification based on convolutional neural network with word and character level," Journal of the Korean Institute of Industrial Engineers, '06, 2018, pp. 180-188.
- [17] F. Chollet, "The sequential model," keras.io, April 12, 2020. [Online]. Available: [https://keras.io/guides/sequential\\_model](https://keras.io/guides/sequential_model). [Accessed July 13, 2020].

# Development of Android-based Internet of Things Application for Data Tracking in Smart Marble Factories

Caglar GURKAN <sup>1,\*</sup>

<sup>1</sup> Izmir Katip Celebi University, Graduate School of Natural and Applied Sciences, Electrical and Electronics Engineering, Turkey

## Abstract

Internet of Things is one of the most important technological components of Industry 4.0. Internet of Things technology has become an even more critical research subject by using it with cloud computing technology that is constantly evolving. In this study, it is aimed to contribute to digital transformation in the marble factories. Therefore, an Android-based Internet of Things application has been developed for data tracking during the drying stage of the marble. NodeMCU, DHT11 temperature and humidity sensor, Firebase Real-Time Database have been used to build the data tracking system. This system has been supported by the Android-based mobile application created in the Android Studio Integrated Development Environment using Java programming language. As a result, losses during the production stage can be minimized by ensuring data tracking during the marble's drying phase.

**Keywords:** *Android; industry 4.0; internet of things; IoT; marble drying system; smart factory.*

## 1. Introduction

The term of the industry is a field of study that has experienced revolution many times today. In this context, the methods and production processes used in industry have been developing day by day. Before the 21st century, there happened three major industrial revolutions in the World [1]. The term Fourth Industrial Revolution (Industry 4.0) was proposed by the Germans at Hannover Fair in 2011 [2]. The main objective of Industry 4.0 technological developments is to achieve advanced industrial automation and higher productivity. This industrial innovation has enabled the build of smart factories that emerged with the Internet of Things (IoT) nowadays. In addition, academic and industrial trends occur around components of smart factories such as cyber-physical systems, IoT, cloud computing, additive manufacturing (3D printing), cybersecurity, autonomous robots, system integration, simulation, virtual reality (VR), augmented reality (AR), big data, data analytics, and artificial intelligence [3]. IoT, one of the most important components of smart factories, is defined as intelligent and self-structuring objects used to ensure that factories' basic equipment and resources communicate with each other. In addition, IoT technology can transfer data. Also, the IoT has been presented as a term used to observe production resources on a cloud server. As a result of data analysis, the decision support mechanism created is used to make production planning for smart factories, follow up the service and maintenance needs of machinery and equipment, and automatically transmit this need and provide the high-quality control process [4-6].

In this paper, temperature and humidity with together time information have been obtained purpose of data tracking in the drying stage of marble in smart marble factories. The NodeMCU embedded system development board, including a built-in WiFi module and the DHT11 temperature and humidity sensor, has been used. Firebase's real-time database has been used in the cloud computing part. An Android-based mobile application has supported the system. The mobile application has been developed in Android Studio integrated development environment (IDE) using Java programming language. As a result, a real-time temperature and humidity sensor data tracking system supported by Android has been developed.

The rest of the submitted paper is organized as follows. In Section 2, the related works on IoT applications have been reviewed. Then, in Section 3, the utilized methodologies have been presented. In Section 4, the obtained results have been reported. Finally, in Section 5, the concluding remarks, including future works, have been presented.

## 2. Related works

Research in the literature on IoT applications is presented in this section.

Terroso-Saenz et al. proposed an IoT application for the control and monitoring of energy consumption of apartments. In the study, sensor data such as room temperature and get instant energy consumption of apartments were obtained [7]. Rashed et al. proposed a wearable embedded system sensor design that contains the WiFi module to track patients' heart rhythm, body temperature, and motion status. Obtained data from the sensors was transferred to the cloud-based server [8]. Taştan created smart irrigation and remote monitoring system for

\*Corresponding author e-mail address: [caglar.gurkan@outlook.com](mailto:caglar.gurkan@outlook.com)

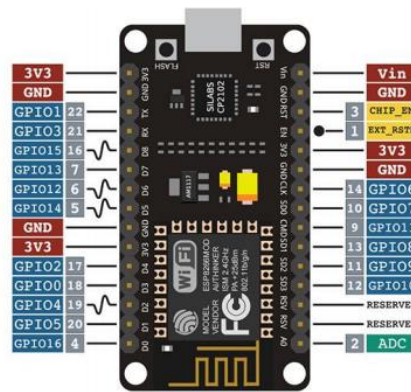
agriculture and landscaping by using DHT22 temperature and humidity sensor and HL-69 soil hygrometer moisture sensor together with NodeMCU [9]. Kumar et al. have tried to take data of a smart manufacturing factory and have done predictive data analysis for energy management based on historical information [10]. Hamilton proposed an IoT-based decision support system in cyber-physical smart manufacturing using wireless sensor networks. It supported this system with a real-time big data analytics algorithm [11].

### 3. Methodology

In this section, the utilized methodologies in the study are presented, including hardware and software details.

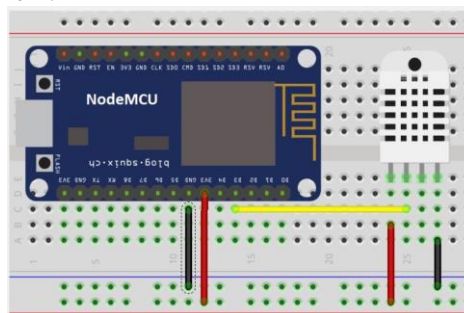
NodeMCU, DHT11 temperature and humidity sensor, and Firebase Real-Time Database have been used for data tracking with the Android-based application during the drying stage of marble in smart marble factories.

NodeMCU is an embedded system development board with a 32-bit processor also that contains the WiFi module. NodeMCU is often used in IoT applications with its low price advantage, low power consumption, and compact card design. NodeMCU development board design and pin diagram are shown in Figure 1. The 32-bit Tensilica L106 processor exists on NodeMCU. The operating frequency of this processor is 80/160 MHz. In addition, NodeMCU has 17 Input/Output pins and 1 ADC pin. The operating voltage is in the range of 3.0~3.6 V, while the operating current is in the range of 12~200 mA. NodeMCU's total flash memory is 4 MB. NodeMCU has a WiFi module and communication system based on the IEEE 802.11 b/g/n network standard. The sleep mode current is less than 10  $\mu$ A, while the standby mode current is less than 10 mA.



**Figure 1.** NodeMCU development board design and pin diagram [9].

The DHT11 temperature and humidity sensor, which contains an 8-bit microprocessor, have been used to obtain real-time humidity and temperature information in the study. For this sensor, which can receive data in the temperature value range of 0~50  $^{\circ}$ C, it is stated that there may be an error of  $\pm 2$   $^{\circ}$ C at the temperature value and  $\pm 5\%$  at the humidity value. Also, the operating voltage and current of this sensor are a range of 3.0~5.5 V and 0.5~2.5 mA, respectively. The DHT11 temperature and humidity sensor and NodeMCU development board connection scheme are shown in Figure 2.



**Figure 2.** The DHT11 temperature and humidity sensor and NodeMCU development board connection scheme [12].

Temperature and humidity information has been obtained from the DHT11 sensor, while the Network Time Protocol (NTP) server has been used for the time information of when this sensor data have been received. Then, all this obtained real-time temperature and humidity data have been transferred from the Arduino Studio IDE to the real-time database using the https address of the Firebase Real-Time Database and the Firebase database identification password. After, this sensor data have been transferred to the mobile application developed in the Android Studio IDE in Java programming language using database query methods.

#### 4. Results and discussion

This section presents the obtained results, including the layout and usage details of an Android-based mobile application.

Real-time temperature and humidity data have been received from two different workstations. These workstations have been controlled by creating two CardView designs, including hints as workstation 1 and workstation 2 in the user interface of the Android-based mobile application, as shown in Figure 3 (a). The workstation 1 and workstation 2 CardViews have been used inside ScrollView purpose of adapting different phone screen sizes. If these CardView designs are clicked, new user interfaces are passed, which are shown in Figure 3 (b) and Figure 3 (c). The CardView designs consist of three TextViews, which include temperature, humidity, and time information. The CardViews, which include TextViews, have been used inside RecyclerView, so the CardView has been designed as an adapter in a different XML file. Finally, the temperature, humidity, and time information have been transferred to these TextViews inside CardView using database query methods.

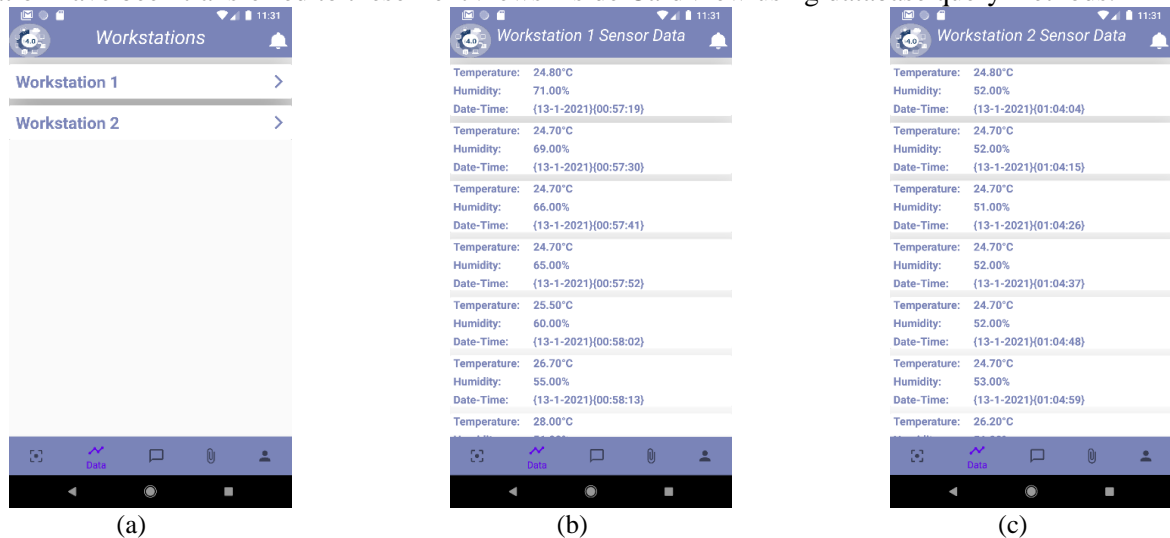


Figure 3. (a) Location of Workstations, (b) Workstation 1 sensor data, (c) Workstation 2 sensor data.

#### 5. Conclusion and future works

Nowadays, with the development of internet technology, the number of applications using this technology are incessantly increased. These applications provide serious time savings to their users. Examples of these applications include e-commerce, online news sites, social media, and the industry. In this study, a contribution has been to the use of the internet in the industry, i.e. IoT. Real-time temperature and humidity data have been received from two different workstations using NodeMCU, DHT11 temperature and humidity sensor, Firebase Real-Time Database, NTP server, and Android mobile technology. As a result, an end-to-end structure has been created for data tracking during the drying stage of the marble in smart marble factories. To do in future studies can be summarized as follows. First, the number of workstations can be increased. Increasing the number of workstations allows the end-user to achieve higher data tracking system performance. New sensor structures, such as DHT22 and SI7021 temperature and humidity sensors, can be used. DHT22 and SI7021 sensor structures operate in a wider temperature and humidity range with high sensitivity than the DHT11 sensor. In this way, more exact sensor data can be obtained. IOS and web versions of the Android-based mobile application can be developed. Consequently, the developed IoT system becomes accessible for more end users.

#### Declaration of Interest

The authors declare that there is no conflict of interest.

#### Acknowledgments

This work has been supported by the Izmir Katip Celebi University of Scientific Research Projects Coordinatorship, Grand no: 2021-TYL-FEBE-0009. Additionally, I would like to thank my advisor, Assoc. Prof. Dr. Merih Palandöken for his guidance and support throughout the study.

## References

- [1] K. Çelik, S. Güteryüz, and H. Özköse, “4. Endüstri Devrimine Kuramsal Bakış,” *Avrasya Sos. ve Ekon. Araştırmaları Derg.*, vol. 5, no. 9, pp. 86–95, 2018.
- [2] M. A. K. Bahrin, M. F. Othman, N. H. N. Azli, and M. F. Talib, “Industry 4.0: a review on industrial automation and robotic,” *Jurnal Teknologi*, vol. 78, no. 6–13. Penerbit UTM Press, pp. 137–143, Jun. 28, 2016, doi: 10.11113/jt.v78.9285.
- [3] N. S. Pamuk and M. Soysal, “Yeni Sanayi Devrimi Üzerine Bir İnceleme,” *Veriml. Derg.*, no. 1, pp. 41–66, 2018.
- [4] M. Hanefi Calp, E. Bahçekapılı, and M. Berigel, “Endüstri 4.0 Kapsamında Akıllı Fabrikaların İncelenmesi,” *IMISC*, Jan. 2019. doi: 10.6084/M9.FIGSHARE.7564886.V1.
- [5] D. Zhang, J. Wan, C. H. Hsu, and A. Rayes, “Industrial technologies and applications for the Internet of Things,” *Computer Networks*, vol. 101. pp. 1–4, 2016, doi: 10.1016/j.comnet.2016.02.019.
- [6] Z. Shu, J. Wan, D. Zhang, and D. Li, “Cloud-Integrated Cyber-Physical Systems for Complex Industrial Applications,” *Mob. Networks Appl.*, vol. 21, no. 5, pp. 865–878, 2016, doi: 10.1007/s11036-015-0664-6.
- [7] F. Terroso-Saenz, A. González-Vidal, A. P. Ramallo-González, and A. F. Skarmeta, “An open IoT platform for the management and analysis of energy data,” *Futur. Gener. Comput. Syst.*, vol. 92, pp. 1066–1079, 2019, doi: 10.1016/j.future.2017.08.046.
- [8] A. Rashed *et al.*, “Integrated IoT medical platform for remote healthcare and assisted living,” in *2017 Proceedings of the Japan-Africa Conference on Electronics, Communications, and Computers, JAC-ECC 2017*, 2018, vol. 2018-Janua, pp. 160–163, doi: 10.1109/JEC-ECC.2017.8305801.
- [9] M. Taştan, “Nesnelerin İnterneti Tabanlı Akıllı Sulama ve Uzaktan İzleme Sistemi,” *Eur. J. Sci. Technol.*, pp. 229–236, 2019, doi: 10.31590/ejosat.525149.
- [10] M. Kumar, V. M. Shenbagaraman, R. N. Shaw, and A. Ghosh, “Predictive data analysis for energy management of a smart factory leading to sustainability,” in *Lecture Notes in Electrical Engineering*, 2021, vol. 661, pp. 765–773, doi: 10.1007/978-981-15-4692-1\_58.
- [11] S. Hamilton, “Real-Time Big Data Analytics, Sustainable Industry 4.0 Wireless Networks, and Internet of Things-based Decision Support Systems in Cyber-Physical Smart Manufacturing,” *Econ. Manag. Financ. Mark.*, vol. 16, no. 2, pp. 84–94, 2021, doi: 10.22381/emfm16220215.
- [12] “NodeMcu DHT11 & MQTT « osoyoo.com.” <https://osoyoo.com/2018/01/17/nodemcu-dht11-mqtt/> (accessed Jun. 15, 2021).



# A Flower Status Tracker and Self Irrigation System (FloTIS)

Rumeysa KESKİN, Furkan GÜNEY, M. Erdal ÖZBEK\*

İzmir Kâtip Çelebi University, Faculty of Engineering and Architecture,  
Department of Electrical and Electronics Engineering, İzmir, Turkey

## Abstract

The Internet of Things (IoT) provides solutions to many daily life problems. Smartphones with user-friendly applications make use of artificial intelligence solutions offered by deep learning techniques. In this work, we provide a sustainable solution to automatically monitor and control the irrigation process for detected flowers by combining deep learning and IoT techniques. The proposed flower status tracker and self-irrigation system (FloTIS) is implemented using a cloud-based server and an Android-based application to control the status of the flower which is being monitored by the local sensor devices. The system detects changes in the moisture of the soil and provides necessary irrigation for the flower. In order to optimize the water consumption, different classification algorithms are tested. The performance comparisons of similar works for example flower case denoted higher accuracy scores. Then the best generated deep learning model is deployed into the smartphone application that detects the flower type in order to determine the amount of water required for the daily irrigation for each type of flower. In this way, the system monitors water content in the soil and performs smart utilization of water while acknowledging the user.

**Keywords:** *Automatic irrigation system; deep learning; IoT.*

## 1. Introduction

There are many types of plants that are very susceptible to water, temperature, soil pH, etc., for their good growth health and overall development. One of the factors to be considered essential to the health of the plant is water and each plant grows healthy in different ranges of this factor. The lack of water or excessive water can make plants dry or rotten which results in insufficient growth or death. It is hardly feasible that everyone knows the appropriate information about the growth of plants for their needs. Even with just the correct execution of the irrigation, the factors of time, water, and a lot of money can be saved [1]. Therefore, the automatic irrigation system has become very convenient due to insufficient water resources and time requirements.

The efficiency of the systems can be increased through the use of machines where every object or “thing” can be connected by the Internet of Things (IoT) concept. The IoT-based network enables the communication between machine-to-machine (M2M), that is the network of the embedded sensors, actuators, and electronic devices for the purpose of collecting and exchanging data with other devices and systems over the Internet [2]. There are several related pieces of research that have been discussed regarding automatic plant irrigation systems developed and implemented with IoT technology. For example, an IoT-based irrigation system uses a regression algorithm to optimize and control the process of water consumption [3]. Another research has implemented a Web-based plant monitoring application that helps users to monitor the plant status of temperature and humidity measurements with a wireless sensor network by sending a short message (i.e., SMS) to the user's smartphone of any unexpected changes in measurements [4].

On the other hand, any irrigation system requires the identification of the plant under investigation. Recent advances in machine learning technology help to develop automatic identification systems by the acquisition of millions of digital plant photos with smartphones, leading to mobile-based automatic plant identification applications. There have been many approaches developed to identify different flower species considering the main features of flowers such as color, texture, and shape. The support vector machine (SVM) has been widely used as the most effective machine learning-based classifier in image classification [5]. However, hand-crafted traditional discriminative features such as histogram of oriented gradients [6], scale-invariant feature transform [7] used in the SVM classification method cannot be easily applied to numerous flower classes. Moreover, the classification method applied to a single flower dataset is not guaranteed to achieve the same performance on a different flower dataset due to the complexity of the problem. Besides, the extracted features might not be generalizable to other flower images with changing conditions such as rotation, scaling, and illumination.

The deep learning methods, especially convolutional neural network (CNN) architectures have recently gained preference on image classification due to their superior performance in accuracy compared to the classical machine learning methods which rely on mostly hand-crafted methods. The CNN takes the intensities of pixels

\*Corresponding author e-mail address: [merdal.ozbek@ikcu.edu.tr](mailto:merdal.ozbek@ikcu.edu.tr)

from an input image and gradually adjusts the parameters (model weights) during training until optimizing the algorithm in order to neural network make a prediction as closely as possible. The detection of a flower from its image depends on the efficient models of the deep learning architectures based on a CNN. There have been studies to classify from various images [8, 9], and various types of plants including flowers [10, 11]. In those works, efficient deep learning architectures like Inception V3 [12], MobileNet [13], and ResNet-50 [14] have been used to obtain higher recognition rates [9, 11, 15]. Moreover, the information of the model obtained from one type of flower can also be transferred to another by transfer learning. Thus the present study uses CNNs along with transfer learning to efficiently recognize flower species in real-time. In order to implement the control and traceability of the system, and display real-time information to the users, the sensors reading the moisture level of the soil are integrated to an Android application on a smartphone.

The remainder of the paper is organized as follows: Section 2 gives the background information and mentions the key concepts. Section 3 explains the system architecture and implementation of the proposed flower recognition-based flower status tracker and self-irrigation system (FloTIS). Section 4 reports the accuracy performance of the models for flower classification and demonstrates the irrigation system integrated with the designed Android smartphone application. Section 5 summarizes and concludes the paper.

## 2. Background

The proposed system combines the concepts of deep learning to detect the flower type, and IoT to transfer the flower data in order to check the status and control the automatic irrigation of the flower via smartphones.

The deep learning models for image classification utilizes mostly the CNN architectures. A typical CNN is composed of three stages [16]. In the first stage, several convolutions are performed in parallel to produce a set of linear activations. The so-called convolution layer uses filters that perform convolution operations and outputs feature maps. Each linear activation is run through a nonlinear activation function in the following stage. In the third stage, a pooling function is used to modify the output of the layer. The pooling is essentially a down-sampling operation where commonly used values of the maximum or the average are taken. The concatenation of layers provides to work with data that has a grid-structured topology and scales to large sizes. This is followed by a final fully connected layer that outputs the desired classification class scores. This overall approach has been the most successful on two-dimensional cases, i.e., on images.

The CNN model is first introduced to perform recognition of hand-written digits from images and it has shown significant performance improvement compared to earlier state-of-the-art machine learning techniques [17]. Later it has been extended by many studies and many architectures have been proposed with an increased number of layers that can achieve lower error rates [9, 18].

On the other hand, IoT aims to create a better environment for humanity, where objects around us are organized according to our needs and act without explicit instructions. Therefore, the fundamental promise of IoT is to monitor and control the connected things [19]. This is mainly built up by sensors to collect information, identifiers to identify the source of data, software to analyze the data, and most importantly the Internet connectivity to communicate and notify. The explosion of smart and mobile devices further boosted the usability of IoT-based solutions for everyday life problems.

Initially, the dominant technology behind the IoT was Radio-Frequency Identification (RFID). Today with further technological achievements, a diverse set of architectures and enabling technologies drive the standardization of IoT in several application domains. There are several surveys that focus on technologies such as wireless sensor networks, communication protocols, cloud platforms, fog computing [20, 21, 22]. Concepts combined with technology are reflected on the applications name such as smart home, smart city, or smart farming. Particularly for agriculture, sensing for soil moisture and nutrients, controlling water usage for plant growth, and determining custom fertilizer are some simple uses of IoT [23].

A natural solution to using these smart systems is employing smartphones. Simple applications in smartphones aid in monitoring and controlling the objects together with the underlying technologies of IoT. A common operating system in smartphones is Android. The software tool generating those applications helps to handle the high computation for deep learning classification using the cloud services, necessary communication protocols to access and control the objects as well as notify users at any step of the usage.

## 3. System Architecture and Implementation

This section describes the theoretical foundation and system implementation of our proposed system in detail.

### 3.1. Flower Recognition System

In order to classify the type of flower and determine the required water based on the soil moisture level, the first step is image classification. Image classification is the pioneering work that boosted the deep learning architectures based on CNNs after AlexNet [24]. Today, most of the deep learning architectures utilize the pre-trained models on large-scale datasets such as ImageNet [25] which contains 1.4 million images over 1000 object classes for training their models using a technique called transfer learning. The transfer learning method keeps the parameters of the CNN model's previous layer and removes the last layer, then retrains the last layer for new categories. This helps to train networks for classification with a small number of training samples like the 20-category flower dataset that we used for this study.

In order to determine the most effective model for flower classification, experiments were carried out with three different CNN algorithms.

**MobileNet V1:** MobileNets are mobile-first computer vision models designed by Google researchers in order to run onto smartphones with limited memory. MobileNets uses depthwise separable convolution architecture that significantly reduces the parameters when compared to networks with the same depth in the networks. This lightweight network structure provides an efficient model for mobile and low-power embedded vision applications [26]. MobileNet V1 network structure has 28 layers and requires only one-eighth of the computation cost. This architecture, as seen in Figure 1, has an image input size of 224-by-224 with 3 channels.

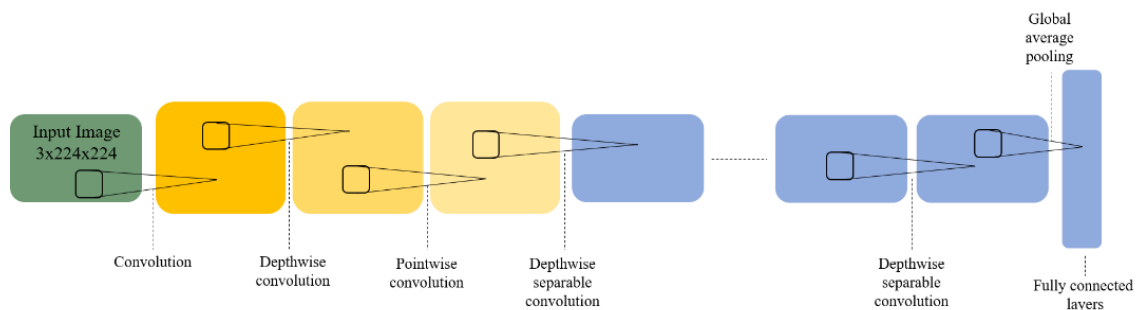


Figure 1. MobileNet V1 architecture.

**Resnet-50:** The ResNet-50 (Residual Network-50) is the first deep-CNN (>50 layers) architecture, as seen in Figure 2, with 50-layer deep that utilized residual learning [27] to manage more complex image recognition tasks and improve the accuracy of the model. In deep networks, with the increasing layers, accuracy gets saturated and then degrades rapidly. Residual learning has been introduced to address this degradation problem. The residual networks are made up of residual blocks that comprise a skip connection to make it easier to learn the identity function and help to solve the problem of degradation of training accuracy. Residual networks are the deepest ever presented on ImageNet and still have lower complexity than VGG networks [28]. This network structure has an image input size of 3-by-224-by-224.

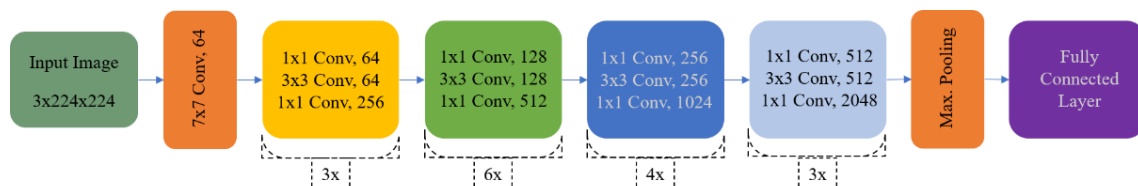


Figure 2. Resnet-50 architecture.

**Inception V3:** Inception V3 is a CNN structure developed by Keras [29], which is pre-trained in ImageNet. The Inception V3 network consists of 48-layers of convolutional, pooling, and fully connected layers as shown in Figure 3 with an image input size 3-by-299-by-299. Compared to the previous versions (Inception V1 [30] and V2 [31]), the Inception V3 network structure uses a convolution kernel splitting method [32] to achieve different scale perceptions. Through the splitting method, it does not only reduce the number of network parameters and computational costs without hampering accuracy but also increases the network depth and extracts the spatial features more effectively.

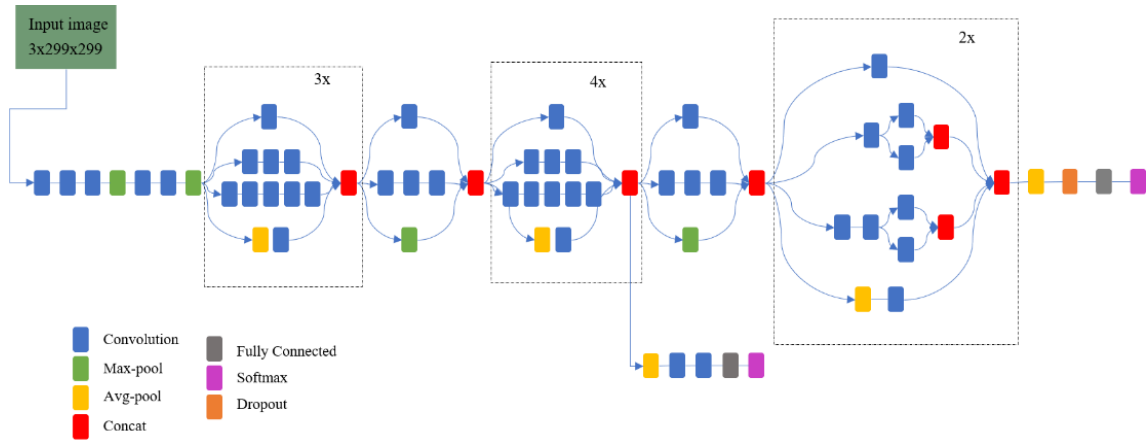


Figure 3. Inception V3 architecture.

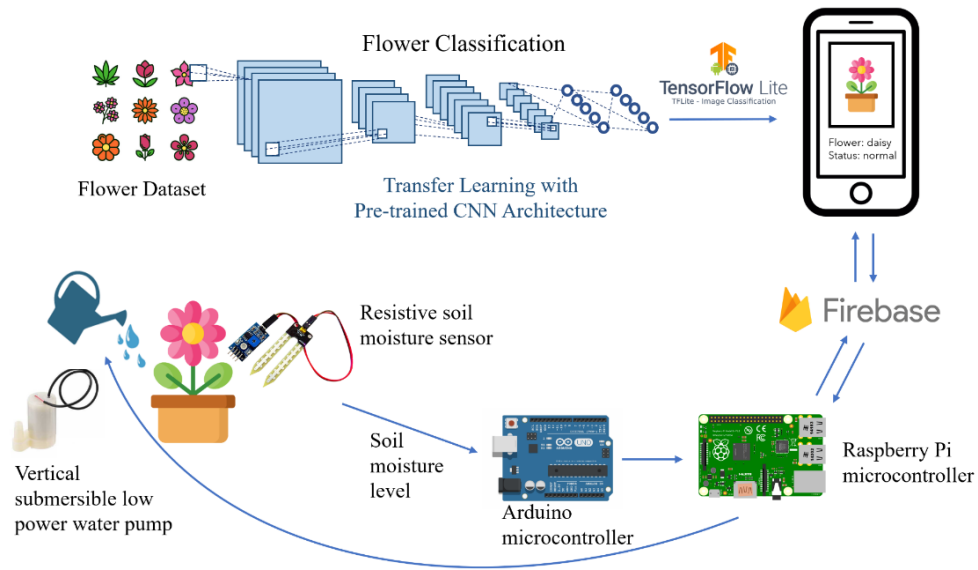
### 3.2. Implementation

The system utilizes two microcontrollers for electronic device monitoring, data collection, and data saving; Arduino UNO [33] and Raspberry Pi [34]. The irrigation system is composed of a vertical submersible low-power water pump to supply water and a resistive soil moisture sensor to detect the humidity level of the soil. The sensor sends the moisture level of the soil to the Raspberry Pi through the Arduino UNO with serial communication. Then, the received value is compared with the predefined soil moisture range of the flower given in Table 1 determined from [35], for every 12 hours. The flower water status is then forwarded to the smartphone application through the Firebase [36] server. If the moisture level is below the desired range, the automated irrigation system provides the required measure of water for 2 seconds till it reaches the determined water content range.

Table 1. Soil moisture levels of flowers in our dataset.

Flower	0% - 20%	21% - 40%	41% - 60%	61% - 80%
	Dry soil ●	Drained soil ●	Moist soil ●●	Wet soil ●●●●
Azalea		●	●	
Bluebell		✓		
Buttercup				✓
Cactus	✓	✓		
Coltsfoot			✓	
Cowslip			✓	
Crocus	✓	✓		
Daffodil			✓	
Daisy			✓	
Dandelion		✓		
Fritillary		✓		
Geranium			✓	
Iris		✓		
Lily Valley		✓		
Pansy			✓	
Snowdrop		✓		
Sunflower		✓		
Tigerlily				
Tulip		✓		
Windflower			✓	

The overall proposed smart flower status tracker and self-irrigation system is depicted as a block diagram and is presented in Figure 4.



**Figure 4.** Schematic of the flower status tracker and self-irrigation system on smartphones.

#### 4. Results

In this study, we used the Oxford-17 Flower Dataset [37] with each species has 80 flower images. Three more types of flower images (80 images per type) were also added to obtain 20 species. This new dataset is split into two parts where 70% are used as a training set, 30% are used as a validation set. For the classification of images, the three aforementioned models were selected as MobileNet V1, ResNet-50, and Inception V3. The optimizer for training these models is the Stochastic Gradient Descent (SGD) [38] optimizer with a selected batch size of 32 and a learning rate of 0.001. The training of all the models is completed in 50 epochs where the validation accuracy has at the peak level. Then the model is exported to TensorFlow-Lite [39] model and is deployed into the application to perform the classification on Android-based smartphone devices and manage to detect flowers.

The training and validation accuracy of the models with the transfer learning technique is presented in Table 2. The best validation accuracy is found with the Inception V3 model; therefore, this model is deployed to the Android application.

**Table 2.** Performance evaluation scores.

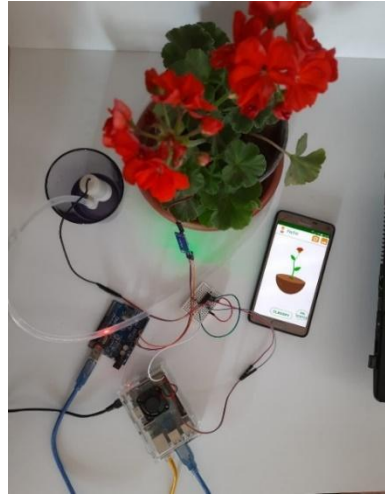
Model	Train Accuracy	Validation Accuracy
MobileNet V1	0.9758	0.8973
ResNet-50	0.9880	0.9010
Inception V3	<b>0.9955</b>	<b>0.9312</b>

**Table 3.** Performance comparison of our proposed model and other state-of-the-art models on Oxford-17 Flower Dataset.

Model	Accuracy
VGG 16-transfer [40]	0.8353
VGG 19-transfer [40]	0.8471
VA ResNet-50 [41]	0.8570
Fusion Descriptor and SVM [7]	0.8617
Our proposed model	<b>0.9312</b>

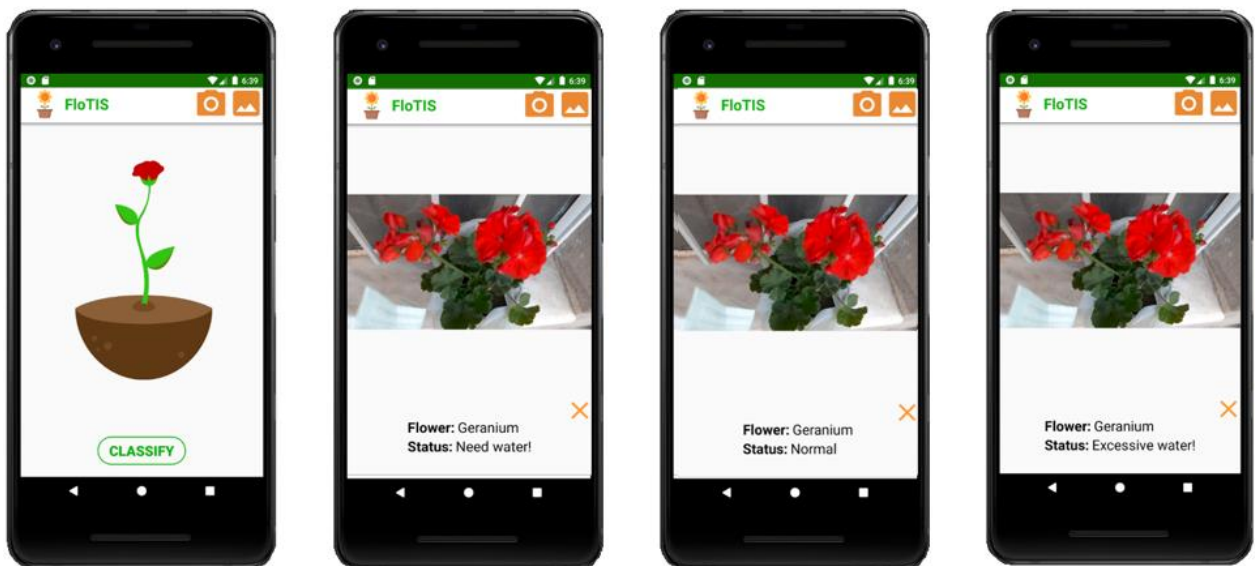
A comparison of the different models on the Oxford-17 Flower Dataset is also performed. Based on the accuracy values presented in Table 3, our proposed Inception V3 model has a significant improvement as compared to other methods.

Figure 5 presents the setup prototype of FloTIS. The system is tested on the geranium. The flower name is identified by taking a photo with the camera of the smartphone from the application menu. The photo is predicted as "geranium" and displayed by the application.



**Figure 5.** Setup for the prototype of FloTIS.

The flower name information is then sent to the controller through the server. As the soil moisture levels have been stored for each flower, the requirement for geranium soil moisture is moist soil and the level should be between 41% and 60% as given in Table 1. In this work, three categories are selected to notify the user and then to supply water to the flower. These are determined as "need water!", "normal", "excessive water!". For example, when the measured value is 23%, the water is insufficient, then the notification is "need water!". When the flower is watered and the moisture level is increased to 56%, it is notified as "normal". When the moisture level is measured as 65% that exceeds the upper limit for the geranium, the notification becomes "excessive water!". The corresponding statuses of the water displayed on the smartphone application are shown in Figure 6.



**Figure 6.** Water statuses of the recognized flower by the smartphone application.

## 5. Conclusion

In this study, an automatic self-irrigation and flower status tracker system is proposed. It combines deep learning architectures to identify the flower, a detection and irrigation system to sense and then provide the plant with a certain level of water and hence prevent overwatering, into a smartphone application named FloTIS integrating all systems into an IoT framework.

The advantage of the system lies in detecting the flower type using image recognition and then providing the correct execution of watering for different flower types. The user-friendly FloTIS application displays corresponding water levels of the flower while it controls and tracks the irrigation system managing the desired levels accordingly.

As the flower image dataset used in this work is limited, only three deep learning architectures were selected for flower identification. A comparison of the other classification algorithms denoted the efficiency of the proposed method in terms of the accuracy scores. Note also that those were pre-trained algorithms and they can be used easily in a higher number of dataset images wherever available. Therefore, the proposed system can be implemented on a large scale for farming, especially for water conservation due to prevailing conditions and water shortages.

## Declaration of Interest

The authors declare that there is no conflict of interest.

## References

- [1] A. Patil, M. Beldar, A. Naik, and S. Deshpande, "Smart farming using Arduino and data mining," In 3<sup>rd</sup> International Conference on Computing for Sustainable Global Development (INDIACom), 2016, pp. 1913-1917.
- [2] R. Ratasuk, B. Vejlgard, N. Mangalvedhe, and A. Ghosh, "NB-IoT system for M2M communication," In Proc. IEEE Wireless Communications and Networking Conference, 2016, pp. 428-432.
- [3] A. Kumar, A. Surendra, H. Mohan, K. M. Valliappan, and N. Kirthika, "Internet of things based smart irrigation using regression algorithm," In Proc. International Conference on Intelligent Computing, Instrumentation and Control Technologies, 2017, pp. 1652-1657.
- [4] M. Mancuso and F. Bustaffa, "A wireless sensors network for monitoring environmental variables in a tomato greenhouse," In IEEE International Workshop on Factory Communication Systems, 2006, pp. 107-110.
- [5] H. H. Lee, X. H. Li, K. W. Chung, and K. S. Hong. "Flower image recognition using multi-class SVM," Applied Mechanics and Materials, vol. 284-287, pp. 3106-3110, 2013.
- [6] W. Zhou, S. Gao, L. Zhang, and X. Lou. "Histogram of oriented gradients feature extraction from raw Bayer pattern images," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 67, no. 5, pp. 946-950, 2020.
- [7] W. Liu, Y. Rao, B. Fan, J. Song, and Q. Wang. "Flower classification using fusion descriptor and SVM," In Proc. IEEE International Smart Cities Conference (ISC2), 2017, pp. 1-4.
- [8] M. Hussain, J. J. Bird, and D. R. Faria, "A study on CNN transfer learning for image classification," In UK Workshop on Computational Intelligence, 2018, pp. 191-202.
- [9] N. Sharma, V. Jain, and A. Mishra, "An analysis of convolutional neural networks for image classification," Procedia Computer Science, vol. 132, pp. 377-384, 2018.
- [10] I. Gogul and V. S. Kumar, "Flower species recognition system using convolution neural networks and transfer learning," In 4<sup>th</sup> International Conference on Signal Processing, Communication and Networking, 2017, pp. 1-6.
- [11] X. Xia, C. Xu, and B. Nan, "Inception-v3 for flower classification," In 2<sup>nd</sup> International Conference on Image, Vision and Computing, 2017, pp. 783-787.
- [12] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," In Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2818-2826.
- [13] D. Sinha and M. El-Sharkawy. "Thin mobilenet: An enhanced mobilenet architecture," In IEEE 10<sup>th</sup> Annual Ubiquitous Computing, Electronics & Mobile Communication Conference, 2019, pp. 280-285.
- [14] K. He, X. Zhang, S. Ren, J. Sun, "Deep residual learning for image recognition", In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778.
- [15] W. Wang, Y. Li, T. Zou, X. Wang, J. You, and Y. Lou, "A novel image classification approach via dense-MobileNet models," Mobile Information Systems, vol. 2020, Article ID 7602384, 8 pages, 2020.
- [16] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, Cambridge, 2016.
- [17] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, "Handwritten digit recognition with a backpropagation network," in: Advances in Neural Information Processing Systems (NIPS), vol. 2, 1989, pp. 396-404.
- [18] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," Journal of Big Data, vol. 8, no. 53, pp. 1-74, 2021.
- [19] A. Rayes and S. Salam, Internet of Things From Hype to Reality: The Road to Digitization, 2nd, Springer, 2019.
- [20] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 2347-2376, Fourthquarter 2015.
- [21] I. U. Din et al., "The Internet of things: A review of enabled technologies and future challenges," IEEE Access, vol. 7, pp. 7606-7640, 2019.



- [22] A. Khanna and S. Kaur, "Internet of things (IoT), applications and challenges: A comprehensive review," *Wireless Personal Communications*, vol. 114, pp. 1687-1762, 2020.
- [23] H. F. Atlam, R. J. Walters, and G. B. Wills, "Internet of things: State-of-the-art, challenges, applications, and open issues," *International Journal of Intelligent Computing Research (IJICR)*, vol. 9, no. 3, pp. 928-938, 2018.
- [24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in: *Advances in Neural Information Processing Systems (NIPS)*, vol. 25, 2012, pp. 1097-1105.
- [25] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," 2015, pp. 211-252.
- [26] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," arXiv:1704.04861, 2017.
- [27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770-778.
- [28] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv:1409.1556, 2014.
- [29] A. Gulli and S. Pal, *Deep Learning with Keras*, Packt Publishing, 2017.
- [30] S. M. Sam, K. Kamardin, N. N. A. Sjarif, and N. Mohamed, "Offline signature verification using deep learning convolutional neural network (CNN) architectures GoogLeNet Inception-v1 and Inception-v3," *Procedia Computer Science*, vol. 161, pp. 475-483, 2019.
- [31] S. R. Bose and V. S. Kumar, "Efficient inception V2 based deep convolutional neural network for real-time hand action recognition," *IET Image Processing*, vol. 14, pp. 688-696, 2019.
- [32] M. Kim and L. Rigazio, "Deep clustered convolutional kernels," In *Proceedings of the 1st International Workshop on Feature Extraction: Modern Questions and Challenges at NIPS 2015*, PMLR vol. 44, pp. 160-172, 2015.
- [33] Y. A. Badamasi, "The working principle of an Arduino," In *11th International Conference on Electronics, Computer and Computation (ICECCO)*, 2014, pp. 1-4.
- [34] N. S. Yamanoor, and S. Yamanoor, "High quality, low cost education with the Raspberry Pi," In *IEEE Global Humanitarian Technology Conference (GHTC)*, 2017, pp. 1-5.
- [35] [Online]. Available: <https://www.groww.fr/en/plants>.
- [36] L. Moroney, "The firebase realtime database," in *The Definitive Guide to Firebase*, Apress, 2017, pp. 51-71.
- [37] M.-E. Nilsback and A. Zisserman, "A visual vocabulary for flower classification," In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006, pp. 1447-1454.
- [38] L. Bottou, "Stochastic gradient descent tricks," In: Montavon G., Orr G. B., Müller K. R. (eds) *Neural Networks: Tricks of the Trade. Lecture Notes in Computer Science*, vol. 7700. Springer, Berlin, Heidelberg, 2012, pp. 421-436.
- [39] O. Alsing, "Mobile object detection using tensorflow lite and transfer learning," Master thesis, KTH Royal Institute of Technology, 2018.
- [40] Y. Wu, X. Qin, Y. Pan, and C. Yuan, "Convolution neural network based transfer learning for classification of flowers," In *Proc. IEEE 3<sup>rd</sup> International Conference on Signal and Image Processing*, 2018, pp. 562-566.
- [41] S. Cao and B. Song, "Visual attentional-driven deep learning method for flower recognition," *Mathematical Biosciences and Engineering*, vol. 18, no. 3, pp. 1981-1991, 2021.

# Turkish Character Usage in Text Classification

Ali Aycan KOLUKISA <sup>1</sup>

<sup>1</sup> Graduate of Izmir Katip Celebi University, Graduate School of Natural and Applied Sciences, Software Engineering, Turkey  
MSc student of Dokuz Eylul University, Institute of Social Sciences, Management Information Systems, Turkey

## Abstract

This study is prepared to examine the effects of Turkish character usage on text data by using multiple classifiers. Regression Classifiers, SVM, NB-Classifiers, and ANN are frequently used in supervised learning methods, especially in classification problems. Regression classifiers generally come in two types: as Linear and Logistic. There are also more than one type of Naive Bayes classifier. In our study, after mentioning the properties of Linear Regression and Logistic Regression classifiers in general terms, why Logistic Regression is much more suitable for this study is explained. Then, with the usage of "Logistic Regression", "LinearSVC", "MultinomialNB", "ComplementNB", "BernoulliNB" and "Perceptron" classifiers, the analyzing part starts. Our datasets consist of abstracts-parts from 64 Turkish articles, which have 4 different classes as Physical Sciences, Social Sciences, Educational Sciences, and Economics Administrative Sciences. The data files are all in CSV file format, however, two different data files were prepared. One with original Turkish characters, and the other with its English equivalent formation targeting the Turkish characters "Ç, ç, Ö, ö, Ü, ü, Ş, ş, İ, ı, ğ". In its English-like equivalent file, these were replaced with "C, c, O, o, U, u, S, s, I, i, g" respectively.

**Keywords:** *Accuracy rate; bag of words; English characters; logistic regression; Turkish characters.*

## 1. Introduction

As it is known, Regression Classifiers, SVM, NB-Classifiers, and ANN are frequently used in supervised learning methods, especially in classification problems. Regression classifiers generally come in two types as Linear and Logistic. It is possible to mention that these two classifiers have some positive and negative aspects according to their characteristics. It is also seen that there are more than one type of Naive Bayes classifier. In our study, firstly, after mentioning the properties of Linear Regression and Logistic Regression classifiers in general terms, it is explained why the Logistic Regression classifier is much more suitable for this study. Afterward, the analyzing part takes place with the usage of Logistic Regression, "LinearSVC", "MultinomialNB", "ComplementNB", "BernoulliNB" classifiers, and "Perceptron" classifiers.

Our datasets consist of abstracts-parts from 64 Turkish articles, which have 4 different class-labels such as Physical Sciences (= FEN), Social Sciences (= Sosyal), Educational Sciences (= Egitim), and Economics and Administrative Sciences (= IIBF). In collecting the data, 4 different journals have been used for each class label and 4 articles have been taken from each journal. The journal names used in this study will be given at the end of this paper. The data files have been prepared in CSV file format. And we have prepared two different types of data files. One with the original Turkish characters, and the other one with its English equivalent formation. In the second one, we have changed the original characters of the Turkish language "Ç, ç, Ö, ö, Ü, ü, Ş, ş, İ, ı, ğ" into its English-like equivalents "C, c, O, o, U, u, S, s, I, i, g" respectively. Hereby, two different-named data files, which can be regarded the same in terms of their contents but differ in the use of Turkish characters, have been made ready for accuracy analysis by the above-mentioned classifiers.

## 2. Classifiers Used for Text Classification

The main classifier supposed to be used in this study is Logistic Regression. However, also other classifiers are added to the study to be able to see how the other classifiers act with the same datasets.

### 2.1. Logistic Regression Classifier

Regression analysis is an analysis method used to examine the effect or effects of one or more independent variables on a dependent variable [1]. On the other hand, when we look at the working principles of Regression classifiers, it is seen that generally two types of results can be obtained depending on more than one variable. These results, which are generally confused as 0 - 1, are encountered especially in linear type regression classifiers. However, this is a disadvantage of linear regression classifiers as it is possible for output categories to take values between 0 and 1, such as 0.8 or 0.4. It is generally seen that these problems are overcome by setting the threshold

\*Corresponding author e-mail address: [kolukisa.aycan@gmail.com](mailto:kolukisa.aycan@gmail.com)

value. On the other hand, if the desired results due to more than one variable are desired to be higher than 0 and 1, logistic regression is preferred because linear regression is seen to be insufficient.

At first glance, it can be assumed that logistic regression classifiers operate like linear regression classifiers. However, it appears that there are subtypes of logistic regression classifiers that can adapt to more than one output. These are the Binary Logistic Regression, Multinomial Logistic Regression, and Ordinal Logistic Regression classifiers [2]. In this way, it can give more stable results than Linear Regression.

On the other hand, when compared with the Linear Regression Classifier, there are differences in terms of Cost Function. While in Linear Regression, algorithms such as Mean Square Error, Mean Absolute Error, and Root Mean Square Error are used as cost functions, these algorithms cause various irregularities when applied in Logistic Regression [3]. For this reason, Softmax Function, which can sometimes be named as Logistics Cost Function, is generally used in Logistic Regression. On the other hand, it can be seen that due to the existing similarities of the Softmax function, it is also considered as the general form of the sigmoid function used in probability calculations on binary variables [4]. However, since the Softmax Function, which is mostly used in multiple classification problems, is a non-linear classifier [5], it takes the input data in the layer preceding it and determines which class these inputs are closer to, unlike linear regression classifiers that can distinguish with a single line, by making probability calculations [6].

Therefore, considering the above reasons, it would be appropriate to say that it would be more appropriate to prefer Logistic Regression since there are 4 different class labels of the datasets used in the study.

## 2.2. Other Classifiers

It is seen that statistical methods such as Regression, Logistic Regression, Time Series Analysis, and Bayesian approaches are generally used in classification problems [7]. In addition to the Logistic Regression classifier, "LinearSVC", "MultinomialNB", "ComplementNB", "BernoulliNB", and Perceptron classifiers are also used to be able to see how other classifiers act with the same datasets.

## 3. Datasets

The datasets in this study were generally prepared using academic journals in Turkish that have open access on the DergiPark<sup>1</sup> website. A ready-to-use dataset was not employed. A total of 64 articles were used. These articles have 4 different class tags. These are in the form of Science (FEN), Social Sciences (Sosyal), Educational Sciences (Egitim), and Economic and Administrative Sciences (IIBF), respectively, and articles in field journals have been used. 4 different journals were used for each field, and 4 articles (abstract parts only) were taken from each journal. In order to classify the articles, the abstract parts were taken and recorded in the data file. Although each article abstract consists of many sentences, it constitutes only 1 sample of data in the study. Therefore, there are 64 article abstracts belonging to 4 different classes in total, and there are 16 article abstracts in each class, although their lengths differ. These datasets were saved in the form of a CSV file with the name "Makale4x16(tr)" for original Turkish characters. And then in the same file, the Turkish specific characters "Ç, ç, Ö, ö, Ü, ü, Ş, ş, İ, ı, ğ" were determined and changed into their English equivalents "C, c, O, o, U, u, S, s, I, i, g" and saved as a different CSV file named as "Makale4x16". Therefore, 2 datasets consisting of  $16 \times 4 = 64$  article abstracts for each, whose contents and word numbers and sequences are exactly the same, but differ only in terms of the use of Turkish characters, were made ready for analysis. The general distribution of these datasets used in the study is as follows.



Figure 1. English-equivalent Formation

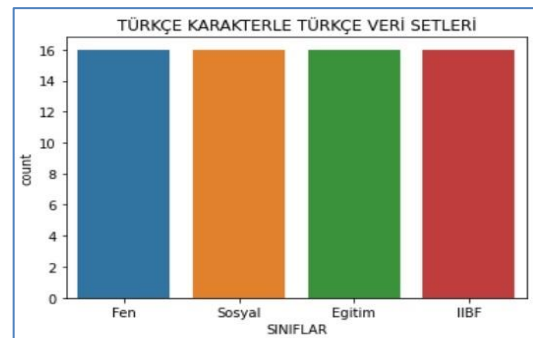


Figure 2. Original Turkish Characters

<sup>1</sup> <https://dergipark.org.tr/tr/>

As it is understood from Figures 1 and 2, the datasets have exactly the same qualification, except the Turkish character usage in the text parts of them.

#### 4. Tools and Environment

To be able to analyze the datasets, PYTHON codes are preferred. The operating system environment is 64-bit Windows 8.1 with 10 GB of RAM - Intel Celeron 2957U@1.4 GHz. In order to run the PYTHON codes, the SPYDER interface (Figure 4) that comes with ANACONDA is preferred.

The libraries used in the application such as Pandas, Scikit-learn, Seaborn, etc. were loaded first into SPYDER via the ANACONDA command line (CMD) before the operation. The version information of SPYDER is 4.1.4 and PYTHON version used in this study is 3.8.3 (64-bit) as seen. And to be able to analyze the files, two different CSV files were loaded by using the pandas library command, as `pd.read_csv('Makale4x16(tr).csv')` and `pd.read_csv('Makale4x16.csv')`.

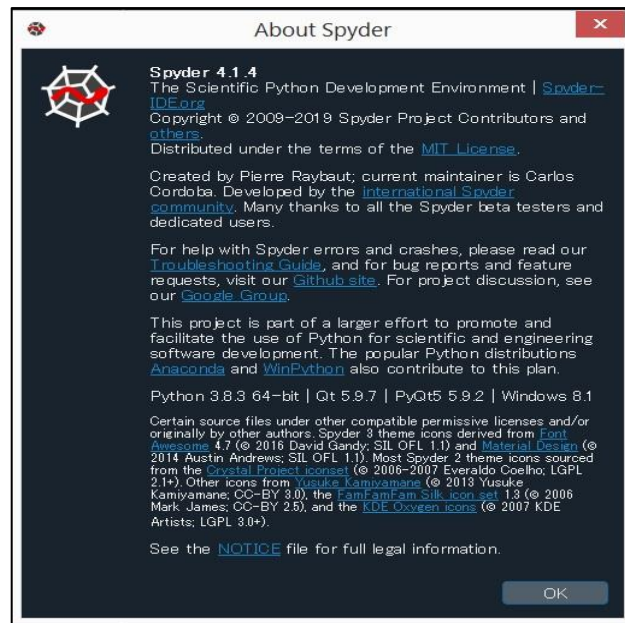


Figure 3. Spyder Environment

#### 5. Operation and Analysis

In our application, the model preparation was done first. The necessary libraries were imported and included in the application, then the modeling of Logistic Regression and other classifiers were created with Python codes. First, "Makale4x16.csv" file has been prepared to be subjected to Logistic Regression analysis. Then, the models of "LinearSVC", "MultinomialNB", "ComplementNB", "BernoulliNB" and "Perceptron" classifiers were also created and added with Python codes. However, since it is not possible for the machine to directly read the string type (textual) data, first of all, this data is converted to numerical form. For this, two methods were used. In the Logistic Regression model, these textual data were converted into numerical form by using TfidfVectorizer. For the other classifiers, the Bag of Words (BOW) model was prepared by using the CountVectorizer. Thus all the textual data was transformed into numerical data so that the machine can understand. After our models and codes were made ready for all the classifiers, the analysis phase started. First, the dataset with English characters was analyzed (Figure 4).

```
In [1]: runfile(
...     ..., '_Makale4x16.py')
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 64 entries, 0 to 63
Data columns (total 2 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   TEXTLER     64 non-null    object
1   SINIFLAR    64 non-null    object
dtypes: object(2)
memory usage: 1.1+ KB

LogisticRegression() Doğruluğu : % 43.75
LinearSVC() Doğruluk Oranı : % 75.0
MultinomialNB() Doğruluk Oranı : % 81.25
ComplementNB() Doğruluk Oranı : % 87.5
BernoulliNB() Doğruluk Oranı : % 37.5
Perceptron() Doğruluk Oranı : % 37.5
```

**Figure 4.** Results of English-equivalent Formation

```
In [2]: runfile(
...     ..., '_Makale4x16(tr).py')
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 64 entries, 0 to 63
Data columns (total 2 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   TEXTLER     64 non-null    object
1   SINIFLAR    64 non-null    object
dtypes: object(2)
memory usage: 1.1+ KB

LogisticRegression() Doğruluğu : % 50.0
LinearSVC() Doğruluk Oranı : % 75.0
MultinomialNB() Doğruluk Oranı : % 75.0
ComplementNB() Doğruluk Oranı : % 87.5
BernoulliNB() Doğruluk Oranı : % 37.5
Perceptron() Doğruluk Oranı : % 50.0
```

**Figure 5.** Results of Original Turkish Characters

As seen in Figure 4, the accuracy rate of Logistic Regression was 43.75%, Bernoulli Naive Bayes was 37.5% and Perceptron was 37.5%, and as understood, the accuracy rates of these three classifiers were generally below 50%. On the other hand, Linear Support Vector Machine achieved an accuracy rate of 75%, Multinomial Naive Bayes was 81.25%, and Complement Naive Bayes was 87.5%, achieving an overall success rate of 75% and above. Now, with the above codes, let's replace only the part of the file to be analyzed with "Makale4x16(tr).csv". Here, It is better to mention again that no changes have been made to the codes and they are the same as in the previous section.

This time, "Makale4x16(tr).csv" file - with Turkish original characters, was analyzed and the following results were obtained. As it is seen from the results at Figure 5, although the basic contents of our two files are the same, the detail has increased with the use of Turkish characters, and the machine classifiers have given a certain reaction to this. This time, the accuracy rate of Perceptron and Logistic Regression classifiers increased within the Turkish original character dataset file analysis and reached to 50%. However, "MultinomialNB" decreased to 75%. There has been no change in the accuracy rates of "LinearSVC", "ComplementNB" and "BernoulliNB" classifiers.

So, what does this situation tell us? Based on the above results, the use of Turkish characters increases the in-text details to a certain extent, and accordingly, Perceptron, which is the most basic and simple form of deep learning algorithms, can draw new teachings from this change, and it is not only limited to Perceptron, it can be said that it also makes sense for the Logistic Regression classifier.

## 6. Conclusion

In this study, Turkish article abstracts with the same content were prepared in 2 different CSV files with Turkish and English characters, and the effects of using Turkish characters on machine learning were examined. Although the contents of both files are the same, the results show that Perceptron and Logistic Regression classifiers, which are frequently used in deep learning, have a positive response to the Turkish characters, while Multinomial Naive Bayes has a negative response under the same conditions. On the other hand, "LinearSVC", "ComplementNB" and "BernoulliNB" classifiers show no reaction to the use of Turkish characters. The results are summarized in Figures 6 and 7.





Makale4:16csv [3]	
1	TEXTLER, SINIFLAR
2	"Galaksiler, kule cekim kuvvetiyle bir arada bulunan yildizlar, gaz, toz ve karanlik maddeden meydana gelen sistemlerdir. Evrende milyarlarca galaksi bulunmaktadır. Her bir galaksinin tek tek incelenmesinin maliyeti yuksek oldugundan galaksi siniflandirmasi astronomik veri analizinde onemli bir yer tutmaktadır. Galaksiler morfolojilerine ve spektral ozelliklerine gore siniflandirilmaktadir. Yeni seti icindeki gicli orantuyu ortaya cikarmayi amaclayan makine ogrenme yontemleri mevcut veriyi analiz ederek dogal gruplari heruz tespit edilmesini olan galaksilerin hangi gruba ait oldugunu tahmin etmek amacıyla kullanilabilir. Bu da gerek arastirmacılara gerekse astronomlara zaman ve maliyet acısından kazanc saglayacaktır. Bu calisma da Shapley Konsantrasyonu bölgesindeki 4215 galaksi, 5 degisken (enlem, boylam, parlaklik, hiz ve hizdaki sapma) dikkate alınarak siniflandirilmistir. IDL programlama ile dogal gruplari tespit edilen galaksiler Weka programi ile makine ogrenme algoritmaları kullanılarak siniflandirilmistir. Bayes Siniflandirici yontemlerinden Naive Bayes ve Bayes net, Karar Agacları yontemlerinden J48, J47 ve Random Forest algoritmaları, Yapay Sinir Agilarından Çok Katmanlı Algılayıcılar ve Destek Vektor siniflandirici yontemleri kullanilmistir. Elde edilen siniflandirma sonuclari dogal gruplarla karşılastirilmis ve yontemlerin tahmin performansları degerlendirilmistir."/>
3	"Bu calisma, Kocaeli il sınırlari icinde yer alan Yuvasik Baraj Golu'nun yuzey suyu kalitesini ve kirlilik problemlerini ortaya koymak uzere bazı fiziko-kimyasal ozelliklerini incelemek ve trofik durumunun belirlenmesi amacıyla yapilmistir. 5 farkli istasyondan farkli derinliklerde iki donem (Eylul 2016 ve Mayıs 2017) hamsu numuneleri alınmistir. Arastirma sonucunda Yuvasik Baraj Golu'nun su kalite parametrelerinin ortalama degerlerinin kalite kriterlerine gore su kalite sinifi I (yuksekkalite, cok iyi) - II (as kirlenmis, iyi) araliginda olduğu tespit edilmiştir. Orofikasyon kriterlerine gore golum trofik duzeyinin toplam asot (TN) ve toplam fosfor (TP) konsantrasyonu acısından mezotrofik, klorofilla acısından oligotrofik, isk gecirgenligi acısından ise donemsel olarak mezotrofik seviyede olduğunu göstermiştir. Ortalama Trofik durum indeksi (TSI) degeri 44.1 olarak hesaplanmiş ve golum trofik seviyesinin mezotrofik olduğu belirlenmiştir."/>
4	"Meyve ve sebzeler saglikli bir yasam icin tuketilmesi gereken esansiyel urunler arasında icerdikleri yuksek vitamin, mineral ve antioksidan gibi faydalı maddeler bakımından da ilk sirada bulunmaktadır. Iyi etkileri sayilamayacak kadar fazla olsa da taze tuketilmeyenlerinde besin degerleri dusmekte ve mikroorganizmalarca istila edilmektedirler. Bu durum sonucunda kuf olusumu gozlenmekte ve hem saglik acısından hem de gida sektorunde ekonomik acidan birtakim sorunlar olmaktadır. Meyve ve sebzelerin tazeliğinin saglik ve gida sektorune ek olarak ekonomik anlamda yarattigi onem gor onunde bulundurulacak yapılan deneyler sonucu tazelik parametresinin iyon hareketliliği ile olan etkisi ortaya cikarilmistir. Gerceklestirilen deneysel calismalar sonucunda elde edilen veriler yardimiyla tazelik ile iyon hareketliliği arasında bir iliski kurularak bu deney sonuclarinin pratikte kullanilabilirliğini saglamak amaci ile bir cihaz gelistirilmistir. Yapilan deneyleri takiben gelistirilen cihaz tazelik-iyon hareketliliği iliskisini belirleyebilen bir sensör olarak tasarlanmistir. Sensör olcülerinin degerlendirilmesi icin deneysel olcüler ile elde edilen tazelik ve bozulma degeri verilerini iceren bir yazilim gelistirilmis ve yazilimdan yararlanarak tazelik, tasarlanan sensör ile belirlenebilmiştir. Elde edilen deneysel olcum sonuclari ve sensör olcum sonuclari karşılastirilmis, sonuclarin birbirini destekler nitelikte olduklari aynı zamanda farkli kosullarda tekrarlanan gozlemlerde de belirlenmiştir."/>
5	"Pasif isi transferi iyilestirme metodlarında isi transferi kat sayisi ve Nusselt sayisini maximize ederken, basinc dusumunu minimize eden yaklasimi tespit edebilmek icin bir cok parametrenin optimizasyonunun yapılması gerekmektedir. Bu sebepten oturu, deneysel ve sayisal calismalara bagli olarak empirik korelasyonlar elde edilmektedir. Bu calismada dikdortgenel finlerin isi transferi davranisi deneysel ve yapay sinir aglari metodlari ile ortaya konulmuştur. Yapay sinir aglari metodolojisi ile elde edilen sonuclar korelasyon ile kiyaslanmistir. Ayrica, tanimlanan problem icin yapay sinir ağı uygulamasında farkli eğitim algoritmalarının ve Katman sayısının sonuclar üzerindeki etkisi arastirilmistir. Elde edilen sonuclara gore YSA yontemi, korelasyon yonteminden daha hizli ve daha dogru sonuç vermektir. Diger yandan YSA yaklasiminin dogruluğunun arttirilmasi icin uygun eğitim algoritmasının secimi, uygun katman sayısının tespiti yani uygun mimarinin elde edilmesi onem arz etmektedir. Tanimlanan bu problem icin, 10-5-1 ağına sahip Bayesian Regularization algoritması 87.6 ortalama yuzde hata ve 0.029 RMSE ile iyi senaryo olarak belirlenmiştir. Maximum ortalama hata %56.3 ile Levenberg- Marquardt algoritmasında 10-12-1 ağı ile elde edilmiştir."/>
6	"Günümüzde isletmeler gerek piyasaya tutunmak gerekse her geçen gun gelismekte olan teknolojiyi yakalamak adına yagun bir rekabet icerisinde bulurlar. Yagun rekabet ortamı mevcut müşteriye tutma ve yeni müşteri kazanma amacını da beraberinde getirmektedir. Hava yolu isletmelerinde yolculara beklentilerinin otesinde hizmet sunma noktasında kabin ekibinin etkisi buyuktur. Bir hava yolu isletmesinde 3764 kabin memurunun 2015 yılında performans degerlendirmeleri incelenmiştir. Yapılan bu performans degerlendirmelerinin sonucunda karne duzeyleri belirlenmektedir. Bu calismanin amaci 2015 yılındaki karne duzeyleri icin; kabin memurlarının yetkinlik bazlı degerlendirme puanları ile demografik ozellikleri arasında anlamlı bir kural olusturmaktır. Bu calismada, acik kaynak kodlu JAVA dilinde gelistirilmis WEKA programi ile veri madenciligi yontemlerinden karar agaci algoritmaları kullanilmistir. Olusturulan karar agaci algoritmalarından siniflandirma dogruluğu acısından en basarili algoritma olarak Random Forest ve ikinci olarak J48 algoritması tespit edilmiştir. Random Forest algoritma çıktisi gorsel bir sonuc vermiyep algoritma adimlarini gorulmeyecek sekilde vererek karmaşık bir yapı olusmasından dolayi calisma J48 algoritmasına gore yorumlanmıştır. Ayrica, WEKA programında nitelik secimi ozelligi ile InfoGainAttributeEval algoritması ile "Ranker" metodu uygulanması sonucunda ciktilarin J48 algoritması ciktilari ile aynı dogrultuda olduğu tespit edilmiştir. Bu baglamda kabin memurlarının karne duzeylerinin korne duzeylerini en basarili algoritma olarak Random Forest ve ikinci olarak J48 algoritması tespit edilmiştir. Olusturulan karar agaci algoritmalarından siniflandirma dogruluğu acısından en basarili algoritma olarak Random Forest ve ikinci olarak J48 algoritması tespit edilmiştir. Urunlerden izole edilen kulfelerin mikroskop altinda morfolojilerinin incelenmesi sonucu Penicillium, Fusarium, Trichoderma ve Aspergillus sp. gibi makrotoksin uretebilen funguslar ile ilgili belirlenmesi ve bunların negatif Staphylococcus sp. ile uyuşmazlığı tespit edilmesi acısından önemli bir sikh olarak degerlendirilmistir. Bu
7	"Canakale-Ezine yoresi sut ve urunleri uretim kapasitesi yani sira icerdigi turistik tarihi alanlari ve coğrafi konumu nedeniyle onemli bir bolgedir. Gunumuzde dunyada ve ulkemizde dogal beslenme, dogal urunler ve ev yapimi urunlerin tuketimi konusunda bir hassasiyet olusmasi nedeniyle halk pazarlarında da ev yapimi urunler tercih edilmektedir. Bu calismada Ezine yore pazarlarında ev yapimi olduğu belirtilerek satilan tereyağların Esherichia coli, koagulaz pozitif Staphylococcus sp ve Salmonella sp. varligi ile toplam aerobik mezofilik bakterisi sayisi, koliform grubu bakterisi sayisi ve toplam maya-kuf sayisi incelenmiştir. Satisa sunulan tereyağların mikrobiyal kalitesinin Turk Gida Kodeksi Mikrobiyolojik Kalite Kriterler Teblig'inde belirlenen limit degerlere uygun olduğu belirlenmiştir. Bununla birlikte urunlerin maya-kuf yukunu 106 kob/g seviyesinde olduğu tespit edilmiştir. Urunlerden izole edilen kulfelerin mikroskop altinda morfolojilerinin incelenmesi sonucu Penicillium, Fusarium, Trichoderma ve Aspergillus sp. gibi makrotoksin uretebilen funguslar ile ilgili belirlenmesi ve bunların negatif Staphylococcus sp. ile uyuşmazlığı tespit edilmesi acısından önemli bir sikh olarak degerlendirilmistir. Bu

Normal text file

length: 103.992 lines: 73

Ln: 1 Col: 1 Sel: 0 | 0

Windows (CR LF) UTF-8

INS



# Social Distancing Automation Software Based on Cloud

Ali Şahin DEMİR<sup>1</sup>, Sevcan EMEK<sup>1,\*</sup>

<sup>1</sup> Manisa Celal Bayar University, Faculty of Engineering, Department of Computer Engineering, Turkey

## Abstract

This study aims to follow the social distance between people working together in a workplace. It is extremely important to provide a healthy workplace by both employers and employees during the pandemic. One of the measures taken during the pandemic is to ensure social distance. In this study, an IoT (Internet of Things) based circuit design is created that measure the distance between people working together. Social distancing automation software has been developed that stores information of employees such as name, surname, identity id, and the department they work in the cloud. Measurements taken from the circuit, which will track the distance of the employees, can be transferred to the cloud with this automation application. Both employees and employers will be able to observe social distance tracking by providing access to the cloud.

**Keywords:** *Cloud; IoT; social distancing automation.*

## 1. Introduction

One of the first measures taken within the scope of the measures taken by occupational health and safety professionals in the workplaces within the scope of the coronavirus epidemic is to make transition markings in accordance with the social distance rule in the workplaces. However, apart from these markings, it becomes mandatory to take other health and safety precautions. It is necessary to reevaluate the working methods and forms by considering the social distancing rule in the workplaces and to make a work organization in accordance with this rule when possible [1]. For this reason, various applications, including social distancing management, are being developed [2-4]. In areas such as public transport, airports, shopping malls, schools, hospitals, tourist attractions, restrooms, office buildings, which are fast service areas, studies are observed for the use of technologies for social distance. Sensing technology solutions are implemented to help understand visitor traffic flow, crowd density management, and behavior. Time-based measurements are suitable for measuring the actions of people in a closed environment. In order to provide a healthy space environment, areas where customers are dense, are determined, and it is observed how far people are from each other. Applications such as conference room management, mass crowd management system, building automation based on real-time signal processing are being developed to manage social distance correctly. Companies looking to improve people counting, crowd monitoring, industrial security, and other applications use sensor technology for real-time decision making and signal processing [5]. Within the scope of this study, it is planned to achieve the following targets in large-scale production areas as a sector:

- Creating social intervals by measuring distances during the epidemic period
- To use personnel performance more efficiently
- Providing controlled access to production areas
- Protecting the safety of personnel in hazardous areas
- Perceiving the mobility with the report submitted to the personnel affairs department
- To ensure that the system works integrated with other systems
- Analyzing the relationship of personnel-distance with the collected data.

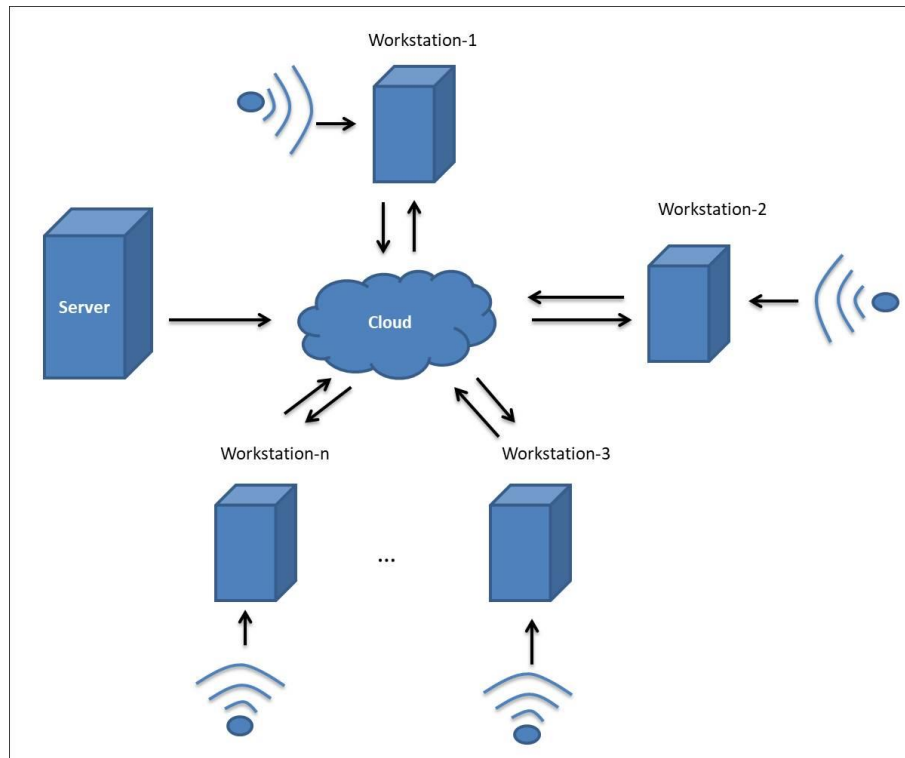
This study is organized as follows: Section 2 explains the architecture of social distancing automation application between integrated circuit and cloud. Section 3 describes how to implement the model planned in this study. Conclusion section suggests that this planned study has the potential to be implemented in a real workplaces.

## 2. Methodology

The social distancing automation consists of desktop software and workstation software. Desktop software runs on the host computer under the control of the administrator. All reporting, personnel information, and settings are provided by desktop software. The workstation software is carried out under the control of the responsible personnel in different departments in the workplace. Multiple workstations can be set up in the workplace. Desktop software and station software are written by using Python programming language.

\*Corresponding author e-mail address: [sevcan.emek@cbu.edu.tr](mailto:sevcan.emek@cbu.edu.tr)

The rangefinder is an integrated-based electronic circuit that measures ultrasonic sound, and it runs with external energy. Data is obtained by placing it in the areas where the distance should be measured. The controls used in the rangefinder are carried out with Arduino programming. The Dropbox web server, which provides a cloud storage service, is used to store the data. Personnel information stored in the cloud can be controlled by administrators with access permission. In desktop software which runs on the server and workstation software, all addresses where files are read and written are the physical address of the Dropbox. The relationship among server, workstation, rangefinder, and cloud is shown in Figure 1.

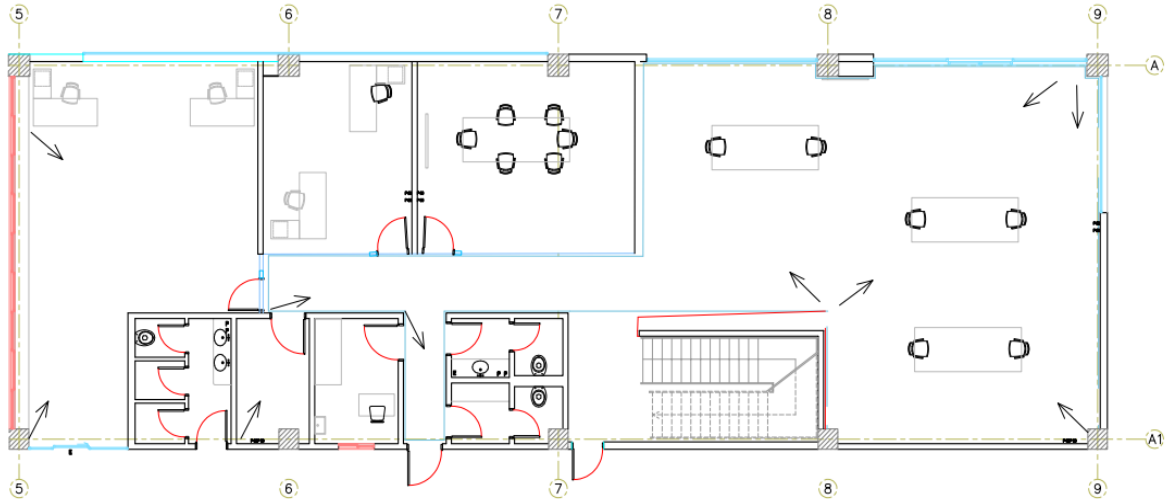


**Figure 1.** The architecture of the social distancing automation

In Figure 1, the interaction between the server and workstations is provided by the cloud. n number of workstations located in different areas will be able to receive data from sensors located in those areas and transfer them to the cloud.

### 3. Implementation

Rangefinders, which will be placed in different areas in the factory, can instantly send distance violations in that area to the nearest workstation. The model of the workplace is shown in Figure 2. The rangefinders placed in the areas marked with arrows in Figure 2 are designed to measure the distance of personnel entering that area. Measurements are sent to the cloud by the workstation. The identification number given to the personnel by the enterprise is defined at the workstations. It is envisaged to record the entry and exit of personnel in all areas at workstations. For example, in a factory whose model is shown in Figure 2, it is prevented that anyone other than authorized personnel is out of their area. The workstation sends this id number personnel information to the cloud when it enters any area determined by the arrow with the personnel card. After this notification, social distancing automation includes staff entry into that area and distance measurements by the workstation. Figure 3b indicates the real distance in detail when the data was received from the desktop software.



**Figure 2.** The modeling of the workplace for the social distancing automation

The workstation user interface is shown in Figure 3a. Authorized personnel can control the personnel information of their own unit with the workstation software.

**Cloud Data**

**AREA POINT**

Get the Person Info By ID

Get the Person Info By ID

Get Info

Name and Surname :  
Person ID :  
Work Area :

THE DISTANCE HAS BEEN SENT AUTOMATICALLY TO THE CLOUD

**Load Data**

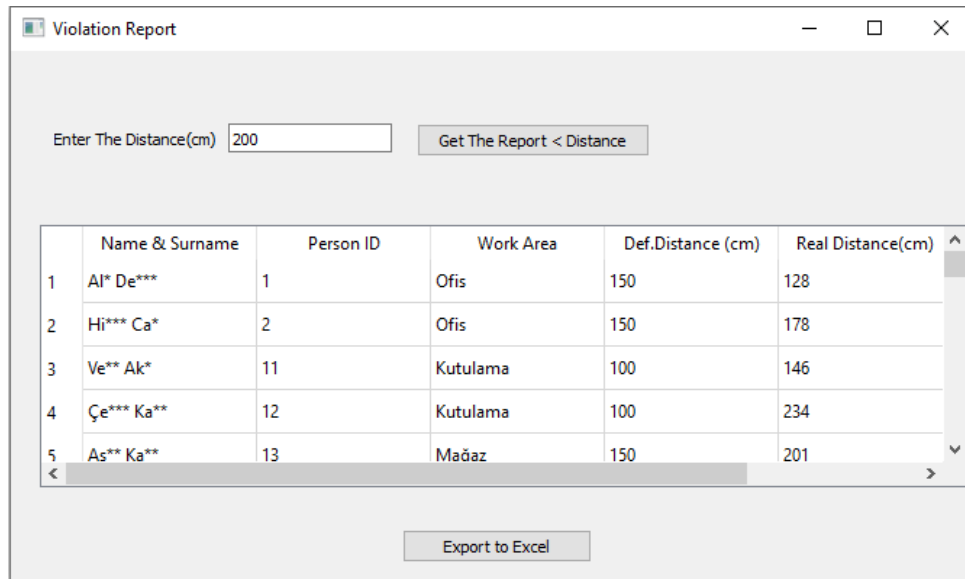
Get The Distances From The Cloud

	Name&Surname	ID	Work Area	Real Distance (cm)
1	Al* De***	1	Ofis	128
2	Hi*** Ca*	2	Ofis	178
3	As** Şi***	3	Üretim	135
4	Ze**** Uy****	4	Üretim	317
5	Ha*** Be***	5	Depo	330
6	Er*** Zo*	6	Depo	268
7	Se*** Te***	7	Depo	95

Clear The Table

**Figure 3.** a) Workstation user interface, b) Personnel information transmitted to the server

Figure 4 shows the violation report. Violation report indicates who entered the areas that should not be entered, how often, and at what distance. Whether the personnel stays within the defined distance can be observed in the violation report.



	Name & Surname	Person ID	Work Area	Def.Distance (cm)	Real Distance(cm)
1	Al* De***	1	Ofis	150	128
2	Hi*** Ca*	2	Ofis	150	178
3	Ve** Ak*	11	Kutulama	100	146
4	Çe*** Ka**	12	Kutulama	100	234
5	As** Ka**	13	Mađaz	150	201

**Figure 4.** *Violation report*

#### 4. Conclusion

This study aims to control the people working together to work in a healthy environment during the pandemic. In this study, social distancing automation that includes desktop software, workstation software, and an integrated circuit has been developed. Cloud-based social distancing automation software has the infrastructure to be applied to a workplace. The validation of this study continues to be tested with sample data. This application has the potential to work integrated with the personnel attendance tracking system and access control systems. It is also possible to develop an application that will make the machines stop when the personnel enters the area where they should not be. In this proposed study, it is foreseen that the risk of transmission of diseases such as Covid-19 will be kept at a minimum level by maintaining the social distance of the personnel. It is thought that the enterprise, whose social distance is maintained, will also contribute to public health. It is aimed to create awareness of protecting their own safety by being more careful in the office where the personnel will enter knowing that the distance is measured. If the enterprise analyzes the areas that are violated, it may have the opportunity to reduce the hazard levels or relocate the areas where the personnel are frequently and constantly. If the system works stably and the reporting is followed regularly, the personnel and work areas reach the level of being more visible and more manageable.

#### Declaration of Interest

There is no conflict of interest.

#### References

- [1] T.C. Aile Çalışma ve Sosyal Hizmetler Bakanlığı, “Yeni Koronavirüs Salgını Kapsamında İş Sağlığı ve Güvenliği Profesyonellerinin İşyerlerinde Aldıracağı Tedbirler,” [Online]. Available: <https://ailevecalisma.gov.tr/media/42183/yeni-koronavirus-salgini-kapsaminda-is-sagligi-ve-guvenligi-profesyonellerinin-isyerlerinde-aldiracagi-tedbirler.pdf> [Accessed July 16, 2021].
- [2] A. Gad, G. ElBary, M. Alkhdher, and M. Ghazal, “Vision-based approach for automated social distance violators detection,” 2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT), ‘2020, pp. 1-5, doi: 10.1109/3ICT51146.2020.9311969.
- [3] S. Siddiqui, M. Z. Shakir, A. A. Khan, and I. Day, “Internet of things (IoT) enabled architecture for social distancing during pandemic”, *Frontiers in Communications and Networks*, vol. 2, 2021. <https://doi.org/10.3389/frcmn.2021.614166>
- [4] X. V. Wang, and L. Wang, “A literature survey of the robotic technologies during the COVID-19 pandemic”, *Journal of Manufacturing Systems*, 2021. <https://doi.org/10.1016/j.jmsy.2021.02.005>
- [5] Ainstein, “5 IoT implementations for social distancing management,” [Online]. Available: <https://ainstein.ai/ebook-5-iot-implementations-for-social-distancing-management/> [Accessed July 16, 2021].

# Estimation of Scattering Parameters of U-Slotted Rectangular RFID Patch Antenna with Machine Learning Models

Ismail AKDAG<sup>1</sup>

<sup>1</sup> Izmir Katip Celebi University, Graduate School of Natural and Applied Sciences, Electrical and Electronics Engineering, Turkey

## Abstract

In this study, machine learning-based models have been used to estimate the return loss parameters of the operational resonant frequency of the U-slotted UHF RFID antenna. The data set utilized, consisting of 544 instances, has been collected from the simulation software as a consequence of the parametric evaluation of the antenna design parameters. Distinct machine learning methods have been used on two different types of output data, complex and linear scattering parameters, and the models' prediction performance has been evaluated. In the single-output regression models, a mean-square error value of 0.25% with an  $R^2$  value of 95.54% was obtained with the Random Forest regression model, and a mean-square error value of 0.85% has been obtained with an  $R^2$  value of 91.32% in the multiple-output regression technique.

**Keywords:** *Machine learning, radio frequency identification, regression, RFID, scattering parameters.*

## 1. Introduction

Machine learning methods have been used frequently in many fields such as science, economics, engineering, and healthcare. Machine learning is a powerful tool that can be used to predict desired data with statistical and mathematical methods. Like machine learning, antenna design is also carried out as a result of many mathematical and statistical studies. The dimensions and weights of wireless systems have shrunk due to advancements in circuit technology, and as a result, antennas, which are crucial parts of wireless systems, have downsized. Antenna designs are often created using 3D electromagnetic simulation software. These applications offer verified antenna performance data by employing the required mathematical methodologies. Many pieces of information, such as an antenna's return loss, far field results, input impedance value, gain, and radiation pattern, may be acquired using 3D electromagnetic simulation tools. As previously stated, because the antenna design is the result of several mathematical and statistical methods, the time spent by simulation programs to provide the necessary performance data increases as the complexity of the antenna topology increases. Also, while designing the antenna, many parametric studies are also carried out. However, with data sets created correctly with machine learning techniques, accurate result data can be obtained in a shorter time. For this reason, the motivation of this study is to save time while determining the optimum design in complex structures by using different machine learning models. While performing the study, an RFID (radio frequency identification) antenna design has been chosen. RFID antennas are used with RFID systems to identify the desired object, person, or any device. As a general structure, an RFID tag containing an RFID antenna is placed on an object to be identified or tracked. The end user obtains the necessary information about the object by providing the necessary communication with the RFID reader and another RFID antenna connected to this reader. In the literature, there are examples of antenna design with artificial intelligence methods such as machine learning [1], [2], artificial neural networks, and deep learning.

In the study conducted by Muñoz et al., [3], the SVR technique has been used for estimating the antenna array design in 2016. In 2019, Khan et al. [4] used a machine learning algorithm to optimize the slot width and length in a microstrip antenna structure by taking into account the near-field radiation of antennas. Fei-Yan et al., in 2018, have used the SVM technique based on density optimization and hybrid kernel function for modeling the antenna operating resonant frequency [5]. Deep learning studies have produced substantial excellent outcomes in feature extraction and classification; [6], [7] and provided a high advantage over manual feature extraction and classification algorithms. In addition, deep learning algorithms have also been used in segmentation [8], [9], multi-object tracking [10], [11], and biomedical [12], [13] applications. To give an example for biomedical applications, Phasukkit et al. [14] proposed a triple coaxial-half-slot antenna scheme with deep learning-based temperature prediction for hepatic microwave ablation. In 2020, machine learning models were used for estimating the scattering parameters of RFID antenna by Akdag et al. [15]. In the study conducted by Koziel et al. in 2021 [16], a novel approach to global optimization of multi-band antennas has been presented. The main component of the framework in the study is the knowledge-based inverse surrogate constructed at the level of response features. With this study, the average optimization cost is only 150 full-wave antenna analyses while ensuring precise allocation of the antenna resonance at target frequencies. Also, in literature, there are studies for optimization

\*Corresponding author e-mail address: [ismail.akdag@ikcu.edu.tr](mailto:ismail.akdag@ikcu.edu.tr)

methods with the simulation-driven antenna design procedure. In 2021, Zhou et al. [17] presented work about a trust-region parallel Bayesian optimization method for simulation-driven antenna design problems. The Bayesian optimization method has also been used by Calik et al. in 2021 [18] for modeling frequency selective surfaces with the fully-connected regression model for automated architecture determination and parameter selection. In 2021, Koziel et al. [19] presented the improved modeling of microwave structures using performance-driven fully-connected regression surrogates. With surrogates, simulation-driven design procedures can be accelerated, and the CPU cost of electromagnetic analyses can be decreased.

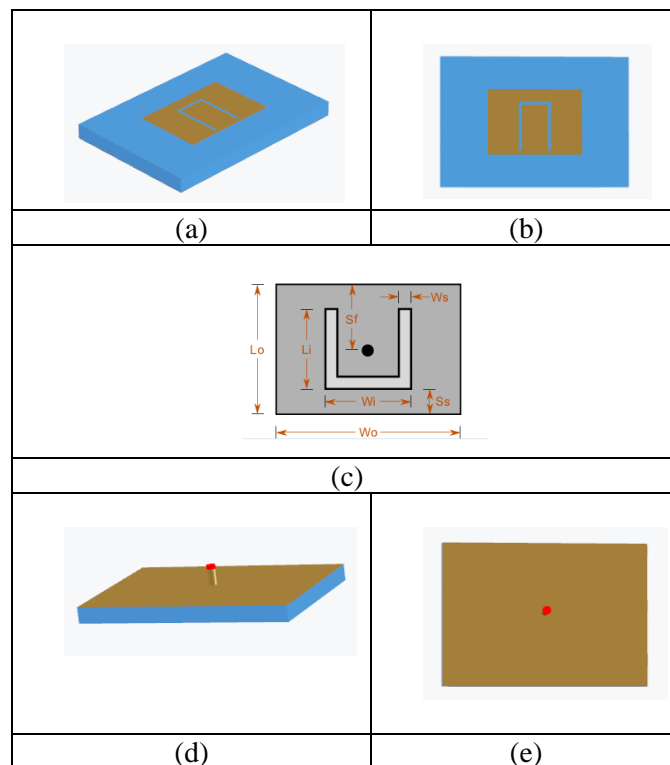
As a result, different artificial intelligence and machine learning models have been used frequently in the field of antenna design, as in many areas in the literature, and provide reliable data.

## 2. Methodology

In this section, the antenna design and input-output data used in these models are presented together with the machine learning models. While the input parameters in the models are the antenna design parameters, the output parameters are the linear and complex states of the scattering parameter  $S_{11}$ . Also, detailed information about the data set created for the antenna is given in this section.

### 2.1. U-Slotted RFID Antenna Design

The antenna design used in the study was obtained through the Antenna Magus program. Antenna Magus has a dataset with many antenna design data in it and verified models can be simulated by importing them into CST Studio Suite. Because of their simplicity and compatibility with circuit board technology, microstrip antennas, also known as patch antennas, are highly common in the microwave frequency range. One of the most utilized microstrip antennas is the pin-fed rectangular patch employed in the study. The necessary parameters for antenna design have been presented in Figure 1.



**Figure 1.** Proposed antenna design; a) Perspective view of the antenna, b) Top layer of antenna, c) Design parameters of antenna top layer, d) Bottom perspective view of the antenna, e) Bottom view of the antenna.

The top layer of the antenna design (Fig. 1.a and Fig. 1.c) contains the radiating part of the U-slot patch antenna. In the obtained antenna design, PEC material with a thickness of 0.035 mm was used as the conductor, and the thickness of the substrate material is 2.8 mm. The antenna's operating frequency can be changed by adjusting the length of the patch on the antenna. At the same time, the width of the patch has an effect on the antenna bandwidth.

The bandwidth of the antenna can be changed with the length of the U-shaped slot structure on the patch antenna. For the ground structure of the antenna, PEC material with the same thickness has been used and placed in such a way that it covers the same area as the substrate material. The design parameters of the antenna have been shown in Table 1 in detail.

**Table 1. Proposed antenna design parameters**

$W_i$	$L_i$	$S_s$	$W_s$	$W_0$	$L_0$	$D_f$	$H_s$
5.83	10.12	2.6	0.57	19.4	13.5	1.52	2.8

## 2.2. U-Slotted Patch RFID Antenna Design Scattering Parameters Machine Learning Algorithms

A data set has been created for U-slotted patch RFID antenna design with parametric studies, and the detailed data set has been presented in Table 2. While the geometric parameters of the antenna have been used for input data, the scattering parameter calculated for related input has been used for output data.

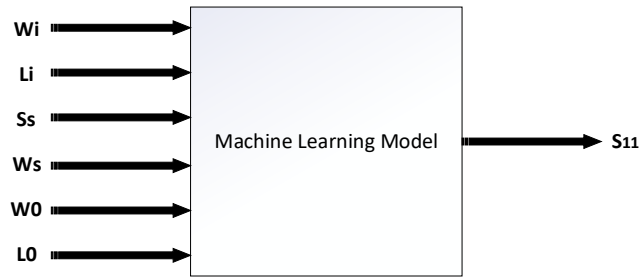
**Table 2. Antenna Design Parameters Data Set**

Parameter		Step Size
$W_i$	[2 25] (mm)	1.5 mm
$L_i$	[10.12 10.13] (mm)	0.01 mm
$S_s$	[0 15] (mm)	1.2 mm
$W_s$	[0.2 1.2] (mm)	0.07 mm
$W_0$	[19.48 30] (mm)	2.6 mm
$L_0$	[6 30] (mm)	1.6 mm
<b>Total Data</b>	544	

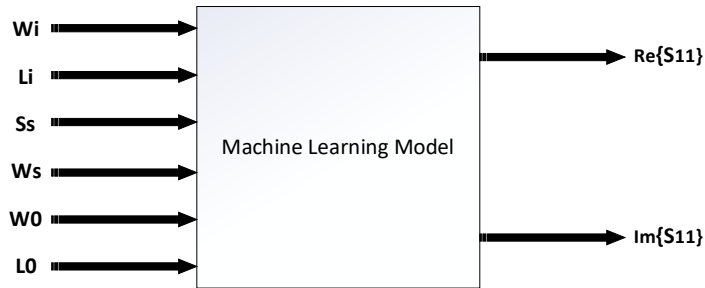
The design parameters have been determined as in Table 2. Here,  $W_i$  is the width of the inner slot, and  $L_i$  is the length of the slot,  $W_s$  is the thickness of the slot.  $W_0$  and  $L_0$  values indicate the outer length and width of the antenna, respectively. The  $S_s$  value indicates the distance of the slot from the lowest part of the patch on the antenna. The data set contains 544 data and is divided as 34%-66% as test and training data.

In the 3D electromagnetic simulation program, the return loss,  $S_{11}$  value of the antenna can be obtained in both linear and complex form. While the linear scattering parameter can be evaluated as a single value as the output value, the complex scattering parameter has two parts, imaginary and real. Therefore different machine learning models have been constructed for different types of output data. In Figure 2, input and output values are shown in a single-output machine learning model, while Figure 3 shows a multi-output machine learning model. In both models, the input values are the design parameters of the antenna, while in Figure 2, the output data is the linear scattering parameter, and in Figure 3, the output data is the complex scattering parameter. Polynomial Regression, Random Forest, Gradient Boosting, Bayesian Ridge, and Voting Regressor have been used for the single output machine learning model, and Multiple Output Regression method has been used for the multiple output machine learning model. The simulation performance of the U-slotted RFID patch antenna has been evaluated on 544 different data. As a result of these simulations, the  $S_{11}$  reflection coefficient data, which determines the operating frequency of the antenna, have been obtained. Scattering parameters have been obtained in two different types, linear and complex, and when the data set has been examined, it has been seen that the data were suitable for regression methods. Although the instance of data in the data set is small, better results can be obtained with regression models by expanding the data set with more simulations.





**Figure 2.** Machine Learning model, input (RFID antenna design parameters), and output( linear scattering parameter value)



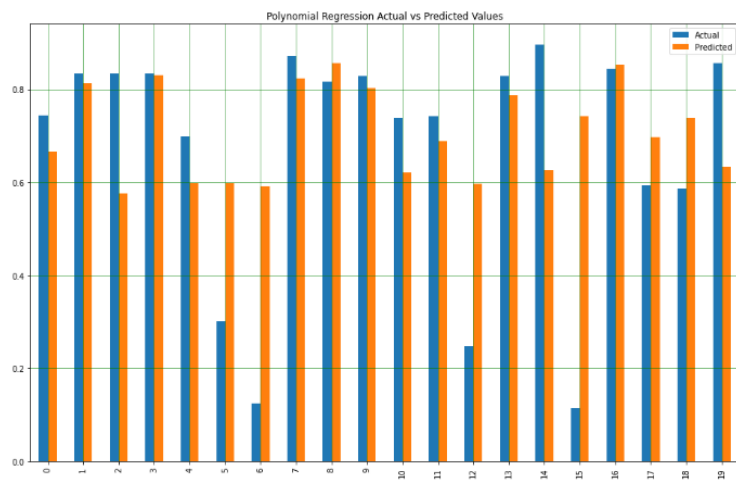
**Figure 3.** Machine learning model, input (RFID antenna design parameters), and output (complex scattering parameter values)

### 3. Numerical Results

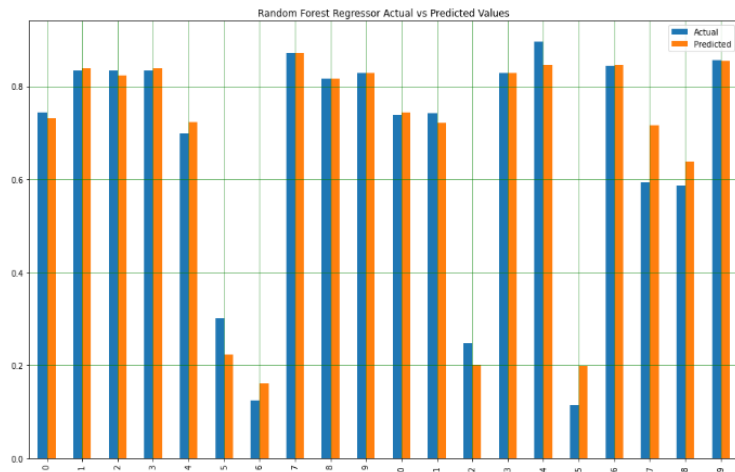
The findings of the various approaches used for the machine learning models depicted in Figures 2 and 3 are provided in this section. Machine learning methods have been written in Python programming language with the Sci-kit Learn library, and the prediction performances obtained from different methods have been compared.

#### 3.1. Regression results for single output $S_{11}$ value

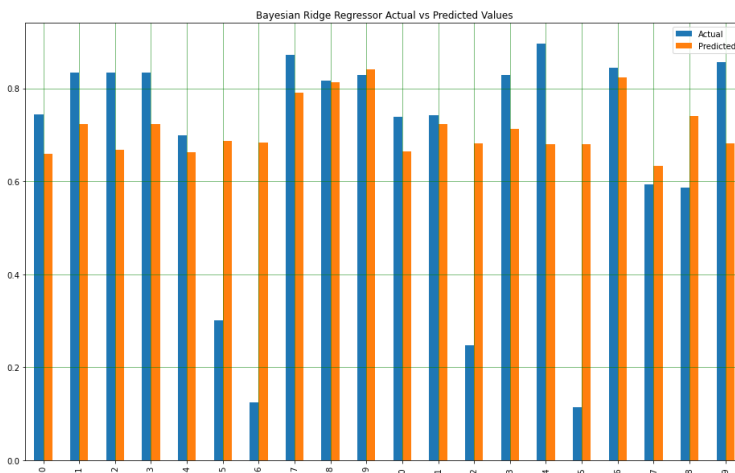
The estimation performance of different methods for the single output machine learning model of U-slotted RFID patch antenna design has been presented in this section. For seeing the estimation performance, 20 sample test instances have been used, and the actual and estimated output values have been presented for Polynomial Regression, Random Forest, Bayesian Ridge, and Gradient Boosting and Voting Regressor methods in Figure 4 – Figure 8, respectively.



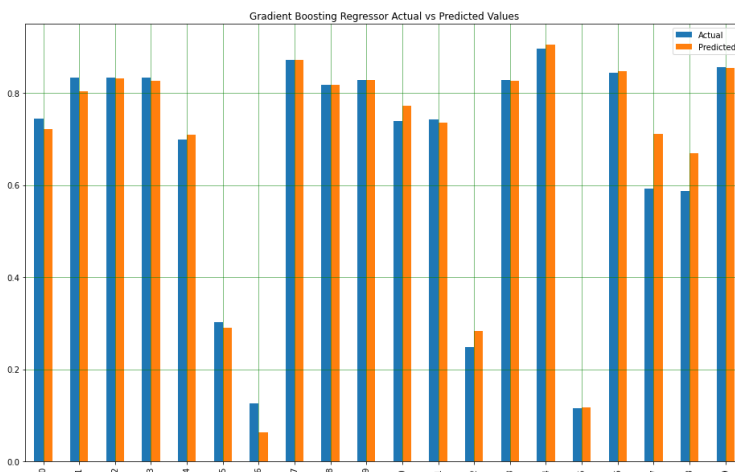
**Figure 4.** Polynomial Regressor Actual / Estimated Data



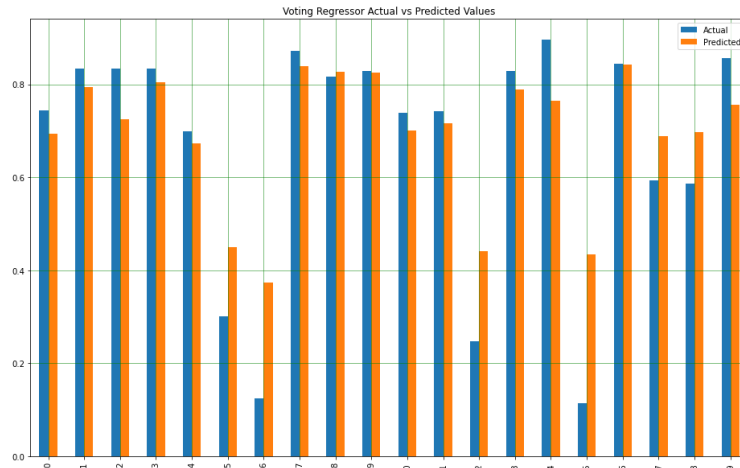
**Figure 5.** *Random Forest Regressor Actual / Estimated Data*



**Figure 6.** *Bayesian Ridge Regressor Actual / Estimated Data*



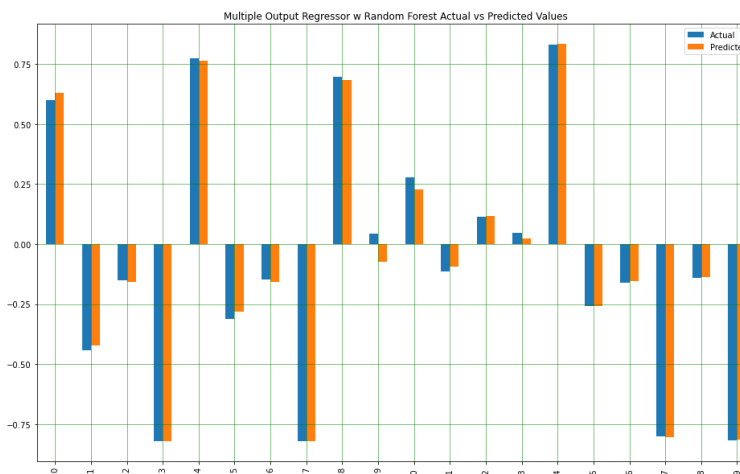
**Figure 7.** *Gradient Boosting Regressor Actual / Estimated Data*



**Figure 8.** Voting Regressor Actual / Estimated Data

### 3.2. Regression results for multiple output complex $S_{11}$ value

The estimation performance of the multi output regression method for  $S_{11}$  estimation of the presented U-slotted patch RFID antenna design is discussed in this section. Because the output value is composed of two data points, the Multi-Output (Figure 9) regression approach has been used. For 20 sample test instances, actual and estimated output values have been presented in Figure 9.



**Figure 9.** Multiple Output Regressor Actual / Estimated Data

## 4. Results and Conclusion

In this study, the estimation of the scattering parameters of a sample RFID antenna design obtained from the Antenna Magus program has been studied. The data set has been created by parametrically changing the input data in the antenna geometry with the help of a 3D electromagnetic simulation program and has a total of 544 instances. The scattering parameter data were obtained in two different forms, linear and complex. While the linear scattering parameter data has a single element, the complex scattering parameter has two parts, real and imaginary. For this reason, Polynomial Regression, Random Forest, Bayesian Ridge and Gradient Boosting methods are used for linear scattering parameter estimation, while multiple output regression method is used for complex scattering parameter estimation. The prediction performance performances obtained from single and multiple output machine learning methods are presented in detail.

Results	Polynomial Regressor
R2 Score %	18.2861811956603
Mean Squared Error %	4.621287953849866
Root Mean Squared Error %	21.497181103228087
Mean Absolute Error %	16.74708846728713
Maximum Error %	62.71797723250162
Results	Random Forest Regressor
R2 Score %	95.54023769162794
Mean Squared Error %	0.25221983422489297
Root Mean Squared Error %	5.022149283174415
Mean Absolute Error %	2.7945147333333327
Maximum Error %	36.18767999999999
Results	Gradient Boosting Regressor
R2 Score %	88.29153424072864
Mean Squared Error %	0.6621669695910813
Root Mean Squared Error %	8.137364251347492
Mean Absolute Error %	3.942763213148881
Maximum Error %	47.97144815549858
Results	Bayesian Ridge Regressor
R2 Score %	6.409634448190449
Mean Squared Error %	5.292960667480311
Root Mean Squared Error %	23.006435333359036
Mean Absolute Error %	17.724106280510114
Maximum Error %	58.29358260760041
Results	Voting Regressor
R2 Score %	76.21253383604501
Mean Squared Error %	1.3452893579642409
Root Mean Squared Error %	11.598660948420903
Mean Absolute Error %	9.07985815233816
Maximum Error %	38.64203616952579
Results	Multi Output Regressor
R2 Score %	91.3280284583893
Mean Squared Error %	0.8547496425562304
Root Mean Squared Error %	9.245267127326448
Mean Absolute Error %	4.448649173722576
Maximum Error %	---

Figure 10. Regression models comparison table

Table 3. Multi-output regression technique test data and output values

Input Parameters $L_i, W_i, L_0, W_0, S_s, W_s$	Estimated Output Parameters ( $S_{11real}, S_{11img}$ )	Actual Output Parameters ( $S_{11real}, S_{11img}$ )
[10.13,5,10,19.5,2.6,0.3]	[0.784904 -0.193443]	[0.799569 -0.117085]
[10.13,10,25,19.5,2.6,1.2]	[0.8240994 -0.2392208]	[0.826512-0.240428]
[10.12,6.5,13.5,19.48,2.5,0.65]	[0.124008 -0.089705]	[0.133303-0.123932]

When the estimation performances of different machine learning methods are examined, it is seen that the best estimation performance is obtained in the Random Forest method. Figure 10 includes the comparison of the estimation performances of all methods. Here, it is seen that the estimation performance is not good for Bayesian Ridge and Polynomial Regression methods. For this reason, it would be more appropriate to use the Random Forest method for the single output machine learning model. With the multiple output regression method, 91.32%  $R^2$  value has been obtained. Table 3 presents the actual input and actual output / estimated output values for the sample data. However, expanding the number of instances in the data set used in this study will result in more precise results. At the same time, utilizing machine learning's predictive performance, these approaches may be applied to various antenna designs, as well as developing antenna calculation software.

### Declaration of Interest

The authors declare that there is no conflict of interest.

### References

- [1] C. R. M. Silva and S. R. Martins, "An Adaptive Evolutionary Algorithm for UWB Microstrip Antennas Optimization Using a Machine Learning Technique," *Microw. Opt. Technol. Lett.*, vol. 55, no. 8, pp. 1864–1868, 2013, doi: <https://doi.org/10.1002/mop.27692>.

- [2] Z. Zheng, X. Chen, and K. Huang, "Application of support vector machines to the antenna design," *Int. J. RF Microw. Comput. Eng.*, vol. 21, no. 1, pp. 85–90, 2011, doi: <https://doi.org/10.1002/mmce.20491>.
- [3] J. Á. Muñiz, R. G. Ayestarán, J. Laviada, and F. Las-Heras, "Support vector regression for near-field multifocused antenna arrays considering mutual coupling," *Int. J. Numer. Model. Networks Devices Fields*, vol. 29, pp. 146–156, 2016.
- [4] T. Khan and C. Roy, "Prediction of slot-position and slot-size of a microstrip antenna using support vector regression," *Int. J. RF Microw. Comput. Eng.*, vol. 29, no. 3, p. e21623, 2019, doi: <https://doi.org/10.1002/mmce.21623>.
- [5] S. Fei-Yan, T. Yu-Bo, and R. Zuo-Lin, "Modeling the resonant frequency of compact microstrip antenna by the PSO-based SVM with the hybrid kernel function," *Int. J. Numer. Model. Electron. Networks, Devices Fields*, vol. 29, no. 6, pp. 1129–1139, 2016, doi: <https://doi.org/10.1002/jnm.2171>.
- [6] K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," *IEEE Int. Conf. Comput. Vis. (ICCV 2015)*, vol. 1502, 2015, doi: 10.1109/ICCV.2015.123.
- [7] R. Geirhos, D. Janssen, H. Schütt, J. Rauber, M. Bethge, and F. Wichmann, "Comparing deep neural networks against humans: object recognition when the signal gets weaker," 2017.
- [8] W. Zhu *et al.*, "AnatomyNet: Deep Learning for Fast and Fully Automated Whole-volume Segmentation of Head and Neck Anatomy," *Med. Phys.*, vol. 46, 2018, doi: 10.1002/mp.13300.
- [9] F. Liu, Z. Zhou, H. Jang, A. Samsonov, G. Zhao, and R. Kijowski, "Deep convolutional neural network and 3D deformable approach for tissue segmentation in musculoskeletal magnetic resonance imaging: deep learning approach for segmenting MR image," *Magn. Reson. Med.*, vol. 79, 2017, doi: 10.1002/mrm.26841.
- [10] B. Zhang, S. Li, Z. Huang, B. H. Rahi, Q. Wang, and M. Li, "Transfer learning-based online multiperson tracking with gaussian process regression," *Concurr. Comput. Pract. Exp.*, vol. 30, no. 23, p. e4917, 2018, doi: <https://doi.org/10.1002/cpe.4917>.
- [11] J. Son, M. Baek, M. Cho, and B. Han, "Multi-object tracking with quadruplet convolutional neural networks," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 3786–3795, 2017, doi: 10.1109/CVPR.2017.403.
- [12] Y. Moazzen, A. Çapar, A. Albayrak, N. Çalık, and B. U. Töreyn, "Metaphase finding with deep convolutional neural networks," *Biomed. Signal Process. Control*, vol. 52, pp. 353–361, 2019, doi: <https://doi.org/10.1016/j.bspc.2019.04.017>.
- [13] A. Albayrak *et al.*, "A whole slide image grading benchmark and tissue classification for cervical cancer precursor lesions with inter-observer variability," 2018.
- [14] P. Phasukkit and T. Wongketsada, "Triple coaxial-half-slot antenna scheme with deep learning-based temperature prediction for hepatic microwave ablation: finite element analysis and in vitro experiment," *IEEE Access*, vol. 9, pp. 79572–79587, 2021, doi: 10.1109/ACCESS.2021.3083088.
- [15] İ. Akdağ, C. Göçen, M. Palandöken, and A. Kaya, "Estimation of the scattering parameter at the resonance frequency of the UHF band of the e-shaped RFID antenna using machine learning techniques," in *2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, 2020, pp. 1–5, doi: 10.1109/ISMSIT50672.2020.9254362.
- [16] S. Koziel and A. Pietrenko-Dabrowska, "Global EM-driven optimization of multi-band antennas using knowledge-based inverse response-feature surrogates," *Knowledge-Based Syst.*, vol. 227, p. 107189, 2021, doi: <https://doi.org/10.1016/j.knosys.2021.107189>.
- [17] J. Zhou *et al.*, "A trust-region parallel bayesian optimization method for simulation-driven antenna design," *IEEE Trans. Antennas Propag.*, vol. 69, no. 7, pp. 3966–3981, 2021, doi: 10.1109/TAP.2020.3044393.
- [18] N. Calik, M. A. Belen, P. Mahouti, and S. Koziel, "Accurate modeling of frequency selective surfaces using fully-connected regression model with automated architecture determination and parameter selection based on bayesian optimization," *IEEE Access*, vol. 9, pp. 38396–38410, 2021, doi: 10.1109/ACCESS.2021.3063523.
- [19] S. Koziel, P. Mahouti, N. Calik, M. A. Belen, and S. Szczepanski, "Improved modeling of microwave structures using performance-driven fully-connected regression surrogate," *IEEE Access*, vol. 9, pp. 71470–71481, 2021, doi: 10.1109/ACCESS.2021.3078432.

# Optimization of Process Parameters for Green Composites in Abrasive Water Jet Machining Process Using Neuro-Regression Analysis

Serap TANRIVERDİ<sup>1</sup>, Levent AYDIN<sup>1,\*</sup>

<sup>1</sup> İzmir Katip Çelebi University, Faculty of Engineering and Architecture, Mechanical Engineering, Turkey

## Abstract

This study aims to develop a design procedure for optimizing the abrasive water jet machining (AWJM) process in green composites. Multiple non-linear neuro-regression analysis has been performed methodically to overcome insufficient approaches to modeling-design-optimizing green composites in AWJM. First, the model generation process is carried out according to three criteria: linearity, order, and functions used in the model. Next,  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$ , and  $R^2_{\text{validation}}$  values have been checked for the validity of the models. Then, the machining parameters have been optimized by applying a numerical non-linear global optimization algorithm, Simulated Annealing. Pressure within the pumping system (PwPS), stand-off distance (SoD), and nozzle speed (NS) are design variables; surface roughness (Ra) and process time (PT) are objective functions of introduced mathematical optimization problems. The numerical result shows that the optimum process parameters obtained are PwPS (150 MPa), SoD (3.5 mm), and NS (125 mm/min). This novel optimization approach is also feasible for another modeling design optimization problem. The proposed design can be used as a systematic framework for parameter optimization in environmentally conscious manufacturing processes.

**Keywords:** *Abrasive water jet machining; green composite; neuro-regression; optimization.*

## 1. Introduction

Polymer-based composites have been widely used in manufacturing industries for several years. However, the rising concerns towards the ecological issues and the need for more flexible materials led to the usage of polymer-based composites filled with natural organic fillers. Therefore, this type of material is often referred to as green composites (GC) or natural filler reinforcement composites (NFRC). These natural fillers are usually drawn from natural plant, animal, and renewable sources and have exceptional merits over the synthetic fillers/fibers such as low cost, renewable in nature, less abrasive, easy to be destroyed, lower specific weight, environment friendly, and also non-toxicity.

The past literature reveals the application of various forms of natural fillers such as flax, cellulose, cotton, sisal, kenaf, curaua, jute, banana, roselle, pineapple, bamboo, rice, and wood as reinforcing agents in order to improve the mechanical properties and obtain the properties needed in actual applications [1].

Peng P. and She D. study bamboo, a material, and various a compilation have been made about its potential and applications in the fields. In its internal structure, the hemicellulose structure was examined, and physical and thermal analysis by purification was made. The film layer is produced from pure hemicellulose [2].

In the research of Oksman and Selin [3], it is shown that the elastic modulus of wood fibers is approximately 40 times higher than that of polyethylene, and the strength is about 20 times higher. Nevertheless, many works devoted the mechanical behavior and machinability of the natural fiber-reinforced polymer composites none of the studies were found in the literature on machining behavior of the wood dust-based polymer composite.

Getu and Sahu [4] manufactured the composites which are undergone for testing of bending rigidity, shear, and absorbency. The tests conducted on the green composites developed reveal that they are suitable for usage fit for all users' fields. Furthermore, bagasse, banana, and sisal fibers have excellent tensile strength, allowing them to be used without finishing. Finally, it is concluded that it significantly reduces production and processing costs while also preserving the environment because the user and the environment will not be harmed during the manufacturing, processing, or disposal of this composite.

Sorgun [5] manufactured the composites with polypropylene matrix using two different particles (100  $\mu\text{m}$  below, 100-200  $\mu\text{m}$ ) obtained by grinding sandalwood. With the addition of 5% SA into PP, the tensile strength of PP increased by 10.2%. However, it was observed that the tensile strength decreased when the amount of SA added into the PP was increased. The highest value for the modulus of elasticity was determined in PP composite reinforced with less than 100-micron SA particles. 20% SA reinforced PP elasticity module increased 29.1% compared to PP elasticity module. When SA is added to PP, a decrease in the bending strength of PP has been determined in general, especially 100  $\mu\text{m}$  above PP. The opposite was seen in the bending modulus. Increasing the amount of SA increased the PP bending modulus [5].

\*Corresponding author e-mail address: [levent.aydin@ikcu.edu.tr](mailto:levent.aydin@ikcu.edu.tr)

Similarly, Gökdemir [6] manufactured the composites with polypropylene matrix using two different particles (100  $\mu\text{m}$  below, 100-200  $\mu\text{m}$ ) obtained by sugar beet pulp. It was observed that the tensile strength decreased when the amount of SP added into the PP was increased. In addition, it was observed that when the SP amount was increased, the modulus of elasticity increased. Thus, in general, as the amount of SP increased, bending strength decreased [6].

Raju et al. investigate the feasibility of using groundnut shell particles in the manufacture of composite panels. Groundnut shell particles were used to make composites with loadings of 20 percent, 30 percent, 40 percent, 50 percent, and 60 percent (by weight). The results showed that the panels might be made with up to 30% peanut husk without impairing their usage. Furthermore, because of its mild mechanical qualities, low moisture content, and low water absorption, groundnut shell particles can be used to substitute for wood in fabricating particle boards in the indoor environment [7].

AWJM is one of the most widely used non-conventional machining processes. This process is a very effective method for precision machining hard and brittle materials, and it is a non-contact, less inertia, and faster machining process. This offers various advantages such as reduced waste materials, less heat-affected zone, higher flexibility, versatility, and minimal force during machining [1]. However, the performance characteristic of this process is directly or indirectly influenced by the process parameters, which directly affects the efficiency of the manufacturing process. Thus, optimization of process parameters is a vital task to achieve green AWJM.

On the other hand, several researchers have profited from various optimization approaches for the AWJM process during the machining of different materials, such as the Taguchi method on machining of SKD61 mold steel [8], transformation-induced plasticity (TRIP) sheet steels, glass sheets, and glass/epoxy composite artificial neural network (ANN) model on machining of AA 7075 aluminum alloy [9]; ANN-genetic algorithm (GA) on machining of 6063-T6 aluminum alloy [10]; artificial neural network (ANN)-simulated annealing (SA) and SA-GA on machining of AA 7075 aluminum alloy [11, 12]; neuro-fuzzy approach on machining of 6063-T6 aluminum alloy [13]; analysis of variance (ANOVA) and Derringer-Suich multi-criteria decision modeling approach on machining of AISI 4340 and aluminum 2219 [14]; Taguchi-fuzzy decision method on machining of coal [15]; RSM with sequential approximation optimization (SAO) method on machining of alumina ceramic [16]; neural network (NN) model on machining of titanium [17]. However, none of the literature researchers have worked on optimizing process parameters for green AWJM on machining of WDFRP composites [1].

This study aims to introduce a design procedure for optimizing the abrasive waterjet machining (AWJM) process in green composites. First, multiple non-linear neuro-regression analysis was methodically performed to overcome the inadequate approaches to numerical part. The values of  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$ , and  $R^2_{\text{validation}}$  were checked for the validity of the models. Then, processing parameters are optimized by applying numerical non-linear global optimization algorithm, Simulated Annealing. Pressure within the pumping system (PwPS), stand-off distance (SoD), and nozzle speed (NS) are design variables; surface roughness (Ra) and processing time (PT) are objective functions of the introduced mathematical optimization problems. In order to perform this procedure, we used the experimental data in the study [1], which has been investigated before in the literature on this subject.

## 2. Methodology

### 2.1. Modelling

At the commencing of the current research, to reap the most efficient values for operational parameters, the modeling system has been implemented to receive the most potent mathematical model. In this way, a combination of artificial neural networks and regression analysis is used. In this approach, all data is divided into three sets comprising 80%, 15%, and 5% of the given data. The data's first, second, and third portions are used for training, testing, and validation, respectively. The training process aims to minimize the error between the experimental and predicted values by modifying the regression models and their coefficients given in Table 1. After the training part, the checking out procedure was once used to eliminate the uncertainties generated via the regression items. The feasibility of suitable models similar to  $R^2$  values is then checked by testing out if the models' severe elements are within the targeted interval of each running parameter. Next, the model analyses are made to appreciate main explanations such as model linearity, model order, and functions, affecting each the ability of prediction and the model's feasibility. After analyses are finished, a modified non-linear model is determined to use for primary regression analyses of output parameters. These modified items are shown in Table 2.



**Table 1.** Multiple Regression Model Types Including Linear, Second and Third Order; Exponential, Trigonometric, Logarithmic and Polynomial.

Model Name	Nomenclature	Formula
Multiple Linear	L	$F = a_0 + a_1x_1 + a_2x_2 + a_3x_3$
Second Order Multiple Nonlinear	SONL	$F = a_0 + a_1x_1 + a_2x_2 + a_3x_3 + a_4x_1^2 + a_5x_2^2 + a_6x_3^2 + a_7x_1x_2 + a_8x_1x_3 + a_9x_2x_3$
Third Order Multiple Nonlinear	TONL	$F = a_0 + a_1x_1 + a_2x_2 + a_3x_3 + a_4x_1^2 + a_5x_2^2 + a_6x_3^2 + a_7x_1x_2 + a_8x_1x_3 + a_9x_2x_3 + a_{10}x_1^3 + a_{11}x_2^3 + a_{13}x_3^3 + a_{14}x_1^2x_2 + a_{12}x_1^2x_3 + a_{13}x_2^2x_1 + a_{14}x_2^2x_3 + a_{15}x_3^2x_1 + a_{16}x_3^2x_2 + a_{17}x_1x_2x_3$
Second Order Multiple Nonlinear Logarithm	SOMNL	$F = a_0 + a_1Logx_1 + a_2Logx_2 + a_3Logx_3 + a_4Logx_1^2 + a_5Logx_2^2 + a_6Logx_3^2 + a_7Logx_1x_2 + a_8Logx_1x_3 + a_9Logx_2x_3$
Third Order Multiple Nonlinear Logarithm	TOMNL	$F = a_0 + a_1Logx_1 + a_2Logx_2 + a_3Logx_3 + a_4Logx_1^2 + a_5Logx_2^2 + a_6Logx_3^2 + a_7Logx_1x_2 + a_8Logx_1x_3 + a_9Logx_2x_3 + a_{10}Logx_1^3 + a_{11}Logx_2^3 + a_{13}Logx_3^3 + a_{14}Logx_1^2x_2 + a_{12}Logx_1^2x_3 + a_{13}Logx_2^2x_1 + a_{14}Logx_2^2x_3 + a_{15}Logx_3^2x_1 + a_{16}Logx_3^2x_2 + a_{17}Logx_1x_2x_3$
Second Order Multiple Nonlinear Trigonometric	SOMNT	$F = a_0 + a_1Sinx_1 + a_2Sinx_2 + a_3Sinx_3 + a_4Sinx_1^2 + a_5Sinx_2^2 + a_6Sinx_3^2 + a_7Sinx_1x_2 + a_8Sinx_1x_3 + a_9Sinx_2x_3$
Third Order Multiple Nonlinear Trigonometric	TOMNT	$F = a_0 + a_1Sinx_1 + a_2Sinx_2 + a_3Sinx_3 + a_4Sinx_1^2 + a_5Sinx_2^2 + a_6Sinx_3^2 + a_7Sinx_1x_2 + a_8Sinx_1x_3 + a_9Sinx_2x_3 + a_{10}Sinx_1^3 + a_{11}Sinx_2^3 + a_{13}Sinx_3^3 + a_{14}Sinx_1^2x_2 + a_{12}Sinx_1^2x_3 + a_{13}Sinx_2^2x_1 + a_{14}Sinx_2^2x_3 + a_{15}Sinx_3^2x_1 + a_{16}Sinx_3^2x_2 + a_{17}Sinx_1x_2x_3$
Second Order Multiple Nonlinear Exponential	SOMNE	$F = a_0 + a_1e^{x_1} + a_2e^{x_2} + a_3e^{x_3} + a_4e^{x_1^2} + a_5e^{x_2^2} + a_6e^{x_3^2} + a_7e^{x_1x_2} + a_8e^{x_3x_2} + a_9e^{x_1x_3}$
Third Order Multiple Nonlinear Exponential	TOMNE	$F = a_0 + a_1e^{x_1} + a_2e^{x_2} + a_3e^{x_3} + a_4e^{x_1^2} + a_5e^{x_2^2} + a_6e^{x_3^2} + a_7e^{x_1x_2} + a_8e^{x_3x_2} + a_9e^{x_1x_3} + a_{10}e^{x_2^3} + a_{11}e^{x_1^3} + a_{12}e^{x_3^3} + a_{13}e^{x_2^2x_1} + a_{14}e^{x_2^2x_3} + a_{15}e^{x_1^2x_2} + a_{16}e^{x_1^2x_3} + a_{17}e^{x_3^2x_1} + a_{18}e^{x_3^2x_2}$

**2.2. Optimization**

Optimization is the process of obtaining the most appropriate solution by providing certain constraints in line with the given purpose or objectives. To express it mathematically; Optimization can be briefly defined as minimizing or maximizing a function. Also, optimization can maximize productivity, strength, reliability, longevity, efficiency, and utilization. The techniques used in an optimization problem can be categorized into traditional and non-traditional. The traditional method starts with the initial solution and with each successive iteration converges to the optimal solution. This convergence depends on the selection of initial approximation. These methods are not suited for discontinuous objective function. So, the need for a non-traditional method was felt [18-20]. The most widely used non-traditional optimization methods are genetic algorithms, simulated annealing, and particle swarm. The reliability of the outcomes taken from a non-traditional (stochastic)

optimization evaluation can also be improved via utilizing a couple of methods. The most difficult mathematical optimization problems have the following issues:

- Multiple non-linear objective functions,
- Objective functions having many local extremum points,
- Mixed-integer (discrete)-continuous nature of the design variables, and
- Non-linear constraints [18].

In this paper, the optimization scenarios mentioned include the challenges given in the first three items.

**2.2.1 Problem Definition**

By using the above-described methods, the optimal analysis of process parameters of green composite in AWJM was organized as follows;

- The data shown in Table 3 are from the reference study [1]. They have modeled the processing time and surface roughness input parameters with Response Surface Method.
- Three base functional structures were proposed for modeling, and the boundedness of the functions was evaluated for appropriateness in terms of  $R^2_{training}$ ,  $R^2_{testing}$ , and  $R^2_{validation}$  values.
- A new updated non-linear model is generated for each of the output parameters from the result of the base models, and then, these modified models are also tested in terms of  $R^2$  values.
- Two different optimization scenarios were introduced using the appropriate models, which were solved by a direct search method.

**2.2.2 Optimization Scenarios**

**Scenario (a)** In this optimization problem, all the design variables are assumed to be real numbers for all the objective functions, and the search space is continuous. For this case, the constraints are  $150 < PwPS < 300$ ,  $1.5 < SoD < 3.5$ ,  $125 < NS < 225$ . The main goal is to get optimum values for objective functions. Mathematically, limits of the objective function can also be obtained by this approach.

**Scenario (b)** Based on only the prescribed experimental setup, more specific optimization problem can also be defined as involving (i) optimization of objective functions, (ii) all the design variables are assumed to be real numbers, and (iii) the constraints are  $PwPS \in \{150, 225, 300\}$ ;  $SoD \in \{1.5, 2.5, 3.5\}$ ;  $NS \in \{125, 175, 225\}$ .

**Table 2. Model Formula for Output.**

Output Name	Modified Model Formula	Model
Ra	$F = a_0 + a_1 \text{Log}x_1 + a_2 \text{Log}x_2 + a_3 \text{Log}x_3 + a_4 \text{Log}x_1^2 + a_5 \text{Log}x_2^2 + a_6 \text{Log}x_3^2 + a_7 \text{Log}x_1x_2 + a_8 \text{Log}x_1x_3 + a_9 \text{Log}x_2x_3$	$\begin{aligned} & -1.094721042052923 \\ & + 0.048718366176467354\text{Log}[x_1] \\ & + 0.024359183088233656\text{Log}[x_1^2] \\ & - 0.047892896045495015\text{Log}[x_2] \\ & + 0.03550664102887537\text{Log}[x_1x_2] \\ & - 0.023946448022747608\text{Log}[x_2^2] \\ & + 0.030129863589708566\text{Log}[x_3] \\ & + 0.019986916583215445\text{Log}[x_1x_3] \\ & + 0.021228411293550627\text{Log}[x_2x_3] \\ & + 0.015064931794854283\text{Log}[x_3^2] \end{aligned}$
PT	$F = a_0 + a_1 \text{Log}x_1 + a_2 \text{Log}x_2 + a_3 \text{Log}x_3 + a_4 \text{Log}x_1^2 + a_5 \text{Log}x_2^2 + a_6 \text{Log}x_3^2 + a_7 \text{Log}x_1x_2 + a_8 \text{Log}x_1x_3 + a_9 \text{Log}x_2x_3$	$\begin{aligned} & -1.9849045387424702 \\ & - 0.01071578210330012\text{Log}[x_1] \\ & - 0.005357891051649882\text{Log}[x_1^2] \\ & - 0.12693642357351628\text{Log}[x_2] \\ & - 0.010586200398944654\text{Log}[x_1x_2] \\ & - 0.06346821178675836\text{Log}[x_2^2] \\ & + 0.15758114439411514\text{Log}[x_3] \\ & + 0.03462614296942368\text{Log}[x_1x_3] \\ & + 0.11375733676330331\text{Log}[x_2x_3] \\ & + 0.07879057219705755\text{Log}[x_3^2] \end{aligned}$

**Table 3.** *Experimental Data [1].*

Exp. No	Inputs			Outputs	
	PwPS (MPa)	SoD (mm)	NS (mm/min)	Ra	PT
1	300	1.5	175	0.2761	0.4012
2	150	2.5	225	0.1980	0.4125
3	300	2.5	125	0.2570	0.1058
4	150	3.5	175	0.1290	0.1853
5	300	2.5	225	0.2981	0.4663
6	150	1.5	175	0.1889	0.3250
7	225	2.5	175	0.1957	0.2568
8	225	3.5	225	0.2456	0.3425
9	225	2.5	175	0.2214	0.3805
10	225	1.5	125	0.1953	0.1569
11	150	2.5	125	0.1190	0.1137
12	225	1.5	225	0.2640	0.4472
13	300	3.5	175	0.2583	0.1665
14	225	2.5	175	0.2012	0.2912
15	225	3.5	125	0.1630	0.1026

### 3. Results

#### 3.1 Determination of main effects on surface roughness (Ra)

In this section, the effect of machining parameters on Ra has been determined using the *NonlinearModelFit* solver of Wolfram Mathematica for the experimental Ra. It has been observed from Table 3 that the variables PwPS (x1) and NS (x3) have positive effects and SoD (x2) harms the Ra. The process variables PwPS and NS are the most influencing parameters and can predict the Ra within the control limits. The value of  $R^2$  is calculated to be 0,996642 for Ra (See Table 4). Higher the  $R^2$  coefficient gives a satisfactory model to the experimental data. Finally, the adequacy and fitness of the model are calculated by adjusted  $R^2$  values. The high value of adjusted  $R^2$  (0,996642) for Ra indicates that the number of experimental data used to develop the model is compatible with the relevant model. It is seen that the surface roughness is mainly affected by PwPS and NS. Surface roughness is increased significantly from 0.125728 to 0.285823  $\mu\text{m}$  as PwPS is increased by 150 to 300 MPa. As for NS, a slight increment of surface roughness occurred when NS is increased from 125 to 225 mm/min with 0.125728 to 0.26733  $\mu\text{m}$ .

Meanwhile, surface roughness is observed to decrease as SoD increased from 1.5 to 3.5 mm. Since PwPS and NS showed higher percentage contribution than other factors (SoD), they can be considered the most significant to the surface roughness. Therefore, the calculations are performed by increasing the PwPS and NS from 150 to 300 MPa and simultaneously by decreasing SoD from 3.5 to 1.5 mm. Finally, the optimal surface roughness is obtained when machining parameters set at PwPS, SoD, and NS are 300 MPa, 1.5 mm, and 225 mm/min, respectively (See Table 5).

#### 3.2 Determination of main effects on process time (PT)

In this section, the effect of machining parameters on PT has been determined similar to that of Ra. It has been observed from Table 3 that the variables NS (x3) have a positive effect, SoD (x2) and PwPS (x1) harm the PT. The process variables NS are the most influencing parameters and can be used to predict the PT within the control limits. The  $R^2$  is 0.979413 for PT. It has been examined from the Table 3 that the process time is mainly affected

by NS. The processing time is increased significantly from 0.0785736 to 0.478729 s as NS was increased by 125 to 225 mm/min.

Meanwhile, the processing time is observed to decrease as SoD increased from 1.5 to 3.5 mm. Since NS shows a higher percentage contribution than the other factors (SoD), they can be considered most significant to the processing time. Therefore, the alternative calculations are also performed by increasing the NS from 125 to 225 mm/min and simultaneously decreasing SoD from 3.5 to 1.5 mm. Finally, the optimal process time is obtained when machining parameters are set at SoD of 1.5 mm and NS of 225 mm/min.

**Table 4.** The candidate model results and their  $R^2$  values.

Minimum Value of PT	Model	$R^2_{\text{training}}$	$R^2_{\text{testing}}$	$R^2_{\text{validation}}$
0.0775151	TOMNL	0.995707	0.519515	-0.161361
0.815783	SONL	0.995707	-0.285542	-3.06543
0.027278	TONL	0.995707	0.0795861	-2.11266
-0.509278	SOMNT	0.995707	-0.462265	0.45073
-0.701173	TOMNT	0.995707	-1.07981	-8.572680
-0.369327	TOMNT	0.995707	-0.0326963	-2.12837
-7.08517	SOMNT	0.995707	0.619977	-6.36817
-0.104237	TOMNE	0.960279	$-2.02549 \times 10^6$	-2.36169
$-1.02783368 \times 10^{11}$	SOMNE	0.960279	$-2.82425 \times 10^6$	0.435343
0.0300375	M1	0.987399	0.653279	0.974805
-0.147157	M2	0.995707	0.862176	-2.47803
0.0696715	M3	0.987399	0.653279	0.974805
-0.017173	M4	0.995707	0.034263	-1.32716
-0.0595981	M5	0.987399	-1.21209	-6.18308

**Table 5.** Results of the optimization problems for the selected models.

Response Parameters	Goal	Optimum			Target	Model
		PwPS	SoD	NS		
Ra	Minimum	150	3.5	125	0.119	0.112589
PT	Minimum	150	3.5	125	0.1026	0.077516

**4. Conclusions**

This work proposed a design optimization based on non-linear multiple neuro regression analysis for machining of a biocomposite in AWJM processes. Thus, a novel approach based on a modeling-design-optimization process to design an optimum surface roughness and process time has been introduced. The purpose of the research study is to reveal the regression model investigated as the best model to predict the experimental results of Ra, PT on input parameters and then optimize inputs. A direct search technique, Simulated Annealing, is used including stochastic approaches, during the optimization process. The numerical results that optimum cutting parameters obtained are 150 MPa, 3.5 mm, and 125 mm/min for PwPS, SoD, and NS, respectively. The results also indicate that the parameters PwPS and NS are the most significant factors for Ra, while only parameter NS for PT. Therefore, it is concluded that the developed mathematical model is significant and adequate for process parameter selection and prediction of AWJM output parameters on the machining of green composites. Thus, this approach can be used as a systematic framework model for predicting response parameters in green manufacturing applications and helps in selecting optimal machining parameters in practical work for green manufacturing industries.

**Declaration of Interest**

The authors declare that there is no conflict of interest.

## References

- [1] B. Jagadish, S. Bhormik, A. Ray, "Prediction and optimization parameters of green composites in AWJM process using response surface methodology," *The International Journal of Advanced Manufacturing Technology*, vol. 87, Nov., pp. 1359-1370, 2016.
- [2] P. Peng, D. She, "Isolation, structural characterization, and potential applications of hemicelluloses from bamboo: A Review," *Carbohydrate Polymers*, vol. 112, July, pp.701-720, 2014.
- [3] K. Oksman, J. F. Selin, *Plastics and composites from polylactic acid*, in *Natural Fibers, Plastics and Composites*. Boston, MA: Springer US, 2004.
- [4] A. Getu, O. Sahu, "Green composite material from agricultural waste," *International Journal of Agricultural Research and Reviews*, vol. 2, no.5, June, pp. 56-62, 2014.
- [5] A. Sorgun, "Manufacturing and characterization of sandalwood filled polypropylene composite," MSc. Thesis, pp. 1-35, 2019.
- [6] B. Gökdemir, "Investigation of usability of sugar beet pulp in biocomposite production, MSc. Thesis, pp. 14-49, 2020.
- [7] G. U. Raju, S. Kumarappa, V. N. Gaitonde, "Mechanical and physical characterization of agricultural waste reinforced polymer composites," *Journal of Materials and Environmental Science*, vol. 3, no. 5, June, pp. 907-916, 2012.
- [8] F. C. Tsai, B. H. Yan, C. Y. Kuan, F. Y. Huang, "A Taguchi and experimental investigation into the optimal processing conditions for the abrasive jet polishing of SKD61 mold steel," *Int. J. of Machine Tools and Manuf.*, vol. 48, no. 7-8, June, pp. 932–945, 2008.
- [9] U. Caydas, A. Hasçalık, "A study on surface roughness in abrasive water jet machining process using artificial neural net-works and regression analysis method," *Journal of Materials Processing Technology*, vol. 202, no. 1-3, June, pp. 574–582, 2008.
- [10] D. S. Srinivasu, N. R. Babu, "A neuro-genetic approach for selection of process parameters in abrasive waterjet cutting considering variation in diameter of focusing nozzle," *Applied Sofy Computing*, vol. 8, no. 1, Jan., pp. 809–819, 2008.
- [11] A. M. Zain, H. Haron, S. Sharif, "Estimation of the minimum machining performance in the abrasive waterjet machining using integrated ANN-SA," *Expert Systems with Applications*, vol. 38, no. 7, July, pp. 8316–8326, 2011.
- [12] A. M. Zain, H. Haron H, S. Sharif, "Optimization of process parameters in the abrasive waterjet machining using integrated SA–GA," *Applied Soft Computing*, vol. 11, no. 8, Dec., pp. 5350–5359, 2011.
- [13] J. J. R. Jegarah, N. R. Babu, "A soft computing approach for controlling the quality of cut with abrasive waterjet cutting system experiencing orifice and focusing tube wear," *Journal of Materials Processing Technology*, vol. 185, no. 1-3, April, pp. 217–227, 2007.
- [14] A. Iqbal, N. U. Dar, G. Hussain, "Optimization of abrasive water jet cutting of ductile materials," *Journal of Wuhan University of Technol Mater. Sci. Ed.*, vol. 26, no. 1, March, pp. 88–92, 2011.
- [15] V. Sharma, S. Chattopadhyaya, S. Hloch, "Multi response optimization of process parameters based on Taguchi-fuzzy model for coal cutting by water jet technology," *Int J Adv Manuf Technol*, vol. 56, March, pp. 1019–1025, 2011.
- [16] Z. Yue, C. Huang, H. Zhu, J. Wang, P. Yao, Z. Liu, "Optimization of machining parameters in the abrasive waterjet turning of alumina ceramic based on the response surface methodology," *Int J Adv Manuf Technol*, vol. 71, Feb., pp. 2107–2114, 2014.
- [17] H. S. Ergur, Y. Oysal, "Estimation of cutting speed in abrasive water jet using an adaptive wavelet neural network," *Journal of Intelligent Manufacturing*, vol. 26, June, pp. 403–413, 2013.
- [18] İ. Polatoglu, L. Aydın, B. Ç. Nevruz, S. Ozer, "A Novel Approach for the Optimal Design of a Biosensor," *Analytical Letters*, vol. 53, no. 9, Jan., pp. 1428-1445, 2020.
- [19] Jose, J. P.; Thomas, S.; Kuruvilla, J. ; Malhotra, S. K. ; Goda, K., & Sreekala, M. S. (2012). *Advances in polymer composites: macro-and micro composites—state of the art, new challenges, and opportunities*. Polymer Composites; Wiley: Weinheim, Germany, 1, 3-16.
- [20] Kumar, R., Jagtap, H. P., Rajak, D. K., & Bewoor, A. K. (2020). *Traditional and Non-Traditional Optimization Techniques to Enhance Reliability in Process Industries*. In *AI Techniques for Reliability Prediction for Electronic Components* (pp. 67-80). IGI Global.

APPENDIX

Model Name	Model
TOMNL	$  \begin{aligned}  & -2.806528296855611 + 0.24576645769296207\text{Log}[x1] + 0.04895764577773693\text{Log}[x1]^2 \\  & - 0.015108744635239061\text{Log}[x1]^3 - 2.3375909888328423\text{Log}[x2] \\  & + 0.37156869342381355\text{Log}[x1]\text{Log}[x2] \\  & - 0.007974927955401436\text{Log}[x1]^2\text{Log}[x2] + 0.7544078020289587\text{Log}[x2]^2 \\  & - 0.7427468406531201\text{Log}[x1]\text{Log}[x2]^2 + 3.7945776182670783\text{Log}[x2]^3 \\  & + 0.06454365663401485\text{Log}[x3] + 0.07552076059510957\text{Log}[x1]\text{Log}[x3] \\  & + 0.00015695539522465223\text{Log}[x1]^2\text{Log}[x3] \\  & + 0.2976846759003845\text{Log}[x2]\text{Log}[x3] \\  & + 0.1123559335083702\text{Log}[x1]\text{Log}[x2]\text{Log}[x3] \\  & - 1.2931833357284106\text{Log}[x2]^2\text{Log}[x3] + 0.022028436007974182\text{Log}[x3]^2 \\  & + 0.0005379959833349629\text{Log}[x1]\text{Log}[x3]^2 \\  & + 0.13856252558673193\text{Log}[x2]\text{Log}[x3]^2 - 0.015170136062410909\text{Log}[x3]^3  \end{aligned}  $
SONL	$  \begin{aligned}  & -0.8877456875662202 + 0.008241494259063021x1 - 0.000011435172427547529x1^2 \\  & - 0.28367413885945597x2 - 0.0009099999999999931x1x2 \\  & + 0.04247284490495425x2^2 + 0.006194517222810836x3 \\  & - 0.0000069866666666666543x1x3 + 0.001296913796198187x2x3 \\  & - 0.000015430862038018096x3^2  \end{aligned}  $
TONL	$  \begin{aligned}  & -0.26427947911334526 + 0.0016701981156721608x1 + 0.00000490004321900638x1^2 \\  & - 1.436362198383393 \times 10^{-8}x1^3 - 0.1457247612928245x2 \\  & + 0.00021909952693669683x1x2 - 0.000001889806824953462x1^2x2 \\  & - 0.021970532359016786x2^2 - 0.000025565211545939657x1x2^2 \\  & + 0.008686457799320446x2^3 + 0.0016337594337852773x3 \\  & + 0.000008022624456241609x1x3 - 9.224429991622914 \times 10^{-9}x1^2x3 \\  & + 0.0003024891345997836x2x3 - 8.157681575534157 \times 10^{-8}x1x2x3 \\  & - 0.00003936983895849437x2^2x3 + 0.000007097804370059959x3^2 \\  & - 2.470454083985247 \times 10^{-8}x1x3^2 + 0.000003107469335376899x2x3^2 \\  & - 4.267499092338159 \times 10^{-8}x3^3  \end{aligned}  $
SOMNT	$  \begin{aligned}  & 0.3464813339886176 + 0.3831843213555272\text{Cos}[x1] - 0.6100846360385241\text{Cos}[x1]^2 \\  & - 0.49513849393352827\text{Cos}[x2] - 0.15714002906460747\text{Cos}[x1]\text{Cos}[x2] \\  & - 0.3962049453066922\text{Cos}[x2]^2 + 0.38278250379852247\text{Cos}[x3] \\  & - 0.10791087170147802\text{Cos}[x1]\text{Cos}[x3] + 0.6377570713976038\text{Cos}[x2]\text{Cos}[x3] \\  & - 0.4390785559194609\text{Cos}[x3]^2  \end{aligned}  $
TOMNT	$  \begin{aligned}  & 0.1686792618833355 + 0.26568673069337423\text{Cos}[x1] + 0.24262614319066322\text{Cos}[x1]^2 \\  & - 0.08130731078829158\text{Cos}[x1]^3 - 0.02442678715695238\text{Cos}[x2] \\  & - 0.34383374206498113\text{Cos}[x1]\text{Cos}[x2] + 0.08780713412470981\text{Cos}[x2]^2 \\  & + 0.06281327558625166\text{Cos}[x1^2x2] - 0.06868399476131787\text{Cos}[x1x2^2] \\  & - 0.030587651869661527\text{Cos}[x2^3] - 0.016378143999534658\text{Cos}[x3] \\  & - 0.14736509640310413\text{Cos}[x1]\text{Cos}[x3] + 0.1619725525663956\text{Cos}[x2]\text{Cos}[x3] \\  & - 0.30212965117886503\text{Cos}[x3]^2 + 0.039158811483846\text{Cos}[x1^2x3] \\  & + 0.02613820585098387\text{Cos}[x1x2x3] - 0.1347167581855755\text{Cos}[x2^2x3] \\  & + 0.02164619613189358\text{Cos}[x1x3^2] - 0.04538789588955367\text{Cos}[x2x3^2] \\  & + 0.10386828830712264\text{Cos}[x3^3]  \end{aligned}  $
TOMNT	$  \begin{aligned}  & 0.010956406333690832 - 0.012880869509242655\text{Sin}[x1] + 0.014056843488704831\text{Sin}[x1]^2 \\  & + 0.04217709102028845\text{Sin}[x1^3] + 0.014644987047005769\text{Sin}[x2] \\  & - 0.021633011430284007\text{Sin}[x1]\text{Sin}[x2] + 0.01039746477383733\text{Sin}[x2]^2 \\  & + 0.003971444244157852\text{Sin}[x1^2x2] + 0.008361173625667672\text{Sin}[x1x2^2] \\  & + 0.01214925494581947\text{Sin}[x2^3] - 0.03934453616300522\text{Sin}[x3] \\  & + 0.044824142835632026\text{Sin}[x1]\text{Sin}[x3] - 0.03896417187256429\text{Sin}[x2]\text{Sin}[x3] \\  & + 0.07476462148286063\text{Sin}[x3]^2 + 0.00881518563270643\text{Sin}[x1^2x3] \\  & - 0.0017988745823680223\text{Sin}[x1x2x3] - 0.044227527857748326\text{Sin}[x2^2x3] \\  & + 0.03145113293156068\text{Sin}[x1x3^2] + 0.09168632058997019\text{Sin}[x2x3^2] \\  & + 0.18111218870491896\text{Sin}[x3^3]  \end{aligned}  $
SOMNT	$  \begin{aligned}  & -1.506481581746801 + 0.6633113978936133\text{Sin}[x1] + 0.1985238688002158\text{Sin}[x1]^2 \\  & - 1.2260355188064873\text{Sin}[x2] - 0.942131397477979\text{Sin}[x1]\text{Sin}[x2] \\  & + 0.001238132309443814\text{Sin}[x2]^2 - 5.608721192285001\text{Sin}[x3] \\  & - 0.43609873206144806\text{Sin}[x1]\text{Sin}[x3] - 0.5709493827365565\text{Sin}[x2]\text{Sin}[x3] \\  & - 3.031580877733221\text{Sin}[x3]^2  \end{aligned}  $





# Modeling and Design Optimization to Determine the Mechanical Properties of a Recent Composite

Naciye Burcu KARTAL\*

*Department of Mechanical Engineering, Izmir Kâtip Çelebi University, Cigli, Izmir, Turkey*

## Abstract

This study proposes an appropriate optimization model for determining a new composite material's mechanical properties by neuro-regression analysis. This new composite material is obtained by combining hemp and polypropylene fibers. It was developed for the sector of upholstered furniture. First, different multiple regression models have been tried for input and output values. The  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$ ,  $R^2_{\text{validation}}$ , and minimum, maximum values were determined for each model. Then, the stochastic optimization approach is used to predict and optimize the mechanical properties of the new biocomposite system. Finally, multiple non-linear models determine the maximum tensile strength and elongation achievable within the constraints. It is found what the optimum input parameters are needed to achieve maximum tensile strength and elongation at break values of the material and that the type of scenario and the choice of constraints for design variables are critical in the optimization problem.

**Keywords:** *Composite material; mechanical properties; neuro-regression analysis; optimization*

## 1. Introduction

A composite material is made up of two materials that have distinct physical and chemical properties. When they are combined, they form a specialized material to perform a specific function, such as becoming stronger, lighter, or more resistant to electricity. One constituent is called the reinforcing phase, and the one in which it is embedded is called the matrix [1]. The matrix is reinforced with an engineered, man-made, or natural fiber, particle, or flake form reinforcing material. The matrix covers the fibers from environmental and exterior damage and transmits the load between the fibers. In turn, the fibers yield strength and stiffness to the matrix, preventing cracks and fractures.

Ciupan et al. [2] has studied the use of artificial neural networks (ANN) to predict certain mechanical properties of new composite material. The material is intended to be used to construct structural elements of upholstered furniture (chairs, armchairs, sofas) in place of wood. Ciupan and his group presented that optimizing these element's shapes using numerical simulation necessitates knowledge of the material's mechanical properties. These properties consist of tensile strength, elongation at break, Young's modulus, and Poisson's ratio. They conducted tests on the material samples and aimed to investigate how far ANN can predict the tensile strength and elongation at the break of the previously discussed composite material. Eventually, they concluded these results: In the case of elongation at break, the degree of fit between experimentally predicted output variables and those simulated using the ANN is greater than in the case of tensile strength. Throughout the recall phase, the outputs in group 2 represent the average values of the outputs used for training. For example, the simulated output  $(\sigma_M, \epsilon_M) = (25.03, 3.13)$  that correlates to the input (1,0) is equivalent to the average of the outputs in the training set that correlates to the same input (1,0) and which were experimentally measured.

A study about the standard test methods for polymer matrix composite materials [3] defines the in-plane tensile properties of polymer matrix composite materials reinforced with high-modulus fibers. A mechanical testing machine grips a thin flat strip of material with a constant rectangular cross-section and monotonically loads it in tension while recording load. This test method is intended to generate tensile property data for material specifications, R&D, quality assurance, and structural design and analysis. This study used an interlaboratory testing program in which nine different laboratories tested an average of five specimens from six different materials and lay-up configurations. This study produced precision statistics for tensile strength, modulus, and failure strain. The data was all normalized concerning an average thickness. The study states that the values of  $Sr/X$  and  $SR/X$  exemplify the repeatability and the reproducibility coefficients of variation, respectively. These averages allow for a relative comparison of the tension test parameters' repeatability (within laboratory precision) and reproducibility (between laboratory precision).

Traditional modeling methods, such as response surface methodology, do not have the same advantages as neuro-regression analysis. Based on only the data, neuro-regression analysis can be used to model the behavior of complex systems [4]. On the other hand, RSM is based on model structure assumptions and requires coefficient estimation [5-9]. To maximize or minimize objective functions, stochastic optimization methods are used. Stochastic optimization is crucial in the analysis, design, and performance of modern systems [10].

\*Corresponding author e-mail address: [nb.kartal71@gmail.com](mailto:nb.kartal71@gmail.com)

The main goal is to use Neuro-Regression steps to create optimization models for determining the mechanical properties of new composite material. In practical experiments, it is critical to estimate accurate values expressed in tensile strength and elongation. The stochastic optimization approach is used to predict and optimize the mechanical properties of a material. Multiple non-linear models are used to determine the maximum tensile strength and elongation achievable within the constraints.

## 2. Materials and Method

### 2.1. Modeling

In the modeling phase, a hybrid method is used to assess the accuracy of the predictions, which integrates the benefits of regression analysis and artificial neural networks. In this method, all of the data is divided into three sets, each containing 80%, 15%, and 5% of the total data, with the first portion used for training, the second for testing, and the third for validation. The objective of the training procedure is to minimize the error between the experimental and predicted values by adjusting the regression models and their coefficients, as shown in Table 1. The prediction results are then obtained by minimizing the effects of regression model discrepancies during the testing step. First, this procedure yields information about the prediction capacity of the candidate models. Second, the boundedness of the candidate models for prescribed values must be checked to determine whether or not the model is realistic. In this case, the maximum and minimum values of the models in the given interval for each design variable are calculated after obtaining the appropriate models in terms of  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$ ,  $R^2_{\text{validation}}$ . This procedure determines whether the chosen models meet the numerous criteria required for reality [11].

The logarithm cannot be used in the modeling of this study because some of the inputs take the value of 0. Also, it is not easy to understand some variables because they have a non-linear relationship. So, the hybrid models shown in Table 2 are tried to obtain better  $R^2$  values.

### 2.2. Optimization

A structure's optimization can be defined as achieving the best designs by reducing the specified single or multi-objective that corresponds to all constraints. There are two kinds of optimization techniques: traditional and nontraditional. Traditional optimization techniques, such as constrained variation and Lagrange multipliers, only apply to continuous and differentiable functions. Traditional optimization techniques cannot be used to solve engineering design problems due to their specificity. Stochastic optimization methods such as genetic algorithms (GA), particle swarm optimization (PS), and simulated annealing (SA) are advantageous in these cases. Because of the features of stochastic methods, correct solutions cannot be achieved, and using multiple methods with different phenomenological principles for the same optimization problem increases the solution's reliability [8].

This study's optimization scenarios include the following challenges: multiple non-linear objective functions, objective functions having many local extremum points, mixed-integer(discrete)- continuous nature of the design variables, non-linear constraints. To solve these optimization scenarios, Nelder-Mead Algorithm, Differential Evolution Algorithm, Simulated Annealing Algorithm, and Random Search Algorithm have been selected. For more information, please see the reference articles given in the subsections.

**Table 1.** Multiple regression model types [11]

Model Name	Nomenclature	Formula
Multiple linear	L	$a[1] + x_1 a[2] + x_2 a[3]$
Multiple linear rational	LR	$(a[1] + x_1 a[2] + x_2 a[3]) / (b[1] + x_1 b[2] + x_2 b[3])$
Second order multiple nonlinear	SON	$a[1] + x_1 a[2] + x_1^2 a[3] + x_2 a[4] + x_1 x_2 a[5] + x_2^2 a[6]$
Second order multiple non-linear rational	SONR	$(a[1] + x_1 a[2] + x_1^2 a[3] + x_2 a[4] + x_1 x_2 a[5] + x_2^2 a[6]) / (b[1] + x_1 b[2] + x_1^2 b[3] + x_2 b[4] + x_1 x_2 b[5] + x_2^2 b[6])$
Third order multiple non-linear	TON	$a[1] + x_1 a[2] + x_1^2 a[3] + x_1^3 a[4] + x_2 a[5] + x_1 x_2 a[6] + x_1^2 x_2 a[7] + x_2^2 a[8] + x_1 x_2^2 a[9] + x_2^3 a[10]$
First order trigonometric multiple non-linear	FOTN	$a[1] + a[2] \text{Cos}[x_1] + a[3] \text{Cos}[x_2] + a[4] \text{Cos}[x_3] + a[5] \text{Sin}[x_1] + a[6] \text{Sin}[x_2] + a[7] \text{Sin}[x_3]$

<b>First order trigonometric multiple non-linear rational</b>	FOTNR	$(a[1] + a[2] \cos[x_1] + a[3] \cos[x_2] + a[4] \sin[x_1] + a[5] \sin[x_2]) / (b[1] + b[2] \cos[x_1] + b[3] \cos[x_2] + b[4] \sin[x_1] + b[5] \sin[x_2])$
<b>Second order trigonometric multiple non-linear</b>	SOTN	$a[1] + a[2] \cos[x_1] + a[3] \cos[x_1]^2 + a[4] \cos[x_2] + a[5] \cos[x_1] \cos[x_2] + a[6] \cos[x_2]^2 + a[7] \cos[x_3] + a[8] \sin[x_1] + a[9] \cos[x_1] \sin[x_1] + a[10] \cos[x_2] \sin[x_1] + a[11] \sin[x_1]^2 + a[12] \sin[x_2] + a[13] \cos[x_1] \sin[x_2] + a[14] \cos[x_2] \sin[x_2] + a[15] \sin[x_1] \sin[x_2] + a[16] \sin[x_2]^2$
<b>Second order trigonometric multiple non-linear rational</b>	SOTNR	$(a[1] + a[2] \cos[x_1] + a[3] \cos[x_1]^2 + a[4] \cos[x_2] + a[5] \cos[x_1] \cos[x_2] + a[6] \cos[x_2]^2 + a[7] \sin[x_1] + a[8] \cos[x_1] \sin[x_1] + a[9] \cos[x_2] \sin[x_1] + a[10] \sin[x_1]^2 + a[11] \sin[x_2] + a[12] \cos[x_1] \sin[x_2] + a[13] \cos[x_2] \sin[x_2] + a[14] \sin[x_1] \sin[x_2] + a[15] \sin[x_2]^2) / (b[1] + b[2] \cos[x_1] + b[3] \cos[x_1]^2 + b[4] \cos[x_2] + b[5] \cos[x_1] \cos[x_2] + b[6] \cos[x_2]^2 + b[7] \sin[x_1] + b[8] \cos[x_1] \sin[x_1] + b[9] \cos[x_2] \sin[x_1] + b[10] \sin[x_1]^2 + b[11] \sin[x_2] + b[12] \cos[x_1] \sin[x_2] + b[13] \cos[x_2] \sin[x_2] + b[14] \sin[x_1] \sin[x_2] + b[15] \sin[x_2]^2)$
<b>First order logarithmic multiple nonlinear</b>	FOLN	$a[1] + a[2] \log[x_1] + a[3] \log[x_2]$
<b>First order logarithmic multiple non-linear rational</b>	FOLNR	$(a[1] + a[2] \log[x_1] + a[3] \log[x_2]) / (b[1] + b[2] \log[x_1] + b[3] \log[x_2])$
<b>Second order logarithmic multiple non-linear</b>	SOLN	$a[1] + a[2] \log[x_1] + a[3] \log[x_1]^2 + a[4] \log[x_2] + a[5] \log[x_1] \log[x_2] + a[6] \log[x_2]^2$
<b>Second order logarithmic multiple non-linear rational</b>	SOLNR	$3(a[1] + a[2] \log[x_1] + a[3] \log[x_1]^2 + a[4] \log[x_2] + a[5] \log[x_1] \log[x_2] + a[6] \log[x_2]^2) / (b[1] + b[2] \log[x_1] + b[3] \log[x_1]^2 + b[4] \log[x_2] + b[5] \log[x_1] \log[x_2] + b[6] \log[x_2]^2)$

**Table 2.** Hybrid models

Model Name	Nomenclature	Formula
Hybrid	H1	$a[1] + 2 x_1 a[2] + [x_1]^2 a[3] + 2 x_2 a[4] + 2 x_1 x_2 a[5] + [x_2]^2 a[6] + a[7] \cos[x_1] + a[8] \cos[x_1]^2 + a[9] \cos[x_2] + a[10] \cos[x_1] \cos[x_2] + a[11] \cos[x_2]^2$
Hybrid	H2	$a[1] + x_1 a[2] + x_1^5 a[3] + x_1^3 a[4] + 7x_2 a[5] + x_1 x_2 a[6] + x_1^4 x_2 a[7] + x_2^6 a[8] + 12 x_1 x_2^2 a[9] + x_2^5 a[10]$

2.2.1 Nelder-mead algorithm

The Nelder–Mead optimization algorithm is one of the most basic direct search methods. As a result, it is not necessary any derivative knowledge and begins with simplex to minimize the function. The iteration continues until the simplex becomes flat. This means that the function's resulting value is nearly identical at all vertices. The Nelder-Mead algorithm's iteration steps are ordering, centroid, and transformation [4].

2.2.2 Differential evolution algorithm

Differential evolution algorithm is one of the appropriate stochastic optimization methods. The differential evolution algorithm's productive parameters are population size, crossover, and scaling factor. Thus, it deals with a population of solutions rather than iterating over them. Although it does not satisfy the global optimum points for all optimization problems, the differential evolution algorithm is proposed in the literature to be robust and efficient [11].

2.2.3 Simulated annealing algorithm

The simulated annealing algorithm is another common search method that is based on the physical annealing of metal. During the melting process, the material shifts to a lower energy state and becomes tougher. Because of the algorithm's inherent structure, it is more effective at determining the global optimum. In addition, it can solve optimization problems that are continuous, mixed-integer, or discrete [12].

2.2.4 Random search algorithm

At this stage, the traditional random search algorithm employs a local optimization method from each starting point to approach a local extremum point. The proposed version of the algorithm includes some booster subroutines such as the conjugate gradient, principal axis, Levenberg Marquardt, Newton, QuasiNewton, and non-linear interior-point method in the localization of the values of all variables for the objective function. This step evaluates the fitness function with symbolic variables, and the procedure is repeated several times. [12].

### 2.3. Problem Definition

The appropriate model for determining the mechanical properties of new composite material is arranged by employing neuro-regression analysis.

- The design variables, where  $x_1$ : Layout of the layers,  $x_2$ : Angle of the tensile ( $^\circ$ ) depicted in Table 3, is the data referenced from the main study.
- 11 candidate functional constructs have been suggested to model the experimental data of new composite material have been tested for the appropriate ones in terms of  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$ ,  $R^2_{\text{validation}}$  values, and then boundedness of the functions is also checked.
- Using the appropriate models, two distinct optimization strategies were implemented, and four different direct search approaches were used to solve these problems.

### 2.4. Optimization Scenarios

#### Scenario 1

In this optimization problem, the objective functions define tensile strength and elongation of the material, the design variables are all supposed to be real numbers, and the search space is infinite. For this case,  $1 < \text{layout of the layers} < 2$  and  $0 < \text{angle of the tensile} < 90$ . The main goal is to maximize the tensile strength and the elongation of the material. This approach can also be used to calculate the limits of the objective function.

**Table 3.** Tensile strength (Mpa),  $\sigma_M$  and elongation at break (%),  $\epsilon_M$  of the new composite material prepared by different layout of the layers and angle of the tensile ( $^\circ$ ) [2]

No.	Layout of the layers	Angle of the tensile ( $^\circ$ )	Tensile strength (Mpa), $\sigma_M$	Elongation at break (%), $\epsilon_M$
1	1	0	19.90	3.01
2	1	0	19.91	3.20
3	1	0	22.00	2.65
4	1	0	23.20	3.40
5	1	0	23.50	2.74
6	1	0	24.20	3.23
7	1	0	25.40	3.35
8	1	0	26.30	3.44
9	1	45	13.50	2.58
10	1	45	14.70	2.41
11	1	45	14.75	2.63
12	1	45	15.85	3.82
13	1	45	16.60	3.65
14	1	45	17.10	2.62
15	1	45	17.30	3.44
16	1	45	17.30	3.44
17	1	45	17.30	3.44
18	1	45	17.30	3.44
19	1	45	17.30	3.44
20	1	45	17.30	3.44
21	1	45	17.30	3.44
22	1	45	17.30	3.44
23	1	45	17.30	3.44
24	2	0	12.00	2.35
25	2	0	13.30	1.68
26	2	0	13.30	1.68
27	2	0	13.30	1.68
28	2	0	13.30	1.68
29	2	0	13.30	1.68
30	2	0	13.30	1.68
31	2	45	18.50	2.30
32	2	45	20.20	3.06

33	2	45	20.20	3.06
34	2	45	20.20	3.06
35	2	45	20.20	3.06
36	2	45	20.20	3.06
37	2	45	20.20	3.06
38	2	45	20.20	3.06
39	2	45	20.20	3.06
40	2	45	20.20	3.06
41	2	45	20.20	3.06
42	2	45	20.20	3.06
43	2	45	20.20	3.06
44	2	45	20.20	3.06
45	2	90	17.40	2.88
46	2	90	19.50	3.62

### Scenario 2

In addition to knowledge learned from scenario 1, a more applicable problem case must be introduced. For this reason, a new optimization problem is defined, which considers the maximization of the tensile strength and the elongation of the material. All design variables are presumed to be real numbers at the intervals:  $1 < \text{layout of the layers} < 2$  and  $0 < \text{angle of the tensile} < 90$ . Moreover, to examine constructional and experimental constraints {strength, elongation}  $\in$  integers are appropriate.

### 3. Result and Discussion

In this study, 11 several regression models (Table 1) with two parameters have been tested for two outputs, and the results are listed in Table 4 and 5 in order to understand the model's capability to explain the process by estimating  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$ ,  $R^2_{\text{validation}}$  values for various regression models and the model's functional limitation by estimating the maximum and minimum values created by the respective model.

In Table 4, the suitability of the candidate models in terms of training, testing and validation coefficients, and boundedness, the following conclusions were made: Training coefficients of all models are quite high ( $>0.97$ ) while the test and validation coefficients are very low ( $<0$  &  $<0.77$  respectively) for L and FOTN models. In addition, the testing and validation values of the LR model are not much but low ( $<85$  both of them), so the LR model can not provide the appropriate model too. Therefore, 11 usable model in terms of fit capabilities at the first stage falls to 8. Furthermore, as previously stated, It is anticipated to satisfy the boundedness criterion for use in model optimization. From this point of view, models SON, TON, SOTN, and H2 are also not suitable. As a result, models SONR, FOTNR, SOTNR, and H1 satisfy all the desired criteria and are also regarded as more realistic. The limitations of the models: for SONR is  $-9.52 \times 10^{13} - 2.19 \times 10^{15}$ , for FOTNR is  $-3.26 \times 10^{15} - 1.64 \times 10^{14}$ , for SOTNR is  $-11.76 - 26.24$  and for H1 is  $-19.09 - 29.11$ . Therefore, it was concluded that the H1 model function best describes the "tensile strength" parameter.

A similar explanation can be made for Table 5: While 11 models may be suitable for training and testing coefficients for the "elongation at break," L, FOTN, FOTNR, SOTN, SOTNR, and H1 models are not available in terms of the third criterion, boundedness. Except for the three criteria given above, these five models, when it is discussed which one is more realistic, the H2 model in terms of simplicity, value range, and fit capabilities is more suitable than other models. Furthermore, alternative formulations such as LR, SON, SONR, and TON are available for use because they perform similarly.

**Table 4.** Results of the Neuro-regression models for the tensile strength

Models	$R^2_{\text{training}}$	$R^2_{\text{testing}}$	$R^2_{\text{validation}}$	Max (MPa)	Min (MPa)
L1	0.97	-0.66	0.038	19.34	17.37
LR1	0.99	0.72	0.77	18.20	13.31
SON1	0.99	0.95	0.90	22.88	1.73
SONR1	0.99	0.95	0.90	$2.19 \times 10^{15}$	$-9.52 \times 10^{13}$
TON1	0.99	0.95	0.90	22.88	-0.94
FOTN1	0.97	-0.73	0.094	23.15	18.37
FOTNR1	0.99	0.95	0.99	$1.64 \times 10^{14}$	$-3.26 \times 10^{15}$
SOTN1	0.99	0.95	0.90	22.88	-4.07
SOTNR1	0.99	0.95	0.90	26.24	-112.67
H1	0.99	0.95	0.90	26.11	-19.04
H2	0.99	0.95	0.90	22.00	1.39

**Table 5.** Results of the Neuro-regression models for the elongation at break

Models	R <sup>2</sup> <sub>training</sub>	R <sup>2</sup> <sub>testing</sub>	R <sup>2</sup> <sub>validation</sub>	Max (%)	Min (%)
L2	0.98	0.57	0.57	3.90	2.44
LR2	0.98	0.87	0.85	3.33	1.85
SON2	0.98	0.87	0.85	3.27	1.85
SONR2	0.98	0.87	0.85	3.54	1.85
TON2	0.99	0.87	0.85	3.36	1.85
FOTN2	0.97	-0.73	0.094	23.15	18.37
FOTNR2	0.99	0.87	0.85	$3.92 \times 10^{13}$	$-3.87 \times 10^{14}$
SOTN2	0.99	0.95	0.90	22.88	-4.07
SOTNR2	0.99	0.87	0.85	$3.91 \times 10^{14}$	$-4.05 \times 10^{14}$
H1	0.99	0.87	0.85	4.52	-0.59
H2	0.99	0.87	0.85	3.56	2.97

In Table 6, the model of "tensile strength " H1 is taken as the objective function, and the results are listed for several optimization scenarios. This table uses DE, NM, SA, and RS algorithms for each scenario, and the results are compared. For example, all algorithms determined the maximum "Tensile Strength" value for the first scenario was 26.55, while the corresponding x2 variables of "Suggested Design" values differed. This gives us four different alternative input parameter triplets to obtain the highest tensile strength. In the second scenario, the problem description is similar, but the input parameters (Layout of the layers, angle of the tensile) are forced to be integers. Using the DE and SA algorithms, the maximum tensile strength value was reduced to 1%, and the input values were  $x_1 = 1$  and  $x_2 = 44$ .

Similarly, in Table 7, the model of "elongation at break " H2 is taken as the objective function, and the results are listed for two several optimization scenarios. This table involves the calculation based on DE, NM, SA, and RS algorithms for each scenario, and the results are compared. The maximum "Elongation at Break" value determined by all algorithms for the first scenario is 3.58, and the corresponding "Suggested Design" is the same. This gives us only one input parameter triplet to obtain the highest elongation at break. In the second scenario, the problem description is similar, but the input parameters (Layout of the layer, angle of the tensile) are forced to be integers. In this case, using the DE, NM, and SA algorithms, the maximum tensile strength was reduced to 8%, and the input values were  $x_1 = 2$  and  $x_2 = 75$ .

**Table 6.** Optimization problem results of the selected model for tensile strength

Objective Function	Scenario No	Constraints	Optimizaiton Algorithm	Max Tensile Strength	Suggested Design
$1.898150158244077 + 1.4678873258539737 x_1 + 0.8469133355818345[x_1]^2 + 0.05490833491406189 x_2 + 0.10167237594192526 x_1 x_2 - 0.0016739290005225325 [x_2]^2 - 4.67959362010483 \text{Cos}[x_1] + 10.99388196366126 \text{Cos}[x_2] + 18.894432221360166 \text{Cos}[x_1] \text{Cos}[x_2]$	1	$1 < x_1 < 2,$ $0 < x_2 < 90$	DE	26.54	$x_1=1.0, x_2=43.98$
			NM	26.12	$x_1=1.0, x_2=62.83$
			SA	26.55	$x_1=1.0, x_2=43.98$
			RS	26.53	$x_1=1.0, x_2=50.26$
	2	$1 < x_1 < 2,$ $0 < x_2 < 90,$ $\{x_1, x_2, x_3\}$ $\in \text{Integers}$	DE	26.53	$x_1=1, x_2=44$
			NM	23.13	$x_1=2, x_2=69$
			SA	26.53	$x_1=1, x_2=44$
			RS	22.27	$x_1=2, x_2=57$

**Table 7.** Optimization problem results of the selected model for elongation at break

Objective Function	Scenario No	Constraints	Optimization Algorithm	Max Elongation at Break	Suggested Design
18.56976245187751 + 5.332625748774333 $x_1$ - 0.6897015434525597 $x_1^3$ - 0.3255438000564282 $x_1^5$ - 0.17116361963166152 $x_2$ + 0.011875685280791307 $x_1 x_2$ + 0.018449736564968358 $x_1^4 x_2$ + 0.00008170504540320082 $x_1 x_2^2$ - 9.502786632380143 $\times 10^{-10} x_2^5$ - 7.268614002730438 $\times 10^{-12} x_2^6$	1	$1 < x_1 < 2,$ $0 < x_2 < 90$	DE NM SA RS	3.58 3.58 3.58 3.58	$x_1=1.74, x_2=69.56$ $x_1=1.74, x_2=69.56$ $x_1=1.74, x_2=69.56$ $x_1=1.74, x_2=69.56$
	2	$1 < x_1 < 2,$ $0 < x_2 < 90,$ { $x_1, x_2, x_3$ } $\in$ Integers	DE NM SA RS	3.50 3.50 3.50 3.49	$x_1=2, x_2=75$ $x_1=2, x_2=75$ $x_1=2, x_2=76$ $x_1=2, x_2=74$

#### 4. Conclusion

Designing optimal products from a new composite material necessitates a thorough understanding of the material's mechanical properties. Unfortunately, these properties are difficult to deduce from their constituents. Furthermore, the properties of the composite cannot be calculated using mathematical formulas from the properties of the constituents. As a result, they can only be determined experimentally using specialized machines and testing methods.

This paper aims to show the possibility of using an optimization method to determine the mechanical properties of tensile strength and elongation at break for new composite material. Results show that the optimization method is convenient to choose an appropriate model for that kind of study. So, it can provide ease of solution to these studies. Using the process variables, the optimization model is proposed to estimate the tensile strength and elongation at break. A novel model based on neuro-regression analysis methods to determine optimum mechanical properties has been introduced to eliminate this deficiency. First, a thorough investigation of non-linear multiple regression analysis was carried out, including rational forms for linear, quadratic, trigonometric, logarithmic. Second, the limitations of candidate models were validated in order to produce realistic values. Finally, several direct search methods were used, including stochastic approaches, during the optimization phase. Another indication that the choice of the objective function and constraints design variables becomes important in optimization problems. It is satisfying that the best result has been given from hybrid models rather than the standard models.

The following conclusions can be reached after the modeling of the tensile strength,  $\sigma_M$ , and the elongation at break,  $\epsilon_M$ :

- 1) The 11 models are tried, and one appropriate model is found for two outputs. The  $R^2$  testing and  $R^2$  validation values of the models are at the desired level.
- 2) In addition, the maximum tensile strength and elongation at break values of the material were found.
- 3) The input parameters were found, which is needed to achieve maximum tensile strength and elongation at break values of the material.
- 4) The type of scenario and the choice of constraints for design variables are critical in the optimization problem.

#### Declaration of Interest

The author declare that there is no conflict of interest.

#### References

- [1] A. K. Kaw, Mechanics of composite materials. CRC press, 2005.
- [2] E. Ciupan, M. Ciupan and D.C. Jucan, "Determining the mechanical properties of a new composite material using artificial neural networks". International Journal of Engineering Trends and Technology (IJETT), vol.66, no.2, Dec., pp.103, 2018.
- [3] Standard Test Method for Tensile Properties of Polymer Matrix Composite Materials, ASTM std. ASTM D3039/D3039M, 2017.
- [4] I. Polatoğlu, L. Aydın, B.C. Nevruz and S. Ozer, "A Novel Approach for the Optimal Design of a Biosensor". Analytical Letters, pp.1-18, 2020.
- [5] MM Dawoud and H.M. Saleh, Introductory Chapter: Background on Composite Materials. In Characterizations of Some Composite Materials. IntechOpen, 2018.
- [6] M. Gandomi, M. Soltanpour, M. Zolfaghari and A. Gandomi, "Prediction of peak ground acceleration of Iran's tectonic regions



- using a hybrid soft computing techniques”. Journal of Geoscience Frontiers, vol. 7, no. 1, pp.75-82, 2016.
- [7] DK. Rajak, D.D. Pagar, P.L. Menezes and E. Linul, (2019). "Fiber-reinforced polymer composites: Manufacturing, properties, and applications". Reinforced Polymer Composites, vol. 11, no. 10, pp. 1667, 2019
- [8] A. Garg, V. Vijayaraghavan, J.S.L. Lam, P.M. Singru and L.A. Gao, "A molecular simulation based computational intelligence study of a nano-machining process with implications on its environmental performance". Swarm and Evolutionary Computation, vol. 21, pp.54-63, 2015.
- [9] A. Garg, V. Vijayaraghavan, J. Zhang, S. Li and X. Liang, "Design of robust battery capacity model for electric vehicle by incorporation of uncertainties". International Journal of Energy Research, vol. 41, no. 10, pp. 1436–1451, 2017.
- [10] Saman Maroufpoor, Omid Bozorg-Haddad, in Handbook of Probabilistic Models, 2020
- [11] I. Polatoglu, and L. Aydin, "A new design strategy with stochastic optimization on the preparation of magnetite cross-linked tyrosinase aggregates (MCLTA)," Process Biochemistry, vol. 99, pp. 131–138, 2020.
- [12] A. B. Ceylan, L. Aydin, and M. Nil, H. Mamur, İ. Polatoğlu and H. Sözen. "A new hybrid approach in selection of optimum establishment location of the biogas energy production plant," Biomass Conv. Bioref, 2021. <https://doi.org/10.1007/s13399-021-01532-8>

## APPENDIX

Nomenclature	Models
LR1	$19.71182066869301 - 0.37026443768996825x_1 - 0.017647348868625304x_2 - 51.31976284522877 + 51.31976284523468x_1 + 774503.1775437717x_2 / -3.854604998705042 + 3.8546049987053x_1 + 42541.08885662762x_2$
SON1	$24.526185784658686 + 2.3350070372976797x_1 - 3.9740499648135224x_1^2 - 0.34202116402116317x_2 + 0.2922984607984611x_1x_2 - 0.0020594837261504097x_2^2$
SONR1	$2218.5038106203615 + 8332.47861996536x_1 - 5600.060243913964x_1^2 + 31088.892439939827x_2 + 1847.6990367776073x_1x_2 - 1574.950427859876x_2^2 / 346.81933827792716 + 44.61861835370988x_1 - 175.11902955653323x_1^2 + 1196.1381419252248x_2 + 501.84473556335837x_1x_2 - 88.848501524142x_2^2$
TON1	$20.94109188907611 + 4.5824368664639605x_1 - 1.0712803912934319x_1^2 - 1.5651055071037883x_1^3 - 0.18033348484038514x_2 + 0.04459740211289496x_1x_2 + 0.07264047537312718x_1^2x_2 - 0.0019446050613860057x_2^2 + 0.0006617696125818712x_1x_2^2 - 0.000010654947332801006x_2^3$
FOTN1	$9.902429779197746 + 1.3032369041527805\text{Cos}[x_1] - 0.6024526244055706\text{Cos}[x_2] + 11.372423196615832\text{Sin}[x_1] - 1.701292858920586\text{Sin}[x_2]$
FOTNR1	$-72.0344943619616 + 3086.5528215675718\text{Cos}[x_1] + 68.68616422167281\text{Cos}[x_2] + 330.4484534546959\text{Sin}[x_1] - 834.7528817322325\text{Sin}[x_2] / 74.71611907347054 + 158.8667416923987\text{Cos}[x_1] + 13.088655381978818\text{Cos}[x_2] - 105.49720900069308\text{Sin}[x_1] - 6.894967901990746\text{Sin}[x_2]$
SOTN1	$3.0460545502597487 + 0.6800626431272705\text{Cos}[x_1] + 5.6052888673774355\text{Cos}[x_2] + 9.949815188229488\text{Cos}[x_1]\text{Cos}[x_2] + 6.999132839176023\text{Sin}[x_1] + 0.12042831374840418\text{Cos}[x_1]\text{Sin}[x_1] + 3.028187782163123\text{Cos}[x_2]\text{Sin}[x_1] + 4.570566768978463\text{Sin}[x_2] - 10.530448906883716\text{Cos}[x_1]\text{Sin}[x_2] - 2.375890082747342\text{Cos}[x_2]\text{Sin}[x_2] + 2.8328197496013865\text{Sin}[x_1]\text{Sin}[x_2]$
SOTNR1	$(1957.2500595090817 + 5961.234191177309\text{Cos}[x_1] + 939.3962800588763\text{Cos}[x_2] - 8450.710855836645\text{Cos}[x_1]\text{Cos}[x_2] + 987.6027093076096\text{Sin}[x_1] + 3420.4660512907126\text{Cos}[x_1]\text{Sin}[x_1] - 699.6677723130807\text{Cos}[x_2]\text{Sin}[x_1] + 1375.5244699483446\text{Sin}[x_2] - 1735.7546609202786\text{Cos}[x_1]\text{Sin}[x_2] - 7320.150314484926\text{Cos}[x_2]\text{Sin}[x_2] + 486.60921758147333\text{Sin}[x_1]\text{Sin}[x_2]) / (67.38342604363065 + 537.0007633750448\text{Cos}[x_1] + 265.67098828479004\text{Cos}[x_2] - 374.9260855810948\text{Cos}[x_1]\text{Cos}[x_2] + 236.3986433952951\text{Sin}[x_1] - 286.1413334671107\text{Cos}[x_1]\text{Sin}[x_1] - 407.7703408734187\text{Cos}[x_2]\text{Sin}[x_1] + 4.457381391121794\text{Sin}[x_2] + 99.66232882250745\text{Cos}[x_1]\text{Sin}[x_2] - 229.8443151710416\text{Cos}[x_2]\text{Sin}[x_2] - 78.61973683325331\text{Sin}[x_1]\text{Sin}[x_2])$
H1	$1.898150158244077 + 1.4678873258539737x_1 + 0.8469133355818345x_1^2 + 0.05490833491406189x_2 + 0.10167237594192526x_1x_2 - 0.0016739290005225325x_2^2 - 4.67959362010483\text{Cos}[x_1] + 10.99388196366126\text{Cos}[x_2] + 18.894432221360166\text{Cos}[x_1]\text{Cos}[x_2]$
H2	$18.56976245187751 + 5.332625748774333x_1 - 0.6897015434525597x_1^3 - 0.3255438000564282x_1^5 - 0.17116361963166152x_2 + 0.011875685280791307x_1x_2 + 0.018449736564968358x_1^4x_2 + 0.00008170504540320082x_1x_2^2 - 9.502786632380143 \times 10^{-10}x_2^5 - 7.268614002730438 \times 10^{-12}x_2^6$
L2	$19.71182066869301 - 0.37026443768996825x_1 - 0.017647348868625304x_2$
LR2	$-51.31976284522877 + 51.31976284523468x_1 + 774503.1775437717x_2 / -3.854604998705042 + 3.8546049987053x_1 + 42541.08885662762x_2$
SON2	$24.526185784658686 + 2.3350070372976797x_1 - 3.9740499648135224x_1^2 - 0.34202116402116317x_2 + 0.2922984607984611x_1x_2 - 0.0020594837261504097x_2^2$
SONR2	$2218.5038106203615 + 8332.47861996536x_1 - 5600.060243913964x_1^2 + 31088.892439939827x_2 + 1847.6990367776073x_1x_2 - 1574.950427859876x_2^2 / 346.81933827792716 + 44.61861835370988x_1 - 175.11902955653323x_1^2 + 1196.1381419252248x_2 + 501.84473556335837x_1x_2 - 88.848501524142x_2^2$

---

<b>TON2</b>	$20.94109188907611 + 4.5824368664639605x1 - 1.0712803912934319x1^2 - 1.5651055071037883x1^3 - 0.18033348484038514x2 + 0.04459740211289496x1x2 + 0.07264047537312718x1^2x2 - 0.0019446050613860057x2^2 + 0.0006617696125818712x1x2^2 - 0.000010654947332801006x2^3$
<b>FOTN2</b>	$9.902429779197746 + 1.3032369041527805\text{Cos}[x1] - 0.6024526244055706\text{Cos}[x2] + 11.372423196615832\text{Sin}[x1] - 1.701292858920586\text{Sin}[x2]$
<b>FOTNR2</b>	$-72.0344943619616 + 3086.5528215675718\text{Cos}[x1] + 68.68616422167281\text{Cos}[x2] + 330.4484534546959\text{Sin}[x1] - 834.7528817322325\text{Sin}[x2]/74.71611907347054 + 158.8667416923987\text{Cos}[x1] + 13.088655381978818\text{Cos}[x2] - 105.49720900069308\text{Sin}[x1] - 6.894967901990746\text{Sin}[x2]$
<b>SOTN2</b>	$3.0460545502597487 + 0.6800626431272705\text{Cos}[x1] + 5.6052888673774355\text{Cos}[x2] + 9.949815188229488\text{Cos}[x1]\text{Cos}[x2] + 6.999132839176023\text{Sin}[x1] + 0.12042831374840418\text{Cos}[x1]\text{Sin}[x1] + 3.028187782163123\text{Cos}[x2]\text{Sin}[x1] + 4.570566768978463\text{Sin}[x2] - 10.530448906883716\text{Cos}[x1]\text{Sin}[x2] - 2.375890082747342\text{Cos}[x2]\text{Sin}[x2] + 2.8328197496013865\text{Sin}[x1]\text{Sin}[x2]$
<b>SOTNR2</b>	$(1957.2500595090817 + 5961.234191177309\text{Cos}[x1] + 939.3962800588763\text{Cos}[x2] - 8450.710855836645\text{Cos}[x1]\text{Cos}[x2] + 987.6027093076096\text{Sin}[x1] + 3420.4660512907126\text{Cos}[x1]\text{Sin}[x1] - 699.6677723130807\text{Cos}[x2]\text{Sin}[x1] + 1375.5244699483446\text{Sin}[x2] - 1735.7546609202786\text{Cos}[x1]\text{Sin}[x2] - 7320.150314484926\text{Cos}[x2]\text{Sin}[x2] + 486.60921758147333\text{Sin}[x1]\text{Sin}[x2])) / (67.38342604363065 + 537.0007633750448\text{Cos}[x1] + 265.67098828479004\text{Cos}[x2] - 374.9260855810948\text{Cos}[x1]\text{Cos}[x2] + 236.3986433952951\text{Sin}[x1] - 286.1413334671107\text{Cos}[x1]\text{Sin}[x1] - 407.7703408734187\text{Cos}[x2]\text{Sin}[x1] + 4.457381391121794\text{Sin}[x2] + 99.66232882250745\text{Cos}[x1]\text{Sin}[x2] - 229.8443151710416\text{Cos}[x2]\text{Sin}[x2] - 78.61973683325331\text{Sin}[x1]\text{Sin}[x2]))$

---

# Modeling and optimum design for wire electrical discharge machining of $\gamma$ titanium aluminide alloy

Ömer Faruk BÜYÜKYAVUZ<sup>1</sup>

<sup>1</sup> Izmir Katip Celebi University, Department of Electrical and Electronics Engineering, Cigli, Izmir, Turkey

## Abstract

Wire electrical discharge machining (WEDM) of  $\gamma$  titanium aluminide is the subject of the current research. Due to the large number of process variables and sophisticated stochastic process mechanisms, selecting the best machining parameter combinations for increased cutting efficiency and accuracy is a difficult task in WEDM. In general, there is no perfect combination that can produce the fastest cutting speed and the finest surface finish quality at the same time. For this purpose, the data were selected from a literature study. This study describes an attempt to devise a suitable machining technique for achieving the highest possible process criteria yield. To model the machining process, a stochastic optimization method, differential evolution, has been performed. Cutting speed, surface roughness, and wire offset are the three most important criteria that have been used as indicators of process performance. The response characteristics can be predicted as a function of six different control parameters, namely pulse on time, pulse off time, peak current, wire tension, dielectric flow rate, and servo reference voltage. The limitations of the candidate models are checked after the  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$  and  $R^2_{\text{validation}}$  values are calculated to reveal whether the model is realistic. Optimization results are 3.02 mm/min, 2.36  $\mu\text{m}$ , and 0.13 mm for the maximum cutting speed, the minimum surface roughness, and minimum wire offset, respectively. It is shown that the machining model is suitable and that the optimization technique meets practical requirements.

**Keywords:**  $\gamma$  titanium aluminide; modeling; optimization; wire EDM.

## 1. Introduction

In recent years, the use of wire electrical discharge machining (WEDM) has increased dramatically. Wire EDM has a wide range of applications, including the production of various press tools, molds and even electrodes for use in other EDM processes. Wire EDM is currently commonly applied in the aerospace, automotive, and medical industries, as well as in almost every other conductive material machining application. It is considered especially suitable for machining complex contours, for high accuracy and for materials that are not amenable to conventional removal methods [1].

Several attempts to model the process have also been made. Scott et al. [2] have created a factor model to assess the process performance in accordance with the varied control conditions. By introducing the concept of a not-dominated point, the procedure was further optimized. By using regression analysis and subsequently resolving the optimization problem with a viable directional approach, the Liao et al. [3] built a mathematical model. EDM was modeled by Karthikeyan et al. [4] with a full factorial design for processing carbide silicone particle composites and the models significance was verified using the analysis of variance technique. Huang et al. [5] attempted to optimize the WEDM finish-cutting operation. The machined workpiece surface's gap width, surface roughness, and white layer depth are all measured and evaluated. The pulse-on duration and the distance between the wire perimeter and the workpiece surface are two major parameters impacting the machining performance, according to the Taguchi quality design approach and numerical analysis.

We took a new approach to the modeling design-optimization process to optimize the cutting speed, surface roughness, and wire offset input parameters in wire electrical discharge processing. This approach was organized based on a literature study [6] using the Box-Behnken design and regression analysis to obtain the percentage of outputs. First, a detailed study was conducted on multiple nonlinear neuro-regression analysis, including linear, non-linear and their rational forms for the outputs. Second, the boundaries of candidate models were checked to produce realistic values. Finally, a stochastic search methods were implemented methodically.

## 2. Materials and methods

### 2.1. Modeling

To assess the accuracy of the predictions during the modeling phase, a hybrid method including regression analysis is used. All data is divided into three groups in this approach, with the first portion being used for training,

the second for testing, and the third for validation. By modifying the regression models and coefficients during the training phase, the goal is to reduce the errors between the experimental and predicted values.

Following that, the testing stage is used to achieve the prediction results by reducing the effects of regression model inconsistencies. This procedure yields information about the candidate models' ability to anticipate. Third, checking the boundedness of candidate models for prescribed values is critical in determining whether or not the model is realistic. In this case, the maximum and minimum values of the models in the given interval for each design variable are calculated after acquiring the right models in terms of  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$  and  $R^2_{\text{validation}}$ . This procedure evaluates whether the selected models satisfy the many criteria which are necessary for reality [7-9].

## 2.2. Optimization

Substantially, the optimization of a structure may be described as obtaining the best design by minimizing the specified single or multi-objective that corresponds to all of the constraints. There are two types of optimization techniques: traditional and nontraditional. Traditional optimization techniques work for only continuous and differentiable functions, such as constrained variation and Lagrange multipliers. In engineering design problems, traditional optimization techniques cannot be used because of their specificity. In these cases, stochastic optimization methods such as genetic algorithms (GA), particle swarm (PS), and simulated annealing (SA) are favorable. Because of the nature of stochastic methods, the exact solution cannot be obtained and using more than one method with a different phenomenological basis for the same optimization problem increases the reliability of the solution [7].

## 2.3 Problem definition

The optimal design of cutting speed, surface roughness, and wire offset in a wire electrical discharge machining was organized as follows, using the references [8-11]. The data shown in Table 1 were selected from the reference study [6]. They modeled the electrical discharge process input parameters with Box-Behnken design and regression analysis.

- Ten to twenty candidate functional structures were proposed to model the data of the wire electrical discharge process system and were evaluated in terms of the limitation of functions,  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$  and  $R^2_{\text{validation}}$  values.
- One optimization scenario was introduced by using the obtained suitable models and these problems were solved by four different direct search methods.

## 2.4. Optimization Scenario

### Scenario

In this optimization problem, the objective function defines the cutting speed, surface roughness and wire offset, all design variables are assumed to be real numbers, and the search field is continuous. For this case,  $0.8 \mu\text{s} < \text{Pulse on time } (T_{\text{on}}) < 1.6 \mu\text{s}$ ,  $14 \mu\text{s} < \text{Pulse off time } (T_{\text{off}}) < 30 \mu\text{s}$ ,  $120 \text{ A} < \text{Peak current } (I_p) < 220 \text{ A}$ ,  $900 \text{ V} < \text{Servo reference voltage } (SV) < 1380 \text{ V}$ ,  $2 \text{ gm} < \text{Wire tension } (WT) < 10 \text{ gm}$ ,  $7 \text{ kg/cm}^2 < \text{Dielectric flow rate (discharge pressure) } (FR) < 10 \text{ kg/cm}^2$ . The main purpose is to maximize cutting speed, minimize surface roughness and wire offset. Mathematically, the boundaries of the objective function can also be obtained with this approach.

## 3. Results and discussion

Various regression models for cutting speed, surface roughness, and wire offset design in electrical discharge machining have been tested using  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$  and  $R^2_{\text{validation}}$  in the literature. In the reference study [6], Box-Behnken design and regression analysis were used to model outcomes input parameters.

In the present study, more than 20 different regression models with six parameters have been tested, and the most proper ones are listed in Table 2. And additionally with respect to these models predicted outputs and prediction error has been shown in Table 1.

The maximum cutting speed (3.02 mm/min) was obtained for the following optimal conditions;

Pulse on time ( $T_{\text{on}}$ ):1.6  $\mu\text{s}$ , Pulse off time ( $T_{\text{off}}$ ):14  $\mu\text{s}$ , Peak current ( $I_p$ ):220 A, Servo reference voltage (SV):900 gm, Wire tension (WT):2 V, Dielectric flow rate (discharge pressure) (FR):7 kg/cm<sup>2</sup>.

**Table 1.** Input parameters, experimental results, adjusted value and error for training data [6]

Experiment Number	Input Parameters					Responses			Predicted Values			Error			
	T <sub>on</sub> (µs)	T <sub>off</sub> (µs)	I <sub>p</sub> (A)	WT (gm)	SV (V)	FR (kg/cm <sup>2</sup> )	Machining Speed (mm/min)	Average R <sub>a</sub> (µm)	Wire Offset (mm)	Machining Speed (mm/min)	Average R <sub>a</sub> (µm)	Wire Offset (mm)	Machining Speed (mm/min)	Average R <sub>a</sub> (µm)	Wire Offset (mm)
1	1.1	14	120	900	2	7	2.67	2.77	0.142	2.42553	2.6586	0.142529	9.2%	4.0%	0.4%
2	1.1	20	170	1140	6	8.5	2.15	2.78	0.148	1.7869	2.6499	0.146767	16.9%	4.7%	0.8%
3	1.1	30	220	1380	10	10	1.38	2.78	0.15	1.37839	2.6598	0.150725	0.1%	4.3%	0.5%
4	1.6	14	120	1140	6	10	2.24	2.85	0.147	2.45339	2.8647	0.148745	9.5%	0.5%	1.2%
5	1.6	20	170	1380	10	7	2.12	2.81	0.147	2.24005	2.9867	0.149615	5.7%	6.3%	1.8%
6	1.6	30	220	900	2	8.5	2.08	2.97	0.151	2.32903	2.9721	0.150505	12.0%	0.1%	0.3%
7	0.8	14	170	900	10	10	1.84	2.33	0.151	1.73582	2.3688	0.148929	5.7%	1.7%	1.4%
8	0.8	20	220	1140	2	7	1.69	2.48	0.146	1.65183	2.5191	0.144681	2.3%	1.6%	0.9%
9	0.8	30	120	1380	6	8.5	0.98	2.3	0.141	0.939057	2.4498	0.141104	4.2%	6.5%	0.1%
10	1.1	14	220	1380	6	7	2.1	2.82	0.151	2.17147	2.6315	0.147892	3.4%	6.7%	2.1%
11	1.1	20	120	900	10	8.5	1.56	2.64	0.148	1.81435	2.6592	0.146644	16.3%	0.7%	0.9%
12	1.1	30	170	1140	2	10	1.78	2.78	0.146	1.60498	2.6776	0.145485	9.8%	3.7%	0.4%
13	1.6	14	170	1380	2	8.5	2.58	2.87	0.148	2.65672	2.9012	0.147745	3.0%	1.1%	0.2%
14	1.6	20	220	900	6	10	2.56	2.92	0.156	2.3741	2.9354	0.154032	7.3%	0.5%	1.3%
15	1.6	30	120	1140	10	7	2.25	3.03	0.15	1.99165	2.9868	0.147087	11.5%	1.4%	1.9%
16	0.8	14	220	1140	10	8.5	1.69	2.39	0.148	1.70961	2.3783	0.149635	1.2%	0.5%	1.1%
17	0.8	20	120	1380	2	10	1.14	2.3	0.14	1.27845	2.4672	0.14122	12.1%	7.3%	0.9%
18	0.8	30	170	900	6	7	1.07	2.42	0.14	1.33865	2.4921	0.14386	25.1%	3.0%	2.8%

**Table 1 (cont.).** Input parameters, experimental results, adjusted value and error for testing and validation data [6]

Experiment Number	Input Parameters					Responses			Predicted Values			Error			
	T <sub>on</sub> (µs)	T <sub>off</sub> (µs)	I <sub>p</sub> (A)	WT (gm)	SV (V)	FR (kg/cm <sup>2</sup> )	Machining Speed (mm/min)	Average R <sub>a</sub> (µm)	Wire Offset (mm)	Machining Speed (mm/min)	Average R <sub>a</sub> (µm)	Wire Offset (mm)	Machining Speed (mm/min)	Average R <sub>a</sub> (µm)	Wire Offset (mm)
19	1.6	30	170	900	2	10	2.24	3.16	0.151	2.21872	2.9626	0.149116	1.0%	6.2%	1.2%
20	0.8	20	170	1380	6	8.5	1.14	2.55	0.139	1.2765	2.4496	0.144315	12.0%	3.9%	3.8%
21	0.8	14	220	1140	2	8.5	2.1	2.31	0.145	1.88906	2.406	0.146223	10.0%	4.2%	0.8%
22	1.6	14	170	900	2	7	2.9	2.84	0.152	2.97953	2.9436	0.147989	2.7%	3.6%	2.6%
23	1.6	14	220	900	6	8.5	2.87	2.72	0.15	2.777	2.8741	0.149263	3.2%	5.7%	0.5%
24	1.1	30	220	900	6	8.5	1.65	2.62	0.15	1.70465	2.6599	0.153329	3.3%	1.5%	2.2%



**Table 2. Results of the Neuro-regression models**

Models		R <sup>2</sup> Training			R <sup>2</sup> Testing			R <sup>2</sup> Validation			
Output 1	Output 2	Output 3	Output 1	Output 2	Output 3	Output 1	Output 2	Output 3	Output 1	Output 2	Output 3
897.012 +2.66282 ArcTan[x1]-15.6669 ArcTan[x2]+31.3301 ArcTan[x3]-583.628 ArcTan[x4]- 0.493013 ArcTan[x5]- 3.92331 ArcTan[x6]	-28389.7+3.48657/(1+e <sup>x1</sup> )- +85229.3/(1+e <sup>x2</sup> )- 28389.7/(1+e <sup>x3</sup> )- 28389.7/(1+e <sup>x4</sup> )- 0.232106/(1+e <sup>x5</sup> )- 59.9122/(1+e <sup>x6</sup> )	0.127465 +0.00589561 x1- 0.0000698996 x2 + 0.0000502358 x3- 2.84728*10 <sup>-6</sup> x4 +0.000426472 x5 +0.00074837 x6	0.9850	0.9968	0.9997	0.9558	0.8999	0.998	0.9844	0.8043	0.9743

**Table 3.** Results of optimization problems for the selected models.

Outputs	Aim	Constraints	Results	Suggested Design
Output 1	Maximize	$0.8 < x_1 < 1.6, 14 < x_2 < 30, 120 < x_3 < 220, 900 < x_4 < 1380, 2 < x_5 < 10, 7 < x_6 < 10$	3.02141	$x_1=1.6, x_2=14, x_3=220, x_4=900, x_5=2, x_6=7$
Output 2	Minimize	$0.8 < x_1 < 1.6, 14 < x_2 < 30, 120 < x_3 < 220, 900 < x_4 < 1380, 2 < x_5 < 10, 7 < x_6 < 10$	2.36884	$x_1=0.8, x_2=14, x_3=210.748, x_4=900.001, x_5=10, x_6=10$
Output 3	Minimize	$0.8 < x_1 < 1.6, 14 < x_2 < 30, 120 < x_3 < 220, 900 < x_4 < 1380, 2 < x_5 < 10, 7 < x_6 < 10$	0.138276	$x_1=0.8, x_2=30, x_3=120, x_4=1380, x_5=2, x_6=7$

When the table is examined, it is seen that different optimum values emerge for each output. This is an indication that the engineering parameters to be maximized or minimized have different dynamics from each other. For this reason, it reveals how important the different optimization definitions made within the scope of this study are. The minimum surface roughness  $R_a$  (2.36  $\mu\text{m}$ ) was obtained for the following optimal conditions;

Pulse on time ( $T_{on}$ ):0.8  $\mu\text{s}$ , Pulse off time ( $T_{off}$ ):14  $\mu\text{s}$ , Peak current ( $I_p$ ):210.74 A, Servo reference voltage (SV):900.001 gm, Wire tension (WT):10 V, Dielectric flow rate (discharge pressure) (FR):10  $\text{kg}/\text{cm}^2$ . The minimum wire offset (0.13 mm) was obtained for the following optimal conditions; Pulse on time ( $T_{on}$ ):0.8  $\mu\text{s}$ , Pulse off time ( $T_{off}$ ):30  $\mu\text{s}$ , Peak current ( $I_p$ ):120 A, Servo reference voltage (SV):1380 gm, Wire tension (WT):2 V, Dielectric flow rate (discharge pressure) (FR):7  $\text{kg}/\text{cm}^2$ . They are also shown in Table 3. These tables (Table 1, Table 2, Table 3, Table 4) have been prepared to show that we have successfully created the model for outputs and implemented them into the project properly. Results are suitable with intervals of our inputs

### 3. Conclusions

In the present research, wire electrical discharge machining of  $\gamma$  titanium aluminide alloy has been carried out, and an advanced optimization strategy has been proposed to determine the optimal combination of control parameters. A stochastic optimization method is used to construct the WEDM process model. During the training process, several optimization models were studied. It has been found that one model can provide a better prediction for each output.

The wire offset value, together with surface finish and cutting speed, have been evaluated as measurements of process performance for improved dimensional control. A model was developed that will enable one to select the optimum model that will result in maximum cutting speed while maintaining the required surface finish within limits. Additionally, the model is also capable of optimizing the machining process (under multi constraint conditions) while maintaining the surface roughness as well as the internal corner radius within specified limits. The findings of the research and the created technical guidelines in the field of  $\gamma$  titanium aluminide alloy machining will also contribute in the resolution of a variety of hard problems faced by manufacturing engineers in today's manufacturing sectors.

### Declaration of Interest

The authors declare that there is no conflict of interest.

### References

- [1] K. H. Ho, and S. T. Newman, "State of the art electrical discharge machining (EDM)," *International Journal of Machine Tools & Manufacture*, vol. 43, pp. 1287–1300, 2003.
- [2] D. Scott, S. Boyina, and K. P. Rajurkar, "Analysis and optimization of parameter combinations in wire electrical discharge machining," *Int J Prod Res*, vol. 29, pp. 2189–2207, 1991.
- [3] Y. S. Liao, J. T. Huang, and H. C. Su, "A study on the machining – parameters optimisation of wire electrical discharge machining," *J Mater Process Technol*, vol. 71 pp. 487–493, 1997.
- [4] R. Karthikkeyan et al., "Mathematical modeling for electric discharge machining of aluminum -silicon carbide particulate

- composites,” *J Mater Process Technol*, vol. 87, pp. 59–63, 1999.
- [5] J. T. Huang, Y. S. Liao, and W. J. Hsue, “Determination of finish-cutting operation number and machining parameters setting in wire electrical discharge machining,” *J Mater Process Technol*, vol. 87, pp. 69–81, 1999.
- [6] S. Sarkar, S. Mitra, and B. Bhattacharyya, “Parametric optimisation of wire electrical discharge machining of  $\gamma$  titanium aluminide alloy through an artificial neural network model,” *Int J Adv Manuf Technol*, vol. 27, pp. 501–508, 2006. <https://doi.org/10.1007/s00170-0>
- [7] L. Aydin, and H. S. Artem, “Comparison of stochastic search optimization algorithms for the laminated composites under mechanical and hygrothermal loadings,” *Journal of Reinforced Plastics and Composites*, vol. 30, no. 14, pp. 1197–1212, 2011. <https://doi.org/10.1177/>
- [8] S. Ozturk, L. Aydin, and E. Celik, “A comprehensive study on slicing processes optimization of silicon ingot for photovoltaic applications,” *Solar Energy*, vol. 161, pp. 109–24, 2018. doi:10.1016/j.solener. 2017.12.040.
- [9] M. Savran, and L. Aydin, “Stochastic optimization of graphite-flax/epoxy hybrid laminated composite for maximum fundamental frequency and minimum cost,” *Engineering Structures*, vol. 174, pp. 675–87, 2018. doi:10.1016/j.engstruct.2018.07.043.
- [10] M. Akcair, M. Savran, L. Aydin, O. Ayakdas, S. Ozturk, and N. Kucukdogan, “Optimum design of anti-buckling behaviour of graphite/epoxy laminated composites by differential evolution and simulated annealing method,” *Research on Engineering Structures and Materials*, vol. 5, pp. 175–88, 2019.
- [11] M. L. A. Savran, “Optimum design of hybrid graphite-flax/epoxy laminated composites for minimum cost, minimum weight and maximum frequency using modified simulated annealing method,” *Dokuz Eylul University Faculty of Engineering Journal of Science and Engineering*, no. 21, pp.833–44, 2019.

# A Novel, Nelder-Mead Optimization Approach, based on Neuro-regression modeling for the Energy Efficiency Parameters of End Milling Process

Şeyda İŞBİLİR\*

Izmir Katip Celebi University, Department of Mechanical Engineering, Cigli, Izmir, Turkey

## Abstract

Global crises are increasing day by day due to the rapid depletion of energy supplies around the planet. One of the goals of engineering is to prevent this situation by developing innovative solutions to this rapid energy consumption that has disappeared in the world. A solution could be to reduce the energy consumption of the machines that are used during production. In this study, a new design technique based on the neuro-regression approach and non-linear regression modeling was offered as an alternative to Taguchi design to reduce energy consumption. Thus, a cutting parameter optimization model was created to examine the effects of the constraint conditions on energy consumption. The cutting power, the surface roughness of the part, and tool life were handled as objective functions (constraint conditions). First of all, the multiple non-linear regression modeling was created using design variables in end milling. These design variables were determined as spindle rotational speed, feed rate power, radial cut depth, axial cut depth, and cutting speed. Then, objective functions were brought to the proper minimum optimal levels due to this optimization modeling. As a result of the optimization model built with design variables, accurate modeling was achieved in this work by studying several optimization models utilized to optimize the minimum objective functions, which play a significant role in reducing energy consumption in end milling. After the optimization, the maximum value was found as 110.791. At the end of the study, some options of direct search method to maximize and minimize results were applied.

**Keywords:** *End milling; energy efficiency; optimization.*

## 1. Introduction

The industrial sector has been pushed to cut energy consumption and enhance energy efficiency since natural energy resources have been depleted due to rising global energy crises and the resulting climate change. Manufacturing is one of the most energy-consuming industries, so there is much potential for energy-saving options to think about, analyze, and test. Energy resources must be used correctly in the manufacturing industry, and energy efficiency must be continuously improved. For example, a suitable energy-oriented machine tool component design or better machine usage, both in terms of machining strategy and process parameter selection, can save energy in machine tools [1].

Machining is a significant part of the manufacturing industry, which is one of the most critical industries. Machining is generally defined as removing material in the form of chips from a workpiece or part. A mechanical part can be machined using different techniques without significant differences in final results. However, machining methods such as end milling play a significant role in producing and shaping parts. End milling can be used to produce slots, shoulders, die cavities, contours, profiles, and other milling parts. It is widely used to create auto parts, aircraft parts, etc. Machine tools are the primary electricity-consuming devices in milling processes, and they are also the source of carbon dioxide emissions [2].

For this reason, end milling machines consume so much energy. Therefore, studying energy parameters is important since cutting parameters in end milling can enhance energy efficiency. Several studies on optimizing cutting parameters have been published in this field; several of them used surface roughness, cutting force, cutting power, tool life, and material removal rate as optimization criteria [3].

There are many combinations of parameters, such as feed rate, spindle speed, axial or radial depth of cut, to achieve varied results in terms of machined surface quality and tool wear, depending on the machining goal and the choice of cutting tool [4]. In addition, each combination of cutting parameters will provide a varied surface roughness and lifetime of the tool [5].

One way to increase energy efficiency is to increase the machine tool's lifetime and achieve minimum surface roughness and cutting power consumption. Many different studies have been conducted to reach these results. Different optimization methods were used in these studies.

Velchev et al. [6] presented an approach to optimize cutting parameters to minimize direct energy consumption during turning. Negrete-Compesto [7] optimized the cutting parameters (cutting speed, cutting depth, and feed rate) for minimizing electrical energy consumption in turning of AISI 6061 T6 by the Taguchi method.

\*Corresponding author e-mail address: [seydaisbilir@outlook.com](mailto:seydaisbilir@outlook.com)

Mativenga and Rajemi [8] established an energy consumption model for a single pass turning process, including energy during machine setup, machining, tool change, and tool production. The energy model could derive the optimal tool for economic life and cutting speed under a minimum energy criterion. Their model can be improved if machine stand-by power was not used to replace power consumption during their study's tool-changing operations. The minimum energy criterion introduced is exploited to improve and apply a methodology for optimum cutting conditions' selection.

The paper by Rajemi et al. [9] aims to improve a novel model and technique for optimizing the energy footprint for a machined product. They introduced the essential parameters in minimizing energy use and therefore reducing the energy cost and environmental footprint.

Asilturk et al. [10] applied response surface methodology to optimize cutting parameters (spindle speed, feed, depth of cut, and tool radius) and developed a surface roughness model of medical alloy machined on a CNC lathe.

In work presented by Bhushan [11], the optimization of turning cutting parameters to minimize electrical energy consumption and maximize tool life was presented. The response surface methodology (RSM) was applied to establish the cutting parameters' electrical energy consumption and tool life models. Results of the research work showed that electrical energy consumption could be reduced by 13.55%, and tool life can be increased by 22.12% with the optimized cutting parameters. Li et al. [12] integrated Taguchi, the response surface methodology and multi-objective particle swarm optimization to optimize energy saving parameters and selected specific energy consumption to evaluate energy efficiency. The results showed that feed rate is the most significant factor for minimizing electrical energy consumption. A higher feed rate provides minimum electrical energy consumption.

The main objective of this paper is to minimize the surface roughness and cutting power of the tool and maximize the lifetime of the machine tool by using a new neuro-regression analysis for improving energy efficiency in end milling. Furthermore, the direct search approach, modified versions of the Nelder-Mead algorithm have been thoroughly tested and shown.

## **2. Materials and Methods**

### **2.1. Optimization**

The notion of optimization, which is the inherent attribute of achieving the best or most beneficial (minimum or maximum) outcome from a given situation, has enormous significance in human affairs and natural laws. Since the beginning of civilization, the human species has faced countless technological obstacles, including determining the best answer to various issues such as control technologies, power sources constructions, economic applications, mechanical engineering, and energy distribution, amongst others. Optimization problems are ubiquitous in science, and even in our daily lives, we optimize how we get to work every morning or how we should navigate to a new place. There is an objective you want to either maximize or minimize in an optimization problem, and there may be constraints within which you need to operate. This shows how design optimization may aid not just in the human activity of producing optimal designs for products, processes, and systems but also in the understanding and analysis of mathematical and physical phenomena and the solution of mathematical problems. [13]

#### **2.1.1. Regression analysis**

Regression analysis is one of the reliable tools frequently used in economics, science, and engineering. Regression analysis allows us to determine which variables have an effect on the subject we are studying. It helps us determine whether we should improve or ignore these variables. The purpose of regression analysis is to express the response variable as a function of the predictive variables. The duality of the fit and the accuracy of the result depend on the data used [14].

Regression analysis can be examined in different categories according to the number and linearity of the predictive variables: Simple and multiple regression; linear and non-linear regression. The most common form of regression analysis is linear regression, it is a model that assesses the relationship between a dependent variable and an independent variable. Multiple linear regression analysis is essentially similar to the simple linear model, except that multiple independent variables are used in the model. When the model function is not linear in the parameters, an iterative procedure must minimize the sum of squares. This introduces many complications, which are summarized in differences between linear and non-linear least squares [15].

### 2.1.2. Neuro-regression approach

To increase the accuracy of the predictions, a technique that combines the strengths of regression analysis and Artificial Neural Network (ANN) is utilized in the modeling step. With this approach, all the data in our table are divided into training, testing, and validation, approximately 80%, 15%, and 5%. However, it is used by choosing an appropriate regression model. If we have only one variable, the simple regression model is used here. If there is more than one variable, a multiple regression model is used. If the variables contain non-linear terms, the non-linear regression model should be used. First, in the training part, the coefficients in the regression model are adjusted to minimize the error between the experimental and predicted values. Then, the inconsistencies of the regression model in the testing part and the validation part are minimized, and estimated results are tried to be reached. After obtaining the appropriate values, each variable's maximum and minimum values in the given ranges are calculated [14, 15].

In the Mathematica code, some terms are used. "Length" will give you the number of data sets in the experiment. The sum of squares total, denoted SST, is the squared differences between the observed dependent variable and its mean. The sum of squares error or SSE is the summation of each element's testing and prediction response data. "Ybar" is the mean of the training data [15]

### 2.1.3. Regression models

In this study, different models were used to reach the appropriate values. Some of them are multiple linear regression, and some are non-linear regression models. The table below contains these models.

**Table 1.** Regression models name with nomenclature – formula. [14, 15]

Model Name	Nomenclature	Formula
Multiple Linear	L	$a[1] + a[2] x1 + a[3] x2 + a[4] x3 + a[5] x4$
Multiple Linear Rational	LR	$(a[1] + a[2] x1 + a[3] x2 + a[4] x3 + a[5] x4)/(b[1] + b[2] x1 + b[3] x2 + b[4] x3 + b[5] x4)$
Second Order Multiple Nonlinear	SON	$a[1] + 2 x1 a[2] + x1^2 a[3] + 2 x2 a[4] + 2 x1 x2 a[5] + x2^2 a[6] + 2 x3 a[7] + 2 x1 x3 a[8] + 2 x2 x3 a[9] + x3^2 a[10] + 2 x4 a[11] + 2 x1 x4 a[12] + 2 x2 x4 a[13] + 2 x3 x4 a[14] + x4^2 a[15]$
Second-Order Multiple Nonlinear Rational	SONR	$(a[1] + 2 x1 a[2] + x1^2 a[3] + 2 x2 a[4] + 2 x1 x2 a[5] + x2^2 a[6] + 2 x3 a[7] + 2 x1 x3 a[8] + 2 x2 x3 a[9] + x3^2 a[10] + 2 x4 a[11] + 2 x1 x4 a[12] + 2 x2 x4 a[13] + 2 x3 x4 a[14] + x4^2 a[15])/(b[1] + 2 x1 b[2] + x1^2 b[3] + 2 x2 b[4] + 2 x1 x2 b[5] + x2^2 b[6] + 2 x3 b[7] + 2 x1 x3 b[8] + 2 x2 x3 b[9] + x3^2 b[10] + 2 x4 b[11] + 2 x1 x4 b[12] + 2 x2 x4 b[13] + 2 x3 x4 b[14] + x4^2 b[15])$
Third Order Multiple Nonlinear	TON	$a[1] + a[2] 3 x1 + a[3] 3 x1^2 + a[4] x1^3 + a[5] 3 x2 + a[6] 6 x1 x2 + a[7] 3 x1^2 x2 + a[8] 3 x2^2 + a[9] 3 x1 x2^2 + a[10] x2^3 + a[11] 3 x3 + a[12] 6 x1 x3 + a[13] 3 x1^2 x3 + a[14] 6 x2 x3 + a[15] 6 x1 x2 x3 + a[16] 3 x2^2 x3 + a[17] 3 x3^2 + a[18] 3 x1 x3^2 + a[19] 3 x2 x3^2 + a[20] x3^3 + a[21] 3 x4 + a[22] 6 x1 x4 + a[23] 3 x1^2 x4 + a[24] 6 x2 x4 + a[25] 6 x1 x2 x4 + a[26] 3 x2^2 x4 + a[27] 6 x3 x4 + a[28] 6 x1 x3 x4 + a[29] 6 x2 x3 x4 + a[30] 3 x3^2 x4 + a[31] 3 x4^2 + a[32] 3 x1 x4^2 + a[33] 3 x2 x4^2 + a[34] 3 x3 x4^2 + a[35] x4^3$
Fourth Order Multiple Nonlinear	FON	$a[1] 1 + a[2] 4 x1 + a[3] 6 x1^2 + a[4] 4 x1^3 + a[5] x1^4 + a[6] 4 x2 + a[7] 12 x1 x2 + a[8] 12 x1^2 x2 + a[9] 4 x1^3 x2 + a[10] 6 x2^2 + a[11] 12 x1 x2^2 + a[12] 6 x1^2 x2^2 + a[13] 4 x2^3 + a[14] 4 x1 x2^3 + a[15] x2^4 + a[16] 4 x3 + a[17] 12 x1 x3 + a[18] 12 x1^2 x3 + a[19] 4 x1^3 x3 + a[20] 12 x2 x3 + a[21] 24 x1 x2 x3 + a[22] 12 x1^2 x2 x3 + a[23] 12 x2^2 x3 + a[24] 12 x1 x2^2 x3 + a[25] 4 x2^3 x3 + a[26] 6 x3^2 + a[27] 12 x1 x3^2 + a[28] 6 x1^2 x3^2 + a[29] 12 x2 x3^2 + a[30] 12 x1 x2 x3^2 + a[31] 6 x2^2 x3^2 + a[32] 4 x3^3 + a[33] 4 x1 x3^3 + a[34] 4 x2 x3^3 + x3^4 + a[35] 4 x4 + a[36] 12 x1 x4 + a[37] 12 x1^2 x4 + a[38] 4 x1^3 x4 + a[39] 12 x2 x4 + a[40] 24 x1 x2 x4 + a[41] 12 x1^2 x2 x4 + a[42] 12 x2^2 x4 + a[43] 12 x1 x2^2 x4 + a[44] 4 x2^3 x4 + a[45] 12 x3 x4 + a[46] 24 x1 x3 x4 + a[47] 12 x1^2 x3 x4 + a[48] 24 x2 x3 x4 + a[49] 24 x1 x2 x3 x4 + a[50] 12 x2^2 x3 x4 + a[51] 12 x3^2 x4 + a[52] 12 x1 x3^2 x4 + a[53] 12 x2 x3^2 x4 + a[54] 4 x3^3 x4 + a[55] 6 x4^2 + a[56] 12 x1 x4^2 + a[57] 6 x1^2 x4^2 + a[58] 12 x2 x4^2 + a[59] 12 x1 x2 x4^2 + a[60] 6 x2^2 x4^2 + a[61] 12 x3 x4^2 + a[62] 12 x1 x3 x4^2 + a[63] 12 x2 x3 x4^2 + a[64] 6 x3^2 x4^2 + a[65] 4 x4^3 + a[66] 4 x1 x4^3 + a[67] 4 x2 x4^3 + a[67] 4 x3 x4^3 + a[68] x4^4$

First Order Trigonometric Multiple Nonlinear	FOTN	$a[1] + a[2] \text{Cos}[x1] + a[3] \text{Cos}[x2] + a[4] \text{Cos}[x3] + a[5] \text{Cos}[x4] + a[6] \text{Sin}[x1] + a[7] \text{Sin}[x2] + a[8] \text{Sin}[x3] + a[9] \text{Sin}[x4]$
First Order Trigonometric Multiple Nonlinear Rational	FOTNR	$a[1] + a[2] \text{Cos}[x1] + a[3] \text{Cos}[x2] + a[4] \text{Cos}[x3] + a[5] \text{Cos}[x4] + a[6] \text{Sin}[x1] + a[7] \text{Sin}[x2] + a[8] \text{Sin}[x3] + a[9] \text{Sin}[x4] / (b[1] + b[2] \text{Cos}[x1] + b[3] \text{Cos}[x2] + b[4] \text{Cos}[x3] + b[5] \text{Cos}[x4] + b[6] \text{Sin}[x1] + b[7] \text{Sin}[x2] + b[8] \text{Sin}[x3] + b[9] \text{Sin}[x4])$
Second Order Trigonometric Multiple Nonlinear	SOTN	$a[1] + 2 a[2] \text{Cos}[x1] + a[3] \text{Cos}[x1]^2 + 2 a[4] \text{Cos}[x2] + 2 a[5] \text{Cos}[x1] \text{Cos}[x2] + a[6] \text{Cos}[x2]^2 + 2 a[7] \text{Cos}[x3] + 2 a[8] \text{Cos}[x1] \text{Cos}[x3] + 2 a[9] \text{Cos}[x2] \text{Cos}[x3] + a[10] \text{Cos}[x3]^2 + 2 a[11] \text{Cos}[x4] + 2 a[12] \text{Cos}[x1] \text{Cos}[x4] + 2 a[13] \text{Cos}[x2] \text{Cos}[x4] + 2 a[14] \text{Cos}[x3] \text{Cos}[x4] + a[15] \text{Cos}[x4]^2 + 2 a[16] \text{Cos}[x5] + 2 a[17] \text{Cos}[x1] \text{Cos}[x5] + 2 a[18] \text{Cos}[x2] \text{Cos}[x5] + 2 a[19] \text{Cos}[x3] \text{Cos}[x5] + 2 a[20] \text{Cos}[x4] \text{Cos}[x5] + a[21] \text{Cos}[x5]^2 + 2 a[22] \text{Sin}[x1] + 2 a[23] \text{Cos}[x1] \text{Sin}[x1] + 2 a[24] \text{Cos}[x2] \text{Sin}[x1] + 2 a[25] \text{Cos}[x3] \text{Sin}[x1] + 2 a[26] \text{Cos}[x4] \text{Sin}[x1] + 2 a[27] \text{Cos}[x5] \text{Sin}[x1] + a[28] \text{Sin}[x1]^2 + 2 a[29] \text{Sin}[x2] + 2 a[30] \text{Cos}[x1] \text{Sin}[x2] + 2 a[31] \text{Cos}[x2] \text{Sin}[x2] + 2 a[32] \text{Cos}[x3] \text{Sin}[x2] + 2 a[33] \text{Cos}[x4] \text{Sin}[x2] + 2 a[34] \text{Cos}[x5] \text{Sin}[x2] + 2 a[35] \text{Sin}[x1] \text{Sin}[x2] + a[36] \text{Sin}[x2]^2 + 2 a[37] \text{Sin}[x3] + 2 a[38] \text{Cos}[x1] \text{Sin}[x3] + 2 a[39] \text{Cos}[x2] \text{Sin}[x3] + 2 a[40] \text{Cos}[x3] \text{Sin}[x3] + 2 a[41] \text{Cos}[x4] \text{Sin}[x3] + 2 a[42] \text{Cos}[x5] \text{Sin}[x3] + 2 a[43] \text{Sin}[x1] \text{Sin}[x3] + 2 a[44] \text{Sin}[x2] \text{Sin}[x3] + a[45] \text{Sin}[x3]^2 + 2 a[46] \text{Sin}[x4] + 2 a[47] \text{Cos}[x1] \text{Sin}[x4] + 2 a[48] \text{Cos}[x2] \text{Sin}[x4] + 2 a[49] \text{Cos}[x3] \text{Sin}[x4] + 2 a[50] \text{Cos}[x4] \text{Sin}[x4] + 2 a[51] \text{Cos}[x5] \text{Sin}[x4] + 2 a[52] \text{Sin}[x1] \text{Sin}[x4] + 2 a[53] \text{Sin}[x2] \text{Sin}[x4] + 2 a[54] \text{Sin}[x3] \text{Sin}[x4] + a[55] \text{Sin}[x4]^2 + 2 a[56] \text{Sin}[x5] + 2 a[57] \text{Cos}[x1] \text{Sin}[x5] + 2 a[58] \text{Cos}[x2] \text{Sin}[x5] + 2 a[59] \text{Cos}[x3] \text{Sin}[x5] + 2 a[60] \text{Cos}[x4] \text{Sin}[x5] + 2 a[61] \text{Cos}[x5] \text{Sin}[x5] + 2 a[62] \text{Sin}[x1] \text{Sin}[x5] + 2 a[63] \text{Sin}[x2] \text{Sin}[x5] + 2 a[64] \text{Sin}[x3] \text{Sin}[x5] + 2 a[65] \text{Sin}[x4] \text{Sin}[x5] + a[66] \text{Sin}[x5]^2$
First Order Logarithmic Multiple Nonlinear	FOLN	$a[1] + a[2] \text{Log}[x1] + a[3] \text{Log}[x2] + a[4] \text{Log}[x3] + a[5] \text{Log}[x4] + a[6] \text{Log}[x5]$
Second Order Logarithmic Multiple Nonlinear	SOLN	$a[1] + 2 a[2] \text{Log}[x1] + a[3] \text{Log}[x1]^2 + 2 a[4] \text{Log}[x2] + 2 a[5] \text{Log}[x1] \text{Log}[x2] + a[6] \text{Log}[x2]^2 + 2 a[7] \text{Log}[x3] + 2 a[8] \text{Log}[x1] \text{Log}[x3] + 2 a[9] \text{Log}[x2] \text{Log}[x3] + a[10] \text{Log}[x3]^2 + 2 a[11] \text{Log}[x4] + 2 a[12] \text{Log}[x1] \text{Log}[x4] + 2 a[13] \text{Log}[x2] \text{Log}[x4] + 2 a[14] \text{Log}[x3] \text{Log}[x4] + a[15] \text{Log}[x4]^2 + 2 a[16] \text{Log}[x5] + 2 a[17] \text{Log}[x1] \text{Log}[x5] + 2 a[18] \text{Log}[x2] \text{Log}[x5] + 2 a[19] \text{Log}[x3] \text{Log}[x5] + 2 a[20] \text{Log}[x4] \text{Log}[x5] + a[21] \text{Log}[x5]^2$

**2.1.4. Problem definition**

The main problem in this study is to obtain suitable results due to optimizing the objective functions used to increase energy efficiency. Tables 2-4 give the information about the results of surface roughness and cutting power of the machine, the lifetime of the machine, and constraints of design variables of objective functions, respectively. Surface roughness and cutting power should be minimized, and the lifetime of the machine should be increased. These outputs depend on some predictive variables. As a result of controlling these predictive variables, maximum or minimum values of outputs can be reached. These constraint (predictive) variables are feed rate, spindle speed, axial and radial depth of cut, and cutting speed. Output 1 (surface roughness) and output 2 (cutting power) have the same constraint variables. These variables are feed rate, spindle speed, axial and radial depth of cut. The constraint variables of output 3 (lifetime of the machine) are feed rate, spindle speed, axial and radial depth of cut, and cutting speed. Output 1 and output 2 have 25 different data sets; Output 3 has nine different data sets. Therefore, appropriate data must be captured for these output values. For this purpose, at the end of the neuro-regression approach, the  $R^2_{\text{training}}$  value must be greater than 0.90, the  $R^2_{\text{testing}}$  value must be greater than 0.85, and finally, the  $R^2_{\text{validation}}$  value must be greater than 0.85 to fit the accurate model. If we obtain these values, the appropriate model is considered to have been reached.



**Table 2.** Data table of output 1 (surface roughness) and output 2 (cutting power). [2]

No.	n (r/min)	v <sub>f</sub> (mm/min)	a <sub>p</sub> (mm)	a <sub>e</sub> (mm)	R <sub>a</sub> (μm)	Power (kW)
1	800	240	0.5	2	2.141	0.020
2	800	290	1	3	2.793	0.060
3	800	340	1.5	4	2.837	0.100
4	800	380	2	5	3.512	0.200
5	800	420	2.5	6	4.013	0.300
6	1200	240	1	4	2.171	0.069
7	1200	290	1.5	5	2.331	0.134
8	1200	340	2	6	2.424	0.244
9	1200	380	2.5	2	1.924	0.119
10	1200	420	0.5	3	2.187	0.047
11	1600	240	1.5	6	2.024	0.140
12	1600	290	2	2	1.897	0.086
13	1600	340	2.5	3	1.799	0.155
14	1600	380	0.5	4	2.17	0.060
15	1600	420	1	5	2.373	0.150
16	2000	240	2	3	1.533	0.143
17	2000	290	2.5	4	1.565	0.253
18	2000	340	0.5	5	1.684	0.068
19	2000	380	1	6	1.76	0.178
20	2000	420	1.5	2	1.741	0.113
21	2400	240	2.5	5	1.497	0.290
22	2400	290	0.5	6	1.728	0.115
23	2400	340	1	2	1.415	0.085
24	2400	380	1.5	3	1.321	0.185
25	800	240	0.5	2	2.141	0.020

**Table 3.** Output 3 (lifetime of machine tool)  $T_{life}$ .

No.	n (r/min)	v <sub>f</sub> (mm/min)	a <sub>p</sub> (mm)	a <sub>e</sub> (mm)	V <sub>c</sub> (m/s)	T <sub>life</sub> (min)
1	1300	120	2.5	3	0.816	80
2	1300	140	3	6	0.816	62
3	1300	180	3.5	9	0.816	53
4	1700	120	3	9	1.068	49
5	1700	140	3.5	3	1.068	55
6	1700	180	2.5	6	1.068	50
7	2100	120	3.5	6	1.319	20
8	2100	140	2.5	9	1.319	30
9	2100	180	3	3	1.319	25

**Table 4.** Constraints of design variables.

Constraints of Design Variables For Output 1 (Surface Roughness) and Output 3 (Cutting Power)	Constraints of Design Variables For Output 3 (Lifetime of Machine)
800 < Spindle Rotation Speed (n) < 2400	1300 < Spindle Rotation Speed (n) < 2100
240 < Feed Rate ( $V_f$ ) < 420	120 < Feed Rate ( $V_f$ ) < 180
0.5 < Radial Cut Depth ( $a_p$ ) < 2.5	2.5 < Radial Cut Depth ( $a_p$ ) < 3.5
2 < Axial Cut Depth ( $a_e$ ) < 6	3 < Axial Cut Depth ( $a_e$ ) < 9
	0.816 < Cutting Speed ( $V_c$ ) < 1.319

### 2.1.4.1. Optimization process of problem

The neuro-regression approach should be used as a first step. The data sets of the outputs should be divided into 80%, 15%, and 5% as training, testing, and validation randomly. Then the appropriate regression models should be used respectively. In this study, non-linear regression modeling was used for three different outputs. The suitable models for outputs 1-3 are the second-order non-linear, third-order non-linear, and second-order trigonometric non-linear regression functions. Then, the regression coefficients were calculated in line with the regression analysis. Mathematica's *NonlinearModelFit* solver was applied to each model. Some values were obtained as a result of the applied neuro-regression approach. Then, the maximum or minimum desired values of the objective functions were calculated. The modified versions of the Nelder-Mead algorithm have been thoroughly tested on output 1 and output 2 to minimize their results and output 3 to maximize its result more and shown in different scenarios.

### 3. Result and Discussion

In this study, the neuro-regression approach was used. Different trials have been made to achieve appropriate results. Moreover, as a result of these trials, suitable models have been reached in order to reach the minimum and maximum values of the objective functions. Inputs and outputs are taken from tables as experimental data sets. It is known that constraint variables directly affect the results of the objective function. In addition to this, appropriate values were tried to be reached as a result of each model. Our first output, the second order non-linear regression model, which gives the most relevant results among the different models, was found suitable and applied for the surface roughness value. As a result of different models applied for our second output, cutting power, the third-order non-linear regression model gave appropriate results. Minimum values have been found for these two objective function values considering different scenarios.

**Table 5.** Results of output 1.

Model of Output 1	$R^2_{\text{training}}$	$R^2_{\text{testing}}$	$R^2_{\text{validation}}$	Maximum	Minimum
SON	0.997314	0.927675	0.883081	4.83422	-2.97059

**Table 6.** Results of output 2.

Model of Output 2	$R^2_{\text{training}}$	$R^2_{\text{testing}}$	$R^2_{\text{validation}}$	Maximum	Minimum
TON	1	0.90423	0.942348	0.488696	0.0275759

**Table 7.** Results of output 3.

Model of Output 3	$R^2_{\text{training}}$	$R^2_{\text{testing}}$	$R^2_{\text{validation}}$	Maximum	Minimum
L	1	0.1505	0.646551	80.2801	15.1206
SON	1	0.7819	-0.08065	80.6516	15.1206
TON	1	0.2675	-1.09194	92.2776	3.67565
FOLN	1	-0.3127	-0.02565	85.6958	0.225738
SOLN	1	0.6429	0.619779	88.2654	8.67709
FOTN	1	0.0351	0.708655	87.8171	-0.536101
SOTN	1	-1.4135	0.768006	102.188	9.56213
HYBRID	1	0.4996	0.996241	110.791	2.43927

Eight different models have been implemented for the third output, the lifetime of the machine. Due to the limited data set of the objective function, relevant results could not be reached. For this reason, the second-order trigonometric non-linear regression model, which is the modeling that gives the closest results we want to obtain, was used. The following tables show the appropriate models for objective functions and their resulting values,  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$ ,  $R^2_{\text{validation}}$ . Results are maximized and minimized for the best models. As a result of these processes, suitable design variables were determined to bring each output to appropriate values. For each model, the design variables required to achieve optimum results are given in Table 8-10.

**Table 8.** Design variables values for output 1.

Model for Output 1	Design Variables Values for Nminimize			
SON	$x_1 \rightarrow 800.$	$x_2 \rightarrow 240.$	$x_3 \rightarrow 2.5$	$x_4 \rightarrow 2.$

**Table 9.** Design variables values for output 2.

Model for Output 2	Design Variables Values for Nminimize			
TON	$x_1 \rightarrow 800.$	$x_2 \rightarrow 240.$	$x_3 \rightarrow 2.5$	$x_4 \rightarrow 2.$

**Table 10.** Design variables values for output 3.

Model for Output 3	Design Variables Values for Nminimize				
SOTLN	$x_1 \rightarrow 1697$	$x_2 \rightarrow 180.$	$x_3 \rightarrow 3.5$	$x_4 \rightarrow 9.$	$x_5 \rightarrow 1.32$

**Table 11.** Results of optimization based on Modified Nelder-Mead algorithm for output 1.

Model for Output 1	Scenario Number	Algorithm options	Outputs	Design Variables
SON	1 $800 < x_1 < 2400$ $240 < x_2 < 420$ $0.5 < x_3 < 2.5$ $2 < x_4 < 6$	Default	-2.97059	$x_1 \rightarrow 800. x_2 \rightarrow 240. x_3 \rightarrow 2.5, x_4 \rightarrow 2.$
		Random Seed $\rightarrow 200$	-2.97059	$x_1 \rightarrow 800. x_2 \rightarrow 240. x_3 \rightarrow 2.5 x_4 \rightarrow 2.$
		Contract Ratio $\rightarrow 0.10$	-2.97059	$x_1 \rightarrow 800 x_2 \rightarrow 240 x_3 \rightarrow 2.5 x_4 \rightarrow 2.$
		Reflect Ratio $\rightarrow 2$	-2.97059	$x_1 \rightarrow 800 x_2 \rightarrow 240 x_3 \rightarrow 2.5 x_4 \rightarrow 2.$
	2 $800 < x_1 < 2400$ $240 < x_2 < 420$ $0.5 < x_3 < 2.5$ $2 < x_4 < 6$ $(x_1, x_2, x_3, x_4) \in \text{Integers}$	Default	-1.04013	$x_1 \rightarrow 801 x_2 \rightarrow 241 x_3 \rightarrow 2 x_4 \rightarrow 3$
		Random Seed $\rightarrow 200$	-1.04013	$x_1 \rightarrow 801 x_2 \rightarrow 241 x_3 \rightarrow 2 x_4 \rightarrow 3$
		Contract Ratio $\rightarrow 0.10$	-1.04013	$x_1 \rightarrow 801 x_2 \rightarrow 241 x_3 \rightarrow 2 x_4 \rightarrow 3$
		Reflect Ratio $\rightarrow 2$	-1.04013	$x_1 \rightarrow 801, x_2 \rightarrow 241, x_3 \rightarrow 2, x_4 \rightarrow 3$
	3 $x_1 \geq 800    x_1 \leq 2400,$ $x_2 \geq 240    x_2 \leq 420,$ $x_3 \geq 0.5    x_3 \leq 2.5,$ $x_4 \geq 2    x_4 \leq 6$	Default	$-1.33004 \times 10^{245}$	$x_1 \rightarrow -5.285 \times 10^{119}, x_2 \rightarrow 240.$ $x_3 \rightarrow 4.709 \times 10^{122}, x_4 \rightarrow 2.121 \times 10^{121}$
		Random Seed $\rightarrow 50$	$-9.988998 \times 10^{235}$	$x_1 \rightarrow -1.44 \times 10^{115}, x_2 \rightarrow -1.3 \times 10^{116},$ $x_3 \rightarrow 1.290 \times 10^{118}, x_4 \rightarrow 5.81 \times 10^{116}$
		Contract Ratio $\rightarrow 0.9$	$-9.988998 \times 10^{235}$	$x_1 \rightarrow -1.5 \times 10^{115}, x_2 \rightarrow -1.38 \times 10^{116},$ $x_3 \rightarrow 1.290 \times 10^{118}, x_4 \rightarrow 5.81 \times 10^{116}$
		Reflect Ratio $\rightarrow 2$	$-9.812743 \times 10^{276}$	$x_1 \rightarrow -4.605 \times 10^{135}, x_2 \rightarrow 240.$ $x_3 \rightarrow 4.053 \times 10^{138}, x_4 \rightarrow 6.$

**Table 12.** Results of optimization based on Modified Nelder-Mead algorithm for output 2.

Model for Output 2	Scenario Number	Algorithm options	Outputs	Design Variables
TON	1 $800 < x_1 < 2400$ $240 < x_2 < 420$ $0.5 < x_3 < 2.5$ $2 < x_4 < 6$	Default	0.027576	$x_1 \rightarrow 800, x_2 \rightarrow 240, x_3 \rightarrow 2.5, x_4 \rightarrow 2.$
		Random Seed $\rightarrow 111$	0.004081	$x_1 \rightarrow 2400, x_2 \rightarrow 420, x_3 \rightarrow 0.5, x_4 \rightarrow 2.$
		Contract Ratio $\rightarrow 0.9$	0.004081	$x_1 \rightarrow 2400, x_2 \rightarrow 420, x_3 \rightarrow 0.5, x_4 \rightarrow 2.$
		Reflect Ratio $\rightarrow 2$	0.004081	$x_1 \rightarrow 2400, x_2 \rightarrow 420, x_3 \rightarrow 0.5, x_4 \rightarrow 2.$
	2 $800 < x_1 < 2400$ $240 < x_2 < 420$ $0.5 < x_3 < 2.5$ $2 < x_4 < 6$ $\{x_1, x_2, x_3, x_4\} \in \text{Integers}$	Default	0.060015	$x_1 \rightarrow 801, x_2 \rightarrow 284, x_3 \rightarrow 1, x_4 \rightarrow 3$
		Random Seed $\rightarrow 111$	0.056212	$x_1 \rightarrow 801, x_2 \rightarrow 241, x_3 \rightarrow 2, x_4 \rightarrow 3$
		Contract Ratio $\rightarrow 0.7$	0.056212	$x_1 \rightarrow 801, x_2 \rightarrow 241, x_3 \rightarrow 2, x_4 \rightarrow 3$
		Reflect Ratio $\rightarrow 2$	0.056212	$x_1 \rightarrow 801, x_2 \rightarrow 241, x_3 \rightarrow 2, x_4 \rightarrow 3$
	3 $x_1 \geq 800    x_1 \leq 2400,$ $x_2 \geq 240    x_2 \leq 420,$ $x_3 \geq 0.5    x_3 \leq 2.5,$ $x_4 \geq 2    x_4 \leq 6$	Default	$-1.22984 \times 10^{620}$	$x_1 \rightarrow 2400, x_2 \rightarrow 3.3142 \times 10^{206}$ $x_3 \rightarrow 0.5, x_4 \rightarrow 4.72 \times 10^{207}$
		Random Seed $\rightarrow 50$	$-1.22984 \times 10^{620}$	$x_1 \rightarrow 2400, x_2 \rightarrow 3.3142 \times 10^{206}$ $x_3 \rightarrow 0.5, x_4 \rightarrow 4.72 \times 10^{207}$
		Contract Ratio $\rightarrow 0.5$	$-1.22984 \times 10^{620}$	$x_1 \rightarrow 2400, x_2 \rightarrow 3.3142 \times 10^{206}$ $x_3 \rightarrow 0.5, x_4 \rightarrow 4.72 \times 10^{207}$
		Reflect Ratio $\rightarrow 1$	$-1.22984 \times 10^{620}$	$x_1 \rightarrow 2400, x_2 \rightarrow 3.3142 \times 10^{206}$ $x_3 \rightarrow 0.5, x_4 \rightarrow 4.72 \times 10^{207}$

**Table 13.** Results of optimization based on Modified Nelder-Mead algorithm for output 3.

Model for Output 3	Scenario Number	Algorithm options	Outputs	Design Variables
	1 $1300 < x_1 < 2100$ $120 < x_2 < 180$ $2.5 < x_3 < 3.5$ $3 < x_4 < 9$ $0.816 < x_5 < 1.319$	Default	110.791	$x_1 \rightarrow 1709.03, x_2 \rightarrow 120,$ $x_3 \rightarrow 3.5, x_4 \rightarrow 3, x_5 \rightarrow 1.319.$
		Random Seed $\rightarrow 111$	110.791	$x_1 \rightarrow 1709.03, x_2 \rightarrow 120,$ $x_3 \rightarrow 3.5, x_4 \rightarrow 3, x_5 \rightarrow 1.319.$
		Shrink Ratio $\rightarrow 1$	110.791	$x_1 \rightarrow 1709.03, x_2 \rightarrow 120,$ $x_3 \rightarrow 3.5, x_4 \rightarrow 3, x_5 \rightarrow 1.319.$
		Expand Ratio $\rightarrow 3$	110.791	$x_1 \rightarrow 1709.03, x_2 \rightarrow 120,$ $x_3 \rightarrow 3.5, x_4 \rightarrow 3, x_5 \rightarrow 1.319.$
	2 $1300 < x_1 < 2100$ $120 < x_2 < 180$ $2.5 < x_3 < 3.5$ $3 < x_4 < 9$ $0.816 < x_5 < 1.319$ $\{x_1, x_2, x_3, x_4, x_5\} \in \text{Integers}$	Default	98.3408	$x_1 \rightarrow 1332, x_2 \rightarrow 139,$ $x_3 \rightarrow 3, x_4 \rightarrow 5, x_5 \rightarrow 1$
		Random Seed $\rightarrow 300$	98.3534	$x_1 \rightarrow 1464, x_2 \rightarrow 140,$ $x_3 \rightarrow 3, x_4 \rightarrow 5, x_5 \rightarrow 1$
		Shrink Ratio $\rightarrow 3$	98.3534	$x_1 \rightarrow 1464, x_2 \rightarrow 140,$ $x_3 \rightarrow 3, x_4 \rightarrow 5, x_5 \rightarrow 1$
		Expand Ratio $\rightarrow 3$	98.3534	$x_1 \rightarrow 1464, x_2 \rightarrow 140,$ $x_3 \rightarrow 3, x_4 \rightarrow 5, x_5 \rightarrow 1$

**4. Conclusion**

This study aimed to increase the machine's life span while minimizing the surface roughness and cutting power. There were four different variables and 25 data sets for surface roughness and cutting power. For the life span, there were five different variables and 9 data sets. The aim here was to reach a suitable regression model that can give us the necessary optimum output values depending on the variables. The neuro-regression approach obtained an appropriate regression model so that the  $R^2_{\text{training}}$  value must be greater than 0.90, the  $R^2_{\text{testing}}$  value must be greater than 0.85, and finally, the  $R^2_{\text{validation}}$  value must be greater than 0.85. For output 1 (surface roughness), these values were obtained conveniently as 0.997314, 0.927675, and 0.883081 for  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$ , and  $R^2_{\text{validation}}$ , respectively. As the desired values were found to be suitable, the second-order non-linear regression model was

used. After the optimization, the minimum value was found -2.97059.

For output 2 (cutting power), these values were obtained as  $R^2_{\text{training}}$  is equal to 1,  $R^2_{\text{testing}}$  equals 0.90423, and  $R^2_{\text{validation}}$  equals 0.94235. As the desired values were found to be suitable, the third-order non-linear regression model was used. After the optimization process, the minimum value was found as 0.0275759. For output 3 (lifetime of the machine), these values were obtained as  $R^2_{\text{training}}$  is equal to 1,  $R^2_{\text{testing}}$  is equal to 0.4996, and  $R^2_{\text{validation}}$  is equal to 0.996241. An accurate model could not be reached due to the scarcity of a number of data sets for output 3. Instead, a model was used in which the values closer to the desired values are provided. As the desired values were found to be closer, the second-order trigonometric non-linear regression model was used. After the optimization, the maximum value was found as 110.791. At the end of the study, some options of the direct search method to maximize and minimize results were applied. After applied some scenarios, it has been seen that the third scenario is more suitable for output 1 and output 2, and it has been seen that the first scenario is more suitable for output 3.

### Declaration of Interest

The authors declare that there is no conflict of interest.

### References

- [1] P. Albertellia, A. Kesharia, and A. Matta, "Energy oriented multi cutting parameter optimization in face milling," *Journal of Cleaner Production*, vol.137, pp. 1602-1618, 2016
- [2] L. Zhoua, J. Lia, F. Lia, G. Mendisb, and J. W. Sutherlandb, "Optimization parameters for energy efficiency in end milling," *PROCEDIA CIRP*, vol. 69, pp. 312-317, Copenhagen, Denmark, 2018.
- [3] R. Kumar, P. S. Bilga, and S. Singh, "Multi-objective optimization using different methods of assigning weights to energy consumption responses, surface roughness and material removal rate during rough turning operation," *Journal of Cleaner Production*, vol.164, pp.45-57, 2017
- [4] B. A. Khidhir, and B. Mohamed, "Analyzing the effect of cutting parameters on surface roughness and tool wear when machining nickel-based Hastelloy – 276." *Materials Science and Engineering Conference Series*, vol.17, pp. 012043, 2011.
- [5] J. Ribeiro, H. Lopes, L. Queijo, and D. Figueiredo, "Optimization of cutting parameters to minimize the surface roughness in the end milling process using the Taguchi method." *Periodica Polytechnica. Engineering. Mechanical Engineering*; Budapest, vol. 61, pp. 30-35, 2017.
- [6] S. Velchev, I. Kolev, K. Ivanov, and S. Gechevski, "Empirical models for specific energy consumption and optimization of cutting parameters for minimizing energy consumption during turning," *J. Clean. Prod.*, vol. 80, pp. 139-149, 2014.
- [7] C. C. Negrete, "Optimization of cutting parameters for minimizing energy consumption in turning of AISI 6061 T6 using Taguchi methodology and ANOVA," *Journal of Cleaner Production*, vol. 53, pp.195-203, 2013.
- [8] P. T. Mativenga, and M. F. Rajemi, "Calculation of optimum cutting parameters based on minimum energy footprint," *CIRP Ann Manuf Technol*, vol. 60, pp. 149-152, 2011.
- [9] M. F. Rajemi, P. T. Mativenga, and A. Aramcharoen, "Sustainable machining : selection of optimum turning conditions based on minimum energy considerations," *J Clean Prod*, vol. 18, pp, 1059-1065, 2010.
- [10] I. Asilturk, S. Neseli, and M.A. Ince, "Optimisation of parameters affecting surface roughness of Co28Cr6Mo medical material during CNC lathe machining by using the Taguchi and RSM methods," *Measurement*, vol. 78, pp. 120-128, 2016.
- [11] R. K. Bhushan, "Optimization of cutting parameters for minimizing power consumption and maximizing tool life during machining of Al alloy SiC particle composites," *Journal of Cleaner Production*, vol. 39, pp. 242-254, 2013.
- [12] C. Li, Q. Xiao, Y. Tang, and L. Li, "A method integrating Taguchi, RSM and MOPSO to CNC machining parameters optimization for energy saving," *J. Clean. Prod.*, vol. 135, pp. 263-275, 2016.
- [13] A. J. Kulkarni, K. Tai, and A. Abraham, "Probability collectives: a distributed multi-agent system approach for optimization," In: *Intelligent Systems Reference Library*, pp. 86, 2015.
- [14] I. Polatoğlu, L. Aydın, B.C. Nevruz and S. Ozer, "A Novel Approach for the Optimal Design of a Biosensor". *Analytical Letters*, pp.1-18, 2020.
- [15] I. Polatoglu, and L. Aydin, "A new design strategy with stochastic optimization on the preparation of magnetite cross-linked tyrosinase aggregates (MCLTA)," *Process Biochemistry*, vol. 99, pp. 131–138, 2020.

APPENDIX

Model Name For Output 1	Model
SON	$-6.509885663403495 + 0.005618299339813021x1 - 3.177186608513434 \times 10^{-7}x1^2 + 0.044188550823988065x2 - 0.00001925906852920848x1x2 - 0.000045373242358123256x2^2 - 4.828733842049232x3 + 0.0013505663904117497x1x3 + 0.012871712805669044x2x3 - 0.5974390681670687x3^2 + 0.3084440279573145x4 - 0.0001638361563178854x1x4 + 0.0005907347821132539x2x4 - 0.053084237534413624x3x4 - 0.009306210170207618x4^2$
Model Name For Output 2	Model
TON	$0.0873885736338913 + 0.00002303413484075954x1 - 2.472633303621079 \times 10^{-8}x1^2 + 1.901217799089097 \times 10^{-11}x1^3 - 0.00011773726012124803x2 + 9.271486175818838 \times 10^{-9}x1x2 - 1.709038139214161 \times 10^{-10}x1^2x2 - 4.579728866634524 \times 10^{-7}x2^2 + 9.684495546735244 \times 10^{-11}x1x2^2 + 1.930501597745149 \times 10^{-9}x2^3 - 0.020880825356986286x3 + 0.000021022527585904126x1x3 - 3.431749926300749 \times 10^{-9}x1^2x3 - 0.00006880981775731341x2x3 + 1.2381322305199 \times 10^{-7}x1x2x3 + 2.531771100982355 \times 10^{-7}x2^2x3 - 0.01704700883302848x3^2 - 0.000006308008574305507x1x3^2 + 0.000011876273887436986x2x3^2 + 0.0037771244976612574x3^3 + 0.0000795324182137095x4 + 0.000002339053662878461x1x4 + 3.547220566752366 \times 10^{-9}x1^2x4 - 0.00004285498644654708x2x4 + 3.68911641138247 \times 10^{-8}x1x2x4 - 3.11930870044312 \times 10^{-7}x2^2x4 - 0.003852659541721482x3x4 + 0.000001461440134025731x1x3x4 + 0.000009925109721374419x2x3x4 - 0.0011541084133987402x3^2x4 + 0.0015191113584700662x4^2 - 0.000002448557820791596x1x4^2 + 0.0000360448905660144x2x4^2 + 0.0024467728436591216x3x4^2 - 0.001172013876941x4^3$
Model Name For Output 3	Model
L	$198.69752228249897 - 0.021872444963457895x1 - 0.2098362545789043x2 - 14.689949495039151x3 + 0.046687461628923115x4 - 34.924242558607936x5$
SON	$64.64181129905197 + 0.014496276936404161x1 - 0.000006176621808466241x1^2 + 0.12513812861414014x2 - 0.000045885621883926725x1x2 - 0.0004157216592296709x2^2 + 0.1266290320622979x3 - 0.0014366289458181746x1x3 - 0.01944343400322525x2x3 - 0.34088885496069x3^2 + 3.1801123453702664x4 + 0.000744889105442118x1x4 - 0.004855967937663978x2x4 - 0.5082296220692951x3x4 - 0.4210065345862815x4^2 + 23.04638571573809x5 - 0.009850257650718444x1x5 - 0.07317520758202213x2x5 - 2.2979404074579386x3x5 + 1.871412946005615x4x5 - 15.708776136010252x5^2$
TON	$29.893416052350194 + 0.009998210925432327x1 + 9.376243055860865 \times 10^{-7}x1^2 - 1.618567835027597 \times 10^{-9}x1^3 + 0.08706345078592117x2 + 0.000012266608043995348x1x2 - 1.382244525610618 \times 10^{-8}x1^2x2 + 0.0009893619374389213x2^2 - 1.169847073579434 \times 10^{-7}x1x2^2 - 0.000001049713657160243x2^3 + 6.749036088653819x3 + 0.0016134269352232533x1x3 - 5.231257594690665 \times 10^{-7}x1^2x3 + 0.011705349758856239x2x3 - 0.000004647476975322477x1x2x3 - 0.00004890410134454004x2^2x3 + 1.0986152072047146x3^2 - 0.00004068640862146888x1x3^2 - 0.0013582576962779232x2x3^2 + 0.033642863393302753x3^3 + 1.9292715836575525x4 + 0.0007827780599396093x1x4 + 6.157699555843764 \times 10^{-8}x1^2x4 + 0.0027208841720602426x2x4 - 4.687682771145666 \times 10^{-7}x1x2x4 - 0.000016309169038364794x2^2x4 + 0.17863089390260406x3x4 - 0.000016366069153634542x1x3x4 - 0.0010496122326251554x2x3x4 - 0.05254895658354053x3^2x4 - 0.0493071567430364x4^2 - 0.0000392326769705405x1x4^2 - 0.0006867757799132519x2x4^2 - 0.04433749007359964x3x4^2 - 0.023347007155944934x4^3 + 15.904266395356187x5 + 0.0014855044925990678x1x5 - 0.000002579588672612522x1^2x5 + 0.01947007270661679x2x5 - 0.000022031814737786293x1x2x5 - 0.000186455414983069x2^2x5 + 2.564600271016065x3x5 - 0.0008350400272385637x1x3x5 - 0.007417199206766824x2x3x5 - 0.06604840572904243x3^2x5 + 1.2459646316622992x4x5 + 0.00009763750207265046x1x4x5 - 0.000746984439577417x2x4x5 - 0.026183754379352983x3x4x5 - 0.062378910659629486x4^2x5 + 2.3534753013457177x5^2 - 0.004111205141820419x1x5^2 - 0.0351167764440617x2x5^2 - 1.332922694384214x3x5^2 + 0.1548134057316448x4x5^2 - 6.552197319709215x5^3$
FOLN	$45.90174309425046 + 6.166940354629804\text{Log}[x1] + 6.75342737350235\text{Log}[x2] - 58.46700292964906\text{Log}[x3] - 9.162245880230312\text{Log}[x4] - 104.1950039454285\text{Log}[x5]$
SOLN	$13.521477710151082 + 1.8019277442891868\text{Log}[x1] + 0.23921348829604344\text{Log}[x1]^2 + 2.272941128627296\text{Log}[x2] + 0.30268754648928725\text{Log}[x1]\text{Log}[x2] + 0.37330579473965075\text{Log}[x2]^2 + 3.2634283187566187\text{Log}[x3] + 0.432393382640694\text{Log}[x1]\text{Log}[x3] + 0.2944387382572189\text{Log}[x2]\text{Log}[x3] - 5.581917444517096\text{Log}[x3]^2 + 0.36856740788861364\text{Log}[x4] + 0.07472954482071655\text{Log}[x1]\text{Log}[x4] - 0.10582033908667163\text{Log}[x2]\text{Log}[x4] - 5.31571810653157\text{Log}[x3]\text{Log}[x4] - 3.6916571182915106\text{Log}[x4]^2 - 50.98377870921349\text{Log}[x5] - 6.966087885114664\text{Log}[x1]\text{Log}[x5] - 8.0553545871115\text{Log}[x2]\text{Log}[x5] - 15.365463543423846\text{Log}[x3]\text{Log}[x5] + 37.66337831228693\text{Log}[x4]\text{Log}[x5] - 107.59435628174604\text{Log}[x5]^2$
FOTN	$10.64081282241565 + 5.034370478891869\text{Cos}[x1] + 5.257553787711318\text{Cos}[x2] - 10.820874858752916\text{Cos}[x3] - 2.6575799620321767\text{Cos}[x4] + 23.40478067680203\text{Cos}[x5] - 17.964586796717672\text{Sin}[x1] - 0.11024584068657649\text{Sin}[x2] + 23.65088730257444\text{Sin}[x3] + 5.42737821990518\text{Sin}[x4] + 11.4753228\text{Sin}[x5]$
SOTN	$2.1268778951890837 + 1.0817233708633547\text{Cos}[x1] + 3.6445800900623873\text{Cos}[x1]^2 + 0.3880832315407555\text{Cos}[x2] + 3.304927252\text{Cos}[x1]\text{Cos}[x2] + 5.2056\text{Cos}[x2]^2 - 2.261725936207216\text{Cos}[x3] - 1.325791\text{Cos}[x1]\text{Cos}[x3] - 0.2787001\text{Cos}[x2]\text{Cos}[x3] + 2.335767\text{Cos}[x3]^2 - 0.9418334606330854\text{Cos}[x4] - 3.33052\text{Cos}[x1]\text{Cos}[x4] - 2.220531592\text{Cos}[x2]\text{Cos}[x4] + 1.0549779\text{Cos}[x3]\text{Cos}[x4] + 2.3010\text{Cos}[x4]^2 + 4.175128\text{Cos}[x5] + 2.51310\text{Cos}[x1]\text{Cos}[x5] + 1.486520\text{Cos}[x2]\text{Cos}[x5] - 4.646382\text{Cos}[x3]\text{Cos}[x5] - 2.399166459\text{Cos}[x4]\text{Cos}[x5] + 6.50245\text{Cos}[x5]^2 - 2.1200571\text{Sin}[x1] - 2.8746\text{Cos}[x1]\text{Sin}[x1] - 1.4208\text{Cos}[x2]\text{Sin}[x1] + 1.982694\text{Cos}[x3]\text{Sin}[x1] + 1.0817273\text{Cos}[x4]\text{Sin}[x1] - 6.309403629\text{Cos}[x5]\text{Sin}[x1] + 2.2206202\text{Sin}[x1]^2 - 0.5198319\text{Sin}[x2] + 2.2207702\text{Cos}[x1]\text{Sin}[x2] + 4.4370\text{Cos}[x2]\text{Sin}[x2] + 0.64428\text{Cos}[x3]\text{Sin}[x2] - 1.334958\text{Cos}[x4]\text{Sin}[x2] - 0.402099036\text{Cos}[x5]\text{Sin}[x2] - 0.2044859\text{Sin}[x1]\text{Sin}[x2] + 3.1245758\text{Sin}[x2]^2 + 3.9929145\text{Sin}[x3] + 0.3146275\text{Cos}[x1]\text{Sin}[x3] + 3.3513020\text{Cos}[x2]\text{Sin}[x3] - 4.5746521\text{Cos}[x3]\text{Sin}[x3] + 0.7033\text{Cos}[x4]\text{Sin}[x3] + 6.52852\text{Cos}[x5]\text{Sin}[x3] - 6.86444229\text{Sin}[x1]\text{Sin}[x3] + 1.415727\text{Sin}[x2]\text{Sin}[x3] + 9.3460\text{Sin}[x3]^2 + 2.032636\text{Sin}[x4] + 8.4765371\text{Cos}[x1]\text{Sin}[x4] + 4.2485003\text{Cos}[x2]\text{Sin}[x4] - 2.4465862\text{Cos}[x3]\text{Sin}[x4] - 7.4238472\text{Cos}[x4]\text{Sin}[x4] + 4.7515586\text{Cos}[x5]\text{Sin}[x4] - 4.00293\text{Sin}[x1]\text{Sin}[x4] + 1.83231\text{Sin}[x2]\text{Sin}[x4] - 5.021096\text{Sin}[x3]\text{Sin}[x4] + 15.25544\text{Sin}[x4]^2 + 2.407062\text{Sin}[x5] + 1.02498\text{Cos}[x1]\text{Sin}[x5] + 0.18563\text{Cos}[x2]\text{Sin}[x5] - 2.50865\text{Cos}[x3]\text{Sin}[x5] - 0.88696\text{Cos}[x4]\text{Sin}[x5] + 5.3772\text{Cos}[x5]\text{Sin}[x5] - 2.033612\text{Sin}[x1]\text{Sin}[x5] - 0.760593\text{Sin}[x2]\text{Sin}[x5] + 4.974744\text{Sin}[x3]\text{Sin}[x5] + 1.88647\text{Sin}[x4]\text{Sin}[x5] + 2.5913\text{Sin}[x5]^2$
HYBRID	$16.59202757244946 + 0.03173224x2 - 0.0001105735x2^2 + 2.6612967x3 - 0.004938x2x3 - 0.0779593x3^2 + 0.50618x4 - 0.0030x2x4 - 0.2226835641837039x3x4 - 0.163622x4^2 + 6.661x5 - 0.0145x2x5 - 0.2780144954732378x3x5 + 0.051825641727912805x4x5 - 2.668534322994205x5^2 + 15.996592242917572\text{Cos}[x1] - 0.008617296752240735x2\text{Cos}[x1] + 2.774913789008801x3\text{Cos}[x1] - 2.18612688614873x4\text{Cos}[x1] + 7.627534289932837x5\text{Cos}[x1] + 60.149392252627514\text{Cos}[x1]^2$

# Electron Beam Welding (EBW) of Aerospace Alloy (Inconel 825): Optimization and Modeling of Weld Bead Area

Gamze ÖZAKINCI<sup>1</sup>, Levent AYDIN<sup>2,\*</sup>

*Department of Mechanical Engineering, İzmir Katip Çelebi University, Çiğli, İzmir, Turkey*

## Abstract

This study investigates the optimum weld area on a popular aerospace alloy (i.e., Inconel 825) made by the electron beam welding technique. Welding speed (S), beam current (I), accelerating voltage (V), and beam oscillation (O) are considered as process parameters to study the weld bead area (WA) of the weldments. An instructive study on multiple non-linear neural regression analyses has been done as a basic introduction to neuro regression modeling with artificial neural network (ANN) philosophy. To do this, the experimental prediction has been modeled with 14 predictive functional structures using fundamental regression modal types to test the accuracy of their predictions. To train the program with the chosen model  $R^2_{\text{training}}$ , test it  $R^2_{\text{testing}}$ , verify the accuracy  $R^2_{\text{validation}}$  is used, and check whether the values are within the engineering limits. Optimization algorithms with three different scenarios have been applied. Only one of the 14 models gave realistic results. It has been seen that the scenario types, selection of different constraints, and different models for design variables affect the optimization results.

**Keywords:** *Electron beam welding; neuro-regression modeling; optimization; weld bead area.*

## 1. Introduction

Nickel and its alloys (Inconel), being popular aerospace alloys, have gained popularity in the manufacturing industry because of their excellent properties but processing them has concerns, so researchers had to utilize new methods such as electron beam welding (EBW). In addition, the influence of welding processes and their parameters on weld characteristics must be investigated.

The Ni-base alloys are used in a wide range of applications in engineering systems exposed to extreme conditions, such as highly corrosive and high-temperature environments; for this reason, they are often used as an aerospace alloy and marine, automobile nuclear, chemical processing industries. Most of these applications require the use of welding processes during manufacture. The weld must perform to a level similar to the base metal [1, 2]. However, the processing or fabrication of this form of nickel-based alloys is not easy and has shown some concern. Formation of deleterious phases like laves phase, segregation of alloying element, and the tendency of micro-fissuring are considered the significant concerns affecting welded joints' mechanical characteristics. Hence, the implementation of statistical and soft computing-based approaches is essential [1]. Researchers utilized different welding methods such as friction stir welding (FSW), gas metal arc welding (GMAW), gas tungsten arc welding (GTAW), higher energy-intensive welding techniques like electron beam welding (EBW), and laser beam welding (LBW), to weld nickel/titanium alloys [3].

Mechanical properties and bead geometry of weld play great importance in controlling weld quality. Cross-section of weld-bead geometry defines the distortion and residual stresses induced while the shape of weld bead geometry rules mechanical properties of the weld. In welding, weld quality must remain the same as the base metal. This can be achieved by the appropriate selection and controlling welding variables. Therefore, the implementation of statistical approaches is essential [1, 5].

Researchers have utilized multiple linear regression (MLR) and soft computing-based intelligent approaches like artificial neural network (ANN), adaptive neuro-fuzzy inference system (ANFIS), and fuzzy logic (FL) to predict different responses in different welding processes. *Soft computing* is an approach that brings together the human mind with an environment of uncertainty and imprecision [4]. These approaches can describe the complex and non-linear behavior of the responses concerning its process parameter with success.

The literature on predicting various weld bead characteristics can be discussed as follows.

Palanivel et al. [6] worked on a backpropagation (BP) based ANN approach and RSM for predicting ultimate tensile strength in a titanium tube. They combined the developed model with genetic algorithms to optimize the GTAW process parameters to obtain the best weld bead geometry. Zaharudin et al. [7] developed ANFIS and ANN models to predict the welding strength of resistance spot welded CR780 specimens. Prediction using the ANN model is found accurate than the RSM approach [8]. Gyasi et al. [9] predicted the structural integrity of GMAW welded UHSS welded joints using the ANN approach. Satpathy et al. [10] used regression, ANN, and ANFIS to predicting the joining strength of the aluminum-copper dissimilar welded joints. Narang et al. [11]

\*Corresponding author e-mail address: [levent.aydin@ikcu.edu.tr](mailto:levent.aydin@ikcu.edu.tr)



established a fuzzy logic (FL) model to predict the weld bead geometry of TIG-welded structural steel weldments. Sivagurumanikandan et al. [12] applied RSM and ANN-based modeling methods to study the influence of welding variables on the optimal strength of weldments. Akbari et al. [13] developed a numerical model to estimate the temperature distribution and weld geometry of laser-welded titanium alloy weldments. Anand et al. [14] used different strategies for training the ANN model for predicting weld strength in FSW of Incoloy 800 weldments. Both models gave excellent results in the prediction of the welded specimens. ANN model trained with backpropagation (BP) algorithm was used to predict WA on stainless steel by Balasubramanian et al. [15].

After the literature studies in the relevant area, we have introduced a novel approach on a modeling design-optimization process to see the optimum weld area of Inconel 825, which is influenced by the welding speed (S), beam current (I), accelerating voltage (V) and beam oscillation (O):

- A detailed study on multiple non-linear neuro-regression analyses, including linear, quadratic, trigonometric, logarithmic, and their rational forms for the output of our process, has been performed.
- The boundedness of the candidate models has been checked to provide generating realistic values.
- The different direct search methods have been performed methodically, including stochastic optimization algorithms (modified versions of differential evaluation, Nelder-Mead, random search, and simulated annealing algorithms).

## 2. Materials and Methods

### 2.1. Modeling

In the modeling phase, the predictions are tested with the regression analysis. Regression models generally estimate the degree of correlation between input and output variables and determine their relationship form. Linear regression is mainly fitted by the least-squares method, but it can also be fitted by other methods, such as minimizing the "underfitting" in some other specifications or minimizing the penalty version of the least square loss function (such as Ridge regression). Linear regression is divided into two categories: simple and multiple linear regression [5].

With this method, the data was split into three parts; 84%, 10%, and 6% of the chosen data calculated for  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$ , and  $R^2_{\text{validation}}$ , respectively. This process reduces the error between predicted and the experimental results by changing the models in *Table 1* for regression and splitting the data into the correct sections. However, the resulting  $R^2$  values are not sufficient to confirm the accuracy of the results; therefore, apart from the results found from the candidates, maximum and minimum predicted values have been found and checked if the results are within the engineering limits. In addition, optimization algorithms with three different scenarios have been tried, and the optimum number of inputs investigated [16, 17].

### 2.2. Optimization

In essence, the optimization of the structure can be described as obtaining the best design by minimizing or maximizing a single specified goal or multiple goals corresponding to all constraints. Optimization techniques can be divided into traditional and non-traditional. Traditional optimization techniques are only suitable for continuous and differentiable functions, such as constraint changes and Lagrangian multipliers. In engineering design problems, traditional optimization techniques cannot be used due to particularities. In these cases, stochastic optimization methods such as genetic algorithm (GA), particle swarm (PS), and simulated annealing (SA) are advantageous. However, due to the nature of stochastic methods, accurate solutions cannot be obtained, and using more than one method with different phenomenological foundations for the same optimization problem will increase the solution's reliability.

The mathematical optimization problem handled in this article has the following issues:

- Multiple non-linear objective functions,
- Objective functions having many local extremum points,
- Mixed-integer (discrete) - continuous nature of the design variables
- Non-linear constraints.

The optimization scenarios discussed in this study include the condition given in the first three items. In addition, four different optimization algorithms (MDE, MNM, MSA, MRS) have been selected to solve optimization problems.

**Table 1:** Multiple regression model types including linear, quadratic, trigonometric, logarithmic, their rational forms and hybrid forms

Model Name	Nomenclature	Formula
Multiple linear	L	$a[1] + a[2] x_1 + a[3] x_2 + a[4] x_3 + a[5] x_4$
Multiple linear rational	LR	$(a[1] + x_1 a[2] + x_2 a[3] + x_3 a[4]) / (b[1] + x_1 b[2] + x_2 b[3] + x_3 b[4])$
Second order multiple nonlinear	SON	$a[1] + a[2] x_1 + a[3] x_2 + a[4] x_3 + a[5] x_4 + a[6] x_1^2 + a[7] x_2^2 + a[8] x_3^2 + a[9] x_4^2 + a[10] x_1 x_2 + a[11] x_1 x_3 + a[12] x_1 x_4 + a[13] x_2 x_3 + a[14] x_2 x_4 + a[15] x_3 x_4$
Second order multiple nonlinear rational	SONR	$(a[1] + 2 x_1 a[2] + x_1^2 a[3] + 2 x_2 a[4] + 2 x_1 x_2 a[5] + x_2^2 a[6] + 2 x_3 a[7] + 2 x_1 x_3 a[8] + 2 x_2 x_3 a[9] + x_3^2 a[10] + 2 x_4 a[11] + 2 x_1 x_4 a[12] + 2 x_2 x_4 a[13] + 2 x_3 x_4 a[14] + x_4^2 a[15]) / (b[1] + 2 x_1 b[2] + x_1^2 b[3] + 2 x_2 b[4] + 2 x_1 x_2 b[5] + x_2^2 b[6] + 2 x_3 b[7] + 2 x_1 x_3 b[8] + 2 x_2 x_3 b[9] + x_3^2 b[10] + 2 x_4 b[11] + 2 x_1 x_4 b[12] + 2 x_2 x_4 b[13] + 2 x_3 x_4 b[14] + x_4^2 b[15])$
Third order multiple nonlinear	TON	$a[1] + 3 x_1 a[2] + 3 x_1^2 a[3] + x_1^3 a[4] + 3 x_2 a[5] + 6 x_1 x_2 a[6] + 3 x_1^2 x_2 a[7] + 3 x_2^2 a[8] + 3 x_1 x_2^2 a[9] + x_2^3 a[10] + 3 x_3 a[11] + 6 x_1 x_3 a[12] + 3 x_1^2 x_3 a[13] + 6 x_2 x_3 a[14] + 6 x_1 x_2 x_3 a[15] + 3 x_2^2 x_3 a[16] + 3 x_3^2 a[17] + 3 x_1 x_3^2 a[18] + 3 x_2 x_3^2 a[19] + x_3^3 a[20] + 3 x_4 a[21] + 6 x_1 x_4 a[22] + 3 x_1^2 x_4 a[23] + 6 x_2 x_4 a[24] + 6 x_1 x_2 x_4 a[25] + 3 x_2^2 x_4 a[26] + 6 x_3 x_4 a[27] + 6 x_1 x_3 x_4 a[28] + 6 x_2 x_3 x_4 a[29] + 3 x_3^2 x_4 a[30] + 3 x_4^2 a[31] + 3 x_1 x_4^2 a[32] + 3 x_2 x_4^2 a[33] + 3 x_3 x_4^2 a[34] + x_4^3 a[35]$
First order trigonometric multiple nonlinear	FOTN	$a[1] + a[2] \sin[x_1] + a[3] \sin[x_2] + a[4] \sin[x_3] + a[5] \sin[x_4] + a[6] \cos[x_1] + a[7] \cos[x_2] + a[8] \cos[x_3] + a[9] \cos[x_4]$
First order trigonometric multiple nonlinear rational	FOTNR	$(a[1] + a[2] \sin[x_1] + a[3] \sin[x_2] + a[4] \sin[x_3] + a[5] \sin[x_4] + a[6] \cos[x_1] + a[7] \cos[x_2] + a[8] \cos[x_3] + a[9] \cos[x_4]) / (b[1] + b[2] \sin[x_1] + b[3] \sin[x_2] + b[4] \sin[x_3] + b[5] \sin[x_4] + b[6] \cos[x_1] + b[7] \cos[x_2] + b[8] \cos[x_3] + b[9] \cos[x_4])$
Second order trigonometric multiple nonlinear	SOTN	$a[1] + a[2] \sin[x_1] + a[3] \sin[x_2] + a[4] \sin[x_3] + a[5] \sin[x_4] + a[6] \cos[x_1] + a[7] \cos[x_2] + a[8] \cos[x_3] + a[9] \cos[x_4] + a[10] \sin[x_1]^2 + a[11] \sin[x_2]^2 + a[12] \sin[x_3]^2 + a[13] \sin[x_4]^2 + a[14] \cos[x_1]^2 + a[15] \cos[x_2]^2 + a[16] \cos[x_3]^2 + a[17] \cos[x_4]^2$
Second order trigonometric multiple nonlinear rational	SOTNR	$(a[1] + a[2] \sin[x_1] + a[3] \sin[x_2] + a[4] \sin[x_3] + a[5] \sin[x_4] + a[6] \cos[x_1] + a[7] \cos[x_2] + a[8] \cos[x_3] + a[9] \cos[x_4] + a[10] \sin[x_1]^2 + a[11] \sin[x_2]^2 + a[12] \sin[x_3]^2 + a[13] \sin[x_4]^2 + a[14] \cos[x_1]^2 + a[15] \cos[x_2]^2 + a[16] \cos[x_3]^2 + a[17] \cos[x_4]^2) / (b[1] + b[2] \sin[x_1] + b[3] \sin[x_2] + b[4] \sin[x_3] + b[5] \sin[x_4] + b[6] \cos[x_1] + b[7] \cos[x_2] + b[8] \cos[x_3] + b[9] \cos[x_4] + b[10] \sin[x_1]^2 + b[11] \sin[x_2]^2 + b[12] \sin[x_3]^2 + b[13] \sin[x_4]^2 + b[14] \cos[x_1]^2 + b[15] \cos[x_2]^2 + b[16] \cos[x_3]^2 + b[17] \cos[x_4]^2)$
First order logarithmic multiple nonlinear	FOLN	$a[1] + a[2] \log[x_1] + a[3] \log[x_2] + a[4] \log[x_3] + a[5] \log[x_4]$
First order logarithmic multiple nonlinear rational	FOLNR	$(a[1] + a[2] \log[x_1] + a[3] \log[x_2] + a[4] \log[x_3] + a[5] \log[x_4]) / (b[1] + b[2] \log[x_1] + b[3] \log[x_2] + b[4] \log[x_3] + b[5] \log[x_4])$
Second order logarithmic multiple nonlinear	SOLN	$a[1] + a[2] \log[x_1] + a[3] \log[x_2] + a[4] \log[x_3] + a[5] \log[x_4] + a[6] \log[x_1]^2 + a[7] \log[x_2]^2 + a[8] \log[x_3]^2 + a[9] \log[x_4]^2 + a[10] \log[x_1 x_2] + a[11] \log[x_1 x_3] + a[12] \log[x_1 x_4] + a[13] \log[x_2 x_3] + a[14] \log[x_2 x_4] + a[15] \log[x_3 x_4]$
Second order logarithmic multiple nonlinear rational	SOLNR	$(a[1] + a[2] \log[x_1] + a[3] \log[x_2] + a[4] \log[x_3] + a[5] \log[x_4] + a[6] \log[x_1]^2 + a[7] \log[x_2]^2 + a[8] \log[x_3]^2 + a[9] \log[x_4]^2 + a[10] \log[x_1 x_2] + a[11] \log[x_1 x_3] + a[12] \log[x_1 x_4] + a[13] \log[x_2 x_3] + a[14] \log[x_2 x_4] + a[15] \log[x_3 x_4]) / (b[1] + b[2] \log[x_1] + b[3] \log[x_2] + b[4] \log[x_3] + b[5] \log[x_4] + b[6] \log[x_1]^2 + b[7] \log[x_2]^2 + b[8] \log[x_3]^2 + b[9] \log[x_4]^2 + b[10] \log[x_1 x_2] + b[11] \log[x_1 x_3] + b[12] \log[x_1 x_4] + b[13] \log[x_2 x_3] + b[14] \log[x_2 x_4] + b[15] \log[x_3 x_4])$
Hybrid	H	$a[1] + 3 a[2] \sin[x_1] + 3 a[3] \sin[x_1]^2 + a[4] \sin[x_1]^3 + 3 a[5] \sin[x_2] + 6 a[6] \sin[x_1] \sin[x_2] + 3 a[7] \sin[x_1]^2 \sin[x_2] + 3 a[8] \sin[x_2]^2 + 3 a[9] \sin[x_1] \sin[x_2]^2 + a[10] \sin[x_2]^3 + 3 a[11] \sin[x_3] + 6 a[12] \sin[x_1] \sin[x_3] + 3 a[13] \sin[x_1]^2 \sin[x_3] + 6 a[14] \sin[x_2] \sin[x_3] + 6 a[15] \sin[x_1] \sin[x_2] \sin[x_3] + 3 a[16] \sin[x_2]^2 \sin[x_3] + 3 a[17] \sin[x_3]^2 + 3 a[18] \sin[x_1] \sin[x_3]^2 + 3 a[19] \sin[x_2] \sin[x_3]^2 + a[20] \sin[x_3]^3 + 3 a[21] \sin[x_4] + 6 a[22] \sin[x_1] \sin[x_4] + 3 a[23] \sin[x_1]^2 \sin[x_4] + 6 a[24] \sin[x_2] \sin[x_4] + 6 a[25] \sin[x_1] \sin[x_2] \sin[x_4] + 3 a[26] \sin[x_2]^2 \sin[x_4] + 6 a[27] \sin[x_3] \sin[x_4] + 6 a[28] \sin[x_1] \sin[x_3] \sin[x_4] + 6 a[29] \sin[x_2] \sin[x_3] \sin[x_4] + 3 a[30] \sin[x_3]^2 \sin[x_4] + 3 a[31] \sin[x_4]^2 + 3 a[32] \sin[x_1] \sin[x_4]^2 + 3 a[33] \sin[x_2] \sin[x_4]^2 + 3 a[34] \sin[x_3] \sin[x_4]^2 + a[35] \sin[x_4]^3 + \sin[x_1] a[36] + \sin[x_2] a[37] + \sin[x_3] a[38] + \sin[x_4] a[39] + \sin[x_1]^2 a[40] + \sin[x_2]^2 a[41] + \sin[x_3]^2 a[42] + \sin[x_4]^2 a[43] + \sin[x_1]^4 a[44] + \sin[x_2]^4 a[45] + \sin[x_3]^4 a[46] + \sin[x_4]^4 a[47]$

**2.2.1 Modified Nelder-Mead Algorithm**

One of the search methods is the Nelder–Mead (NM) optimization algorithm. Therefore, it does not require any derivative information and starts with simplex to minimize the function. As a result, the iteration maintained up to the simplex becomes flat. It means that the resulting value of the function is almost the same at all the

vertices. The iteration steps of the Nelder-Mead algorithm are ordering, centroid, and transformation. In the present version of the algorithm, a penalty function is added to the flow to solve the prescribed constrained minimization problem. The construction of the initial working simplex  $S$  is the first step. Second, minimizing the function moves the search course away from the peak, which is the worst function value [16].

### 2.2.2 Modified Differential Evolution Algorithm

Differential evolution (DE) is one of the suitable ways of the stochastic optimization method. It can be used in complex structured engineering design problems to find the optimum result. The productive parameters of the DE algorithm are population size, crossover, and scaling factor. It handles a population of solutions instead of iterating over solutions. The DE algorithm is proposed to be robust and efficient in the literature; it does not satisfy the global optimum points for all optimization problems [17].

### 2.2.3 Modified Simulated Annealing Algorithm

Another search method based on the physical annealing process of metal is simulated annealing (SA). The material moves to a lower energy state throughout the melting process and becomes tougher. Because of the intrinsic structure of the algorithm, it is better at finding the global optimum. Moreover, it can solve continuous, mixed-integer, or discrete optimization problems [17].

### 2.2.4 Modified Random Search Algorithm

The first step in the traditional random search algorithm is to produce a population of random starting points. Then, it uses a local optimization method from each starting point to get closer to a local extremum point at this stage. In the proposed algorithm version, there are some booster subroutines such as the conjugate gradient, principal axis, Levenberg Marquardt, Newton, QuasiNewton, and non-linear interior-point method the localization of the values of all variables for the objective function. In this step, the fitness function is evaluated with the variables being symbolic, and then the process continues again and again. It is also possible to solve mixed-integer-continuous global optimization problems [18].

## 2.3. Problem Description

1. The data given in Table 2 were selected from the reference study [1]. It should be noted that the parameters  $x_1$ - $x_4$  included in the models given in Table 1 correspond in terms of engineering to accelerating voltage, beam current, welding speed, and beam oscillation parameters, respectively.

2. 14 candidate functional constructs have been suggested to model the experimental data have been tested for the proper ones in terms of  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$ , and  $R^2_{\text{validation}}$  values, and then boundedness of the functions is also checked.

3. Three different optimization scenarios have been introduced using the appropriate model obtained, and these problems are solved through four different direct search methods: MDE, MSA, MRS, and MNM

## 2.4. Optimization Scenarios

### *Scenario 1*

In this optimization problem, the objective functions define the weld bead area ( $\text{mm}^2$ ) of a welded Inconel 825 alloy. All the design variables are considered to be real numbers, and the search space is continuous. In this case, the limit values for the system inputs are  $48 < \text{accelerating voltage (kV)} < 60$ ,  $38 < \text{beam current (mA)} < 4$ ,  $900 < \text{welding speed (mm/min)} < 1200$  and  $200 < \text{beam oscillation (Hz)} < 600$ . The main objective is to maximize the weld bead area. In this way, the theoretical limits of the objective function can also be seen mathematically.

Intervals are considered to be real numbers:  $48 < \text{accelerating voltage (kV)} < 60$ ,  $38 < \text{beam current (mA)} < 4$ ,  $900 < \text{welding speed (mm/min)} < 1200$  and  $200 < \text{beam oscillation (Hz)} < 600$ . Additionally, to examine the constraints of design variables {accelerating voltage, beam current, welding speed, beam oscillation}  $\in$  integers are proper.

### *Scenario 2*

Besides knowledge-based in Scenario 1, more applicable problem cases for the weld bead area need to be added. For this purpose, a new optimization problem has been described that assumes the maximization of the

weld bead area. All the design variables in the intervals are considered to be real numbers:  $48 < \text{accelerating voltage (kV)} < 60$ ,  $38 < \text{beam current (mA)} < 46$ ,  $900 < \text{welding speed (mm/min)} < 1200$  and  $200 < \text{beam oscillation (Hz)} < 600$ . Additionally, to examine the constraints of design variables {accelerating voltage, beam current, welding speed, beam oscillation}  $\in$  integers are proper.

*Scenario 3*

Based on only the prescribed experimental setup, the more specific optimization problem can also be defined as involving (I) maximization of the beam oscillation, (II) minimization of accelerating voltage, (III) optimizing the welding speed, (IV) minimizing the beam current, (V) all the design variables are assumed to be real numbers and (VI) the constraints are accelerating voltage  $\in$  {48, 54, 60}; beam current  $\in$  {38, 42, 46}; welding speed  $\in$  {900, 1050, 1200}; and beam oscillation  $\in$  {200, 400, 600}.

**Table 2:** *The data used for optimization operations in the article [1]*

SI. No.	Accelerating voltage(kV), V	Beam Current (mA), I	Welding speed (mm/min), S	Beam Oscillation (Hz), O	Weld bead area (mm <sup>2</sup> ), WA
1	48	38	900	200	4.86
2	60	46	1200	400	5.99
3	54	42	1050	600	5.39
4	48	38	1200	600	3.93
5	48	38	1200	200	3.24
6	60	38	1200	600	5.06
7	60	38	1200	200	5.12
8	48	38	900	600	4.81
9	48	46	900	600	6.06
10	48	46	1200	600	4.92
11	54	42	1050	400	5.33
12	60	38	900	600	5.69
13	60	46	900	600	6.07
14	48	46	900	200	6.15
15	54	42	1050	400	5.54
16	54	42	1050	400	5.38
17	48	46	1200	200	4.52
18	60	38	900	200	5.89
19	60	46	1200	200	5.42
20	60	46	900	200	5.53
21	54	46	1050	400	6.02
22	54	42	1200	400	4.93
23	54	42	1050	400	5.38
24	54	42	1050	600	5.32
25	54	38	1050	400	5.33
26	54	42	900	400	5.64
27	54	42	1050	200	5.56
28	48	42	1050	400	4.89
29	54	42	1050	400	5.63
30	60	42	1050	400	5.67

### 3.Results and Discussion

In this study, 14 different regression models with four parameters were tested for one output (see *Table 1*), and the results are listed in *Tables 3* to understand how the model successfully explained the process to estimate the  $R^2_{\text{training}}$ ,  $R^2_{\text{testing}}$  and  $R^2_{\text{validation}}$  values of various regression models and determine the functional limitation (boundedness) of the model by estimating the maximum and minimum values generated by the corresponding model.

In order to understand that we have trained the program correctly, we should pay attention that the training value is  $<0.90$ , the testing value is  $<0.90$ , and the validation value is  $<0.85$ . After completing these processes, we check whether the maximum and minimum values we find for the inputs are within the engineering limits, the model that meets all the conditions can be used.

In *Table 3*, the suitability of the candidate models in terms of training, testing, validation coefficients, and boundedness, the following inferences were made:

- Training coefficients of all models are pretty high ( $>0.99$ ) while the test coefficients are high for only LR, SON, FOTNR, SOTNR, FOLNR, and H. Therefore, the number of usable models in the testing phase decreases from 14 to 6.
- In the next stage, the compatibility of the validation value is examined. At this stage, we only have 1 model that meets this requirement, and that is SON.
- Besides, as mentioned in the previous section, it is also expected to meet the boundedness criterion for use in optimizing the model. When viewed from this angle, only one model is suitable, which is SON. As a result, only the SON model can meet all the desired criteria and be considered a realistic model.

**Table 3:** Results of the Neuro-regression models for fitting performance and boundedness.

Models	$R^2_{\text{Training}}$	$R^2_{\text{Testing}}$	$R^2_{\text{Validation}}$	Max (mm <sup>2</sup> )	Min (mm <sup>2</sup> )
L	0.99	0.74	0.80	6.50	4.19
LR	0.99	0.97	0.43	6.13	3.56
SON	0.99	0.97	0.87	6.20	3.56
SONR	0.99	0.50	0.87	6.12	4.70
TON	0.99	0.87	0.32	6.76	2.71
FOTN	0.99	0.79	0.89	6.50	-5.18
FOTNR	0.99	0.98	-22.2	$4.48496 * 10^7$	$-2.01089 * 10^{13}$
SOTN	0.99	0.79	0.89	6.45	-3.44
SOTNR	0.99	0.97	0.85	13.40	$-5.0818 * 10^{11}$
FOLN	0.99	0.75	0.81	6.50	4.17
FOLNR	0.99	0.96	0.32	6.12	3.58
SOLN	0.99	0.79	0.89	6.46	4.09
SOLNR	0.99	0.86	0.42	$5.58908 * 10^{10}$	3.87
H	0.99	0.95	0.93	11.63	-8.80

In *Table 4*, the model of SON is taken as the objective function, and the results are listed for three different optimization scenarios. This table uses MDE, MSA, MRS, and MNM algorithms for each scenario, and the results were compared. According to all algorithms for the first scenario, the maximum input values were calculated, and that gave us four different alternative input parameter triplets to achieve the weld beam area. The problem definition is the same in the second scenario, but the input parameters are forced to be integers. In this case, the input values have changed but still were close to numbers from scenario 1. In terms of reliability, achieving the very close results for the four direct search methods used in scenarios 1 and 2 increases the possibility that obtained values will be the global optimum. When we look at the results of the third scenario, we can say that we have reached the optimum parameter values similar to scenario 2. However, the best solution proposal of each algorithm is quite different from each other. This shows that it is essential to use more than one different phenomenological-based algorithm when there is a need to produce alternative optimum solutions for this problem and similar studies.

**Table 4:** Optimization problem results

Objective function	Scenario no	Constrains	Optimization Algorithm	Suggested Design (max)
$-2.20064 + 0.728057 x_1 - 0.00512518 x_1^2 - 0.663085 x_2 - 0.00883957 x_1 x_2 + 0.0131558 x_2^2 + 0.0019017 x_3 + 0.000244055 x_1 x_3 + 0.0000806661 x_2 x_3 - 0.0000105548 x_3^2 - 0.0060513 x_4 + 3.41694 \cdot 10^{-6} x_1 x_4 + 0.0000941879 x_2 x_4 + 2.57166 \cdot 10^{-6} x_3 x_4 - 6.12662 \cdot 10^{-7} x_4^2$	1	$48 < x_1 < 60;$ $38 < x_2 < 46;$ $900 < x_3 < 1200;$ $200 < x_4 < 600;$	MDE	$x_1 = 54.6809$ $x_2 = 46$ $x_3 = 971.143$ $x_4 = 600$
			MSA	$x_1 = 54.6776$ $x_2 = 46$ $x_3 = 971.116$ $x_4 = 600$
			MRS	$x_1 = 54.6809$ $x_2 = 46$ $x_3 = 971.142$ $x_4 = 600$
			MNM	$x_1 = 54.6809$ $x_2 = 46$ $x_3 = 971.142$ $x_4 = 600$
	2	$48 < x_1 < 60;$ $38 < x_2 < 46;$ $900 < x_3 < 1200;$ $200 < x_4 < 600;$ $\{x_1, x_2, x_3, x_4\} \in \text{Integers}$	MDE	$x_1 = 56$ $x_2 = 45$ $x_3 = 982$ $x_4 = 599$
			MSA	$x_1 = 55$ $x_2 = 45$ $x_3 = 937$ $x_4 = 503$
			MRS	$x_1 = 52$ $x_2 = 45$ $x_3 = 1015$ $x_4 = 599$
			MNM	$x_1 = 56$ $x_2 = 45$ $x_3 = 982$ $x_4 = 599$
	3	$x_1 \in \{48, 54, 60\},$ $x_2 \in \{38, 42, 46\},$ $x_3 \in \{900, 1050, 1200\},$ $x_4 \in \{200, 400, 600\}$	MDE	$x_1 = 54$ $x_2 = 46$ $x_3 = 1006$ $x_4 = 550$
			MSA	$x_1 = 55$ $x_2 = 45$ $x_3 = 1010$ $x_4 = 555$
			MRS	$x_1 = 55$ $x_2 = 49$ $x_3 = 1006$ $x_4 = 550$
			MNM	$x_1 = 53$ $x_2 = 45$ $x_3 = 1001$ $x_4 = 550$

**4. Conclusion**

In this research, electron beam welding of Inconel 825 is performed to investigate the influence of welding process variables WA. Thirty welding experiment results have been taken from Ref [1] on four weld factors. First, predictive modeling for WA is developed using non-linear multiple regression analysis employing the philosophy of the popular method ANN, and the effectiveness is investigated. Then, the obtained results were checked whether the selected models are also bounded or not in the engineering parameter intervals. Finally, modified versions of four direct search methods (Differential Evolution, Simulated Annealing, Random Search, and Nelder-Mead) were used during the optimization process. From this investigation, the followings are some of the critical conclusions obtained.

The training values selected to train the program should be as many as possible so that the testing and validation values are as accurate as possible. Nevertheless, it is not enough to just have a large amount of data in the training part; the input values of the selected data must also include the maximum and minimum input values because the program has difficulty estimating a value, not in the range. Another factor affecting the consistency of the estimated values is that the outputs are very variable; the model may give closer results with outputs close to each other. During model selection, estimates are used because there is no information about which model will give better results, different results are obtained by trial and error, and which model gives more accurate results is continued. If we had bad results from the direct search methods, we could use the particular options that apply to Nelder-Mead: ReflectRatio, Tolerance, ShrinkageRatio, Randomseed, and more not needed.

**Declaration of Interest**

The authors declare that there is no conflict of interest.

**References**

- [1] B. Choudhury, and M. Chandrasekaran, "Electron beam welding of aerospace alloy (inconel 825): a comparative study of RSM and ANN modeling to predict weld bead area," *Optik - International Journal for Light and Electron Optics*, Elsevier GmbH, vol. 219, 2020
- [2] J. W. Sowards, and J. Caron, "Weldability of nickel-base alloys," *Comprehensive Materials Processing*, Elsevier, Oxford, vol. - 1, no. 6, pp. 151-179, 2014.
- [3] B. Choudhury, and M. Chandrasekaran, "Investigation on welding characteristics of aerospace materials –a review," *Selection and Peer-review under responsibility of the Committee Members of International Conference on Advancements in Aeromechanical Materials for Manufacturing*, Elsevier, vol. 4, pp. 7519–7526, 2017.

- [4] M. Chandrasekaran, M. Muralidhar, C. M. Krishna, and U. S. Dixit, "Application of soft computing techniques in machining performance prediction and optimization: a literature review," *Int J Adv Manuf Technol*, Springer-Verlag London Limited, vol. 46, pp. 445–464, 2009.
- [5] F. Khademi, S. M. Jamal, N. Deshpande, and S. Londhe, "Predicting strength of recycled aggregate concrete using artificial neural network, adaptive neuro fuzzy inference system and multiple linear regression," *International Journal of Sustainable Built Environment*, Elsevier BV, vol. 5, no. 2, pp. 355–369, 2016.
- [6] R. Palanivel, I. Dinaharan, and R. F. Laubscher, "Application of an artificial neural network model to predict the ultimate tensile strength of friction welded titanium tubes," *J. Braz. Soc. Mech. Sci. Eng.*, Springer, vol. 41, no. 111, 2019.
- [7] M. F. A. Zaharuddin, D. Kim, and S. Rhee, "An ANFIS based approach for predicting the weld strength of resistance spot welding in artificial intelligence development," *J. Mech. Sci. Technol.*, Springer, vol. 31, no. 11, pp. 5467–5476, 2017.
- [8] K. Anand, R. Shrivastava, K. Tamilmannan, and P. Sathiya, "A comparative study of artificial neural network and response surface methodology for optimization of friction welding of incoloy 800 H." *Acta Metall. Sin. (Engl. Lett.)*, Springer, vol. 28, no. 7, pp. 892–902, 2015.
- [9] E. A. Gyasi, P. Kah, H. Wu, and M. A. Kesse, "Modeling of an artificial intelligence system to predict structural integrity in robotic GMAW of UHSS fillet welded joints," *Int. J. Adv. Manuf. Technol.*, Springer, vol. 93, pp. 1139–1155, 2017.
- [10] M. P. Satpathya, S. B. Mishraa, and S. K. Sahoob, "Ultrasonic spot welding of aluminum-copper dissimilar metals: a study on joint strength by experimentation and machine learning techniques," *J. Manuf. Process.*, Elsevier, vol. 33, pp. 96–110, 2018.
- [11] H. K. Narang, U. P. Singh, M. M. Mahapatra, and P. K. Jha, "Prediction of the weld pool geometry of TIG arc welding by using fuzzy logic controller," *Int. J. Eng. Sci. Technol.*, MultiCraft, vol. 3, no. 9, pp. 77–85, 2011.
- [12] N. Sivagurumanikandan, S. Saravanan, G. S. Kumar, S. Raju, and K. Raghukandan, "Prediction and optimization of process parameters to enhance the tensile strength of Nd: YAG laser welded super duplex stainless steel," *Optik*, Elsevier GmbH., vol. 157, pp. 833–840, 2018.
- [13] M. Akbari, S. Saedodin, A. Panjehpour, M. Hassani, M. Afrand, and M. J. Torkamany, "Numerical simulation and designing artificial neural network for estimating melt pool geometry and temperature distribution in laser welding of Ti6Al4V alloy," *Opt. – Int. J. Light Electron. Opt.*, 2016.
- [14] K. Anand, B. K. Barik, K. Tamilmannan, and P. Sathiya, "Artificial neural network modeling studies to predict the friction welding process parameters of incoloy 800H joints," *Eng. Sci. Technol. Int. J.*, Elsevier BV, vol. 18, pp. 394–407, 2015.
- [15] K. R. Balasubramanian, G. Buvanashakaran, and K. Sankaranarayanan, "Modeling of laser beam welding of stainless steel sheet butt joint using neural networks," *CIRP J. Manuf. Sci. Technol.*, Elsevier, vol. 3, pp. 80–84, 2010.
- [16] I. Polatoglu, and L. Aydin, "A new design strategy with stochastic optimization on the preparation of magnetite cross-linked tyrosinase aggregates (MCLTA)," *Process Biochemistry*, vol. 99, pp. 131–138, 2020.
- [17] I. Polatoglu, L. Aydin, B. Ç. Nevruz, and S. Ozer, "A novel approach for the optimal design of a biosensor," *Anal. Lett.*, pp. 1–18, 2020. doi.org/10.1080/00032719.2019.1709075.
- [18] A. B. Ceylan, L. Aydin, and M. Nil, H. Mamur, İ. Polatoğlu and H. Sözen. "A new hybrid approach in selection of optimum establishment location of the biogas energy production plant," *Biomass Conv. Bioref.*, 2021. <https://doi.org/10.1007/s13399-021-01532-8>



APPENDICES

Nomenclature	Models
<b>L</b>	$1.05302 + 0.0606329 x_1 + 0.0866439 x_2 - 0.00260376 x_3 + 0.00029121 x_4$
<b>LR</b>	$(-1.42443 \cdot 10^6 + 42920.6 x_1 - 4984.72 x_2 - 302.626 x_3 + 68.1846 x_4) / (-212520. + 7358.63 x_1 - 1833.02 x_2 - 38.4445 x_3 + 11.5423 x_4)$
<b>SON</b>	$-2.20064 + 0.728057 x_1 - 0.00512518 x_1^2 - 0.663085 x_2 - 0.00883957 x_1 x_2 + 0.0131558 x_2^2 + 0.0019017 x_3 + 0.000244055 x_1 x_3 + 0.0000806661 x_2 x_3 - 0.0000105548 x_3^2 - 0.0060513 x_4 + 3.41694 \cdot 10^{-6} x_1 x_4 + 0.0000941879 x_2 x_4 + 2.57166 \cdot 10^{-6} x_3 x_4 - 6.12662 \cdot 10^{-7} x_4^2$
<b>SONR</b>	$(0.999275 + 1.92477 x_1 + 1.0596 x_1^2 + 1.9466 x_2 + 2.29488 x_1 x_2 + 1.17203 x_2^2 + 0.536888 x_3 + 24.0295 x_1 x_3 + 14.7801 x_2 x_3 + 1.31359 x_3^2 + 1.32937 x_4 + 10.1391 x_1 x_4 + 6.95023 x_2 x_4 - 5.89455 x_3 x_4 + 0.545452 x_4^2) / (1.00445 + 2.57484 x_1 + 3.65083 x_1^2 + 2.39654 x_2 + 8.45453 x_1 x_2 + 1.38953 x_2^2 + 11.0051 x_3 + 1.26633 x_1 x_3 - 0.25641 x_2 x_3 + 0.523906 x_3^2 + 6.0619 x_4 + 4.87717 x_1 x_4 + 3.59808 x_2 x_4 - 1.51806 x_3 x_4 + 0.289919 x_4^2)$
<b>TON</b>	$-11.8606 + 0.224046 x_1 + 0.00488066 x_1^2 - 0.0000194012 x_1^3 - 0.138815 x_2 - 0.00203656 x_1 x_2 - 0.000152265 x_1^2 x_2 + 0.00150761 x_2^2 - 0.000147576 x_1 x_2^2 + 0.00017423 x_2^3 + 0.00711543 x_3 + 0.000204546 x_1 x_3 + 4.53442 \cdot 10^{-7} x_1^2 x_3 + 0.0000324893 x_2 x_3 + 7.21174 \cdot 10^{-6} x_1 x_2 x_3 + 1.55683 \cdot 10^{-6} x_2^2 x_3 + 4.53854 \cdot 10^{-6} x_3^2 - 7.37788 \cdot 10^{-9} x_1 x_3^2 - 8.21767 \cdot 10^{-8} x_2 x_3^2 - 1.36093 \cdot 10^{-8} x_3^3 + 0.0670044 x_4 - 0.0010011 x_1 x_4 - 1.36903 \cdot 10^{-7} x_1^2 x_4 - 0.00130697 x_2 x_4 + 0.0000359974 x_1 x_2 x_4 - 4.92474 \cdot 10^{-6} x_2^2 x_4 - 0.0000768777 x_3 x_4 - 7.3771 \cdot 10^{-7} x_1 x_3 x_4 - 7.41981 \cdot 10^{-7} x_2 x_3 x_4 + 7.01739 \cdot 10^{-8} x_3^2 x_4 + 0.000070046 x_4^2 + 3.50838 \cdot 10^{-7} x_1 x_4^2 + 8.10061 \cdot 10^{-7} x_2 x_4^2 + 3.92122 \cdot 10^{-9} x_3 x_4^2 - 1.05786 \cdot 10^{-7} x_4^3$
<b>FOTN</b>	$0.665315 - 4.20932 \cos[x_1] - 0.366313 \cos[x_2] + 0.172438 \cos[x_3] - 0.0653933 \cos[x_4] - 1.26012 \sin[x_1] + 0.310245 \sin[x_2] + 0.882573 \sin[x_3] + 0.0241337 \sin[x_4]$
<b>FOTNR</b>	$(-950.462 - 772.317 \cos[x_1] - 9.46081 \cos[x_2] - 45.0644 \cos[x_3] + 12.786 \cos[x_4] - 661.373 \sin[x_1] - 20.6108 \sin[x_2] - 44.7748 \sin[x_3] + 16.1305 \sin[x_4]) / (-171.53 - 138.883 \cos[x_1] - 2.17954 \cos[x_2] - 7.43295 \cos[x_3] + 2.30761 \cos[x_4] - 117.893 \sin[x_1] - 3.41943 \sin[x_2] - 6.79714 \sin[x_3] + 2.97475 \sin[x_4])$
<b>SOTN</b>	$0.407069 - 1.18265 \cos[x_1] + 1.28031 \cos[x_1]^2 - 0.241127 \cos[x_2] + 0.393791 \cos[x_2]^2 + 0.413412 \cos[x_3] + 0.329156 \cos[x_3]^2 - 0.0575362 \cos[x_4] + 0.745032 \cos[x_4]^2 - 2.15281 \sin[x_1] - 1.4521 \sin[x_1]^2 + 0.315919 \sin[x_2] + 0.628466 \sin[x_2]^2 + 0.750764 \sin[x_3] + 0.700909 \sin[x_3]^2 - 0.280635 \sin[x_4] + 0.362098 \sin[x_4]^2$
<b>SOTNR</b>	$(2.31883 - 6.25099 \cos[x_1] + 11.8668 \cos[x_1]^2 + 0.711155 \cos[x_2] + 1.33971 \cos[x_2]^2 + 1.39785 \cos[x_3] + 1.1556 \cos[x_3]^2 + 0.901391 \cos[x_4] + 2.47489 \cos[x_4]^2 + 9.87237 \sin[x_1] - 8.54795 \sin[x_1]^2 - 0.15083 \sin[x_2] + 1.97912 \sin[x_2]^2 + 2.46102 \sin[x_3] + 2.16323 \sin[x_3]^2 + 1.28901 \sin[x_4] + 0.843944 \sin[x_4]^2) / (0.175012 - 0.382517 \cos[x_1] + 2.69676 \cos[x_1]^2 + 0.452111 \cos[x_2] + 0.354456 \cos[x_2]^2 + 0.200628 \cos[x_3] + 0.498388 \cos[x_3]^2 + 0.1765 \cos[x_4] + 0.260945 \cos[x_4]^2 + 1.38414 \sin[x_1] - 1.52174 \sin[x_1]^2 - 0.115724 \sin[x_2] + 0.820556 \sin[x_2]^2 + 0.219855 \sin[x_3] + 0.676624 \sin[x_3]^2 + 1.02702 \sin[x_4] + 0.914067 \sin[x_4]^2)$
<b>FOLN</b>	$-3.5355 + 3.29747 \log[x_1] + 3.64491 \log[x_2] - 2.67695 \log[x_3] + 0.12492 \log[x_4]$
<b>FOLNR</b>	$(-5061.54 + 2010.94 \log[x_1] - 200.597 \log[x_2] - 283.919 \log[x_3] + 20.2834 \log[x_4]) / (-810.22 + 344.877 \log[x_1] - 71.2596 \log[x_2] - 36.5198 \log[x_3] + 3.36826 \log[x_4])$
<b>SOLN</b>	$-378.91 + 67.2296 \log[x_1] - 11.8256 \log[x_1]^2 - 164.035 \log[x_2] + 26.8554 \log[x_2]^2 - 20.4811 \log[x_1 x_2] + 81.1211 \log[x_3] - 11.69 \log[x_3]^2 + 41.6929 \log[x_1 x_3] + 14.2396 \log[x_2 x_3] - 5.26173 \log[x_4] + 0.0303843 \log[x_4]^2 + 9.02977 \log[x_1 x_4] - 26.5893 \log[x_2 x_4] + 22.5746 \log[x_3 x_4]$
<b>SOLNR</b>	$(0.941802 + 6.08834 \log[x_1] + 40.7971 \log[x_1]^2 - 1.65513 \log[x_2] - 16.8924 \log[x_2]^2 + 3.43321 \log[x_1 x_2] + 0.178359 \log[x_3] - 9.24925 \log[x_3]^2 + 5.2667 \log[x_1 x_3] - 2.47677 \log[x_2 x_3] + 0.82184 \log[x_4] - 0.568154 \log[x_4]^2 + 5.91018 \log[x_1 x_4] - 1.83329 \log[x_2 x_4] + 0.00019938 \log[x_3 x_4]) / (1.13334 + 0.939816 \log[x_1] + 7.07759 \log[x_1]^2 + 1.28176 \log[x_2] - 4.89594 \log[x_2]^2 + 1.22157 \log[x_1 x_2] + 0.616195 \log[x_3] - 1.6354 \log[x_3]^2 + 0.55601 \log[x_1 x_3] + 0.897953 \log[x_2 x_3] + 0.457665 \log[x_4] - 0.174584 \log[x_4]^2 + 0.397481 \log[x_1 x_4] + 0.739424 \log[x_2 x_4] + 0.0738598 \log[x_3 x_4])$

<b>H</b>	$  \begin{aligned}  & 3.72692 - 2.83153 \sin[x_1] - 1.94904 \sin[x_1]^2 + 1.82501 \sin[x_1]^3 - 1.10419 \sin[x_1]^4 + 2.11394 \sin[x_2] - 0.249112 \sin[x_1] \sin[x_2] - 2.77911 \sin[x_1]^2 \sin[x_2] + \\  & 0.266018 \sin[x_2]^2 - 2.79166 \sin[x_1] \sin[x_2]^2 - 0.976365 \sin[x_2]^3 - 0.649314 \sin[x_2]^4 + 0.258553 \sin[x_3] + 0.317103 \sin[x_1] \sin[x_3] + 0.772664 \sin[x_1]^2 \sin[x_3] - \\  & 2.01385 \sin[x_2] \sin[x_3] - 3.47612 \sin[x_1] \sin[x_2] \sin[x_3] - 0.152631 \sin[x_2]^2 \sin[x_3] + \\  & 0.973056 \sin[x_3]^2 + 1.12739 \sin[x_1] \sin[x_3]^2 - 0.0749866 \sin[x_2] \sin[x_3]^2 + 0.143497 \sin[x_3]^3 - 0.0653614 \sin[x_3]^4 + 0.995053 \sin[x_4] - 1.34491 \sin[x_1] \sin[x_4] - \\  & 0.383878 \sin[x_1]^2 \sin[x_4] + 0.575175 \sin[x_2] \sin[x_4] + 2.06164 \sin[x_1] \sin[x_2] \sin[x_4] + 0.220295 \sin[x_2]^2 \sin[x_4] - 0.0890705 \sin[x_3] \sin[x_4] + 0.456294 \sin[x_1] \\  & \sin[x_3] \sin[x_4] - 0.230877 \sin[x_2] \sin[x_3] \sin[x_4] + 0.759084 \sin[x_3]^2 \sin[x_4] + 0.539582 \sin[x_4]^2 + 0.586279 \sin[x_1] \sin[x_4]^2 - 1.08923 \sin[x_2] \sin[x_4]^2 + 0.644453 \\  & \sin[x_3] \sin[x_4]^2 - 0.772504 \sin[x_4]^3 + 1.45937 \sin[x_4]^4  \end{aligned}  $
----------	--

---