

e-ISSN: 2980-177X

JCEEES

Volume: 4 Issue: 1 Year: 2026

Journal of
**Computer &
Electrical and
Electronics
Engineering
Sciences**



Journal of
**Computer & Electrical and
Electronics Engineering Sciences**

EDITORIAL BOARD

EDITOR-IN-CHIEF

Asst. Prof. Fuat TÜRK

Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Türkiye

ASSOCIATE EDITORS-IN-CHIEF

Asst. Prof. Hüseyin AYDİLEK

Department of Electrical and Electronics Engineering, Faculty of Engineering and Architecture, Kırıkkale University, Kırıkkale, Türkiye

Assoc. Prof. Mahmut KILIÇASLAN

Department of Computer Technologies, Vocational School of Nallıhan, Ankara University, Ankara, Türkiye

Assoc. Prof. Selim BUYRUKOĞLU

Department of Computer Engineering, Faculty of Engineering, Çankırı Karatekin University, Çankırı, Türkiye

EDITORIAL BOARD

Prof. Aydın ÇETİN

Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Türkiye

Asst. Prof. Ayhan AKBAŞ

Department of Computer Engineering, Faculty of Engineering, Abdullah Gül University, Kayseri, Türkiye

Asst. Prof. Ebru AYDOĞAN DUMAN

Department of Computer Information Systems Engineering, Faculty of Bucak Technology, Burdur Mehmet Akif University, Burdur, Türkiye

Asst. Prof. Elvan DUMAN

Department of Software Engineering, Faculty of Bucak Technology, Burdur Mehmet Akif University, Burdur, Türkiye

Asst. Prof. Enes AYAN

Department of Electrical and Electronics Engineering, Faculty of Engineering and Architecture, Kırıkkale University, Kırıkkale, Türkiye

Assoc. Prof. Erinc KARATAŞ

Department of Computer Technologies, Vocational School of Elmadağ, Ankara University, Ankara, Türkiye

Asst. Prof. Faruk ULAMIŞ

Department of Electronics and Automation, Vocational School of Hacılar Hüseyin Aytemiz, Kırıkkale University, Kırıkkale, Türkiye

Assoc. Prof. Fatih KORKMAZ

Department of Electrical and Electronics Engineering, Faculty of Engineering, Çankırı Karatekin University, Çankırı, Türkiye

Assoc. Prof. Hüseyin POLAT

Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Türkiye

Assoc. Prof. İbrahim Alper DOĞRU

Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Türkiye

Asst. Prof. İsmail ATACAK

Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Türkiye

Prof. İsmail Rakıp KARAS

Department of Computer Engineering, Faculty of Engineering, Karabük University, Karabük, Türkiye

Journal of
**Computer & Electrical and
Electronics Engineering Sciences**

EDITORIAL BOARD

Assoc. Prof. Levent GÖKREM

Department of Electrical and Electronics Engineering, Faculty of Engineering and Architecture, Tokat Gaziosmanpaşa University, Tokat, Türkiye

Asst. Prof. Mehmet GÜÇYETMEZ

Department of Electrical and Electronics Engineering, Faculty of Engineering and Natural Sciences, Sivas University of Science and Technology, Sivas, Türkiye

Assoc. Prof. Muhammet Tahir GÜNEŞER

Department of Electrical and Electronics Engineering, Faculty of Engineering, Karabük University, Karabük, Türkiye

Assoc. Prof. Murat LÜY

Department of Electrical and Electronics Engineering, Faculty of Engineering and Architecture, Kırıkkale University, Kırıkkale, Türkiye

Asst. Prof. Mürsel Ozan İNCETAŞ

Department of Computer Technologies, Vocational School of Alanya Ticaret ve Sanayi Odası, Alanya Alaaddin Keykubat University, Antalya, Türkiye

Asst. Prof. Mustafa TEKE

Department of Electrical and Electronics Engineering, Faculty of Engineering, Çankırı Karatekin University, Çankırı, Türkiye

Asst. Prof. Mustafa KARHAN

Department of Computer Engineering, Faculty of Engineering, Çankırı Karatekin University, Çankırı, Türkiye

Asst. Prof. Mustafa Yasin ERTEN

Department of Electrical and Electronics Engineering, Kırıkkale University, Kırıkkale, Türkiye

Prof. Necaattin BARIŞCI

Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Türkiye

Prof. Necmi Serkan TEZEL

Department of Electrical and Electronics Engineering, Faculty of Engineering, Karabük University, Karabük, Türkiye

Nuri Alper METİN, PhD

Department of Electronics and Communications, Vocational School of Kırıkkale, Kırıkkale University, Kırıkkale, Türkiye

Assoc. Prof. Ramazan Kürşat ÇEÇEN

Department of Aircraft Technology, Vocational School of Eskişehir, Eskişehir Osmangazi University, Eskişehir, Türkiye

Asst. Prof. Rukiye KARAKIŞ

Department of Software Engineering, Faculty of Technology, Cumhuriyet University, Sivas, Türkiye

Asst. Prof. Saadin OYUCU

Department of Computer Engineering, Faculty of Engineering, Adıyaman University, Adıyaman, Türkiye

Dr. Salih DEMİR

Department of Computer Engineering, Faculty of Open and Distance Education, Ankara University, Ankara, Türkiye

Asst. Prof. Sevilay VURAL

Department of Internal Medicine Sciences, School of Medicine, Yozgat Bozok University, Yozgat, Türkiye

Assoc. Prof. Sinan TOKLU

Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Türkiye

Ufuk TANYERİ, PhD

Department of Computer Technologies, Vocational School of Nallıhan, Ankara University, Ankara, Türkiye

Yunus KÖKVER, PhD

Department of Computer Technologies, Elmadağ Vocational School, Ankara University, Ankara, Türkiye

Asst. Prof. Zafer CİVELEK

Department of Electrical and Electronics Engineering, Faculty of Engineering, Çankırı Karatekin University, Çankırı, Türkiye

Journal of
**Computer & Electrical and
Electronics Engineering Sciences**

EDITORIAL BOARD

ENGLISH LANGUAGE EDITOR

Assoc. Prof. Esra Güzel TANOĞLU

Department of Molecular Biology and Genetics, Hamidiye Health Sciences Institute, University of Health Sciences, İstanbul, Türkiye

STATISTICS EDITOR

Prof. Turgut KÜLTÜR

Department of Physical Therapy and Rehabilitation, Faculty of Medicine, Kırıkkale University, Kırıkkale, Türkiye

LAYOUT EDITOR

Şerife KUTLU

Engineer, MediHealth Academy Publishing, Ankara, Türkiye

Journal of

Computer & Electrical and Electronics Engineering Sciences

Dear Colleagues,

Academic journals play a vital role in the dissemination of scientific knowledge and provide researchers with a platform to share their work with the wider academic community. In this context, ensuring sustainability, consistency, and quality is essential for the continuous development of scientific publishing.

I am pleased to present the seventh issue of the Journal of Computer and Electrical-Electronics Engineering Sciences. Since the publication of our first issue, our journal has continued to grow with a strong commitment to academic integrity, ethical publishing standards, and the dissemination of original research in the fields of Computer Engineering and Electrical-Electronics Engineering.

We continue to implement a rigorous double-blind peer-review process and adhere strictly to ethical publishing principles, including transparency, academic integrity, and the protection of personal data. By maintaining our program of publishing two issues per year, we aim to ensure consistency while continuously improving the scientific contribution and overall quality of our journal.

Over the past year, we have observed a steady increase in both the number and quality of submitted articles, reflecting the growing interest in our journal. We are committed to supporting innovative, original, and high-quality research and providing a reliable publication platform for researchers.

Going forward, we will continue to focus on improving the quality, visibility, and impact of our journal while strengthening our editorial processes and academic standards. We sincerely thank you for your continued support and look forward to your future contributions.

Sincerely and with best wishes,

Assoc. Prof. Fuat TÜRK
Editor-in-Chief

Journal of
**Computer & Electrical and
Electronics Engineering Sciences**

CONTENTS

Volume: 4 Issue: 1 Year: 2026

ORIGINAL ARTICLES

- Application and performance evaluation of the dynamic bit-level encoding algorithm (DEA) in text compression: a cross-domain perspective..... 1-8**
Erdal, E., Ergüzen, A., & Önal, A.
- Development of a personalized cardio exercise and diet tracking mobile application: CardioFit IOS..... 9-16**
Uyan, Y., Sarısoy, F., Yılmaz, B., & Jira, H.
- Investigation of fine-tuned BERT models for sentiment analysis in COVID-19 tweets using a fuzzy logic-based ensemble approach..... 17-26**
Evgin, M. S., & Toklu, S.
- Intelligent diagnosis of tomato leaf diseases using YOLOv8..... 27-33**
Candan, H., Aydılek, H. & Erten, M. Y.

REVIEW

- Detection of Alzheimer's disease from magnetic resonance images with deep learning 34-45**
Güngör, A. & Barışçı, N.

Application and performance evaluation of the dynamic bit-level encoding algorithm (DEA) in text compression: a cross-domain perspective

✉ Erdal Erdal, ✉ Atilla Ergüzen, ✉ Alperen Önal*

Department of Computer Engineering, Faculty of Engineering and Natural Sciences, Kırıkkale University, Kırıkkale, Türkiye

Cite this article as: Erdal, E., Ergüzen, A., & Önal, A. (2026). Application and performance evaluation of the dynamic bit-level encoding algorithm (DEA) in text compression: a cross-domain perspective. *J Comp Electr Electron Eng Sci*, 4(1), 1-8.

Received: 15.08.2025

Accepted: 04.09.2025

Published: 25.04.2026

ABSTRACT

The exponential growth of digital data has heightened the need for efficient, lossless compression techniques to improve storage and transmission performance. This study investigates the applicability of the dynamic bit-level encoding algorithm (DEA), originally designed for image compression, to text compression. DEA employs a dynamic, frequency-based bit-level coding scheme that models inter-character relationships and constructs two-level Huffman trees for high- and low-frequency character groups. Unlike conventional algorithms such as Huffman, LZ77, and LZ78, DEA adapts its coding structure to local data patterns, aiming to enhance compression efficiency across heterogeneous text datasets. Experiments were conducted on nine datasets-including plain text, SQL queries, and HTML documents-using compression ratio (CR) and space saving (SS) as evaluation metrics. DEA achieved an average CR of 2.72 and SS of 62%, outperforming Huffman (1.87 CR, 43% SS), LZ77 (1.67 CR, 30% SS), and LZ78 (1.73 CR, 14% SS) across all datasets. These results demonstrate DEA's robustness and adaptability to diverse text structures. The findings suggest DEA as a practical alternative for applications requiring high-efficiency, lossless compression, such as archival systems, log management, messaging protocols, and real-time data transmission. Future work will focus on memory optimization, parallelization for large-scale applications, and integration into hybrid compression frameworks.

Keywords: Text compression, lossless compression, dynamic encoding algorithm, cross-domain compression

INTRODUCTION

In the contemporary era, the generation of digital data is undergoing an exponential growth, giving rise to substantial challenges in the domains of data storage and transmission (Dhulavvagol et al., 2024). The explosion of data has led to a strain on physical storage resources and has also exerted pressure on system resources, including network bandwidth, processing time, and energy consumption (Gastón et al., 2013). In this context, data compression techniques enhance storage efficiency by encoding digital data to occupy less space, thereby improving data transmission performance.

Data compression can be categorized into two primary classifications: lossy and lossless (Beemkumar et al., 2024). Lossless compression methods are preferred for sensitive data, such as text, where structural integrity is paramount (Gupta et al., 2017). Text compression is a process that represents character sequences with shorter bit sequences (Keskin et al., 2023). It is widely used in many areas, such as natural language processing, data archiving, messaging protocols, and log analysis. In such applications, the utilization of algorithms that offer both high CRs and low computational costs is of critical importance.

Conventional algorithms, including Huffman encoding, LZ77, and LZ78, have been demonstrated to be effective by capitalizing on the statistical and structural characteristics inherent in text data (Huffman, 1997). These encoding algorithms form the foundation of contemporary compression algorithms. However, the optimization of these methods for specific data types can impose various limitations in cross-domain applications. Consequently, the evaluation of algorithms developed for disparate domains on text data is imperative for two primary reasons. Firstly, it facilitates the identification of the potential inherent in novel methodologies. Secondly, it enables the assessment of the generalizability of these algorithms.

This study evaluates the performance of the dynamic bit-level encoding algorithm (DEA) (Erdal & Önal, 2025), initially developed for image compression, in text compression. We test the algorithm in comparison with traditional text compression methods, such as Huffman, LZ77, and LZ78. The goal is to determine whether DEA is effective with text data, under what conditions it provides advantages, and what its limitations are.

Corresponding Author: Alperen Önal, 238802017@kku.edu.tr



This work is licensed under a Creative Commons Attribution 4.0 International License.

The paper begins with an overview of existing compression techniques, followed by an introduction to the DEA algorithm's basic structure. Then, the experimental setup and metrics used are explained, and the results are evaluated in detail to discuss DEA's potential in text compression. The final section summarizes the findings and suggests future research directions.

BACKGROUND AND RELATED WORK

Data compression techniques use coding strategies to reduce file size by taking advantage of statistical properties and repetitive structures (Gopinath & Ravisankar, 2020). Lossless compression algorithms are designed to ensure that the original data can be fully recovered. Huffman encoding is one of the most widely used methods for this purpose (Huffman, 1952). It assigns short and long bit sequences based on the frequency of characters; more frequently occurring characters are represented by shorter codes. This approach provides a highly effective CR, especially in texts with uneven character distributions. However, because it operates on a symbol-based system, Huffman coding is limited in its ability to consider local context or larger patterns.

The LZ77 and LZ78 algorithms belong to the Lempel-Ziv family (Welch, 1984). They replace repetitive structures in data with pointer structures by performing a broader context analysis. The LZ77 algorithm identifies repetitive sequences by referencing previous data with a sliding window approach (Ziv & Lempel, 1977). LZ78, on the other hand, builds a dictionary-based structure that represents previously encountered sequences through indexes (Ziv & Lempel, 1978). These techniques produce effective results, especially in long texts or data sets with frequent repetition of certain patterns. However, parameters such as dictionary size and update strategy directly affect the performance of these algorithms. These classical algorithms all offer different advantages and limitations depending on the data type and pattern density. This necessitates developing new algorithms and conducting comparative performance analyses.

Huffman Encoding Algorithm

The Huffman coding algorithm, developed by David A. Huffman in 1952, is a statistical lossless data compression method (Huffman, 1952). It works by representing more frequently occurring symbols with shorter bit sequences and less frequently occurring symbols with longer ones. This minimizes the total bit length of the data, thereby achieving compression. Huffman coding first calculates the frequencies of characters and then creates a binary tree structure based on these frequencies.

The Huffman algorithm uses a prefix-free code structure, meaning no code can be the beginning of another code. This feature facilitates decoding compressed data and ensures that it can be decoded only once. The Huffman tree created during the encoding process is referenced during both encoding and decoding. Typically stored alongside the data, this tree enhances the algorithm's universality but may slightly reduce the CR.

The algorithm's greatest strengths are its low complexity and its ability to deliver highly effective results based on the statistical distribution of data. Huffman coding is particularly effective in texts with uneven symbol frequencies. However,

its tendency to treat each character independently and its inability to evaluate contextual patterns within the data can result in poorer performance in more complex data structures.

For this reason, Huffman coding is often used with other techniques or supported by more advanced, artificial intelligence-based, or pattern recognition methods. Nevertheless, its simple structure, high speed, and deterministic analysis feature make it an attractive option, especially for systems with limited processing power.

LZ77 Algorithm

The LZ77 algorithm, developed in 1977 by Abraham Lempel and Jacob Ziv, is a lossless compression method (Ziv & Lempel, 1977). It is based on identifying repeating data patterns and representing these repetitions with references, or pointers. LZ77 uses a sliding window approach to identify repeating sequences within previous data blocks, expressing these repetitions with a triplet structure that specifies the position and length of the data.

The algorithm does not consider both past and future data; it only uses a specific portion of the past window as a reference. The longest matching sequences within the window are identified and replaced with "distance, length, character" tokens. This structure provides a highly effective CR, especially when sequential repetitions are present in the data.

The LZ77 algorithm's greatest advantage is its ability to capture long and frequent repetitions in data effectively and perform compression at the sequence level rather than the character level. However, this method may require large buffers to increase the CR, which increases processing time proportionally to the window size. Additionally, its performance may be limited in short texts or data with low repetition rates.

Today, various LZ77 variants have been developed, forming the basis for common compression formats such as ZIP and GZIP. Thus, LZ77 remains one of the most important reference algorithms in text compression, both theoretically and practically.

LZ78 Algorithm

Developed in 1978 by Abraham Lempel and Jacob Ziv, the LZ78 algorithm is an extension of the LZ77 algorithm (Ziv & Lempel, 1978). It uses a dynamically generated dictionary structure to compress data. Each new sequence is evaluated based on its presence in the dictionary. If the sequence is not present, it is added as a new entry. Compressed data consists of an index pointing to the previous sequence in the dictionary combined with the new character.

Unlike LZ77, which references past data, LZ78 creates a dictionary that grows with each new sequence of symbols. This enables the algorithm to capture direct repetitions and more complex patterns within the data. As the algorithm learns each new sequence, the CR increases, making it particularly effective for large data sets.

The biggest advantage of LZ78 is its ability to adapt to data over time thanks to its constantly updated dictionary. However, controlling the size of the dictionary is also necessary. Efficient memory management is critical for the algorithm to work properly. Additionally, properly synchronizing the dictionary may require storing additional structures for data analysis.

The LZ78 algorithm formed the basis for many modern compression methods, particularly laying the groundwork for more advanced algorithms, such as LZW (Lempel-Ziv-Welch). Thus, LZ78 is considered one of the most fundamental algorithms, retaining its importance in both theoretical studies and practical data compression systems.

Application of DEA in Image Compression

In the field of image compression, lossless compression requires that data be represented in smaller sizes while preserving its integrity. Traditional algorithms often offer limited performance in this context. Increasing data diversity and resolution levels has heightened the need for more flexible, data-centric encoding methods. The DEA algorithm was developed to address this need. It offers a dynamic bit-level encoding approach that can easily adapt to different image types, particularly thanks to its adaptive structure (Erdal & Önal, 2025). DEA performs frequency-based encoding and works with an innovative strategy that includes contextual analysis based on inter-character relationships.

DEA's fundamental principle is evaluating characters likely to follow each other in sequence. This results in a dynamic coding dictionary, unlike the traditional Huffman encoding algorithm, which uses a fixed dictionary. The DEA analyzes these relationships to represent frequently recurring patterns with shorter bit sequences, ensuring that rarely encountered structures are encoded optimally. This contributes to more efficient CRs, particularly in high-resolution images with a wide range of colors. DEA's adaptive structure increases the CR and enables the algorithm to automatically configure itself according to different image contents.

The algorithm creates two groups within the application: a "palette" (group 1) for frequently used characters and a secondary group (group 2) for less common ones. The algorithm dynamically determines these two groups by analyzing the frequency with which each character occurs. Separate Huffman trees are created for each group and used during the encoding process according to which group each character belongs to. This structure enables DEA to encode frequently used characters with short bit sequences while preventing the unnecessary length of the representation of rarely used characters. Thus, DEA provides a bit-level optimized structure. This structure allows for effective compression of the entire image and its sub-segments. The flowchart of the DEA algorithms presented below in [Figure 1](#).

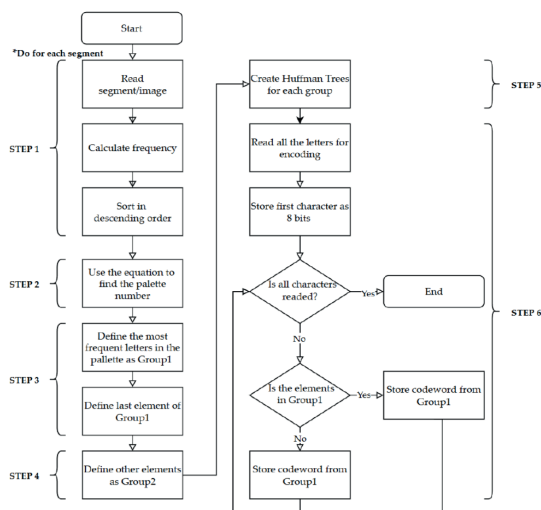


Figure 1. DEA flowchart (Erdal & Önal, 2025)

DEA: Dynamic bit-level encoding algorithm

The applicability of this algorithm in image compression has been enhanced by a segmentation-based preprocessing step. DEA processed images divided into segments based on color and texture similarities separately, thus enabling encoding on homogeneous data sets. This approach maximizes DEA's pattern recognition capabilities and increases compression efficiency. Additionally, the data structures created after segmentation (header and data blocks) ensure that the compressed data can be reconstructed without loss.

We tested the DEA algorithm in comparison with current lossless and lossy algorithms, such as JPEG2000, JPEG-LS, and BPG (Erdal & Önal, 2025). The results showed that DEA offers significant advantages, particularly for data sets with high color density and pattern repetition rates. Furthermore, the S+DEA version, which operates alongside segmentation, has been found to deliver superior performance across all datasets except for certain medical images. These results demonstrate that DEA is an innovative approach offering a comprehensive compression framework when combined with segmentation and data structures, not merely an encoding algorithm.

Gap in the Literature and the Need for Cross-Domain Evaluation

A review of the extant literature on data compression reveals that numerous algorithms have been developed for specific data types. Performance evaluations are typically conducted only for the designated domain. Algorithms that operate on different data types, such as image compression and text compression, are often addressed separately. As a result, the cross-domain validity of these algorithms is largely overlooked. However, the increasing demands of modern data processing have led to a growing need for adaptive and general-purpose compression techniques capable of operating with multiple data types concurrently.

In this context, the most dynamic and adaptive compression algorithms in the extant literature are either limited to fixed coding structures or dependent on specific pattern types. Classical algorithms such as Huffman, LZ77, and LZ78 have been demonstrated to be most effective in text or data sets with specific characteristics. However, the efficacy of these algorithms may be constrained in the context of alternative data types. This situation suggests that confining the evaluation of testing algorithms to their specific domains impedes the assessment of their potential application areas.

A notable lacuna in the extant literature pertains to the evaluation of algorithms developed for disparate domains in the context of their applicability to diverse data types. For instance, while the DEA algorithm has demonstrated notable efficacy in the domain of image compression, its effectiveness on character-based and structurally patterned data, such as text, remains to be systematically investigated. However, the dynamic structure of the DEA, its frequency-based coding approach that is sensitive to character sequences, and its ability to process on a segment basis render it a potential candidate for text data.

The primary motivation behind this study is to extend the boundaries of existing algorithms, assess their cross-domain validity, and identify novel application domains. In particular, the adaptability of algorithms with unique approaches, such as the data envelopment analysis (DEA), to different content types, such as text, should be analyzed without being limited

to a single data type. Such cross-evaluations are critical for both testing the generalizability of the algorithm and increasing the diversity of applications in the literature.

In summary, the present study undertakes two primary objectives. Firstly, it evaluates the performance of DEA on text data. Secondly, it proposes a model in terms of cross-domain validity. Thus, the study aims to fill an important gap in the literature. Consequently, the efficacy of DEA, which has been demonstrated to exhibit high efficiency in the context of image compression on various data types, including text, will be systematically investigated.

METHODS

Ethics

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data. All procedures were carried out in accordance with the ethical rules and principles.

Dynamic Bit-Level Encoding Algorithm (DEA)

Algorithm overview: The DEA is an innovative encoding method that is adaptable to different data types, lossless, and provides high efficiency. In contrast to the fixed-structure classical methods, the algorithm autonomously adjusts its configuration in a manner that is sensitive to local patterns and character sequences within the data. Initially developed for the purpose of image compression, DEA has the potential to offer a viable solution for character-based data types, such as text data, due to its context-sensitive structure.

The fundamental principle of DEA is to analyze the characters that are likely to follow each character and incorporate these relationships into the model based on frequency. In this process, character pairs that occur with high frequency are represented by shorter bit sequences, while character pairs that occur with low frequency are represented by longer sequences. This structure, in contrast to the classical Huffman algorithm, offers data-specific adaptation rather than a fixed symbol-frequency distribution. Consequently, the compression process becomes both more precise and more effective.

The algorithm delineates two distinct groups for each character: the first group (palette) consists of high-frequency characters, and the second group consists of low-frequency characters. Huffman trees are created separately for each group, and the encoding is performed based on which group the character belongs to. This structure ensures that frequently used characters are encoded efficiently while preventing waste in the encoding of low-frequency characters. Consequently, DEA attains bit-level optimized compression.

A salient feature of DEA is its reliance on a mathematical model that dynamically determines the palette size and character groups. This model determines the optimal bit distribution for each data segment, thereby ensuring the algorithm's compatibility with diverse data structures. The flexibility to work on a segment-based or entire data level makes DEA a strong candidate for both image and text data.

In summary, DEA represents a contemporary coding strategy that transcends conventional fixed coding methodologies, demonstrating a capacity for adaptability to the inherent dynamics of data. This approach facilitates the attainment of substantial compression rates, particularly in datasets

characterized by a high prevalence of repetitive patterns. In this regard, the present study explores an innovative algorithm for its applicability to different content types, such as images and text.

Original design for image compression: The DEA was initially developed for the purpose of lossless image compression. It was designed with an innovative structure that aims to overcome the limitations of traditional coding algorithms. This is particularly evident in scenarios involving images with intricate color distributions or dense textures. The efficacy of conventional compression methods is diminished, resulting in substantial performance degradation with respect to both storage and transmission. The DEA addresses this issue by analyzing the transition probabilities between characters and assigning dynamic bit lengths, offering a statistically adaptive coding approach.

The algorithm is predicated on the division of an image into smaller segments that exhibit analogous color and texture characteristics, with these segments then being encoded in a distinct manner. This approach facilitates the compression of each segment in the most optimal manner, based on its unique statistical characteristics. In the design of DEA, each pixel value is converted to an ASCII character, and the frequency relationships between characters are analyzed. Subsequently, two distinct Huffman trees are generated based on these relationships. This configuration facilitates the generation of abbreviated codes for frequently occurring characters and streamlined, extended codes for infrequent characters. Consequently, DEA provides a lossless solution that maintains both a high CR and data integrity.

Byte-level implementation details: The byte-level applicability of the DEA algorithm has been meticulously engineered to enhance the CR and optimize processing time. In practice, the RGB components (red, green, blue) of each pixel are processed separately. As these values are in the range of 0-255, they are directly represented by 8 bits (1 byte). These values are then converted to ASCII characters to initiate the algorithm's character frequency analysis process. Consequently, each color component is evaluated as an independent input sequence for character-based statistical analysis.

The algorithm calculates transition frequencies on these byte-level character sequences to determine the frequency of subsequent characters for each character. According to the resulting transition matrices, two distinct character sets (group 1 and group 2) are delineated for each character. Group 1 comprises characters that are frequently reiterated, while group 2 encompasses characters that are less frequently encountered. Huffman trees are created for each group, enabling shorter bit sequences to be assigned to frequently used characters, while optimized but longer bit sequences are used for rare characters. The flowchart presented in [Figure 2](#) provides a detailed visual representation of the steps summarized.

This structure at the byte level enables the algorithm to adapt dynamically to variations in image data, even when processing on a segment basis and encountering different contents in each segment. Furthermore, during the encoding process, the initial character is recorded directly with an 8-bit ASCII code, while subsequent characters are encoded at the byte level using the relevant Huffman trees. Characters that

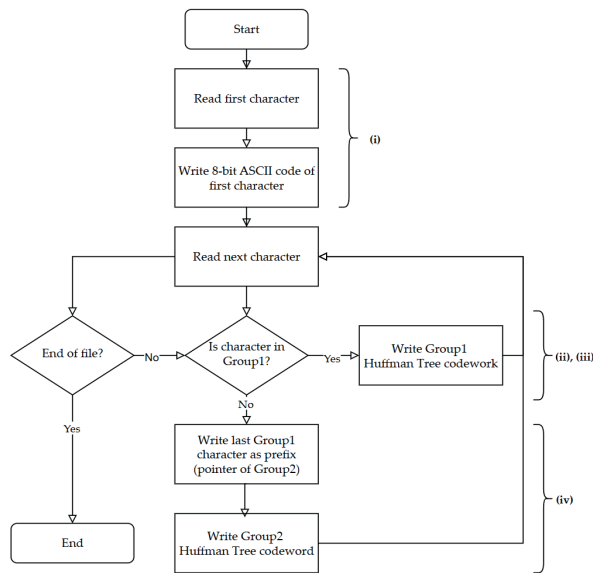


Figure 2. Flowchart of the DEA process example steps (Erdal & Önal, 2025)
DEA: Dynamic bit-level encoding algorithm

do not belong to group 1 are directed to group 2 through a specialized “flag code” and encoded via the Huffman tree. The byte-level modular structure of the system renders both the decoding process and the reconstruction of compressed data highly efficient and lossless.

The DEA algorithm’s modular, dynamic, and data-type-independent structure renders it suitable for utilization as an encoding component within the core of any desired compression algorithm. Its capacity to execute two-level Huffman encoding, predicated on inter-character transition frequencies, facilitates DEA’s expeditious adaptation to both fixed-structure and statistically variable data sets. Its byte-level operation enables direct application to text, images, biomedical signals, and other data types, facilitating seamless integration into compression frameworks. This feature enables DEA to function not only as a compression algorithm but also as a flexible encoding infrastructure that can be integrated into various applications.

EXPERIMENTAL SETUP

Datasets Used

To evaluate the proposed algorithm with greater accuracy and comprehensiveness, nine distinct data sets were utilized, exhibiting variation in terms of content and size. These datasets encompass a variety of data types, including structured, semi-structured, and unstructured formats, comprising diverse data structures such as plain text, SQL queries, and HTML documents. The nomenclature and dimensions (in bytes) of the datasets utilized in the experiments are enumerated in [Table 1](#).

Data set number	Data set name	Size (Byte)	Character type
1	Alice	148.481	UTF-8
2	Genome	5.244.846	UTF-8
3	Pizza Chill English 50	52.428.799	ISO-8891-1
4	Pizza Chill English 100	104.851.942	ISO-8899-1
5	enwik9	100.000.000	ISO-8859-1
6	SQL Query	22.210.031	ISO-8859-9
7	SQL Query 2	9.651.518	ISO-8859-9
8	SQL Query 3	11.647.877	ISO-8859-9
9	HTML	12.993	UTF-8

The type and characteristics of the data contained in the data sets directly affect compression performance. [Table 2](#) presents a cross-section of each data set.

Data set number	Data sample
1	Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to
2	>NC_007946.1 <i>Escherichia coli</i> UTI89, complete sequenceAGCTTTTTCATTCTGACTGCAACGGGCAATATGCTCTGTGTGGAT
3	[Redactor’s note: This document uses the ISO 8859-1 Latin1 character set (Windows). The book is composed
4	[Redactor’s note: This document uses the ISO 8859-1 Latin1 character set (Windows). The book is composed
5	<mediawiki xmlns=”http://www.mediawiki.org/xml/export-0.3/” xmlns:xsi=”http://www.w3.org/2001/XMLSchema-instance”
6	SQL Query: -- Tablo yapısı: `products` CREATE TABLE `products` (`id` int(11) NOT NULL, `name` varchar(255)
7	SQL Query2: CREATE TABLE `products` (`id` int(11) NOT NULL, `name` varchar(255) NOT NULL, `description` text
8	SQL Query3: CREATE TABLE `products` (`id` int(11) NOT NULL AUTO_INCREMENT, `name` varchar(255) NOT NULL, `description`
9	HTML: <!DOCTYPE html> <html lang=”tr”> <head> <meta charset=”UTF-8”> <meta name=”viewport” content

Evaluation Metrics

In order to evaluate the performance of the algorithms applied to the data more clearly and to perform comparative analyses, some criteria that have been widely accepted in the literature have been used. In this context, the CR and SS metrics have been preferred, especially for the quantitative expression of compression performance. The utilization of both metrics facilitates the objective interpretation of the results obtained and enables comparative evaluation between disparate algorithms.

The CR is a metric of paramount importance in the evaluation of the efficacy of a compression algorithm. This ratio is a quantitative metric that quantifies the extent to which the compressed data has been reduced in size compared to the original (uncompressed) data. In general, an elevated CR is indicative of the algorithm’s capacity to achieve enhanced data savings and optimize operational efficiency. This is particularly salient in applications that process voluminous data sets, where the CR exerts a pivotal function in reducing storage expenditures and accelerating data transmission times.

The CR is calculated using the following mathematical formula:

$$\text{Compression ratio} = \frac{\text{Original data size}}{\text{Compressed data size}}$$

According to the formula, if a 1000 KB file is compressed to 250 KB, the CR is 4.0. This indicates that the data has been reduced to a quarter of its original size. However, it should be noted that the CR does not always directly indicate the superiority of the algorithm. It is imperative to consider additional factors, such as compression time, algorithm complexity, and data type, to ensure a comprehensive evaluation.

SS is a critical metric that quantifies the amount of space saved on data that has undergone compression as a percentage of the original size. This metric, which is closely related to the CR, provides a more intuitive indication of the extent to which the data has been reduced after compression. This approach is frequently favored, particularly in systems with constrained storage capacity, as it provides a direct illustration of the extent to which the algorithm successfully reduces data storage requirements. The decision to express the results in percentage format was made with the intention of facilitating a more straightforward comparison of the performance of different algorithms.

SS is calculated using the following formula:

$$\text{Space saving} = \left(\frac{\text{Original data size} - \text{Compressed data size}}{\text{Original data size}} \right) \times 100$$

This formula demonstrates the amount of data that has been compressed as a percentage. To illustrate, when a 1000 kilobyte (KB) file is compressed to 300 KB, it is understood that a 70% SS has been achieved. This metric not only completes the CR but also renders compression performance more intelligible by clearly demonstrating to the user the extent to which data has been eliminated.

This study focused exclusively on compression efficiency metrics (CR and SS) to assess the cross-domain applicability of DEA. Computational complexity and runtime performance were not within the current scope but represent important areas for future research.

Implementation Environment and Tools

All development processes, simulations, and performance evaluations of the algorithm proposed in this study were carried out on a laptop computer with an Intel Core i7-4720HQ processor, 16 GB RAM, and 256 GB SSD hardware specifications. The efficacy of the algorithm was demonstrated through its ability to function seamlessly without inducing bottlenecks in processor power and memory usage during tests conducted on substantial datasets.

The NetBeans IDE 22 development environment was utilized during the software development process, and the algorithms were written in the Java programming language. Specifically, Java Development Kit (JDK) 17.0.12 and Java SE Runtime Environment 17.0.12+8-LTS-286 versions were preferred. The Java language facilitated the development of the algorithm in a modular, portable, and scalable manner, thanks to its advantages, including platform independence, automatic memory management, and extensive library support. The modules pertaining to image segmentation, DEA, and the calculation of evaluation metrics were executed in an integrated manner within this environment.

Compression Performance Comparison

The original sizes of the nine distinct data sets utilized in the study, along with the sizes obtained after implementing four distinct coding algorithms-Huffman, DEA, LZ77, and LZ78-to these data, are presented in **Table 3** for comparison.

Four distinct compression algorithms were implemented on nine distinct data sets utilized in the study, and CR values were derived for each. The results obtained are presented in **Table 4** and provide meaningful findings for comparing the overall

Table 3. Data set data sizes

Data set number	Original (Byte)	Huffman encoding (Byte)	DEA (Byte)	LZ77 Byte	LZ78 Byte
1	152.089	87.687	65.916	135.824	174.546
2	5.244.846	1.487.301	1.459.943	3.587.444	4.097.436
3	52.428.800	29.908.512	23.779.013	50.583.488	56.939.454
4	104.857.600	60.171.023	47.856.491	101.142.020	113.803.264
5	1.000.000.000	644.617.698	488.849.555	840.867.648	1.071.890.988
6	22.440.068	14.747.840	6.543.813	6.413.192	4.056.114
7	10.020.370	6.136.903	3.072.309	3.979.624	3.707.214
8	12.047.167	7.648.161	4.346.789	6.662.588	7.028.568
9	13.974	7.787	5.151	10.076	20.016

DEA: Dynamic bit-level encoding algorithm

performance of the algorithms. After the calculation of the CR values separately for each dataset, the average CR values for the algorithms were also determined. In this context, the DEA algorithm exhibited the optimal compression performance, with an average CR value of 2.72. Conversely, the Huffman algorithm attained an average CR value of 1.87, the LZ77 algorithm 1.67, and the LZ78 algorithm 1.73.

Table 4. Compression ratio and average CR value

Data set number	Huffman encoding	DEA	LZ77	LZ78
1	1.73	2.31	1.12	0.87
2	3.53	3.59	1.46	1.28
3	1.75	2.20	1.04	0.92
4	1.74	2.19	1.04	0.92
5	1.55	2.05	1.19	0.93
6	1.52	3.43	3.50	5.53
7	1.63	3.26	2.52	2.70
8	1.58	2.77	1.81	1.71
9	1.79	2.71	1.39	0.70
Average	1.87	2.72	1.67	1.73

CR: Compression ratio

The findings indicate that DEA provides superior CRs in comparison to conventional Huffman and dictionary-based algorithms. This enhancement is attributed to the dynamic nature of DEA's structure, which incorporates the probabilities associated with transitions between characters. This is particularly evident when considering the diversity of data sets, as DEA's statistically sensitive structure renders it more efficient for different content types. In this context, the DEA algorithm has proven to be a significant alternative in terms of compression efficiency, both in theoretical models and in practical applications.

SS values calculated to evaluate the effectiveness of compression algorithms show the percentage of SSs achieved by the applied methods compared to the original data. Pursuant to the experiments conducted, an examination of the mean SS values obtained from nine distinct data sets revealed that the DEA algorithm exhibited the optimal efficiency, yielding a 62% reduction in costs. After this, the Huffman algorithm yielded an average savings value of 43%, the LZ78 algorithm yielded 14%, and the LZ77 algorithm yielded 30% as shown in **Table 5**.

Table 5. Space saving values and average SD values

Data set number	Huffman encoding	DEA	LZ77	LZ78
1	0.42	0.57	0.11	-0.15
2	0.72	0.72	0.32	0.22
3	0.43	0.55	0.04	-0.09
4	0.43	0.54	0.04	-0.09
5	0.36	0.51	0.16	-0.07
6	0.34	0.71	0.71	0.82
7	0.39	0.69	0.60	0.63
8	0.37	0.64	0.45	0.42
9	0.44	0.63	0.28	-0.43
Average	0.43	0.62	0.30	0.14

SD: Standard deviation, DEA: Dynamic bit-level encoding algorithm

The findings indicate that DEA can more efficiently eliminate semantic redundancy within data by dynamically analyzing repetitive patterns and sequential character transitions. Specifically, the 19% increase in efficiency over the classical Huffman algorithm underscores the notion that DEA is not merely a theoretical strength but a practical alternative for data compression purposes. The observation that the LZ77 and LZ78 algorithms yield lower SS values suggests that the dictionary-based structures of these methods are constrained in certain data types and are not universally capable of achieving optimal compression. Overall, the DEA's capacity to yield substantial SSS across diverse data types renders it a versatile and effective compression method.

Statistical Analysis

In this study, the proposed DEA algorithm was analyzed in comparison with the classic Huffman, LZ77, and LZ78 algorithms to objectively evaluate compression performance. Each algorithm was implemented on nine distinct data sets, and the outcomes were assessed using fundamental performance metrics such as CR and SS. The numerical results obtained are presented in tabular form, followed by a general comparison of performance based on arithmetic means. Statistical analyses indicate that DEA provides a significant advantage over other algorithms in terms of both CR and SS values.

Specifically, the DEA algorithm demonstrated superior performance in comparison to all competing algorithms, exhibiting an average CR of 2.72 and an SS value of 62%. The values were calculated as 1.87 CR and 43% SS for the classic Huffman algorithm, 1.67 CR and 30% SS for LZ77, and 1.73 CR and only 14% SS for LZ78. The consistent observation of these differences across all datasets demonstrates that DEA delivers stable performance not only in selected cases but generally across different data structures. The findings indicate that DEA's success in modeling inter-character sequential relationships directly contributes to compression efficiency and offers a statistically significant advantage. This finding suggests that DEA can be regarded as an effective coding infrastructure for both academic and industrial applications.

DEA offers several notable advantages. First, it is an advanced encoding technique capable of efficiently representing repetitive patterns within data, leading to improved compression performance. A key strength of DEA is its data-

type independence, which allows it to operate effectively across diverse datasets without requiring significant structural adjustments. Additionally, when combined with appropriate preprocessing steps—such as normalization or tokenization—DEA can deliver even better results by reducing data variability before encoding. These characteristics make DEA a highly versatile component that can serve as the core of a compression pipeline, ensuring both flexibility and adaptability in various application scenarios. However, its implementation complexity and potential overhead in dictionary management should be considered when integrating DEA into large-scale or resource-constrained systems.

Limitations

The proposed DEA is distinguished by its flexible structure and dynamic encoding approach, which can be applied to different data types. A notable strength of this approach is its ability to generate a data-specific two-level Huffman tree through a systematic analysis of the frequencies of successive characters for each data element. This configuration facilitates high CRs by efficiently diminishing data density in data sets characterized by elevated repetition rates. Furthermore, its capacity to function at the byte level facilitates seamless adaptation to image, text, and other numerical data types. When integrated with segmentation, the encoding of each segment according to its own statistical structure represents another advantage that enhances the algorithm's adaptability and compression performance.

However, it is important to note that DEA does possess certain limitations. Initially, the implementation of discrete frequency analyses and Huffman trees for each character has the potential to prolong processing duration and augment the algorithm's computational intricacy. This phenomenon is particularly evident in datasets characterized by low repetition or high diversity. Consequently, the compression gain provided by the algorithm may be constrained. Furthermore, the group separation and flag encoding mechanism necessitates precise parameter adjustments, which may require additional adaptations to ensure optimal performance in different applications. Consequently, strategic choices based on data type and distribution are imperative when utilizing DEA.

CONCLUSION

In this study, the performance of the DEA, which was originally developed for image compression, was systematically examined in the field of text compression. A series of comparative experiments were conducted on nine distinct datasets, employing the classical Huffman, LZ77, and LZ78 algorithms. The effectiveness of these algorithms was gauged by two key metrics: CR and SS. The findings indicated that DEA exhibited statistically significant superiority over all competing algorithms, with an average CR of 2.72 and an SS value of 62%. These findings suggest that DEA can provide stable and high efficiency not only on specific data sets but also on different data types. The findings indicate that DEA is an algorithm with considerable promise for practical applications. In particular, DEA can provide significant advantages in areas that require low latency, high data volumes, and lossless data transmission. The high CR it provides in areas such as text-based archiving systems, log management, messaging services, and real-time data transmission can contribute to

both reducing storage costs and optimizing bandwidth usage. Furthermore, the algorithm's dynamic structure facilitates rapid adaptation to diverse data types, rendering it applicable in heterogeneous data processing environments. While DEA currently demonstrates robust performance, there are numerous opportunities for further research to enhance the efficacy of the algorithm. First, the algorithm's memory management strategies can be optimized to ensure efficient operation even on systems with low hardware capacity. Furthermore, the development of parallel processing versions will be a significant advancement, particularly for the real-time processing of large-scale data sets. A comprehensive analysis of prevailing compression algorithms will be conducted to identify opportunities for the integration of DEA coding components within the algorithmic framework. This will result in the formulation of a hybrid compression approach that is specific to DEA. The outcomes obtained from this approach will be evaluated in comparison with existing methods. An integration study will demonstrate how DEA can be positioned within the existing algorithm ecosystem and expand its potential areas of application. The Future Work section will move beyond generic statements and include specific, actionable directions such as extending DEA for large-scale text datasets, exploring its adaptability to big data environments, and examining its integration into distributed or cloud-based systems. Furthermore, an integration study will demonstrate how DEA can be positioned within the existing algorithm ecosystem and expand its potential areas of application. The presentation quality of tables and figures will be enhanced by ensuring consistent captions and seamless references within the text, and the reference list will be revised to correct formatting and special character issues in line with the journal's style guide. Finally, the development of an open-source DEA library will contribute to the broader adoption of the algorithm in both academic and industrial domains. This initiative will allow researchers to test DEA in diverse scenarios and enable continuous improvement of the algorithm through feedback from a wide user base.

ETHICAL DECLARATIONS

Ethics Committee Approval

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data.

Peer Review Process

This manuscript was subject to external peer review.

Conflict of Interest

The authors declare no conflicts of interest related to this study.

Financial Disclosure

This work is supported by the Kırıkkale University Department of Scientific Research Projects (2025/034).

Author Contributions

Concept: EE, AE, AÖ; Design: EE, AE, AÖ; Control: EE, AE, AÖ; Resources: EE, AE, AÖ; Materials: EE, AE, AÖ; Data Collection and/or Processing: EE, AE, AÖ; Analysis and/or Interpretation: EE, AE, AÖ; Literature Review: EE, AE, AÖ; Writing the Article: EE, AE, AÖ; Critical Review: EE, AE, AÖ.





Acknowledgments

The author acknowledges the assistance of the DeepL tool, which was used for improving the clarity and fluency of the English in this manuscript.

REFERENCES

- Beemkumar, N., Riyat, S., & Kumar, D. (2024). An Investigation of Lossless and Lossy Data Compression & Source Coding. 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), 1-6. <https://doi.org/10.1109/ICCCNT61001.2024.10725458>
- Dhulavvagol, P. M., Gadagkar, A., KJ, A., Hegade, G., Poonia, R., & Totad, S. G. (2024). Lossless text compression using recurrent neural networks. *Procedia Computer Science*, 235, 3340-3349. <https://doi.org/10.1016/j.procs.2024.04.315>
- Erdal, E., & Önal, A. (2025). Enhanced framework for lossless image compression using image segmentation and a novel dynamic bit-level encoding algorithm. *Applied Sciences*, 15(6), 2964. <https://doi.org/10.3390/app15062964>
- Gastón, B., Pujol, J., & Villanueva, M. (2013). A realistic distributed storage system that minimizes data storage and repair bandwidth (Versiyon 1). *arXiv*. <https://doi.org/10.48550/ARXIV.1301.1549>
- Gopinath, A., & Ravisankar, M. (2020). Comparison of lossless data compression techniques. 2020 *International Conference on Inventive Computation Technologies (ICICT)*, 628-633. <https://doi.org/10.1109/ICICT48043.2020.9112516>
- Gupta, A., Bansal, A., & Khanduja, V. (2017). Modern lossless compression techniques: review, comparison and analysis. 2017 *Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, 1-8. <https://doi.org/10.1109/ICECCT.2017.8117850>
- Hoffman, R. (1997). Compression Algorithms for Symbolic Data. In: R. Hoffman, Data Compression in Digital Systems (ss. 53-74). Springer US. https://doi.org/10.1007/978-1-4615-6031-9_4
- Huffman, D. (1952). A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, 40(9), 1098-1101. <https://doi.org/10.1109/JRPROC.1952.273898>
- Keskin, S., Seveli, O., & Okatan, E. (2023). Single and binary performance comparison of data compression algorithms for text files. *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, 12(3), 783-796. <https://doi.org/10.17798/bitlisfen.1301546>
- Welch. (1984). A technique for high-performance data compression. *Computer*, 17(6), 8-19. <https://doi.org/10.1109/MC.1984.1659158>
- Ziv, J., & Lempel, A. (1977). A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory*, 23(3), 337-343. <https://doi.org/10.1109/TIT.1977.1055714>
- Ziv, J., & Lempel, A. (1978). Compression of individual sequences via variable-rate coding. *IEEE Transactions on Information Theory*, 24(5), 530-536. <https://doi.org/10.1109/TIT.1978.1055934>

Development of a personalized cardio exercise and diet tracking mobile application: CardioFit IOS

 Yaren Uyan¹,  Fatih Sarisoy*²,  Büşra Yılmaz³,  Harun Jira⁴

¹Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Türkiye

²Personal Trainer, Grand National Assembly of Türkiye, Ankara, Türkiye

³Department of Physical Education and Sport, Faculty of Sport Sciences, Gazi University, Ankara, Türkiye

⁴Kahramanmaraş Metropolitan Municipality Secondary School, Ministry of Education, Kahramanmaraş, Türkiye

Cite this article as: Uyan, Y., Sarisoy, F., Yılmaz, B., & Jira, H. (2026). Development of a personalized cardio exercise and diet tracking mobile application: CardioFit IOS. *J Comp Electr Electron Eng Sci*, 4(1), 9-16.

Received: 18.10.2025

Accepted: 07.02.2026

Published: 25.04.2026

ABSTRACT

Aims: This study emphasizes the urgent need for integrated and adaptive platforms in the growing mobile health sector and aims to consolidate them into a common mobile platform. The main goal was to develop CardioFit IOS, an innovative IOS app designed to overcome limitations in existing mHealth tools. It provides a comprehensive, personalized health management platform that promotes sustainable healthy behaviors and user goal achievement.

Methods: Developed as a native IOS app using Swift and Firebase for secure data management and authentication, CardioFit IOS features an adaptive personalization engine powered by K-Means clustering. The engine analyzes user physiological data and in-app activities to form clusters, generating and refining personalized exercise, nutrition, and hydration plans beyond standard advice. It also includes real-time tracking of daily physical activity, diet, and water intake.

Results: CardioFit IOS unifies multiple health monitoring features into a single intuitive interface, improving user satisfaction by eliminating the need for multiple apps. Its AI-driven personalization via dynamic clustering delivers wellness strategies responsive to evolving user data, boosting engagement through continuous monitoring, feedback, and progress visualizations to enhance adherence to healthy routines.

Conclusion: CardioFit IOS represents a significant advance in mHealth, blending seamless integration with intelligent personalization. By leveraging advanced clustering and robust infrastructure, it supports users in achieving health and fitness goals, underscoring adaptive AI's value in personalized digital health interventions.

Keywords: Diet, firebase, fitness, healthy life, IOS, swift

INTRODUCTION

The pervasive integration of mobile phones and applications into daily life has propelled the popularity of fitness and healthy living apps. Research shows that mobile health (mHealth) applications effectively encourage physical activity and healthy dietary habits, leading to improved health outcomes and greater patient engagement (Basto & Ferreira, 2025; Pradal-Cano et al., 2020). These platforms are increasingly recognized as accessible, scalable solutions for promoting behavior change and managing chronic conditions (Ridwan et al., 2025).

Despite their widespread adoption and potential, many existing mHealth apps suffer from fragmentation and poor integration (Eaton et al., 2024; Mescher et al., 2024). Users often juggle multiple apps to track aspects of their health—such as exercise, diet, and hydration—resulting in a disjointed, inefficient experience. This fragmentation undermines adherence and long-term engagement, as evidenced by high dropout rates (Blasiak et al., 2022; Mazéas, 2023). Moreover, current systems rarely integrate diverse health data modalities,

limiting holistic insights and personalized recommendations (Gougeh & Žilić, 2024; Smith et al., 2025).

To address these limitations, CardioFit IOS's provides an innovative, comprehensive solution with a fully integrated user experience (Susaiyah et al., 2024). It consolidates essential health and fitness tracking—dietary intake, hydration, and physical activity—into a single platform, eliminating the need to manage multiple apps (Zahedani et al., 2023). This approach aligns with growing interest in personal health monitoring systems that aggregate data from various sources to deliver real-time alerts, comprehensive tracking, and tailored insights (Mahato et al., 2024; Secara & Hordiiuk, 2024).

Beyond simple integration, CardioFit IOS evaluates collected data holistically, adapting to each user's daily context. It surpasses conventional multi-module apps by generating fully personalized plans aligned with individual lifestyles and health goals (Moz et al., 2023). Such personalization is key to boosting user engagement and mHealth effectiveness, ultimately enhancing motivation and sustaining healthy habits.

Corresponding Author: Fatih Sarisoy, fatihsarisoy08@gmail.com



This work is licensed under a Creative Commons Attribution 4.0 International License.

Related Work & Motivation

A survey of mobile health applications in the literature reveals that CardioFit IOS core modules exercise tracking, water intake, and diet logging are present individually across various apps. However, no existing application integrates all three into a unified framework tailored to the individual user. While CardioFit IOS incorporates similar functions to those in current apps, it innovatively redesigns their interplay.

Most applications deliver these modules separately, requiring users to switch between apps to track exercise in one, log water intake in another, and record diet in a third. This fragmentation undermines practicality and long-term adherence. CardioFit IOS stands out not only by consolidating these features but also by interconnecting the data and adapting it to users' daily contexts, surpassing conventional multi-module apps. Moreover, it offers a free trial of all modules initially, ensuring broad accessibility (Table 1).

Review of Features in Other Applications

Apple Health: Apple Health serves as a foundational platform for health monitoring on the IOS operating system. However, the application primarily focuses on passive data collection. It does not aim to provide personalized experiences or influence behavior change based on the user's individual needs.

MyFitnessPal: While MyFitnessPal is strong in dietary tracking, it does not offer a personalized experience for fitness monitoring. In addition, water intake is managed as a separate module, resulting in a fragmented user experience.

Lifesum: Although Lifesum offers nutrition suggestions, these recommendations remain largely predefined for certain user segments and do not provide truly individualized dietary programs.

Fitbit: Fitbit is advanced in terms of physical measurements and sensor-based tracking; however, its recommendation features function primarily within the Fitbit device ecosystem, limiting accessibility for users who do not own compatible devices.

WaterMinder: WaterMinder focuses solely on hydration tracking. While effective for a single-purpose use case, it is limited in promoting comprehensive behavior change or providing personalized health guidance.

METHODS

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data. All procedures were carried out in accordance with the ethical rules and principles.

The CardioFit IOS application was developed for IOS devices using the Swift programming language and UIKit framework

for user interface design (Akoosh et al., 2023; Serantoni et al., 2022). It integrates Firebase for backend services including user authentication via Firebase Authentication and real-time data storage and synchronization in Firebase Realtime Database for user data on dietary intake, hydration, and physical activities (Chaudhari et al., 2023; Gundavarapu et al., 2023; Wiryonoputro & Saputri, 2023). Tailored health information, exercise, and dietary plans were created with input from professional fitness coaches to deliver personalized recommendations based on user-specific parameters (Chatterjee et al., 2022; Naveed et al., 2025). The extensive use of UIKit components ensures a user-friendly and interactive interface, enhancing overall usability (Chatterjee et al., 2022).

Artificial Intelligence Model for Personalization

CardioFit IOS employs a lightweight AI-based segmentation approach not a complex deep learning architecture to personalize the user experience (Chiarito et al., 2022). The model assesses users' fundamental physical characteristics alongside their in-app behavioral patterns, clustering individuals with similar tendencies (Subramaniaswamy et al., 2022). As a result, exercise and nutrition recommendations are dynamically shaped by the shared traits of these clusters, rather than relying solely on predefined formulas. Such personalization is essential for boosting user engagement and the effectiveness of mHealth solutions.

The model applies the K-Means unsupervised clustering algorithm, selected for its interpretability and computational efficiency, particularly in health data analysis.

Data Used

Both physical and behavioral variables were jointly evaluated for training the AI model and assigning users to appropriate clusters. The main variables used include:

- Age
- Weight
- Height
- Body-mass index (BMI)
- Basal metabolic rate (BMR)
- Physical activity level (PAL)
- Daily water consumption
- Daily recommended calorie intake

As users update their physical information or as in-app behavior changes over time, the AI mechanism reorganizes their profiles accordingly. Therefore, the model's output is not static; it continually adapts as user behavior evolves.

K-Means Unsupervised Clustering Algorithm

The K-Means algorithm, first introduced by MacQueen in 1967, has become one of the most widely used clustering methods in data mining (Ikotun et al., 2022). It aims to partition n data

Table 1. Comparison of existing application features with CardioFit

Feature	CardioFit	Apple Health	MyFitnessPal	Lifesum	Fitbit	WaterMinder
Integration of all modules on a single platform	✓	✗	✗	✗	Partial	✗
Personalized exercise planning	✓ (Karvonen + level test)	✗	✗	✗	Partial	✗
Personalized nutrition recommendations	✓	✗	Partial	✓	✗	✗
Daily water intake integration	✓	✗	✓	Partial	Partial	✓
AI-based segment analysis	✓	✗	✗	✗	✗	✗
Weekly/behavior-driven dynamic feedback	✓	✗	✗	✗	✗	✗

points into k clusters specified by the researcher, such that points within the same cluster are as similar as possible while those in different clusters are as distinct as possible (Fränti & Sieranoja, 2018). The algorithm operates through an iterative process in which cluster centers are repeatedly updated until they reach stability.

Academic and Mathematical Description of the K-Means Model

K-Means clustering is an unsupervised learning algorithm commonly used in behavioral segmentation and health data analysis due to its interpretability and low computational cost. The algorithm partitions the dataset $X=\{x_1, x_2, \dots, x_n\}$ into k distinct clusters by minimizing the objective function:
$$J = \sum_{i=1}^k \sum_{x_j \in C_i} \|x_j - \mu_i\|^2$$

Feature scaling was performed using StandardScaler to prevent high-range variables, such as calorie intake, from dominating the clustering process.

The optimal value of k was determined empirically by comparing Silhouette and Davies-Bouldin scores for ($k=2, 3, 4$); among these, $k=3$ achieved the highest cluster separation (Fränti & Sieranoja, 2018).

Construction of the Synthetic Dataset

Since the application was tested by a limited number of users during the early development phase, a synthetic dataset was generated solely for proof-of-concept evaluation of the clustering mechanism. The synthetic data does not aim to represent real-world behavioral distributions and should not be interpreted as empirically validated user data.

The primary purpose of this dataset is to demonstrate the operational behavior of the proposed personalization model and to illustrate how user clusters may emerge under controlled conditions. The absence of real-world data is therefore acknowledged as a significant limitation of the current study (Vara et al. 2022, Ahmed et al. 2020).

The synthetic dataset was created by:

- Considering typical mHealth user distributions reported in the literature,
- Generating realistic values for age, BMI, PAL, water intake, and exercise duration,
- Forming a representative sample of 120 synthetic users.

This dataset was used solely to test the operational behavior of the model and to demonstrate how clusters emerge (Table 2).

Age	BMI	PAL	Water (L)	Exercise (min)	Calorie	Cluster
27	21.44	1.24	1.27	26	2093	2
33	24.86	1.52	1.52	45	2458	1
42	28.42	1.79	2.15	61	2874	0
24	20.11	1.22	1.23	34	1900	2
36	25.88	1.47	1.62	49	2578	1
39	27.66	1.83	2.03	58	2983	0
23	20.54	1.22	1.11	29	1990	2
28	24.91	1.50	1.66	42	2389	1
40	27.94	1.80	2.07	63	2898	0
26	21.73	1.24	1.28	31	2008	2

BMI: Body-mass index, PAL: Physical activity level

To evaluate the clustering model during development, a synthetic dataset was generated. This approach is widely used in early-stage studies of personalized mobile health applications, as it protects patient privacy while enabling robust model testing. The dataset drew from typical mHealth user distributions in the literature to generate realistic values for age, BMI, PAL, water intake, and exercise duration, producing a representative sample of 120 synthetic users. It was used to assess the model’s behavior and demonstrate cluster formation, with daily caloric requirements estimated based on BMI, PAL, and activity duration (Yun. et al., 2025).

Model Limitations

Although the clustering-based personalization mechanism provides an interpretable and computationally efficient solution, several limitations should be acknowledged.

First, the K-Means algorithm relies on Euclidean distance and assumes linear separability, which may limit its ability to capture complex and non-linear behavioral patterns. Additionally, the model remains sensitive to the predefined number of clusters (k), which may influence segmentation outcomes.

Second, the use of a fully synthetic dataset represents an idealized approximation of user behavior and does not reflect real-world variability. As a result, the findings should be interpreted as preliminary and illustrative rather than empirically validated.

Third, the current model is based on cross-sectional profile data and does not incorporate longitudinal behavioral changes over time. Temporal transitions between user clusters and their long-term impact on personalization accuracy remain an open research direction.

Future work will focus on collecting real user data, incorporating time-series analysis, and comparing K-Means with alternative clustering approaches such as Hierarchical Clustering and DBSCAN to enhance robustness and behavioral expressiveness (Ahmed et al. 2020).

Interpretation of Clusters

The model identified three distinct clusters, each guiding a tailored personalization strategy:

- **Sedentary users:** Light and progressive exercise plans.
- **Moderately active users:** Balanced and structured training plans.
- **Athletic/high-activity users:** High-intensity, performance-oriented plans (Table 3).

Evaluation of Model Performance

Clustering performance was assessed using the Silhouette Score, which quantifies cohesion within clusters relative to separation from others. The model achieved a Silhouette Score of 0.609, indicating good cluster separation (Vardakas et al., 2024). Additionally, a Davies-Bouldin Index of approximately 0.7 confirmed strong internal cluster coherence.

Impact of the Model on Application Output

User profiles from the clustering model directly shape CardioFit iOS three core modules:

Table 3. Interpretation of K-Means clustering results

Cluster	BMI	PAL	Exercise duration	Water intake	Daily caloric requirement*	Likely user profile	Recommended goal
Cluster 1 - low activity	20-22	1.2-1.3	30 min	1.0-1.3 L	1900-2200 kcal	Young individuals with normal BMI but low activity level; sedentary workers; beginner-level users	Healthy weight maintenance or gradual weight gain; beginner exercise program
Cluster 2 - moderate activity	23-26	1.4-1.6	45 min	1.4-1.8 L	2300-2600 kcal	Users who walk regularly, have normal to slightly above-normal BMI, moderately active	Weight loss or improved conditioning; intermediate exercise program
Cluster 3 - high activity	27-30	1.7-1.9	60 min	1.9-2.4 L	2700-3200 kcal	Users with athletic background, high energy expenditure; individuals engaged in strength training	Muscle gain, performance improvement, advanced training program

*Daily caloric requirement values are estimated considering BMI, PAL, and activity duration, BMI: Body-mass index, PAL: Physical activity level

- **Exercise module:** Adapts intensity, set duration, and difficulty to the assigned cluster.
- **Diet module:** Personalizes daily calorie targets and meal suggestions by cluster.
- **Water intake module:** Dynamically adjusts reminder frequency and hydration targets based on the profile.

This integration delivers personalized, adaptive recommendations that evolve beyond static lists to behavior- and profile-driven guidance.

Application Modules and Structure

Addressing the fragmentation common in existing mobile health tracking applications which often focus on isolated aspects CardioFit IOS offers a comprehensive, integrated solution. The app is structured around three core modules that enable seamless management of daily health activities within a single platform.

Exercise Tracking Module

The exercise tracking module, a central feature of CardioFit IOS, delivers a personalized program based on onboarding information and user objectives. It monitors key physical activities to help users achieve their health and fitness goals.

Upon entering the exercise section, users select from two main categories: cardio and fitness. Each offers programs at three difficulty levels beginner (4-week plans), intermediate (8-week), and advanced (12-week) with one to three options per level, all developed with input from professional fitness coaches (Iolascon et al., 2021).

Users choose a level based on self-assessment; for those unsure, the International Physical Activity Questionnaire determines a suitable starting point for tailored progression.

Cardio Module Details

The cardio module tracks activities like running, swimming, brisk walking, and cycling, recording and analyzing data such as heart rate and calories burned. Prior to use, the user's target heart rate is calculated via the Karvonen formula using age and resting heart rate (Karvonen & Vuorimaa, 1988; Olsson et al., 2022). Although widely used for prescribing exercise

intensity, this predictive method requires caution due to individual physiological variations.

This approach defines intensities for beginner, intermediate, and advanced plans, with gradual increases as users progress. Such progression enhances aerobic capacity and cardiovascular health safely, optimizing adaptation while minimizing overexertion and injury risks (Milani et al., 2024).

Exercise Intensity Percentages

Exercise intensity is categorized by exertion level, aligning with physiological goals that influence metabolic, hormonal, and cardiorespiratory responses:

- **Low intensity:** Ideal for activity novices, this builds basic movement skills and eases adaptation to exercise (Taylor et al., 2021).
- **Weight control:** Moderately intense to maximize fat burning, boosting energy expenditure for weight management and fat reduction (MacIntosh et al., 2021).
- **Aerobic:** Strengthens the cardiovascular system and enhances fat utilization, improving heart-lung capacity and endurance (Koman et al., 2024).
- **Anaerobic:** High-intensity for building muscle strength and performance, demanding near-maximal effort (Patel et al., 2017).

These categories guide program customization to users' goals and fitness levels.

Exercise Duration Protocols

CardioFit IOS includes structured protocols for 8-week and 12-week exercise programs that progressively increase in frequency, duration, and intensity to optimize adaptation and minimize injury (Rospo et al., 2016) (Table 4, 5).

Sex-Specific Basal Metabolic Rate Calculations: a Rationale for Differentiation

The differentiation in basal metabolic rate calculations between males and females is not an inconsistency but a critical aspect for maintaining accuracy, predicated on fundamental physiological distinctions between the sexes. The utilization

Table 4. Exercise duration protocol for an 8-week program

Weeks	Sessions per week	Warm-up (min)	Stretching (min)	Exercise intensity	Target energy expenditure (kcal)	Session duration (min)
1-2	2	10	5	50% HRMax	150	20
3-4	2	10	5	50% HRMax	200	30
5-6	3	10	5	60% HRMax	250	35
7-8	3	10	5	60% HRMax	300	35

Table 5. Exercise duration protocol for a 12-week program

Weeks	Sessions per week	Warm-up (min)	Stretching (min)	Exercise intensity	Target energy expenditure (kcal)	Session duration (min)
1-2	2	10	5	50% HRMax	200	30
3-4	3	10	5	60% HRMax	200	30
5-8	4	10	5	70% HRMax	300	40
9-12	4	10	5	70% HRMax	400	50

of sex-specific predictive equations, such as the Mifflin-St. Jeor equation, ensures that applications like CardioFit IOS provide metabolically consistent and scientifically grounded estimations of energy expenditure.

Physiological basis for sex-specific BMR calculations:

Variations in BMR between sexes are primarily driven by differences in body composition, hormonal profiles, and average body dimensions:

- **Body composition:** Males typically exhibit a higher proportion of lean body mass, particularly skeletal muscle tissue, compared to females. Given that muscle tissue is metabolically more active than adipose tissue, a greater LBM contributes to a higher absolute BMR in males (Jagim et al., 2023). Conversely, females generally possess a higher percentage of body fat, which is metabolically less active, thereby contributing to a comparatively lower BMR even when adjusted for overall body mass (Ferraro et al., 1992).
- **Hormonal influences:** Sex hormones significantly modulate energy metabolism and body composition. Testosterone in males promotes muscle anabolism, while estrogen in females influences fat distribution and has been observed to impact resting energy expenditure, with studies showing that estrogen administration can increase resting energy expenditure (Weidlinger et al., 2023). These distinct hormonal landscapes contribute to differing metabolic profiles, influencing overall energy homeostasis and metabolism between men and women (Mauvais-Jarvis, 2015, 2023; Sanchez et al., 2024).
- **Anthropometric averages:** On average, males tend to be taller and possess greater body mass than females. While height and weight are direct inputs into BMR predictive equations, the cumulative effect of these anthropometric differences also contributes to the observed variations in BMR between sexes.

Consistency in the Mifflin-St. Jeor equation: The Mifflin-St. Jeor equation systematically accounts for these physiological sex differences through distinct constant terms within its formulation. The development of this predictive equation involved multiple-regression analyses derived from data on a substantial cohort of healthy subjects, leading to empirically established relationships between resting energy expenditure and factors such as weight, height, and age (Mifflin et al., 1990). The equations are as follows:

- **Female BMR:** $BMR = [9.99 \times \text{weight (kg)}] + [6.25 \times \text{height (cm)}] - [4.92 \times \text{age (years)}] - 161$
- **Male BMR:** $BMR = [9.99 \times \text{weight (kg)}] + [6.25 \times \text{height (cm)}] - [4.92 \times \text{age (years)}] + 5$

The distinct constant values (-161 for females and +5 for males) are empirically derived adjustments. These constants

were specifically determined through statistical analysis to optimize the fit between predicted and measured BMRs for each sex, reflecting the integrated effects of body composition, hormonal regulation, and average anthropometrics (Mifflin et al., 1990). Therefore, these sex-specific calculations do not represent an inconsistency; rather, they constitute an academically rigorous and physiologically accurate methodology for estimating BMR, making them entirely appropriate and necessary for precise application within platforms like CardioFit IOS.

Physical Activity Level Multipliers

The physical activity level multipliers are essential for calculating total energy expenditure by adjusting the BMR based on an individual’s activity level (Table 6).

Table 6. Physical activity levels

Physical activity level	Multiplier
Sedentary	1.0-1.39
Lightly active (light exercise or sports, 1-3 days per week)	1.4-1.59
Moderately active (moderate exercise or sports, 3-5 days per week)	1.6-1.89
Very active (intense exercise or sports, 6-7 days per week)	1.9-2.5

Calculation of Total Energy Expenditure

Once the total energy expenditure is calculated, the application determines the user’s daily caloric needs and provides tailored guidance (Bianchetti et al., 2022). This approach helps users maintain energy balance and achieve their specific health goals. It then presents a variety of meal options, including both user-created and recommended meals customized to their preferences. These suggestions account for factors such as physical activity, meal history, height, and weight (Papastratis et al., 2024). The module integrates open-source APIs to compile comprehensive meal lists with nutritional values, images, and total caloric content (Han & Chen, 2024). Daily meal data is recorded in a Firebase database, allowing users to track progress over time and visualize dietary performance through graphs.

Hydration Tracking Module

The CardioFit IOS application enables users to log their daily water consumption alongside dietary intake. Similar to dietary tracking, users can customize their daily target water intake to match personal preferences (Cruz et al., 2021; Pauley et al., 2024). By default, the app calculates an individual’s daily water requirement using a standardized formula based primarily on body weight, offering personalized guidance though more advanced models incorporate additional intrinsic and extrinsic factors, such as activity level or climate (Dolci et al., 2022). This formula is shown below. Meanwhile, wearable technologies and smart devices are emerging as complementary tools for monitoring fluid intake.

Daily Water Requirement Equation

Daily water requirement = body weight x 35 ml

Users can view their daily water consumption in real-time through graphical visualizations powered by data stored daily in a Firebase database. This setup helps them track progress, manage hydration habits, and access detailed analyses to achieve long-term goals (Reeves et al., 2023).

DISCUSSION

The CardioFit IOS application addresses a critical need within the mobile health (mHealth) landscape by offering a comprehensive, integrated platform for exercise, diet, and hydration tracking. Existing literature highlights the prevalent issue of fragmentation in mHealth applications, where users often resort to multiple single-purpose apps, leading to disjointed experiences and diminished long-term adherence (Tomlinson et al., 2013). CardioFit IOS directly counters this by consolidating these essential health management functions into a single ecosystem, fostering a more streamlined and efficient user journey, an approach supported by research advocating for integrated solutions in chronic disease management (Ferreira et al., 2024).

A core strength of CardioFit IOS lies in its personalized approach, driven by an artificial intelligence mechanism utilizing K-Means unsupervised clustering. This contrasts with many existing applications that offer predefined or limited personalization, often failing to adapt to the dynamic and evolving needs of users (Zhu et al., 2021). While personalization is recognized as essential for enhancing user engagement and effectiveness in mHealth (Rivera-Romero et al., 2023), CardioFit IOS's strategy of clustering users based on both physical characteristics and in-app behavioral patterns allows for the dynamic shaping of recommendations, aiming to mitigate the "personalization paradox" the inherent conflict between user modeling and adaptation in behavior change applications (Zhu et al., 2021). The use of K-Means for user segmentation is a recognized technique in customer segmentation and behavior analysis, valued for its interpretability and computational efficiency (Salminen et al., 2023).

The application's detailed exercise module incorporates the Karvonen formula for heart rate-based intensity prescription (Hofmann & Tschakert, 2017) and structured duration protocols, providing a scientifically grounded framework for physical activity (Arora et al., 2023). Similarly, the diet and hydration modules offer personalized targets and recommendations, utilizing established equations for basal metabolic rate and total energy expenditure calculations (Prado-Nóvoa et al., 2024), further enhancing the individualized nature of the application (Abelino et al., 2024).

Despite its advantages, the current implementation of CardioFit IOS acknowledges several limitations, consistent with broader challenges in mHealth AI. The reliance on K-Means clustering, while interpretable, may struggle to capture complex, non-linear behavioral patterns due to its dependence on Euclidean distance, and its performance can be sensitive to noise and the need to specify the number of clusters a priori (Zahra et al., 2015; Zhang et al., 2025). Furthermore, the initial development and evaluation were conducted using a synthetic dataset (Giuffré & Shung, 2023). While useful in early-stage studies for protecting privacy

and evaluating model behavior, such data represent idealized user distributions rather than real-world variability (Breugel et al., 2023). This highlights a general challenge in mHealth app research, where practical implementation faces hurdles related to efficacy, uptake, usability, and patient outcomes (Birkhoff & Moriarty, 2020).

Future work will focus on collecting real-world user data, exploring alternative clustering algorithms, and developing a hybrid model that integrates supervised learning techniques (Rauba et al., 2024).

Limitations

The clustering-based personalization mechanism, while offering a lightweight and interpretable solution, presents inherent limitations. First, K-Means relies on Euclidean distance and may fail to capture non-linear behavioral patterns. Second, the synthetic dataset represents idealized user distributions rather than real-world variability, potentially leading to performance gaps when transitioning to authentic scenar IOS. Third, the current personalization engine depends primarily on profile-level data and does not yet incorporate longitudinal changes in user behavior.

CONCLUSION

This study identified limitations in existing mobile health applications that fragment the user experience, prompting the development of CardioFit IOS. The application provides a personalized, holistic solution for tracking health, diet, and exercise, empowering users to adopt sustainable healthy lifestyle habits. CardioFit IOS introduces key innovations, including integrated monitoring of exercise, diet, and hydration in a single platform. This directly addresses the challenge of juggling multiple apps, while its user-friendly interface enhances accessibility and practicality for daily health management. At the core of the app's effectiveness is its AI mechanism, which uses a lightweight K-Means unsupervised clustering algorithm for dynamic user segmentation. Rather than relying on static recommendations, it analyzes users' physical characteristics and in-app behaviors, grouping those with similar patterns into evolving clusters. As a result, exercise intensities, diet targets, and hydration reminders adapt continuously to each user's profile and cluster traits, delivering truly personalized health plans. In conclusion, CardioFit IOS advances mHealth through its integrated platform and intelligent personalization. By leveraging AI for dynamic, profile-driven recommendations alongside comprehensive tracking, it empowers users to achieve their health and wellness goals. Ongoing refinements based on user feedback and addressing the identified limitations will further enhance this innovative tool.

ETHICAL DECLARATIONS

Ethics Committee Approval

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data.

Peer Review Process

This manuscript was subject to external peer review.

Conflict of Interest

The authors declare no conflicts of interest related to this study.

Financial Disclosure

The authors received no financial support for the conduct or publication of this research.

Author Contributions

Concept: YU, FS, BY, HJ; Design: YU, FS, BY, HJ; Control: YU, FS, BY, HJ; Resources: YU, FS, BY, HJ; Materials: YU, FS, BY, HJ; Data Collection and/or Processing: YU, FS, BY, HJ; Analysis and/or Interpretation: YU, FS, BY, HJ; Literature Review: YU, FS, BY, HJ; Writing the Article: YU, FS, BY, HJ; Critical Review: YU, FS, BY, HJ.

REFERENCES

- Abeltino, A., Riente, A., Bianchetti, G., Serantoni, C., Spirito, M. D., Capezzone, S., ..., & Maulucci, G. (2024). Digital applications for diet monitoring, planning, and precision nutrition for citizens and professionals: a state of the art. *Nutrition Reviews*, 83(2). <https://doi.org/10.1093/nutrit/nuae035>
- Ahmed, M., Seraj, R., & Islam, S. M. S. (2020). The k-means algorithm: a comprehensive survey and performance evaluation. *Electronics*, 9(8), 1295. <https://doi.org/10.3390/electronics9081295>
- Akoosh, L. M. S., Siddiqui, F., Naaz, S., & Alam, M. A. (2023). Machine learning using radial basis function with k means clustering for predicting cardiovascular diseases. In *Lecture notes in electrical engineering* (p. 651). Springer Science+Business Media. https://doi.org/10.1007/978-981-99-5974-7_52
- Arora, N. K., Donath, L., Owen, P. J., Miller, C. T., Saueressig, T., Winter, F., ..., & Belavý, D. L. (2023). The impact of exercise prescription variables on intervention outcomes in musculoskeletal pain: an umbrella review of systematic reviews. *Sports Medicine*, 54(3), 711. <https://doi.org/10.1007/s40279-023-01966-2>
- Basto, P. S., & Ferreira, P. (2025). Mobile applications, physical activity, and health promotion. *BMC Health Services Research*, 25, 1. <https://doi.org/10.1186/s12913-025-12489-z>
- Bianchetti, G., Abeltino, A., Serantoni, C., Ardito, F., Malta, D., Spirito, M. D., & Maulucci, G. (2022). Personalized self-monitoring of energy balance through integration in a web-application of dietary, anthropometric, and physical activity data. *Journal of Personalized Medicine*, 12(4), 568. <https://doi.org/10.3390/jpm12040568>
- Birkhoff, S. D., & Moriarty, H. (2020). Challenges in mobile health app research: Strategies for interprofessional researchers. *Journal of Interprofessional Education & Practice*, 19, 100325. <https://doi.org/10.1016/j.xjep.2020.100325>
- Blasiak, A., Sapanel, Y., Leitman, D., Ng, W. Y., Nicola, R. D., Lee, V. V., ..., & Ho, D. (2022). Omnichannel communication to boost patient engagement and behavioral change with digital health interventions. *Journal of Medical Internet Research*, 24, 11. <https://doi.org/10.2196/41463>
- Bottou, L., & Bengio, Y. (1994). Convergence properties of the k-means algorithms. *Advances in Neural Information Processing Systems*, 7, 585-592.
- Breugel, B. van, Qian, Z., & Schaar, M. van der. (2023). *Synthetic data, real errors: How (not) to publish and use synthetic data*. arXiv. <https://doi.org/10.48550/arxiv.2305.09235>
- Chatterjee, A., Prinz, A., Gerdes, M., Martínez, S., Pahari, N., & Meena, Y. K. (2022). ProHealth eCoach: user-centered design and development of an eCoach app to promote healthy lifestyle with personalized activity recommendations. *BMC Health Services Research*, 22, 1. <https://doi.org/10.1186/s12913-022-08441-0>
- Chaudhari, S., Aparna, R., Ramesh, A. S., Bhat, D., Gaurav, V., & Divya. (2023). *Multi-factor based nutrition management system and recipe recommendation engine*. Research Square. <https://doi.org/10.21203/rs.3.rs-3227663/v1>
- Cirett-Galán, F., Peralta, R. T., & Mora, O. F. G. (2023). *K-means cluster analysis to support diabetic patient care*. Research Square. <https://doi.org/10.21203/rs.3.rs-2461033/v1>
- Cruz, F. O. D. A. M. D., Faria, E. T., Ghobad, P. C., Alves, L. Y. M., & Reis, P. E. D. D. (2021). A mobile app (AMOR Mama) for women with breast cancer undergoing radiation therapy: Functionality and usability study. *Journal of Medical Internet Research*, 23, 10. <https://doi.org/10.2196/24865>
- Dolci, A., Vanhaecke, T., Qiu, J., Ceccato, R., Arboretti, R., & Salmaso, L. (2022). Personalized prediction of optimal water intake in adult population by blended use of machine learning and clinical data. *Scientific Reports*, 12, 1. <https://doi.org/10.1038/s41598-022-21869-y>
- Eaton, C. K., McWilliams, E., Yablon, D., Kesim, I., Ge, R., Mirus, K., ..., & Riekert, K. A. (2024). Cross-cutting mHealth behavior change techniques to support treatment adherence and self-management of complex medical conditions: systematic review. *JMIR Mhealth and Uhealth*, 12. <https://doi.org/10.2196/49024>
- Ferraro, R., Lillioja, S., Fontvieille, A. M., Rising, R., Bogardus, C., & Ravussin, É. (1992). Lower sedentary metabolic rate in women compared with men. *Journal of Clinical Investigation*, 90(3), 780-784. <https://doi.org/10.1172/jci115951>
- Ferreira, E. de S., Oliveira, A. H. M. de, Dias, M. A., Costa, G. D. da, Januário, J. P. T., Botelho, G. M., & Cotta, R. M. M. (2024). Mobile solution and chronic diseases: development and implementation of a mobile application and digital platform for collecting, analyzing data, monitoring and managing health care. *BMC Health Services Research*, 24, 1. <https://doi.org/10.1186/s12913-024-11505-y>
- Fränti, P., & Sieranoja, S. (2018). K-means properties on six clustering benchmark datasets. *Applied Intelligence*, 48(12), 4743-4759. <https://doi.org/10.1007/s10489-018-1238-7>
- Giuffré, M., & Shung, D. (2023). Harnessing the power of synthetic data in healthcare: innovation, application, and privacy. *Npj Digital Medicine*, 6, 1. <https://doi.org/10.1038/s41746-023-00927-3>
- Gougeh, R. A., & Žilić, Ž. (2024). Systematic review of IoT-based solutions for user tracking: towards smarter lifestyle, wellness and health management. *Sensors*, 24(18), 5939. <https://doi.org/10.3390/s24185939>
- Gundavarapu, M. R., Bhavita, M., Sahithi, M., Varsha, N. A., Kumar, R., & Prasanna, Y. L. (2023). IoT-powered intelligent framework for detecting food adulteration: a smart approach. *E3S Web of Conferences*, 430, 1074. <https://doi.org/10.1051/e3sconf/202343001074>
- Han, M., & Chen, J. (2024). *NutrifyAI: An AI-powered system for real-time food detection, nutritional analysis, and personalized meal recommendations*. arXiv. <https://doi.org/10.48550/arxiv.2408.10532>
- Hang, Y., Yin, H., Hu, W., & Zhong, L. (2024). *Large-scale stream k-means based on product-quantized codes*. Research Square. <https://doi.org/10.21203/rs.3.rs-4412715/v1>
- Hofmann, P., & Tschakert, G. (2017). Intensity- and duration-based options to regulate endurance training. *Frontiers in Physiology*, 8. <https://doi.org/10.3389/fphys.2017.00337>
- Ikotun, A. M., Ezugwu, A. E., Abualigah, L., Abuhajja, B., & Jia, H. (2022). K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. *Information Sciences*, 622, 178-210. <https://doi.org/10.1016/j.ins.2022.11.139>
- Iolascon, G., Gimigliano, F., Pietro, G. D., Moretti, A., Paoletta, M., Rivezzi, M., ..., & Piscitelli, P. (2021). Personalized paths for physical activity: Developing a person-centered quantitative function to determine a customized amount of exercise and enhancing individual commitment. *BMC Sports Science Medicine and Rehabilitation*, 13, 1. <https://doi.org/10.1186/s13102-021-00282-4>
- Jagim, A. R., Jones, M. T., Askow, A. T., Luedke, J., Erickson, J. L., Fields, J. B., & Kerksick, C. M. (2023). Sex differences in resting metabolic rate among athletes and association with body composition parameters: a follow-up investigation. *Journal of Functional Morphology and Kinesiology*, 8(3), 109. <https://doi.org/10.3390/jfkm8030109>
- Karvonen, J., & Vuorimaa, T. (1988). Heart rate and exercise intensity during sports activities. *Sports Medicine*, 5(5), 303-311. <https://doi.org/10.2165/00007256-198805050-00002>
- Kayange, H., Mun, J., Park, Y., Choi, J., & Choi, J. (2024). A hybrid approach to modeling heart rate response for personalized fitness recommendations using wearable data. *Electronics*, 13(19), 3888. <https://doi.org/10.3390/electronics13193888>
- Koman, A. M., Chamera-Cyrek, K., Pliszka, M., Janik, I., Gadżala, K., Palacz, K. A., ..., & Sadowska, I. (2024). The beneficial effects of aerobic exercise on human systems and organs: a literature review. *Journal of Education Health and Sport*, 73, 51710. <https://doi.org/10.12775/jehs.2024.73.51710>
- MacIntosh, B. R., Murias, J. M., Keir, D. A., & Weir, J. M. (2021). What is moderate to vigorous exercise intensity? *Frontiers in Physiology*, 12, 682233. <https://doi.org/10.3389/fphys.2021.682233>
- Mahato, K., Saha, T., Ding, S., Sandhu, S. S., Chang, A., & Wang, J. (2024). Hybrid multimodal wearable sensors for comprehensive health monitoring. *Nature Electronics*, 7(9), 735-748. <https://doi.org/10.1038/s41928-024-01247-4>
- Mauvais-Jarvis, F. (2015). Sex differences in metabolic homeostasis, diabetes, and obesity. *Biology of Sex Differences*, 6, 1. <https://doi.org/10.1186/s13293-015-0033-y>
- Mauvais-Jarvis, F. (2023). Sex differences in energy metabolism: natural selection, mechanisms and consequences. *Nature Reviews Nephrology*, 20(1), 56-71. <https://doi.org/10.1038/s41581-023-00781-2>

- Mazéas, A. (2023). *Development and evaluation of a digital intervention based on gamification to promote physical activity of patients with chronic diseases* [Doktora Tezi]. HAL. <https://theses.hal.science/tel-04146764>
- Mescher, T., Hacker, R. L., Martinez, L., Morris, C. D., Mishkind, M. C., & Garver-Apgar, C. E. (2024). Mobile health apps: guidance for evaluation and implementation by healthcare workers. *Journal of Technology in Behavioral Science*. <https://doi.org/10.1007/s41347-024-00441-7>
- Mifflin, M., Jeor, S. S., Hill, L., Scott, B., Daugherty, S. A., & Koh, Y. O. (1990). A new predictive equation for resting energy expenditure in healthy individuals. *American Journal of Clinical Nutrition*, 51(2), 241-247. <https://doi.org/10.1093/ajcn/51.2.241>
- Milani, J. G. P. O., Milani, M., Verboven, K., Cipriano, G., & Hansen, D. (2024). Exercise intensity prescription in cardiovascular rehabilitation: bridging the gap between best evidence and clinical practice. *Frontiers in Cardiovascular Medicine*, 11, 1380639. <https://doi.org/10.3389/fcvm.2024.1380639>
- Moz, S. H., Hosen, Md. A., Santo, Md. N. S., Kabir, Sk. S., Adnan, Md. N., & Galib, S. Md. (2023). Precision cardiometabolic: transforming cardiac care with artificial intelligence-driven dietary recommendations. *Radioelectronic and Computer Systems*, 4, 20-35. <https://doi.org/10.32620/reks.2023.4.02>
- Naveed, M., Samin, O. B., Bilal, M., & Waseem, M. (2025). IoT based health monitoring with diet, exercise and calories recommendation using machine learning. *Human-Centric Intelligent Systems*. <https://doi.org/10.1007/s44230-025-00096-4>
- Olsson, K., Rosdahl, H., & Schantz, P. (2022). Interchangeability and optimization of heart rate methods for estimating oxygen uptake in ergometer cycling, level treadmill walking and running. *BMC Medical Research Methodology*, 22(1), 55. <https://doi.org/10.1186/s12874-022-01524-w>
- Papastratis, I., Konstantinidis, D., Daras, P., & Dimitropoulos, K. (2024). AI nutrition recommendation using a deep generative model and ChatGPT. *Scientific Reports*, 14(1), 14620. <https://doi.org/10.1038/s41598-024-65438-x>
- Patel, H., Alkhwam, H., Madanieh, R., Shah, N., Kosmas, C. E., & Vittorio, T. J. (2017). Aerobic vs anaerobic exercise training effects on the cardiovascular system. *World Journal of Cardiology*, 9(2), 134-138. <https://doi.org/10.4330/wjcv.v9.i2.134>
- Pauley, A. M., Rosinger, A. Y., Savage, J. S., Conroy, D. E., & Downs, D. S. (2024). Every sip counts: understanding hydration behaviors and user-acceptability of digital tools to promote adequate intake during early and late pregnancy. *PLOS Digital Health*, 3(5). <https://doi.org/10.1371/journal.pdig.0000499>
- Pradal, C. L., Lozano-Ruiz, C., Rodríguez, J., Saigi-Rubió, F., Bach-Faig, A., Esquiús, L., ..., & Aguilar, A. (2020). Using mobile applications to increase physical activity: a systematic review. *International Journal of Environmental Research and Public Health*, 17(21), 8238. <https://doi.org/10.3390/ijerph17218238>
- Prado, N. O., Howard, K. R., Laskaridou, E., Zorrilla-Revilla, G., Reid, G. R., Marinik, E. L., ..., & Davy, K. P. (2024). Validity of predictive equations for total energy expenditure against doubly labeled water. *Scientific Reports*, 14, 1. <https://doi.org/10.1038/s41598-024-66767-7>
- Qirtas, M. M., Zafeiridi, E., White, E. B., & Pesch, D. (2024). *Evolving AI for wellness: dynamic and personalized real-time loneliness detection using passive sensing*. arXiv. <https://doi.org/10.48550/arxiv.2402.05698>
- Rabbi, M., Pfammatter, A. F., Zhang, M., Spring, B., & Choudhury, T. (2015). Automated personalized feedback for physical activity and dietary behavior change with mobile phones: a randomized controlled trial on adults. *JMIR Mhealth and Uhealth*, 3(2). <https://doi.org/10.2196/mhealth.4160>
- Ramakrishnan, R., Xing, T., Chen, T., Lee, M.-H., & Gao, J. (2023). *Application of AI in nutrition*. arXiv. <https://doi.org/10.48550/arxiv.2312.11569>
- Rauba, P., Seedat, N., Kacprzyk, K., & Schaar, M. van der. (2024). *Self-healing machine learning: a framework for autonomous adaptation in real-world environments*. arXiv. <https://doi.org/10.48550/arxiv.2411.00186>
- Reeves, B., Carter, B., Roberson, L., & Jordan, D. G. (2023). Comparison of two reminder interventions to achieve adequate water intake and hydration in women: a pilot study. *Journal of Exercise and Nutrition*, 6, 1. <https://doi.org/10.53520/jen2023.103142>
- Ridwan, E. S., Ahmad, O., & Ali, Z. M. (2025). Technology strategies in health promotion: preventive lifestyle interventions to reduce the burden of disease. *Journal of World Future Medicine Health and Nursing*, 3(1), 86. <https://doi.org/10.70177/health.v3i1.1905>
- Rivera, R. O., Gabarrón, E., Roperio, J., & Denecke, K. (2023). Designing personalised mHealth solutions: an overview. *Journal of Biomedical Informatics*, 146, 104500. <https://doi.org/10.1016/j.jbi.2023.104500>
- Rospo, G., Valsecchi, V., Bonomi, A., Thomassen, I. W., Dantzig, S. van, Torre, A. L., & Sartor, F. (2016). Cardiorespiratory improvements achieved by American College of Sports Medicine's exercise prescription implemented on a mobile app. *JMIR Mhealth and Uhealth*, 4, 2. <https://doi.org/10.2196/mhealth.5518>
- Salminen, J., Mustak, M., Sufyan, M., & Jansen, B. J. (2023). How can algorithms help in segmenting users and customers? A systematic review and research agenda for algorithmic customer segmentation. *Journal of Marketing Analytics*, 11(4), 677-698. <https://doi.org/10.1057/s41270-023-00235-5>
- Sanchez, B. N., Volek, J. S., Kraemer, W. J., Sáenz, C., & Maresh, C. M. (2024). Sex differences in energy metabolism: a female-oriented discussion. *Sports Medicine*, 54(8), 2033-2045. <https://doi.org/10.1007/s40279-024-02063-8>
- Secara, I.-A., & Hordiuik, D. (2024). Personalized health monitoring systems: integrating wearable and AI. *Journal of Intelligent Learning Systems and Applications*, 16(2), 44. <https://doi.org/10.4236/jilsa.2024.162004>
- Serantoni, C., Zimatore, G., Bianchetti, G., Abeltino, A., Spirito, M. D., & Maulucci, G. (2022). Unsupervised clustering of heartbeat dynamics allows for real time and personalized improvement in cardiovascular fitness. *Sensors*, 22(11), 3974. <https://doi.org/10.3390/s22113974>
- Smith, K., Ward, T., Lambe, S., Ostinelli, E. G., Blease, C., Gant, T., ..., & Cipriani, A. (2025). Engagement and attrition in digital mental health: current challenges and potential solutions. *Npj Digital Medicine*, 8(1), 398. <https://doi.org/10.1038/s41746-025-01778-w>
- Subramaniaswamy, V., Vijayakumar, V., Srinivasan, D., Balaganesh, V., Damerla, S. B., Bhuvaneshwari, S., & Ravi, L. (2022). Dynamic physical activity recommendation delivered through a mobile fitness app: a deep learning approach. *Axioms*, 11(7), 346. <https://doi.org/10.3390/axioms11070346>
- Susaiyah, A., Härmä, A., Reiter, E., Balloccu, S., & Petković, M. (2024). *Feedback-driven insight generation and recommendation for health self-management*. Research Square. <https://doi.org/10.21203/rs.3.rs-4016799/v1>
- Taylor, J. L., Bonikowske, A. R., & Olson, T. P. (2021). Optimizing outcomes in cardiac rehabilitation: the importance of exercise intensity. *Frontiers in Cardiovascular Medicine*, 8. <https://doi.org/10.3389/fcvm.2021.734278>
- Tomlinson, M., Rotheram-Borus, M. J., Swartz, L., & Tsai, A. C. (2013). Scaling up mHealth: where is the evidence? *PLoS Medicine*, 10(2). <https://doi.org/10.1371/journal.pmed.1001382>
- Vara, N., Mirzabeigi, M., Sotudeh, H., et al. (2022). Application of k-means clustering algorithm to improve effectiveness of the results recommended by journal recommender system. *Scientometrics*, 127, 3237-3252. <https://doi.org/10.1007/s11192-022-04397-4>
- Vardakas, G., Papakostas, I., & Likas, A. (2024). *Deep clustering using the soft silhouette score: Towards compact and well-separated clusters*. arXiv. <https://doi.org/10.48550/arxiv.2402.00608>
- Weidlinger, S., Winterberger, K., Pape, J., Weidlinger, M., Janka, H., Wolff, M. von, & Stute, P. (2023). Impact of estrogens on resting energy expenditure: a systematic review. *Obesity Reviews*, 24(10), e13605. <https://doi.org/10.1111/obr.13605>
- Wiryonoputro, T. N., & Saputri, T. R. D. (2023). Rancang bangun aplikasi untuk ibu menyusui pasca persalinan dengan algoritma Mifflin-St Jeor. *Jurnal Informatika Jurnal Pengembangan IT*, 8(3), 281. <https://doi.org/10.30591/jpit.v8i3.5733>
- Yun, T., Yang, E., Safdari, M., Lee, J., Kumar, V., Mahdavi, S. S., ..., & Matarić, M. J. (2025). *Sleepless nights, sugary days: Creating synthetic users with health conditions for realistic coaching agent interactions*. arXiv. <https://doi.org/10.48550/arxiv.2501.00000>
- Zahedani, A. D., McLaughlin, T., Veluvali, A., Aghaeepour, N., Hosseinian, A., Agarwal, S., ..., & Snyder, M. (2023). Digital health application integrating wearable data and behavioral patterns improves metabolic health. *Npj Digital Medicine*, 6(1), 216. <https://doi.org/10.1038/s41746-023-00956-y>
- Zahra, S., Ghazanfar, M. A., Khalid, A., Azam, M. A., Naeem, U., & Prügel-Bennett, A. (2015). Novel centroid selection approaches for KMeans-clustering based recommender systems. *Information Sciences*, 320, 156-189. <https://doi.org/10.1016/j.ins.2015.03.062>
- Zhang, X., Chen, H., Chen, J., Feng, H., Liu, M., Zhang, X., ..., & Chen, F. (2025). A hybrid machine learning-enhanced MCDM model for transport safety engineering. *Scientific Reports*, 15, 1. <https://doi.org/10.1038/s41598-025-21297-8>
- Zhu, J., Dallal, D. H., Gray, R. C., Villareale, J., Ontañón, S., Forman, E. M., & Arigo, D. (2021). Personalization paradox in behavior change apps. *Proceedings of the ACM on Human-Computer Interaction*, 5, 1-25. <https://doi.org/10.1145/3449190>

Investigation of fine-tuned BERT models for sentiment analysis in COVID-19 tweets using a fuzzy logic-based ensemble approach

 Mustafa Sefa Evgin*,  Sinan Toklu

Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Turkiye

Cite this article as: Evgin, M. S., & Toklu, S. (2026). Investigation of fine-tuned BERT models for sentiment analysis in COVID-19 tweets using a fuzzy logic-based ensemble approach. *J Comp Electr Electron Eng Sci*, 4(1), 17-26.

Received: 19.01.2026

Accepted: 24.02.2026

Published: 25.04.2026

ABSTRACT

Aims: With the beginning of the COVID-19 pandemic, social media applications such as twitter used more than usual because people started to work at their homes rather than offices. Thus, data on this application has become more important to manage crisis of COVID-19. While conventional deep learning methods have shown success in sentiment analysis, they often encounter challenges in capturing the inherent semantic ambiguity and informal linguistic structures prevalent on social media platforms. To find ambiguity on these texts propose ensemble model enhanced by fuzzy logic designed to improve sensitivity and capability.

Methods: Architecture uses BERT model, fine-tuned for specific data to supply dynamic attributes for MLP, LSTM and BiLSTM elements. Their shared executive is regulated via Mamdani Fuzzy Interference System. Then dynamic weights results from defuzzification after mapping prediction of confidence and validation accuracy value on 7 level fine-grained rule set.

Results: Experiment performed on Kaggle Corona NLP dataset resulted in 91.28% accuracy, 91.23% F1 score and 91.38% precision. System's robust performance is demonstrated by Mean Square Error of 0.2301.

Conclusion: Relative analysis demonstrates dominance of this approach against traditional models. Fuzzy Ensemble model proposes more trustworthy solution for obstruse tweets with successfully straining noise and dealing semantic uncertainties which is naturally present in social media data.

Keywords: Sentiment analysis, BERT, fuzzy logic, ensemble learning, COVID-19, natural language processing

INTRODUCTION

Natural language processing has appreciated as need to interpret of unstructured textual information raised. During the COVID-19 pandemic, social media platforms provided a basis for expressing their concerns and psychological situations thus examination of this data become crucial (Singh et al., 2022). Accurate information on this data is essential for politician for strategic perspective. Though current literature confirms the effectiveness of BERT (Devlin et al., 2019) and LSTM (Hochreiter & Schmidhuber, 1997) frameworks, semantic ambiguity and noise are still among the biggest problems on social media data (Dhanalakshmi et al., 2024).

Yet, human language is intrinsically filled with ambiguity and uncertainty. Standard algorithms relying on 'crisp' logic often struggle here, as they try to force text into rigid 'positive' or 'negative' boxes. This binary approach inevitably ignores the nuanced gray areas that are fundamental to natural communication. To mathematically model this type of uncertainty, current literature advocates for the integration of fuzzy theories into natural language processing (Howell & Ertugan, 2017). This approach proves particularly superior in high-stakes fields like medical diagnosis. In such critical contexts, text-based systems enhanced with fuzzy logic have

been shown to deliver significantly more precise and reliable outcomes than standard NLP methods (Omogrebe et al., 2020). In this context, it is widely accepted in the literature that hybrid and fuzzy-based approaches offer a more stable architecture compared to individual deep learning models, especially in datasets where sentiment transitions are not sharp (Sherin et al., 2025). Furthermore, the creation of optimized and compressed representations of texts based on information theory has entered the literature as a new approach to enhancing the performance of machine learning models (Kale et al., 2024).

When the literature is examined, it is observed that a large portion of research in the field of COVID-19 sentiment analysis relies on standard deep learning architectures such as LSTM or CNN, as seen in the study (Karaca & Aslan, 2021). Although these models produce successful results, hybrid approaches in which modern transformer architectures like BERT are used within a Fuzzy Ensemble structure are limited. Even though hybrid studies on COVID-19 tweet datasets have increased recently, gaps still exist. For instance, in a study where sampling methods were employed to address imbalance in the dataset (Kumar et al., 2024), an accuracy rate of 89.00%

Corresponding Author: Mustafa Sefa Evgin, 24833301023@gazi.edu.tr



This work is licensed under a Creative Commons Attribution 4.0 International License.

was achieved by combining BERT and CNN models. Another study conducted on same dataset remained at the level of 86% while using hybrid model. Conversely, research employing conventional deep learning layers like BiLSTM (Schuster & Paliwal, 1997) and GRU saw performance plateau at an 85% accuracy rate (Shahriar & Sarker, 2023). These findings serve as implicit evidence that shifting towards Transformer-based architectures is essential for surpassing this threshold.

Nevertheless, a common limitation in prior research is the reliance on static techniques to handle model uncertainty. In contrast, fuzzy logic has emerged as a superior alternative, particularly for complex tasks like detecting rare textual events (Arslan et al., 2021) where classical methods struggle. Consequently, our work seeks to bridge this gap by integrating fine-tuned BERT architectures with dynamic fuzzy logic weighting.

To handle with challenges, we propose ensemble framework that integrates a fine-tuned BERT with 7-level Mamdani-type fuzzy inference system. Our model starts with fine-tuning on BERT architecture on COVID-19 specific dataset for capturing context dependent shift in vocabulary. Then contextual representation is processed by three parallel individual base learner MLP (Rumelhart et al., 1986), LSTM and BiLSTM for extract diverse architectural features. Sentiment decisions are conducted via dynamic fuzzy weighting mechanism which evaluates both model confidence and validation accuracy. Our work have three contribution: Firstly fined-grained, 7 level fuzzy logic module provides better sensitivity over traditional 3-level systems; secondly dynamic 'reliability filter' mitigates noise with penalizing high confidence but formerly inaccurate prediction; lastly model achieves better performance leap which surpasses soft, hard voting and recent hybrid benchmarks in COVID-19 sentiment analysis domain.

RELATED WORK

This section provides inclusive examinations of theoretical keystone to support our search: deep learning-based sentiment analysis, transformer driven hybrid architectures and the useful application of fuzzy logic.

Deep Learning and Hybrid Approaches

Over the years recurrent neural networks (RNNs) and machine learning method have served as headstone for sentiment analysis, operating as typical tools for mapping text into vector space and classifying. Comprehensive literature reviews in this area emphasize that deep learning models' performance performs quietly better than conventional models for text classification (Miaee et al, 2021). One of the examples for it, analyzing of children's stories where feature engineering integrated with Random Forest algorithms produced notably more steady result than traditional lexicon-based approaches (Bilal et al., 2023). Similarly, several studies confirmed the benefit of weight updated classifiers for text-based sentiment analysis across variety of datasets (Bilal et al., 2024). Furthermore, research emphasize that bringing Deep Belief Network with feature selection considerably improves classification accuracy, mainly by alleviating noise essential in high-dimensional data (Ruangkanokmas et al., 2016).

Transformer-Based (BERT, RoBERTa) Models

As the BERT model became important prominent, researchers focused on merging BERT with other deep learning models

more than before. A prominent purpose includes combining BERT with CNNs for long text classification. Thus, hybrid attitude supports model to instantaneously catch local features by the CNN while keeping global context provided by BERT (Chen et al., 2022).

Similar hybrid approach used to evaluate investor sentiment in the energy sector. While channeling BERT outputs directly to BiLSTM model, this architecture performed significant proficiency in finding the progressive dependencies which essential in financial literature (Cai et al., 2020). Recent study finds a triple hybrid model by implementing RoBERTa (Liu et al, 2019) with BiLSTM and CNN layers to examine ceramic product comments. Results shows that this combined structure achieved considerably higher accuracy when compared individual models operating alone (Yang et al., 2025). Furthermore, research on COVID-19 tweets revealed that combining BERT with LSTM model resulted in better outcome compared with conventional methods (Dhanalakshmi et al., 2024). This hybrid integration showed considerably more efficient than conventional models in catching nuances in the dataset. Similarly, comprehensive research shows that architectures combining BERT with CNN and BiLSTM layers notably exceed conventional Word2Vec-based techniques, achieving better results both accuracy and F1 scores (Bello et al., 2023).

While BERT models show extraordinary potential in contextual learning, current studies emphasize a critical point; they mostly need additional optimization layer. This additional optimization layer is critical for effectively managing high noise and semantic discrepancies which occurs commonly faced in raw data. (Elgabry & Hamdi, 2025).

METHODS

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data. All procedures were carried out in accordance with the ethical rules and principles.

In this section, we present technical infrastructure and methodological details of proposed hybrid model which aimed performing sentiment analysis of COVID-19 tweets with higher accuracy, F1 score and minimized mean square error. Methodology begins with dataset preparation and preprocessing then fine-tuned BERT model used for contextual text representation and parallel deep learning models MLP, LSTM and BiLSTM processing respectively. Finally, probabilities which generated from respectively models are combining via Mamdani-type fuzzy logic system and it offers decision mechanism closer to human thinking and reasoning system than conventional system. Fuzzy system then elaborates principles of proposed ensemble model.

All experimental processes and model training phases performed on Google Colab Pro cloud computing platform. In this platform we used NVIDIA A100 Tensor Core GPU with 40GB of VRAM and High-RAM, approximately 25GB. Software environment configured with Python 3.10 and TensorFlow 2.x and KerasNLP library. Individual models (BERT+MLP, BERT+LSTM, BERT+BiLSTM) have comparable model complexities, ranging from 109.5 million to 109.9 million trainable parameters, with BiLSTM variants displaying highest computational cost during training (869.28 second). In terms of interference latency, deep learning

methods shows high efficiency with an average processing time of approximately 2.00 ms per sample. Rule-based fuzzy logic module performs computational overhead of 22.67 ms per sample due to CPI-bound defuzzification but aggregate systems stay approximately 22.47 ms per second. That means our model processing product roughly 40 tweets per second and it proves that proposed ensemble model is computationally viable for real time social media monitoring application despite increased architectural complexity.

Dataset

In this study, the ‘‘Corona NLP’’ dataset obtained from the Kaggle platform was utilized. Due to its structure harboring varying sentiment intensities regarding COVID-19, this dataset is also preferred as a primary data source in recent studies where language models are tested within the framework of information theory (e.g., in the TexShape architecture) (Kale et al., 2024). The dataset contains tweets posted about COVID-19 and their sentiment labels (positive, negative, neutral). A total of 41,157 training and 3,798 test data points were used.

Training dataset includes 15398 negative, 7713 neutral and 18046 positive tweets. Test dataset includes 1633 negative, 619 neutral and 1546 positive tweets.

Data preprocessing: The following steps were applied to clean the noise from the raw tweet data:

- **Cleaning:** URLs, HTML tags, and ‘‘@user’’ expressions were removed using Regex.
- **Normalization:** All texts were converted to lowercase, and non-ASCII characters were filtered out.
- **Tokenization:** Texts were converted into numerical vectors (Input IDs and Attention Masks) using the bert-base-uncased tokenizer, in accordance with the BERT model. The maximum sequence length was determined as 60.

Proposed Model Architecture: Fuzzy-BERT Ensemble

In this study, a hierarchical ensemble architecture integrating BERT-based contextual feature extraction with a fuzzy logic-based decision fusion mechanism has been developed to detect the sentiment status of COVID-19 tweets. This architecture, the detailed flowchart of which is given in **Figure 1**, fundamentally consists of three main modules: (1) Contextual feature extraction (BERT), (2) Parallel base learners, and (3) Fuzzy decision fusion module. The architecture begins with the retraining of all layers of the pre-trained BERT-base-uncased model as trainable=True. Contrary to the ‘‘frozen weights’’ approach common in the literature, the model was subjected to a fine-tuning process to more accurately represent words such as ‘‘quarantine,’’ ‘‘mask,’’ and ‘‘symptom,’’ whose meanings shifted within the COVID-19 context. Contextual features obtained from the BERT layer were transferred to three parallel sub-models (BERT+MLP, BERT+LSTM, and BERT+BiLSTM) to approach the data from different perspectives. Thus, dense, sequential, and bidirectional information processing capabilities were combined within the same architecture.

In combining the prediction probabilities produced by these sub-models, a Mamdani-type fuzzy inference system (FIS) was used instead of classical weighted average methods. In this system, the effect of each model on the final decision depends not on fixed coefficients, but on dynamic variables

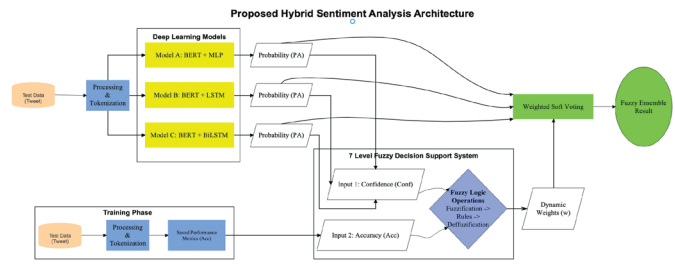


Figure 1. Proposed 7-level (fine-grained) Mamdani fuzzy logic module contextual feature extraction (fine-tuned BERT)
BERT: Bidirectional encoder representations from transformers

such as ‘‘model confidence’’ and ‘‘model accuracy.’’ Triangular Membership Functions (Trimf), shown in Equation (1), were preferred for the fuzzification of these variables due to computational efficiency (Jang et al, 1997):

$$\mu_A(x) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right) \tag{1}$$

In Equation (1) *a*, *b*, and *c* represent the corner points of the triangular membership function. Numerical values are converted into linguistic variables such as ‘‘low,’’ ‘‘medium,’’ and ‘‘high’’ through these functions, and IF-THEN rules based on expert opinion are activated.

In the final stage of the inference mechanism, the fuzzy set formed by the triggered rules needs to be converted into a crisp numerical weight value (*W_i*). For this operation, the centroid (center of gravity) method given in Equation (2) was used in the Defuzzification stage (Mendel, 1995):

$$W_i = \frac{\int \mu_C(z) \cdot z \, dz}{\int \mu_C(z) \, dz} \tag{2}$$

Here, $\mu_C(z)$ denotes the membership value of the aggregated fuzzy set. These calculated weights are combined with the prediction *P_i* of each sub-model to obtain the final output of the ensemble architecture. Thus, both the strong representation capacity of contextual deep learning models and the flexible decision mechanism of fuzzy logic are integrated within a single architecture.

In contrast to traditional Word2Vec or GloVe methods, the BERT (bidirectional encoder representations from transformers) model was employed on the preprocessed tweet texts at the input layer of the architecture to capture the context-dependent meanings of words based on their position within the sentence.

BERT was specifically selected as the primary feature extractor due to its superior bidirectional context-capture capabilities and its proven robustness as a benchmark in recent COVID-19 literature, which allows for a more focused evaluation of the proposed fuzzy ensemble’s impact on handling sentiment uncertainty.

At this stage, all layers of the bert-base-uncased model were set to trainable, and the model was subjected to a fine-tuning process on the COVID-19 dataset; this ensured that the model could distinguish between medical terms with negative connotations and their usage in daily language. In this process, where high-density contextual vectors of dimension (N, 768) were generated for each tweet, the training of the model was carried out using the Adam optimizer. To prevent overfitting, a Dropout rate of 0.3 was applied; furthermore, the training parameters were set as a learning rate of 2e-5, a batch size of 16, and 4 epochs.

Parallel Base Learners

Dynamic vectors which obtained from BERT instantaneously feeds three distinct parallel subordinate model (MLP, LSTM and BiLSTM) to capture semantic dimension of data and utilize the principle of ‘architectural diversity’. While the MLP component of this hybrid design models features through dense and non-linear transformations, it has been observed in the deep learning literature that such hybrid utilization of different architectural paradigms (such as Transformer and MLP) yields significant improvement compared to single-type architectures (Bashar, 2025). The BERT+MLP model was trained for 4 epochs using the Adam optimizer and a learning rate of 2×10^{-5} , in accordance with the parameters specified in **Table 1**.

Parameter	Value
Learning rate	2×10^{-5}
Epoch	4
Batch size	16
Optimizer	Adam
Hidden layer units	128
Hidden activation	ReLU
Dropout rate	0.3
Output activation	Softmax

BERT: Bidirectional encoder representations from transformers, MLP: Multi-layer perceptron, ReLU: Rectified linear units

Parameter	Value
Learning rate	2×10^{-5}
Epoch	4
Batch size	16
Optimizer	Adam
LSTM units	64
Dropout rate	0.3
Output activation	Softmax
Loss function	Sparse categorical cross entropy

BERT: Bidirectional encoder representations from transformers, LSTM: Long short-term memories

Parameter	Value
Learning rate	2×10^{-5}
Epoch	4
Batch size	16
Optimizer	Adam
BiLSTM units	64
Dropout rate	0.3
Output activation	Softmax
Loss function	Sparse categorical crossentropy

BERT: Bidirectional encoder representations from transformers, BiLSTM: Bidirectional long short-term memories

Capturing context in text classification by protecting progressive relationship and forwarding flow of words within the text through memory cells, contrasting conventional models (Airlangga, 2024). BERT+LSTM model was trained 4 epochs with one recurrent layer, 64 neurons, using the Adam optimizer with a learning rate of 2×10^{-5} , based on the parameters presented in **Table 2**.

BiLSTM last branch of parallel base learners integrated in the system but has different features when compared with LSTM. BiLSTM scans text in both forward and backwards directions, so learning the future context of the word; this reduces ‘vanishing gradient’ problem and gives more stable results for maintaining semantic integrity, especially for long sentences (Rahman, 2025). Correspondingly BERT+BiLSTM model is trained for 4 epoch, one recurrent layer and total 128 neuron 64 for the forward pass and 64 for the backward pass with the configuration setting in the **Table 3**, while employing Adam optimizer and with learning rate of 2×10^{-5} .

Each sub-model generates its own probability distribution (P_{model}) regarding the class of the tweet (positive, negative, neutral).

Fuzzy Decision Fusion Module

In this study, a Mamdani-type fuzzy inference system (FIS) has been designed to optimally combine the predictions of the individual models (BERT+MLP, BERT+LSTM, BERT+BiLSTM). Proposed system essentially consists of three sub-layer and its decision mechanism close to human thinking and reasoning system by transforming input values into linguistic variables.

Fuzzification

Normalized triangular membership functions (Trimf) were used between the range of [0,1] for input and output variables of the system. The reason for choosing normalized triangular membership function, it able to reduce quantities cost due to their linear features also offers to quick responses for real time application. The variables expressed in the system with three main component which shown in **Figure 2**.

The first input which is labeled ‘model confidence’ and model confidence is equal to peak SoftMax probability score originated from model’s specific prediction. Divided from conventional three-layer classification, we charted this variable

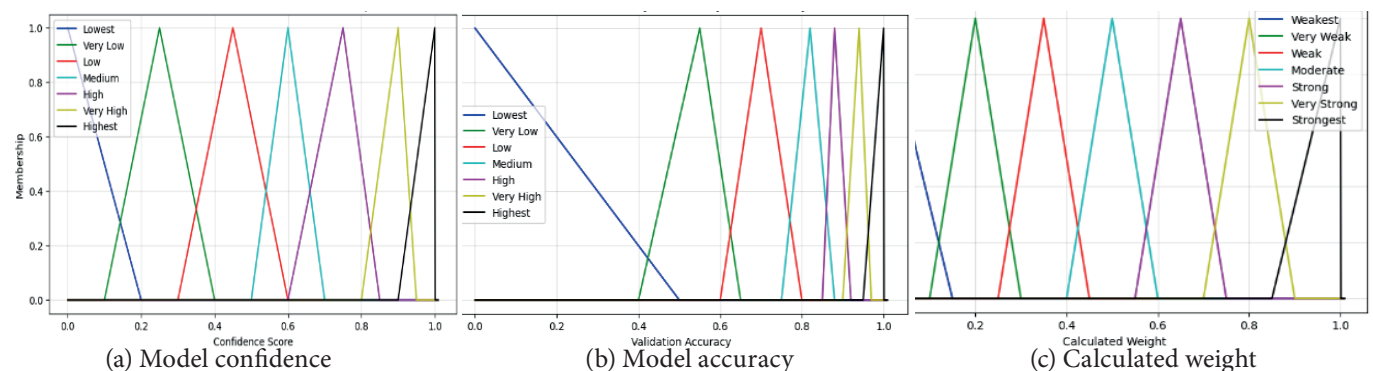


Figure 2. Membership functions of the proposed fuzzy logic system

onto more elaborate seven level scale varying from ‘lowest’ to ‘highest’ for drastically raise the system’s performance.

Model accuracy is a second input, and it reflect model’s collective performance for validation dataset. Reflecting approach taken with confidence scores, we also charted accuracy onto seven level granular scale. Detailed segmentation is designated for handle the model’s uncertainty with higher precision, especially addressing the crucial ambiguity often found in between 50% and 60%.

‘Weight’ decides how much effect a specific model holds over ensemble’s final decision and acts as a final output of our inference system. Dependable with our input variables, this output is categorized into seven-layer spectrum that ranges from ‘weak’ to ‘strong’.

After fuzzification system assesses linguistic terms while using ‘IF-THEN’ conditional statement by domain experts. In harmony with improved 7 level granularity, we used a set of 12 precise rules to rule decision making process. This inclusive rule base provides for dual purpose first it suppresses underperforming models and simultaneously refining weight allocation for nuanced scenarios, such as instances of ‘high accuracy’ and ‘medium confidence’.

Model’s final decision weight is stemmed from the collaboration between calculated input variables and fuzzy logic inference mechanism. Inclusive analysis of rule base and main decision logic chart exhibited in **Table 4**. In this regard, linguistic terms equal to seven tier fuzzy spectrum, expressed from L1 (lowest) throughout L7 (highest).

Table 4. Rule base of the Mamdani fuzzy inference system

Rule no	Input 1: Confidence	Operator	Input 2: Accuracy	Output: Weight
1	l7 (highest)	AND	l7 (perfect)	l7 (strongest)
2	l6 (very high)	AND	l7 (perfect)	l7 (strongest)
3	l5 (high)	AND	l7 (perfect)	l6 (very strong)
4	l7 (highest)	AND	l6 (very good)	l7 (strongest)
5	l5 (high)	AND	l6 (very good)	l6 (very strong)
6	l4 (medium)	AND	l6 (very good)	l5 (strong)
7	l7 (highest)	AND	l5 (good)	l6 (very strong)
8	l4 (medium)	AND	l5 good)	l4 (moderate)
9	l7 (highest)	AND	l4 (average)	l5 (strong)
10	l3 (low)	OR	l3 (low)	l3 (weak)
11	-	-	l2 (very low)	l2 (very weak)
12	l1 (lowest)	OR	l1 (lowest)	l1 (weakest)

Standard protocols (rules 1-9) evaluate both inputs together by the ‘AND’ operator. However, to reinforce system reliability, we applied a distinct strategy for low-performance scenarios.

Especially, rules 10 and 12 uses ‘OR’ operator as fool proofed. This ensures that if accuracy or instantaneous confidence drops below a critical level, model’s effects is automatically limited. With doing this system effectively filters out noise and avoiding unstable interpreters from crooking the final ensemble. Instead of 49 rules, we generated 12 rules for the reason decrease calculation cost and focus on only critic points.

Defuzzification

Defuzzification is a process that converting fuzzy output into clear numerical value with using rules. In this study we used centroid (center of gravity) which is recognized as one of the most common and reliable method in fuzzy inference system. This method calculates geometric center of the area which is formed by all activated members functions and its enabling system to generate precise weight value. Centroid method considers contribution of all active rules, so allowing for smoother and continuous transition in output value. Obtained clear value is assigned as a final constant of the relevant deep learning model in voting process.

RESULTS

In this section experimental results of 7 level fuzzy logic ensemble architecture which is developed for sentiment analysis for COVID-19 tweets with ‘Corona NLP’ dataset are presented in detail. To measure proposed method’s efficiency and resistance against noise we used Accuracy, F1-score, precision, recall and mean square error (MSE) as principal performance sings. Resulting data then subjected to a three-stage analysis for ensuring a multidimensional evaluation of model’s reliability. First performance differences among proposed model, individual models and conventional ensemble models were examined throughout comparative analysis (ablation study). Then classification behaviors of sub-models (BERT+MLP, BERT+LSTM, BERT+ BiLSTM) and ensemble structures were visualized by confusion matrices and (reciever operating characteristic) ROC curves. Finally, case study was performed to test context sensitivity of model beyond arithmetical data.

Ablation Study and Comparative Analysis

To show the efficiency of proposed Fuzzy Logic-based ensemble architecture (fuzzy ensemble), comparative analysis such as ablation study were carried out not only for fine-tuned BERT models but also unweight soft voting and hard voting methods because they are widely used in the literature and **Table 5** provides detailed information of performance metrics for all evaluated methods.

A review of **Table 5** identifies BERT+LSTM as the extraordinary performer among the standalone architectures, obtaining an accuracy of 90.20%. However, the data shows a considerable

Table 5. Performance benchmarking of the proposed framework against standard ensemble and individual models

Model architecture	Model type	Accuracy	F1-score	Precision	Recall	MSE
BERT+MLP	Individual model	89.86%	89.80%	89.86%	89.86%	0.2569
BERT+LSTM	Individual model	90.20%	90.13%	90.25%	90.20%	0.2527
BERT+BiLSTM	Individual model (Best)	89.81%	89.74%	89.90%	89.81%	0.2598
Hard voting	Standard ensemble	90.96%	90.91%	91.05%	90.96%	0.2340
Soft voting	Standard ensemble	91.12%	91.07%	91.22%	91.12%	0.2348
Fuzzy ensemble	Proposed method	91.28%	91.23%	91.38%	91.28%	0.2301

MSE: Mean squared error, BERT: Bidirectional encoder representations from transformers, MLP: Multi-layer perceptron, LSTM: Long short-term memories, BiLSTM: Bidirectional long short-term memories

leap in predictive capability upon the implementation of ensemble strategies. The proposed 7-level fuzzy ensemble framework exceeded both standalone models and the conventional soft voting approach, obtaining an excellent accuracy of 91.28%. McNemar testing at this stage emphasize a specific improvement; the proposed framework successfully corrected 6 complex instances that the soft voting approach had previously misclassified. While the numerical variance might seem slight at first glance, the fundamental metrics reveal a substantial advantage. Proposed methods prove it reliability with boosting precision to 91.38% and reduced mean squared error (MSE) to 0.2301. These figures confirm that 7 level structure overtakes at filtering out false positives, indicating a system that is both accurate and inherently robust against data noise.

Table 6 shows our approach has considerable performance gain and outperformed the benchmark baseline by approximately 15.73% in terms of F1 score. Our approach

also performed better accuracy, F1 score, Precision and Recall against Benchmark research.

Visual Performance Analysis of Sub-Models (CM, ROC, and MSE)

Supplementing the aggregate metrics in Table 5, we scrutinized the confusion matrices (CM) and ROC curves for each sub-architecture (BERT+MLP, BERT+LSTM, and BERT+BiLSTM). This granular analysis is necessary for expose the specific class-based discrimination capabilities of the models and to understand the synergy that allows them to complement one another.

A relative review of the matrices in Figure 3-5 shows that each architecture develops a unique proximity for specific sentiment classes. Especially, the BERT+MLP model (Figure 3a) proved most effective for the ‘positive’ category, securing 1428 correct predictions, while correctly identifying 1490 instances in the ‘negative’ class. In contrast, the BERT+BiLSTM model

Method	Accuracy	F1 score	Precision	Recall	MSE
LSTM + Word2Vec (Karaca & Aslan, 2021)	-	75.50%	88.65%	-	-
BERT + CNN (Kumar et al., 2024)	89%	90%	91%	90%	-
Hybrid model (Shahriar & Sarker, 2025)	86%	86%	86%	86%	-
Fuzzy BERT ensemble	91.28%	91.23%	91.38%	91.28%	0.2301

MSE: Mean squared error, LSTM: Long short-term memories, BERT: Bidirectional encoder representations from transformers

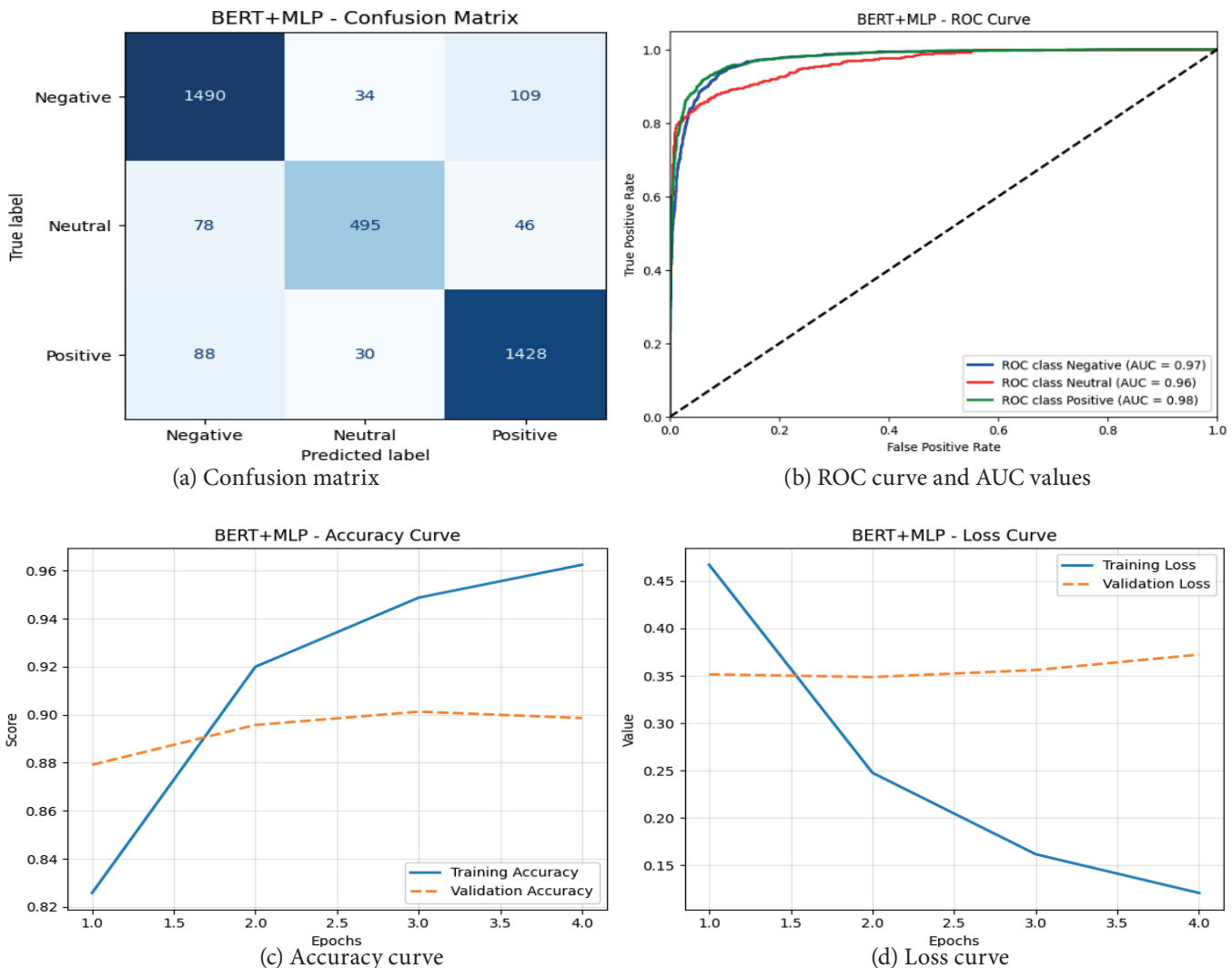


Figure 3. BERT+MLP performance analysis

BERT: Bidirectional encoder representations from transformers, MLP: Multi-layer perceptron, ROC: Receiver operating characteristic, AUC: Area under curve

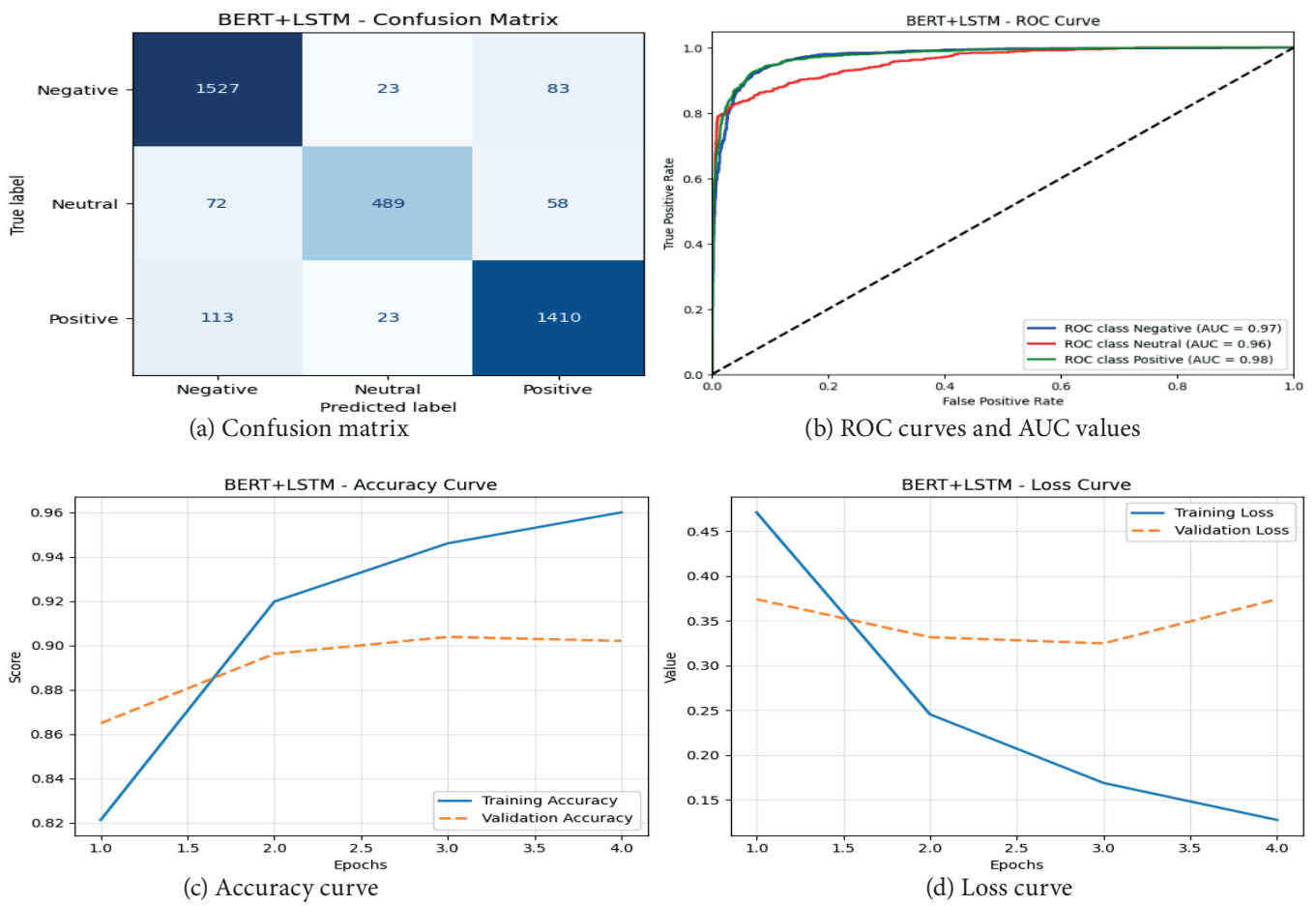


Figure 4. BERT+LSTM performance analysis

BERT: Bidirectional encoder representations from transformers, LSTM: Long short-term memories, AUC: Area under curve, ROC: Receiver operating characteristic

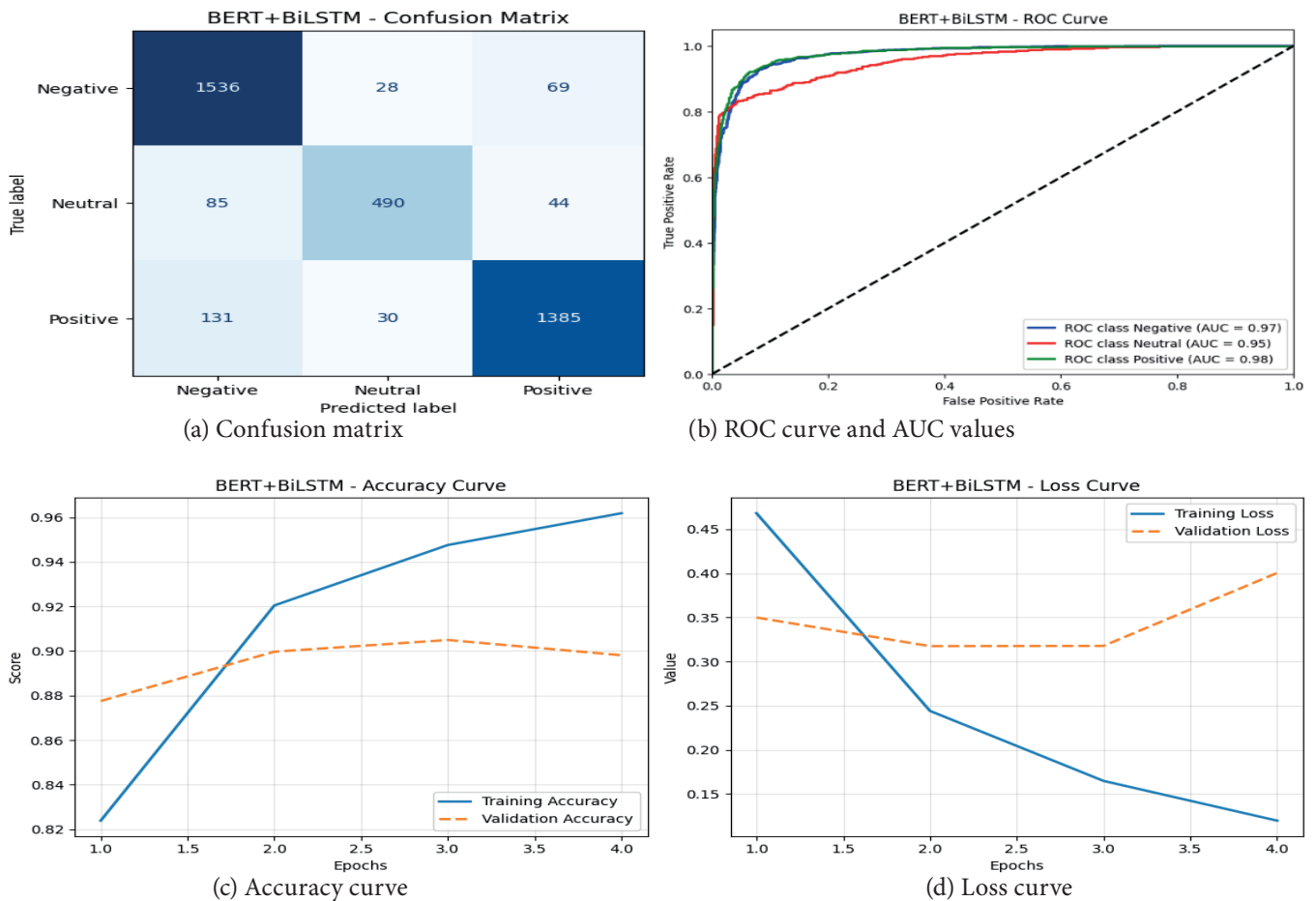


Figure 5. BERT+BiLSTM performance analysis

BERT: Bidirectional encoder representations from transformers, BiLSTM: Bidirectional long short-term memories, ROC: Receiver operating characteristic, AUC: Area under curve

(Figure 5a) exhibits a distinct advantage in identifying ‘negative’ sentiments. With 1536 correct predictions, this architecture demonstrated significantly superior to all competing models in this specific category. However, this same model adopted a more conservative attitude toward the ‘positive’ class, recording 1385 correct predictions covering the MLP variant in this specific regard. In contrast, the BERT+LSTM model (Figure 4a) bridged the gap between these two extremes, exhibiting a balanced performance profile with 1527 correct identifications for negative cases and 1410 for positive ones.

These findings confirm the study’s major hypothesis; though solitary models tend to exhibit blind spot toward specific classes, the ensemble framework succeeds in exploiting this heterogeneity. Moreover, the examination of ROC curves of all sub-model (Figures 3b, 4b and 5b) demonstrated that the area under curve (AUC) continuously decreases in range between 0.95 and 0.98 yet it does not surpass them indicative of state-of-the-art learning ability of fundamental architectures.

Comparative Analysis of Ensemble Methods

To show performance improvement of our model over soft voting method we used McNemar’s statical test. Test shows p-value is 0.1094. Result is slight above standard significance threshold of $\alpha=0.05$.

Figures 6-8 help direct relative analysis between the proposed fuzzy logic framework and traditional ensemble techniques like hard and soft voting. These visuals clearly explain the performance advantages of our approach over standard methods.

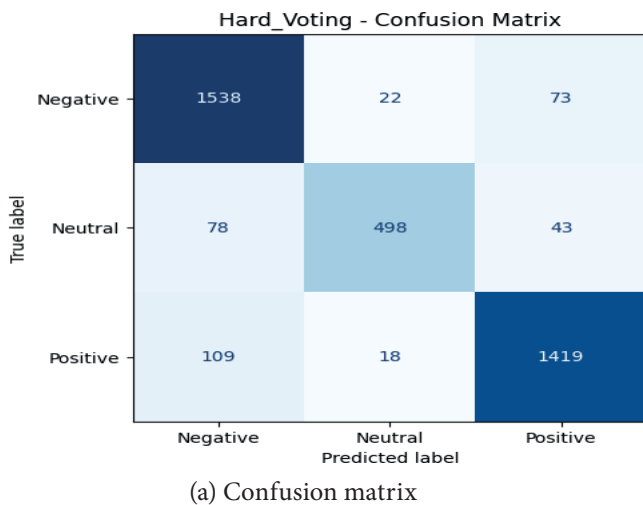


Figure 6. Hard voting

Figure 6 shows that hard voting has 1538 correct prediction for ‘negative’ class and 1419 for ‘positive’. Significantly, because hard voting disregards confidence scores, it introduces hardness that fails to accommodate borderline samples situated along critical decision boundaries.

Soft voting performed slightly better than hard voting while raising positive prediction count to 1418. Nevertheless, our proposed fuzzy ensemble model (Figure 8a) performed better than others with 1420 positive, 1543 negative and 504 neutral correct predictions. It is look like one digit increase but +1 gain serves as a quantitative proof. This represents irony tweet previously misclassified by Soft Voting that our Fuzzy Logic model rules successfully classified it on their correct category.

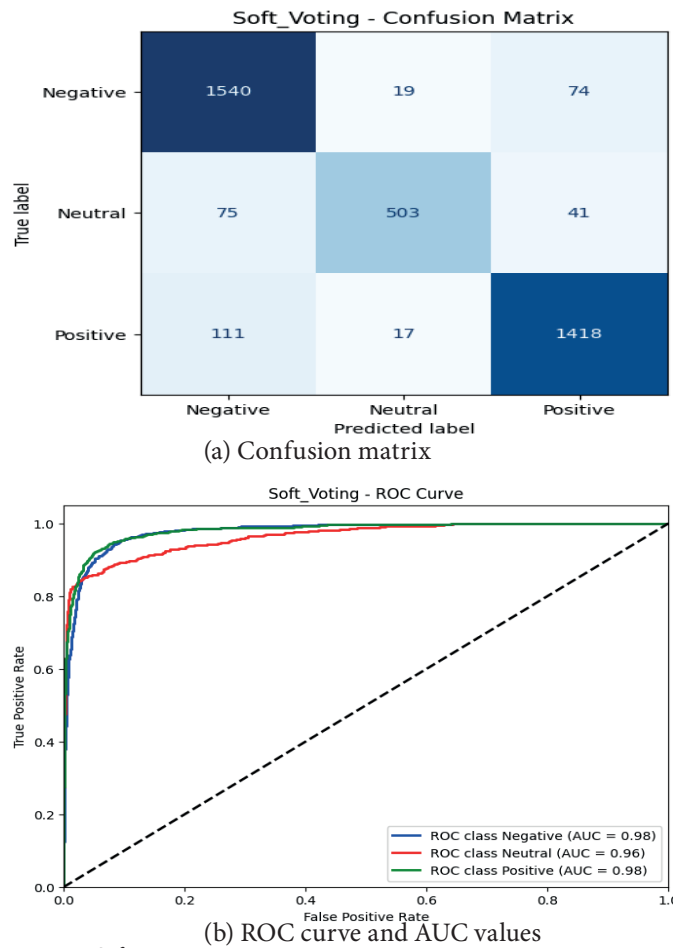


Figure 7. Soft voting

ROC: Receiver operating characteristic, AUC: Area under curve

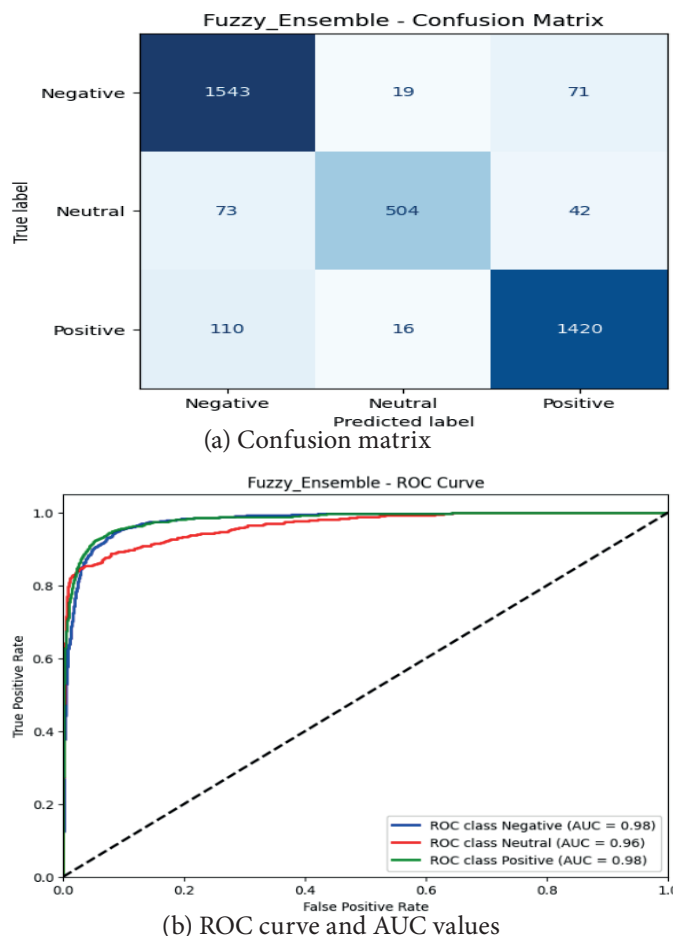


Figure 8. Proposed fuzzy ensemble

ROC: Receiver operating characteristic, AUC: Area under curve

In the visual analysis, the ROC curve for the fuzzy ensemble (Figure 8b) confirms the system's superiority. It delivers peak accuracy together with unmatched classification robustness, a claim validated by AUC values that consistently reach 0.98 across all categories.

DISCUSSION

Comparing with current literature our model shows better advantage. Especially our model surpassed 0.7550 F1 score reported in the reference study for their LSTM model (Karaca & Aslan, 2021) with reaching 0.9123. This performance leap is largely attributed to our dynamic weighting mechanism which effectively lessens 'contextual uncertainty' issue mostly cited in previous search (Alam, 2025). Additionally, although high performance was reported in a study proposing a hybrid RoBERTa-BiLSTM-CNN structure (Yang et al., 2025), the model in question does not include a weighting mechanism based on error rate (MSE) as in our research.

The proposed ensemble architecture aligns with the necessity of "managing uncertainty and ambiguity in natural language," which is emphasized in previous studies (Ming et al., 2024), (Alam, 2025). When more detailed comparison conducted, proposed Fuzzy logic-based model shows better performance from their contemporary rivals for COVID-19 tweets. For instance, while an accuracy rate of 89.00% was achieved with the proposed BERT-CNN structure in a study using a limited time range (March 16-31) of the same data source (Kumar et al., 2024), another study conducted on same dataset remained at the level of 86% while using hybrid model (Shahriar & Sarker, 2025). This indicates that, instead of static weighting, fuzzy logic rules that dynamically process the confidence and past performance values of each sample are more successful in managing the diversity and uncertainty in the dataset. While approaches in the literature such as TexShape rely on weighted linear combinations to balance data processing objectives (Kale et al., 2024), the model proposed in this study has unwearied this balance through linguistic rules based on expert opinion. To cope with the noise inherent and ambiguity available in social media data (Nandi et al., 2025), this ensemble model proposed in the study conducted in 2025 offers a more versatile decision mechanism in ambiguous situations, as opposed to traditional methods.

Traditional soft voting methods are vulnerable to indiscriminate reliance when it comes to model confidence. To solve this problem, the Mamdani method is used as a reliability filter since its configuration provides historical accuracy. Within the protocol, if the model's past performance is not sufficient, even a high-confidence prediction is amerced via the Rule Base. This mechanism enhances noise resistance. Consequently, the observed rise in precision (to 0.9138) stems from the fuzzy logic functioning as a 'safety valve,' which systematically removes false positives.

Conventional type-1 fuzzy logic mostly relies on basic 3-level member structure, but our research enlarges this to 7-level fine-grained frameworks. This increased granularity serves to highly improve the correctional power of the model. To validate this advantage over a plain Soft Voting and to bring out context dependency that raw numbers might miss, we focused on disagreements between respective outputs in cases where the two-voting mechanism were a odd. In this case, our study reveals that the Fuzzy Logic attempted to solve failures when text was particularly highly complex, for example, irony

or slang. Within this framework, 'ambiguity' points out to the semantic 'gray areas' where sentiment transitions are not clear, so it is not mathematically easy to find a 'crisp' logic to create a definitive label. Furthermore, tweets that are 'hard-to-understand' are defined as examples where high-dimensional noise such as slang, irony, humor or context-dependent keywords create conflicts with the literal meaning of the text. For find a solution, our model operates as a sensitive 'sentiment filter' that figures out these ambiguities by assessing sample-specific uncertainty. As an example, the system was tested against tweets that are usually misinterpreted by the traditional architectures. One of the examples is the test set entry: *'Im at a pioneer supermarket in brooklyn and its nice to see thats some places havent completely lost their mind and bought everything and everything covid.'* This text is prone to misinterpretation by shallow models because keywords like 'covid' and 'lost' often trigger negative associations regarding hardship, obscuring the true supportive sentiment.

Standard soft voting faltered here, mislabeling the text as 'neutral' due to an over-reliance on the pessimistic connotations of specific keywords. The proposed fuzzy ensemble, however, successfully decoded the nuance; it recognized the statement as an empathetic observation rather than a complaint, correctly assigning it to the 'positive' class. This instance underscores the utility of our 7-level granular architecture. It functions as a highly sensitive 'sentiment filter,' specifically excelling in ambiguous, gray-area categories like 'neutral' typically the hardest to pinpoint.

Limitations

Even though this model achieves significant performance, several limitations must be acknowledged. First, model success is mostly dependent on fine-tuning process especially adjusted for COVID-19 vocabulary such as "mask", "quarantine" and "symptom", so this can limit its usage for other sentiment analysis. Also, preprocessing process and bert-base-uncased tokenizer restrict evaluation to English language texts. Our 7 level fuzzy logic module is additional computational burden (22.67 ms per sample) during the defuzzification process. It can be usage restriction with wider dataset and real time streaming. We used 12 precise rules instead of 49 rules. It balanced performance but may not catch all possible nuanced scenario.

CONCLUSION

The obtained accuracy rate of 91.28% is at a competitive level with the results of recent studies. All results when compared to benchmark studies above with 91.28% accuracy and 91.38% precision rate achieved in this study. Thus, the improvement is not merely statistical; it represents a tangible qualitative leap over standard methodologies. Our future work will prioritize collection of far more diverse, real word-datasets to unsure the model can adapt effectively to wide range of different context. Generalizability is the primary objective here. At the same time, we intend to benchmark our current architecture against other powerful transformer variants such as DistilBERT and RoBERTa. Future work also can contain different deep learning models such as CNN, RNN and Gans. Finally moving from theory to practice we plan to deploy this model on live tweet streams while integrating Explainable Artificial Intelligence to ensure every decision remains transparent to the user.

ETHICAL DECLARATIONS

Ethics Committee Approval

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data.

Peer Review Process

This manuscript was subject to external peer review.

Conflict of Interest

The authors declare no conflicts of interest related to this study.

Financial Disclosure

The authors received no financial support for the conduct or publication of this research.

Author Contributions

Concept: MSE, ST; Design: MSE, ST; Control: MSE, ST; Resources: MSE, ST; Materials: MSE, ST; Data Collection and/or Processing: MSE, ST; Analysis and/or Interpretation: MSE, ST; Literature Review: MSE, ST; Writing the Article: MSE, ST; Critical Review: MSE, ST.

REFERENCES

- Airlangga, G. (2024). Spam detection in YouTube comments using deep learning models: a comparative study of MLP, CNN, LSTM, BiLSTM, GRU, and attention mechanisms. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 4(4), 1533-1538.
- Alam, M. S., Mrida, M. S. H., & Rahman, M. A. (2025). Sentiment analysis in social media: how data science impacts public opinion knowledge integrates natural language processing (NLP) with artificial intelligence (AI). *American Journal of Scholarly Research and Innovation*, 4(1), 63-100.
- Anwar, Z., Afzal, H., Altaf, N., Kadry, S., & Kim, J. (2024). Fuzzy ensemble of fine-tuned BERT models for domain-specific sentiment analysis of software engineering dataset. *PLOS ONE*, 19(5), e0300279. <https://doi.org/10.1371/journal.pone.0300279>
- Arslan, S., Orman, Z., & Akan, A. (2021). A novel fuzzy logic-based text classification method for tracking rare events on Twitter. *IEEE Access*, 9, 36915-36929. <https://doi.org/10.1109/ACCESS.2021.3062345>
- Bashar, M. K., Monjur, O., Islam, S., Shams, M. G., & Quader, N. (2025). Exploring synergistic ensemble learning: uniting CNNs, MLP-mixers, and vision transformers to enhance image classification. *arXiv*. <https://arxiv.org/abs/2504.09076>
- Bello, A., Ng, S. C., & Leung, M. F. (2023). A BERT framework for sentiment analysis of tweets. *Sensors*, 23(1), 506. <https://doi.org/10.3390/s23010506>
- Bilal, A. A., Erdem, O. A., & Toklu, S. (2023). Applying sentiment analysis on children's stories. *Gazi University Journal of Science Part A: Engineering and Innovation*, 10(1), 71-84.
- Bilal, A. A., Erdem, O. A., & Toklu, S. (2024). Children's sentiment analysis from texts by using weight updated tuned with random forest classification. *IEEE Access*, 12, 70103-70116. <https://doi.org/10.1109/ACCESS.2024.3400992>
- Cai, R., Qin, B., Chen, Y., Zhang, L., Yang, R., Chen, S., & Wang, W. (2020). Sentiment analysis about investors and consumers in energy market based on BERT-BiLSTM. *IEEE Access*, 8, 171408-171416. <https://doi.org/10.1109/ACCESS.2020.3024417>
- Chen, X., Cong, P., & Lv, S. (2022). A long-text classification method of Chinese news based on BERT and CNN. *IEEE Access*, 10, 34094-34105. <https://doi.org/10.1109/ACCESS.2022.3161545>
- Dhanalakshmi, P., Reddy, U. J., Ravikanth, G., Samathoti, P., & Ramu, G. (2024). COVID-19 Twitter data analysis using LSTM and BERT techniques. *International Journal of Engineering Trends and Technology*, 72(1), 219-228.
- Elgabry, M., & Hamdi, A. (2025). Confidence-credibility aware weighted ensembles of small LLMs outperform large LLMs in emotion detection. *arXiv*. <https://arxiv.org/abs/2512.17630>
- Garcia-Plaza, A. P., Fresno, V., & Martinez, R. (2008). Web page clustering using a fuzzy logic based representation and self-organizing maps. In *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*.
- Haroon, N. H., Abdulsada, Z. R., Sharif, H., Ahmed, W. S., Saleem, M., & Jawad, I. A. (2023). Social media analysis using fuzzy natural language processing with an extension of semantic queries. In *Proceedings of AICERA/ICIS*. <https://doi.org/10.1109/AICERA59538.2023.10420086>
- Howells, K., & Ertugan, A. (2017). Applying fuzzy logic for sentiment analysis of social media network data in marketing. *Procedia Computer Science*, 120, 664-670. <https://doi.org/10.1016/j.procs.2017.11.293>
- Jang, J. S. R., Sun, C. T., & Mizutani, E. (1997). *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*. Prentice Hall.
- Kale, K., Esfahanizadeh, H., Elias, N., Baser, O., Médard, M., & Vishwanath, S. (2024). TexShape: information theoretic sentence embedding for language models. *arXiv*. <https://arxiv.org/abs/2402.05132>
- Karaca, Y. E., & Aslan, S. (2021). Sentiment analysis of COVID-19 tweets using LSTM. *Journal of Computer Science (IDAP Special Issue)*, 366-374.
- Kumar, G., Agrawal, R., Sharma, K., Gundalwar, P. R., Kazi, A., Agrawal, P., ..., & Salagrama, S. (2024). Combining BERT and CNN for sentiment analysis: a case study on COVID-19. *IJACSA*, 15(10), 676-686.
- Liu, M., Zhang, H., Xu, Z., & Ding, K. (2024). The fusion of fuzzy theories and natural language processing: a state-of-the-art survey. *Applied Soft Computing*, 162, 111818. <https://doi.org/10.1016/j.asoc.2024.111818>
- Mendel, J. M. (1995). Fuzzy logic systems for engineering: a tutorial. *Proceedings of the IEEE*, 83(3), 345-377. <https://doi.org/10.1109/5.364486>
- Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021). Deep learning-based text classification: a comprehensive review. *ACM Computing Surveys*, 54(3), 1-40. <https://doi.org/10.1145/3439726>
- Nandi, S., Subrahmanyam, P., & Singh, S. (2025). Fuzzy-based ensemble learning for sentiment analysis in social media data. *Global Journal of Engineering Innovations & Interdisciplinary Research*, 5(1), 1-8.
- Rahman, M. M., Shiplu, A. I., Watanobe, Y., & Alam, M. A. (2025). RoBERTa-BiLSTM: a context-aware hybrid model for sentiment analysis. *arXiv*. <https://arxiv.org/abs/2406.00367>
- Singh, C., Imam, T., Wibowo, S., & Grandhi, S. (2022). A deep learning approach for sentiment analysis of COVID-19 reviews. *Applied Sciences*, 12(8), 3709. <https://doi.org/10.3390/app12083709>
- Seth, T., & Muhuri, P. K. (2024). Enriching word embeddings with fuzzy systems for NLP tasks. In *IEEE FUZZ Conference*. <https://doi.org/10.1109/FUZZ-IEEE60900.2024.10611949>
- Sherin, A., Lokesh, S., Deepa, S. N., & Jeya, I. J. S. (2025). Fusion of deep recurrent neural network models and fuzzy decision support system for tweet sentiment analysis. *Automatika*, 66(4), 28-50. <https://doi.org/10.1080/00051144.2025.XXXXXX>
- Xu, C., & Kechadi, M.-T. (2024). An enhanced fake news detection system with fuzzy deep learning. *IEEE Access*, 12, 88009-88022. <https://doi.org/10.1109/ACCESS.2024.3418340>
- Yang, L., Wang, J., & Qiu, W. (2025). RoBERTa-based multi-feature integrated BiLSTM and CNN model for ceramic review analysis. *IEEE Access*, 13, 103681-103692.
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: pre-training of deep bidirectional transformers for language understanding. *arXiv*. <https://arxiv.org/abs/1810.04805>
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ..., & Stoyanov, V. (2019). RoBERTa: a robustly optimized BERT pretraining approach. *arXiv*. <https://arxiv.org/abs/1907.11692>
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), 2673-2681. <https://doi.org/10.1109/78.650093>
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533-536. <https://doi.org/10.1038/323533a0>

Intelligent diagnosis of tomato leaf diseases using YOLOv8

H Hatice Candan*, H Hüseyin Aydilek, M Mustafa Yasin Erten

Department of Electrical and Electronics Engineering, Faculty of Engineering and Natural Sciences, Kırıkkale University, Kırıkkale, Türkiye

Cite this article as: Candan, H., Aydilek, H. & Erten, M. Y. (2026). Intelligent diagnosis of tomato leaf diseases using YOLOv8. *J Comp Electr Electron Eng Sci*, 4(1), 27-33.

Received: 11.11.2025

Accepted: 01.04.2026

Published: 25.04.2026

ABSTRACT

Tomato is one of the most widely cultivated and consumed vegetables worldwide. However, its production is hindered by various pests and pathogens. Rapid and accurate detection of these diseases is crucial for timely intervention and effective management. This study presents a model that employs YOLOv8n, an advanced object detection algorithm, for the identification of tomato leaf diseases. Experimental evaluations were conducted using 2,000 images selected from the Tomato subset of the CCMT dataset, which comprises 5,435 images categorized into five disease classes. The subset was constructed using a stratified sampling strategy to ensure balanced class representation, while low-quality or ambiguous images were excluded to improve annotation reliability. The results demonstrate that YOLOv8n achieved a mean Average Precision of 0.704, a precision of 0.655, and a recall of 0.721 across all classes. The best performance was observed in the healthy class, with an mAP50 of 0.942, a precision of 0.835, and a recall of 0.944. Overall, the findings indicate that AI-based rapid diagnostic systems can serve as an effective solution for early detection of tomato diseases and the prevention of yield losses in agricultural production.

Keywords: Plant leaf disease, disease detection, tomato leaf disease, YOLOv8, object detection

INTRODUCTION

Tomatoes are among the most consumed and widely cultivated vegetables worldwide. Almost every region with suitable climate conditions grows tomatoes. According to FAO and TEPGE data, while world tomato production was approximately 186 million tons in 2022, Türkiye's 13 million tons of production constituted 7% of the world total, placing the country in third place after China and India (FAO, 2023; TEPGE, 2024). According to TEPGE's 2024 report, tomato production in Türkiye in 2023 was approximately 13.3 million tons, 58.3% of which was table tomatoes and 41.7% was tomato paste. In addition, Türkiye's tomato exports in 2023 amounted to 533 thousand tons and provided foreign exchange income of 459 million dollars (TEPGE, 2024). According to TÜİK balance tables, per capita tomato consumption in Türkiye was 106.4 kg in the 2022/23 period, and the sufficiency level was at 117.5% (TÜİK, 2023 ; TEPGE, 2024).

Tomatoes are of great importance to human health due to their nutritional value and antioxidant properties. However, tomato cultivation faces significant challenges from pests and diseases, which negatively affect yield and quality. Farmers need adequate knowledge of tomato diseases to identify and distinguish them manually. Nevertheless, they already encounter numerous difficulties during the cultivation process. Pest attacks and failure to detect diseases on time can lead to significant reductions in yield and quality, causing severe economic losses and potentially resulting in the need to import tomatoes at exorbitant prices. Similar yield-related research highlights the importance of precise agricultural

decision-making for enhancing productivity. For example, Karadaş and Bulut (2024) compared multiple predictive algorithms for estimating tomato yield, finding that the MARS algorithm achieved the highest accuracy and identified key agronomic factors influencing yield, such as irrigation frequency, fertilizer amount, and seedling density.

Therefore, it is crucial to identify tomato diseases rapidly and accurately and assess disease severity for timely implementation of preventive and management strategies. Currently, tomato disease detection relies on visual identification by experts using microscopes and similar tools, a process that is both time-consuming and labor-intensive. Additionally, the accuracy of disease identification depends on the expertise of the personnel involved. Since experts cannot always be present in the field and farmers may lack sufficient knowledge to diagnose such diseases, Artificial Intelligence (AI) and computer-aided methods can be employed to detect tomato leaf diseases (Mugithe et al., 2020).

In the literature, deep learning methods such as YOLO, VGG-16, Faster R-CNN, ResNet, AlexNet, CNN, and MobileNet have been widely used for tomato leaf disease detection (Ibáñez and Reyes-Muñoz, 2023; Adhikari et al., 2018; Kılıçarslan and Paçal, 2023; Li and Wang, 2020; Sakkarvarthi et al., 2022; Hong and Huang, 2020). Similar comparative studies on plant leaf disease detection using transfer learning-based CNN models have also been conducted. For instance, Sazak, Balsak, and Badem (2025) evaluated VGG16, VGG19, AlexNet, MobileNetV1, and MobileNetV2, reporting that

Corresponding Author: Hatice Candan, hatcan95@gmail.com



This work is licensed under a Creative Commons Attribution 4.0 International License.

MobileNetV1 achieved the highest accuracy (99.20%) and superior precision, sensitivity, and F1-score across all classes, with the model integrated into a real-time web-based application for practical use in agriculture. In a study conducted by Ibáñez and Reyes-Muñoz, a model based on a Convolutional Neural Network (CNN) was proposed for identifying and classifying tomato leaf diseases, achieving an accuracy of 99% on both training and test datasets (Ibáñez and Reyes-Muñoz, 2023). Adhikari and others (Adhikari et al., 2018) used the PlantVillage dataset to detect three classes of tomato diseases (late blight, gray spot, and bacterial canker) using the YOLO model, reporting a Mean Average Precision (mAP) of 0.76. Kılıçarslan and Paçal (Kılıçarslan and Paçal, 2023) compared DenseNet, ResNet50, and MobileNet deep learning models for distinguishing between healthy and diseased tomato leaves. Their experimental results indicated that the DenseNet model performed best, achieving an error rate of 0.0269 and an accuracy of 99%.

Li and Wang detected tomato diseases and pests using YOLOv3 algorithm. They concluded that YOLOv3 can accurately and quickly detect tomato diseases in real time with a detection time of 20.39 ms and a detection accuracy performance of 92.39% (Li and Wang, 2020). Sakkarvarthi et al. used a CNN-based model with two convolutional and two pooling layers for tomato plant disease detection and classification. Their proposed model outperformed the pre-trained InceptionV3, ResNet152 and VGG19 models by showing a training accuracy performance of 98% (Sakkarvarthi et al., 2022). Hong et al. (Hong and Huang, 2020) evaluated the performance of ResNet50, Xception, MobileNet, ShuffleNet and Densenet121_Xception models on tomato leaf images captured by mobile applications. Among the tested models, DenseNet_Xception achieved the highest detection accuracy of 97.10%, while the lowest accuracy belongs to ShuffleNet with 83.68% accuracy.

The aim of this study is to provide fast and accurate identification of tomato diseases, which are critical factors for tomato production and yield, by employing the YOLOv8 model, a state-of-the-art object detection algorithm. Unlike earlier versions such as YOLOv3 or other CNN-based approaches (e.g., ResNet, MobileNet, DenseNet), YOLOv8 offers significant advantages in terms of real-time performance, computational efficiency, and adaptability to different vision tasks. These characteristics make it highly suitable for practical agricultural applications where timely detection is essential. Experimental studies were conducted using 2,000 selected images from the “Tomato” subset of the CCMT dataset, which contains a total of 5,435 images across five different disease classes. The performance of the YOLOv8 model was then evaluated, and promising results were obtained.

METHODS

Ethics

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data. All procedures were carried out in accordance with the ethical rules and principles.

Dataset

The dataset used in this study was developed within the scope of the African Food Systems program, supported by the AI4AFS/GR/AFS1233809296 project (Mensah et al., 2023).

It comprises images of cashew, cassava, maize, and tomato plants collected from local farms across various regions of Ghana, representing diseased, pest-affected, and healthy samples. The images were captured using a Canon EOS Rebel T7 DSLR camera equipped with an 18–55 mm lens between October and December 2022, during daylight hours, from multiple angles, and under varying lighting conditions and diverse backgrounds, including white, yellow, brown, gray, and natural settings.

A total of 24,881 raw images were initially collected. To enhance the dataset’s diversity and generalizability, augmentation techniques such as rotation, brightness adjustment, and scaling were applied, resulting in 102,976 images. All images were labeled by experts in plant virology and pathology, and this labeling process was verified by the authors through manual review and consistency checks. Label accuracy was further validated via cross-validation and consensus meetings among the experts. Samples with inconsistent or ambiguous labels were excluded, and the final dataset was structured to include 22 distinct classes across four plant species.

The images were collected using a high-resolution camera device. The original JPG images have varying dimensions, including 400×400, 487×1080, 1080×518, 3024×4032, and 4032×3024 pixels. The tomato subset of the dataset consists of five categories: healthy leaves, leaf blight, leaf curl, septoria leaf spot, and verticillium wilt, as illustrated in Figure 1 and 2. Each category is represented in both raw and augmented forms, totaling 5,435 raw and 27,178 augmented images in JPG format. The images were captured from both front-facing and 180-degree rotated angles, under various lighting conditions including daylight, shadow, and artificial illumination, and against diverse backgrounds.

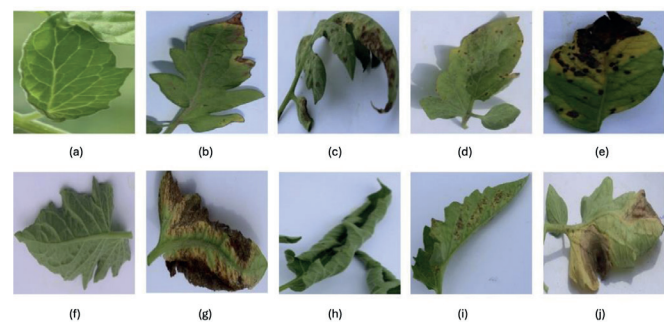


Figure 1. Tomato leaf samples: (a,f) healthy, (b,g) blight, (c,h) curl, (d,i) septoria, (e,j) wilt

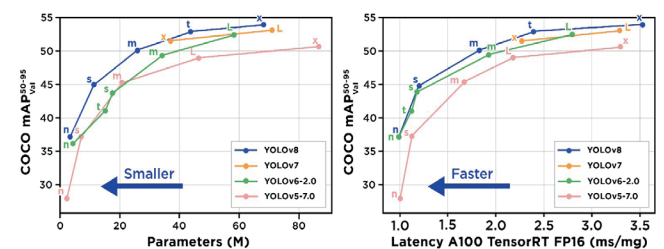


Figure 2. Comparison of YOLOv8 with other YOLO models (Taromi and Haghzad Klidbary, 2025)

Although the tomato subset contains 5,435 raw images, only 2,000 images were used for model training. This decision was made because processing high-resolution images imposes substantial hardware requirements. Considering the limited processing power and memory capacity of the available hardware, the dataset was restricted to 2,000 images in a balanced manner to preserve the representativeness of each

class. This selection aimed to reduce computational load and ensure an efficient and stable training process.

Data Preprocessing Procedure

In this study, tomato leaf diseases were identified using the YOLO algorithm, which is widely applied in deep learning-based object detection tasks. Object detection is a branch of computer vision that focuses on identifying and locating objects of interest within an image or video frame. The objective is not only to determine the presence of specific objects but also to localize them precisely by drawing bounding boxes around their regions.

In supervised learning applications such as object detection and recognition with YOLO, labeling constitutes a crucial preprocessing stage. This process involves assigning appropriate labels to objects within each image and defining the region of interest that encompasses the object through bounding boxes.

Manual labeling of extensive datasets and the determination of bounding boxes require considerable time and human effort. Although various software platforms such as Roboflow provide partial automation, the labeling and bounding box annotation steps in this study were manually executed by the authors using a custom-developed labeling tool. A total of 2000 tomato leaf images were annotated into five distinct classes encompassing healthy leaves, leaf blight, leaf curl, septoria leaf spot, and verticillium wilt as presented in [Table 1](#). The selection of the 2000-image subset was performed using a stratified sampling approach to ensure balanced class representation. Specifically, images were randomly selected from each class while preserving proportional distribution across all five disease categories. Additionally, images with poor quality, excessive blur, or insufficient visibility of disease symptoms were excluded to improve annotation reliability. This selection strategy aimed to maintain dataset diversity while reducing computational load, ensuring that the model was trained on representative and informative samples.

The labeling process was guided by visual disease characteristics commonly reported in the literature, such as lesion color, shape, and distribution patterns on the leaf surface. Although the annotation was performed by the authors, reference images and publicly available dataset guidelines were used to ensure consistency across all classes. In addition, annotations were reviewed multiple times to minimize labeling errors and improve internal consistency.

However, laboratory-based diagnostic methods such as molecular or microscopic analyses were not applied to validate the labels, and no external expert verification was conducted. Therefore, the labeling process may include a degree of subjectivity, particularly in visually similar disease classes. This limitation is inherent in many image-based agricultural datasets.

The relatively limited size of the annotated subset and the absence of expert-validated labeling constitute important limitations of the present study. Nevertheless, the applied labeling procedure provides sufficiently consistent ground-

truth data for model training and evaluation, while highlighting the need for more robust annotation protocols in future research.

YOLO Model

YOLO is a deep learning-based object detection algorithm that employs convolutional neural networks to identify and localize multiple objects within an image. It was first introduced by Redmon et al. (2016) and is designed to perform detection in a single pass through the image, which provides high computational efficiency and strong detection accuracy. Unlike region-based methods that operate in multiple stages, YOLO processes the entire image simultaneously, thereby reducing computational complexity and allowing faster inference. This single-stage approach makes YOLO particularly suitable for real-time applications in which rapid decision-making and efficient computation are essential.

Although newer iterations of the YOLO architecture have been introduced, YOLOv8 was selected for this study because it represents a mature, well-documented, and extensively benchmarked generation that integrates several architectural innovations without the instability or limited community validation often associated with more recent releases. YOLOv8 introduces a fully anchor-free detection mechanism, decoupled classification and regression heads, and a revised feature fusion module that significantly enhance convergence stability, inference speed, and detection accuracy. These advancements collectively provide a robust foundation for real-world applications requiring both precision and computational efficiency. Moreover, YOLOv8 has been widely adopted in various domains such as medical imaging, remote sensing, and agriculture, where it has demonstrated reproducible and verifiable performance across diverse datasets. This makes it particularly suitable for research that demands methodological transparency and replicability.

YOLOv8 employs a convolutional neural network architecture composed of two main components, the backbone and the head. The backbone is based on an enhanced Cross Stage Partial Darknet structure with 53 convolutional blocks, enabling improved information flow between layers. The head comprises several convolutional layers followed by fully connected layers that predict bounding boxes, objectness scores, and class probabilities. A key innovation of YOLOv8 is the integration of a self-attention mechanism within the head, allowing the model to focus on salient regions of the image and refine important features. Furthermore, the use of a feature pyramid network provides strong multi-scale detection capability, enabling the simultaneous recognition of both large and small objects. The schematic diagram of the YOLOv8 architecture is illustrated in [Figure 3](#).

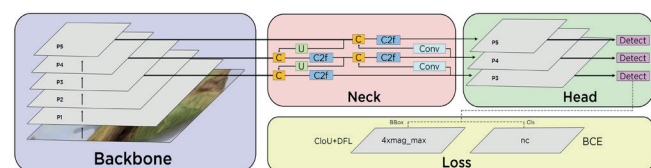


Figure 3. YOLOv8 architecture

Table 1. Number of images per class						
Class type	Healthy	Leaf blight (<i>Alternaria solani</i>)	Leaf curl (<i>Morbus foliorum tomati incurvation</i>)	Septoria leaf spot (<i>Septoria lycopersici</i>)	Verticillium wilt (<i>Verticillium dahliae</i>)	Total
Number of images	426	328	433	466	347	2000

Within the YOLOv8 model family, which consists of five variants (n, s, m, l, and x), the YOLOv8n model was deliberately chosen due to its optimal balance between detection accuracy and computational efficiency. Agricultural image datasets often contain high-resolution images with substantial variation in illumination, texture, and background complexity. YOLOv8n effectively addresses these challenges while requiring considerably less processing power and memory. Its lightweight structure allows deployment on standard GPUs and edge devices without significant accuracy loss. Architectural features such as the anchor-free detection head, decoupled classification and regression branches, and adaptive spatial feature fusion further enhance training stability and generalization capability. Although larger models such as YOLOv8l or YOLOv8x offer slightly improved accuracy, their increased computational cost makes them less suitable for real-time or large-scale agricultural analysis.

In addition, although more recent versions such as YOLOv9 have been introduced, these models are relatively new and have not yet been extensively validated across diverse agricultural datasets. The lack of standardized benchmarks and reproducibility studies for these newer versions may introduce uncertainty in performance evaluation. Therefore, YOLOv8 was preferred due to its architectural maturity, extensive documentation, and proven stability in real-world applications.

Furthermore, the selection of YOLOv8n was guided by the need to balance detection accuracy and computational efficiency. In practical agricultural scenarios, models are often required to operate under limited hardware conditions, making lightweight architectures more suitable for deployment. While larger or newer models may provide marginal accuracy improvements, they typically require significantly higher computational resources.

Performance Evaluation Metrics

Standard quantitative metrics are necessary for only evaluation and comparison of object detection models performances. The two most accepted standard metrics used for these kinds of evaluations are IoU and Mean Average Precision (Rizzoli, 2023; Padilla et al, 2020).

Intersection over Union (IoU)

IoU is rather frequently mentioned among the object detection metrics for assessing localization, as well as for errors in localization. The calculation of IoU is performed using the intersection area shared by the two appropriate bounding boxes set for the same object. Next, the total area covered by the two bounding boxes constitutes the ‘Union,’ and the overlapping occupied by these two bounding boxes works out the ‘Intersection.’ To get the cover between prediction boxes and ground-truth boxes, the intersection should be divided by the union to create a ratio (Figure 4). IoU denotes how well the predicted coordinates of the bounding box overlap with the actual box coordinates. Generally, the higher value of IoU indicates a more accurate prediction. IoU values above 0.5 are treated as positive predictions; below 0.5 are treated as negative.

Mean Average Precision (mAP)

mAP is a well-developed metric for evaluating the performance of various object detection models. To compute average precision, one must take the area under the precision-recall curve mapped over the given set of predictions. The mAP is then calculated by averaging the APs of all classes.

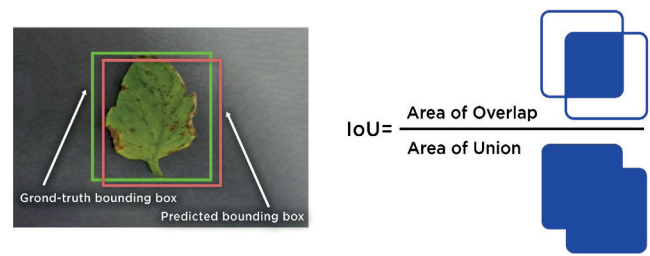


Figure 4. Intersection over union representation

Recall refers to the ratio of correctly predicted instances to the total number of ground truth instances for a class, while precision is the ratio of true positives to the total amount of predictions made by the model. The anomaly of high mAP results from an intelligent object detection model performance.

$$mAP = \frac{1}{n} \sum_{k=1}^{k=n} AP_k$$

The formula to compute the mAP is as follows, where $[AP]_k$ is the AP of class k and n is the number of classes.

Such terms include true positive (TP), false positive (FP), true negative (TN), and false negative (FN); that are used in the calculation of other important metrics such as accuracy, precision, and recall.

Accuracy is the percentage of correctly classified instances:

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

Precision represents the proportion of correctly predicted positive cases:

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

Recall reflects the proportion of positive instances correctly identified by the model:

$$recall = \frac{TP}{TP + FN}$$

F1 Score, the harmonic mean of precision and recall, ranges between 0 and 1, with 1 indicating perfect precision and recall:

$$f1\ score = 2x * \frac{precision * recall}{precision + recall}$$

RESULTS

After dividing the dataset into training, validation, and test sets, the training set was used to train the YOLOv8n model within the PyCharm IDE. The dataset was split into 80% training, 10% testing, and 10% validation. The trained model was then evaluated on the test dataset using the custom-trained weights. Figure 5 provides a flowchart summarizing the entire process.

The model aims to classify each image into one of the predefined classes (‘healthy’, ‘leaf blight’, ‘leaf curl’, ‘septoria leaf spot’, ‘verticillium wilt’), effectively identifying tomato leaf diseases. Table 2 provides details on the hyperparameters used during the training process, while Table 3 presents the model’s evaluation metrics and mAP values for each class.

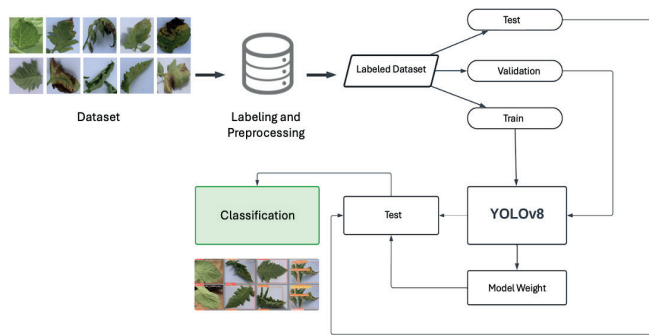


Figure 5. Algorithm flow chart

Hyperparameter	Value
Image size	600x800 pixels
Training time	2282 sec
Epochs	20
Batch size	32
Learning rate	0.01

Class	P	R	mAP50	mAP50-95
All	0.655	0.721	0.704	0.520
Healthy	0.835	0.944	0.942	0.788
Leaf blight	0.650	0.412	0.503	0.293
Leaf curl	0.710	0.800	0.709	0.440
Septoria leaf spot	0.520	0.636	0.606	0.484
Verticillium wilt	0.560	0.812	0.761	0.597

P: Precision, R: Recall

The model’s hyperparameters included a resolution of 600×800 pixels for input images, a total training time of 2282 seconds, and 20 epochs with a batch size of 32. The learning rate was set to 0.01. Due to hardware limitations, the training process was restricted to 20 epochs. Despite the relatively low number of epochs, the training and validation loss curves indicated stable convergence without significant fluctuations.

The results in Table 3 indicate that the YOLOv8n model performed best in detecting healthy leaf samples, with precision and recall values of 0.835 and 0.944, respectively. However, lower performance metrics for classes such as “leaf blight” and “septoria leaf spot” highlight areas for improvement. The overall mAP50 value of 0.704 and mAP50-95 of 0.520 suggest good general performance, with room for optimization.

Figure 6 provide examples of labeled data and predictions made by the YOLOv8n model. Figure 7 displays the precision-confidence, recall-confidence, and precision-recall curves, which further demonstrate the model’s performance at various thresholds.

Precision-confidence curve: The precision reached a perfect score of 1.0 at a confidence threshold of 0.958.

Recall-confidence curve: Recall was 0.96 even at a confidence threshold of 0.0, indicating robust sensitivity.

Precision-recall curve: The model achieved an average precision score of 0.704 at a 0.5 IoU threshold, indicating effective classification across different classes.

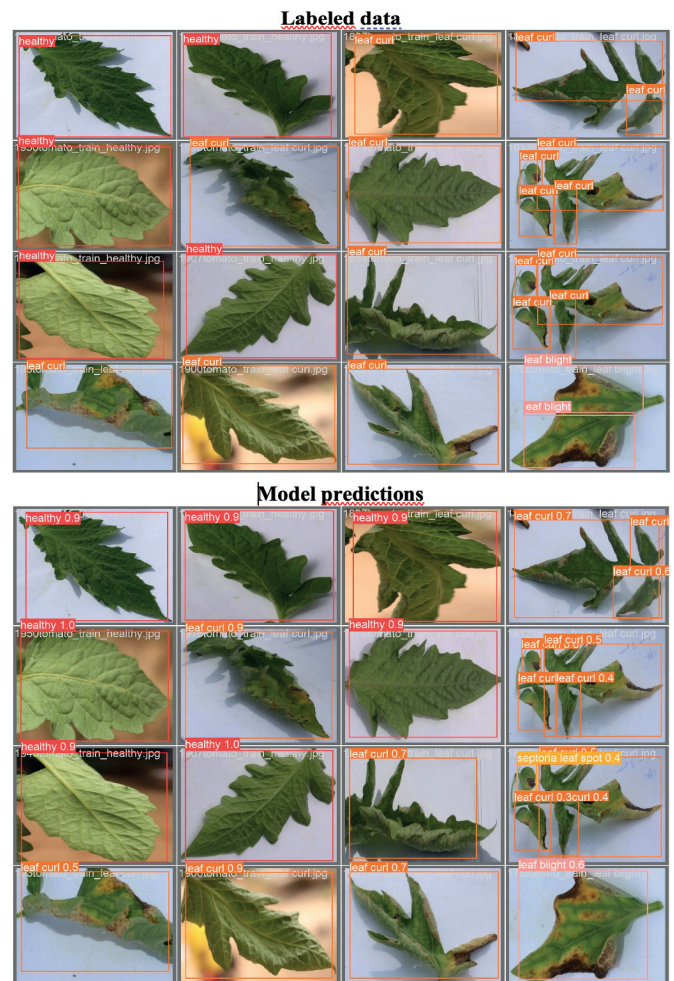


Figure 6. a) Images labeled by us during the model training data preparation process, b) Labels predicted by YOLOv8n model

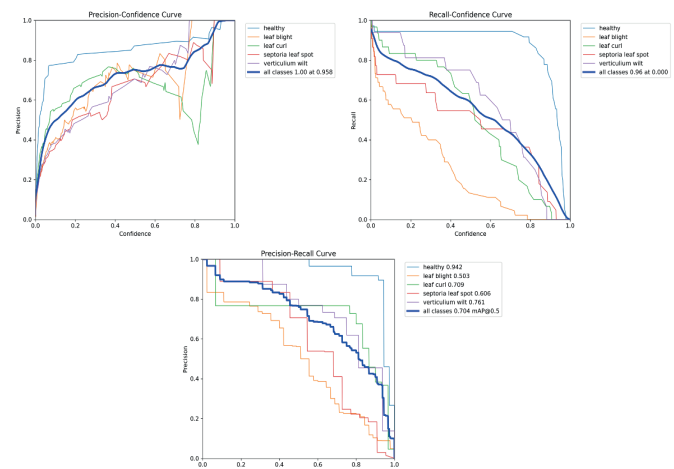


Figure 7. Precision-confidence curve; recall-confidence curve; precision-recall curve, respectively

These findings affirm the model’s ability to detect plant diseases while emphasizing specific classes where further refinement is needed. The relatively lower mAP50 value observed in the leaf blight class (0.503) can be explained by several interrelated factors. First, leaf blight shares highly similar visual symptoms with other diseases such as Septoria leaf spot and Verticillium wilt, which increases inter-class confusion even for advanced deep learning models. Second, class imbalance within the dataset-where leaf blight samples are fewer than Septoria leaf spot samples-may have biased the model toward the majority classes. Third, variations in illumination, background texture, and leaf overlap in field

images reduce lesion contrast and increase detection difficulty. Moreover, YOLOv8's object-detection-based architecture is inherently more sensitive to labeling inconsistencies and background noise compared to pure classification models. Similar difficulties have also been reported in prior works such as (Durmuş et al. 2017). Therefore, the lower detection performance primarily reflects the natural complexity and heterogeneity of real-world agricultural imagery rather than a shortcoming of the proposed model. Future research will focus on mitigating these challenges by applying class balancing, targeted data augmentation, and attention-based feature enhancement strategies.

Although the YOLOv8n model achieved promising overall results, performance varied across classes. The primary reason for selecting the YOLOv8 model in this study is its real-time detection capability, computational efficiency, and robustness to varying image conditions such as illumination, background clutter, and leaf overlap. Unlike models such as DenseNet or ResNet50, which perform only image-level classification, YOLOv8 is capable of both detecting and localizing diseased regions within the leaf images. This dual functionality makes it more suitable for practical field applications and real-time monitoring systems. Moreover, the models in the literature that report extremely high accuracies typically rely on cropped, segmented, or laboratory-captured images under controlled lighting conditions, whereas the YOLOv8 model in this study was trained and tested on unsegmented, natural field images containing greater variability and noise. Therefore, the overall mAP50 value of 0.704 represents a reasonable performance level for a more complex and realistic detection task. This discussion strengthens the justification for model selection and clarifies the contextual differences between the proposed model and previous benchmark studies. The lowest performance was achieved, particularly in the Leaf Blight class, with a recall value of 0.412 and an mAP50 value of 0.503. The Septoria Leaf Spot class remained relatively low, with a precision of 0.520 and an mAP50 value of 0.606. These lower scores may be attributed to several factors. Firstly, there is an imbalance between classes in the dataset; having fewer images in some classes (leaf blight and verticillium wilt classes have fewer images than others) may reduce the generalization ability of the model. Second, leaf blight, Septoria leaf spot, and verticillium wilt share similar visual symptoms, which can lead to misclassifications. Finally, variations in image quality and background conditions introduced additional noise into the model, making it difficult to accurately detect disease-specific patterns. Another important limitation of this study lies in its single-label classification framework, which assumes that each image corresponds to only one disease category. However, in real agricultural environments, multiple diseases or abiotic stress factors may coexist on the same leaf, forming what is known as a "disease complex." In such cases, a single-label detection model like YOLOv8 tends to identify the most dominant visible symptom, potentially overlooking secondary infections. The timing of image capture also plays a crucial role, as disease symptoms evolve over time—early and late infection stages exhibit different color, texture, and lesion characteristics that significantly influence visual diagnosis and model predictions. Although the present framework defines five disease classes and one "healthy" class for academic clarity, real-world implementations would benefit from a hierarchical multi-label structure that first

distinguishes between "healthy" and "diseased" conditions and then differentiates co-occurring disease types. The model's higher accuracy in the "healthy" class further supports this interpretation, suggesting that YOLOv8 is more proficient at detecting the presence of anomalies than distinguishing between visually similar disease categories. Future work will therefore focus on extending the framework to multi-label learning, temporal disease modeling, and stage-aware dataset collection to improve its practical utility under real field conditions.

Unlike previous studies that reported higher accuracy values using larger datasets and heavier models such as DenseNet, ResNet50, or MobileNetV2 (Ibáñez & Reyes-Muñoz, 2023; Kılıçarslan & Paçal, 2023), the proposed approach in this study focuses on achieving a balance between accuracy and computational efficiency. Specifically, the YOLOv8n model—a lightweight and real-time variant of the YOLO family—was intentionally selected to enable practical deployment under limited hardware conditions. While the mean Average Precision (mAP50) of 0.704 is relatively lower than those obtained by more complex architectures trained on larger or augmented datasets, it demonstrates that reliable tomato disease detection can still be achieved with reduced computational cost. This makes the proposed method particularly valuable for resource-constrained environments such as small-scale farms, mobile applications, or edge-based agricultural monitoring systems. Therefore, the novelty of this work lies not in surpassing existing models in raw performance metrics but in providing a feasible, efficient, and transparent framework adaptable to real-world agricultural contexts.

Limitations

In addition, a limitation of this study is that only a single detection model (YOLOv8n) was employed. Ablation studies or experiments with alternative models and hyperparameter settings could provide deeper insights and potentially improve performance. Another practical limitation is that, under real field conditions, multiple diseases or stresses may co-occur on the same leaf, whereas this study assumed a single-label classification approach. This gap may reduce the applicability of the model in complex real-world scenarios.

CONCLUSION

In this study, a YOLOv8n-based deep learning model was successfully developed and tested for tomato leaf disease detection. Utilizing a subset of 2,000 manually annotated images selected from the 5,435-image "Tomato" dataset categorized into five distinct disease classes, the model achieved a promising overall mean Average Precision (mAP50) of 0.704. The highest detection performance was recorded for the healthy class, confirming the model's strong capability in accurately identifying disease-free tomato leaves. The notably higher performance in the healthy class suggests that the model more easily distinguishes between healthy and diseased leaves, while finer discrimination among visually similar disease categories such as leaf blight and septoria leaf spot remains more challenging.

The main contribution of this study lies in demonstrating that a lightweight YOLOv8n model can provide reliable and efficient tomato disease detection, even with a relatively small manually annotated subset. This shows the potential of

integrating real-time AI solutions into agricultural practice to support farmers in early diagnosis and reduce yield losses.

However, the lower detection metrics observed in specific disease categories highlight areas for refinement. Future research could focus on: (i) expanding the labeled dataset to reduce class imbalance, (ii) employing advanced data augmentation techniques, (iii) experimenting with different versions of YOLOv8 (s, m, l, x) and alternative models such as EfficientNet, DenseNet, or ResNet for comparative analysis, (iv) conducting ablation studies on hyperparameters and feature selection, and (v) extending the framework to handle multi-label classification, where multiple diseases may occur on the same leaf under real field conditions.

Overall, this study emphasizes the potential of advanced AI-based solutions like YOLOv8n to improve agricultural practices by enabling rapid and accurate disease detection. By bridging the gap between laboratory-level image datasets and real-world agricultural needs, the proposed approach contributes to more sustainable tomato production and enhanced food security.

ETHICAL DECLARATIONS

Ethics Committee Approval

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data.

Peer Review Process

This manuscript was subject to external peer review.

Conflict of Interest

The authors declare no conflicts of interest related to this study.

Financial Disclosure

The authors received no financial support for the conduct or publication of this research.

Author Contributions

Concept: HC, HA, MYE; Design: HC, HA, MYE; Control: HC, HA, MYE; Resources: HC, HA, MYE; Materials: HC, HA, MYE; Data Collection and/or Processing: HC, HA, MYE; Analysis and/or Interpretation: HC, HA, MYE; Literature Review: HC, HA, MYE; Writing the Article: HC, HA, MYE; Critical Review: HC, HA, MYE.

REFERENCES

- Adhikari, S., Shrestha, B., & Baiju, B. (2018, September). *Tomato plant diseases detection system*. In *Proceedings of the 1st KEC Conference* (Vol. 1, pp. 81-86). Kathmandu Engineering College.
- Durmuş, H., Güneş, E. O., & Kırıcı, M. (2017, August). *Disease detection on the leaves of the tomato plants by using deep learning*. In *2017 6th International Conference on Agro-Geoinformatics* (pp. 1-5). IEEE. <https://doi.org/10.1109/Agro-Geoinformatics.2017.8047016>
- Food and Agriculture Organization of the United Nations. (2023). *FAOSTAT: crops and livestock products*. FAO. <https://openknowledge.fao.org/server/api/core/bitstreams/6e04f2b4-82fc-4740-8cd5-9b66f5335239/content>
- Hong, H., Lin, J., & Huang, F. (2020). *Tomato disease detection and classification by deep learning*. In *Proceedings of the 2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)* (pp. 25-29). IEEE. <https://doi.org/10.1109/ICBAIE49996.2020.00012>
- Ibáñez, J. A. G., & Reyes-Muñoz, A. (2023). Monitoring tomato leaf disease through convolutional neural networks. *Electronics*, *12*(1), 229. <https://doi.org/10.3390/electronics12010229>
- Joher, G., Qiu, J., & Chaurasia, A. (2023). *Ultralytics YOLO (Version 8.0.0)* [Computer software]. Ultralytics. <https://github.com/ultralytics/ultralytics>
- Karadaş, K., & Bulut, O. D. (2024). Comparison of predictive performance of data mining algorithms in predicting tomato yield: a case study in Iğdır. *Kahramanmaraş Sütçü İmam Üniversitesi Tarım ve Doğa Dergisi*, *27*(2), 443-452. <https://doi.org/10.18016/ksutarimdogavi.121585>
- Kılıçarslan, S., & Paçal, İ. (2023). Domates yapraklarında hastalık tespiti için transfer öğrenme metotlarının kullanılması. *Mühendislik Bilimleri ve Araştırmaları Dergisi*, *5*(2), 215-222. <https://doi.org/10.46387/bjesr.1273729>
- Li, J., & Wang, X. (2020). Tomato diseases and pests detection based on improved YOLOv3 convolutional neural network. *Frontiers in Plant Science*, *11*, 898. <https://doi.org/10.3389/fpls.2020.00898>
- Mensah, K. P., Akoto-Adjepong, V., Adu, K., Abra Ayidzoe, M., Asare Bediako, E., Nyarko-Boateng, O., & Opoku, M. (2023). *Dataset for crop pest and disease detection* [Data set]. Mendeley Data. <https://doi.org/10.17632/bwh3zbpkpv.1>
- Mugithe, P. K., Mudunuri, R. V., Rajasekar, B., & Karthikeyan, S. (2020). *Image processing technique for automatic detection of plant diseases and alerting system in agricultural farms*. In *Proceedings of the 2020 International Conference on Communication and Signal Processing (ICCSIP)* (pp. 1603-1607). IEEE. <https://doi.org/10.1109/ICCSIP48568.2020.9182281>
- Padilla, R., Netto, S. L., & da Silva, E. A. (2020). *A survey on performance metrics for object-detection algorithms*. In *Proceedings of the 2020 International Conference on Systems, Signals and Image Processing (IWSSIP)* (pp. 237-242). IEEE. <https://doi.org/10.1109/IWSSIP48289.2020.9145130>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 779-788). IEEE. <https://doi.org/10.1109/CVPR.2016.91>
- Rizzoli, A. (2023, April 24). *The ultimate guide to object detection*. V7 Labs Blog. <https://www.v7labs.com/blog/object-detection-guide>
- Sakkarvarthi, G., Sathianesan, G. W., Murugan, V. S., Reddy, A. J., Jayagopal, P., & Elsis, M. (2022). Detection and classification of tomato crop disease using convolutional neural network. *Electronics*, *11*(21), 3618. <https://doi.org/10.3390/electronics11213618>
- Sazak, S., Balsak, S. C., & Badem, H. (2025). Transfer öğrenme temelli bitki yaprak hastalıklarının tespiti için karşılaştırmalı bir çalışma. *Kahramanmaraş Sütçü İmam Üniversitesi Tarım ve Doğa Dergisi*, *28*(1), 154-170. <https://doi.org/10.18016/ksutarimdogavi.1571202>
- Taromi, A. D., & Klidbary, S. H. (2025). A novel data-driven algorithm for object detection, tracking, distance estimation, and size measurement in stereo vision systems. *Multimedia Tools and Applications*, *84*(12), 11041-11061.
- Tarım Ekonomisi ve Politika Geliştirme Enstitüsü. (2024). *Domates ürün raporu 2024* (Yayın No: 396). T.C. Tarım ve Orman Bakanlığı.
- Türkiye İstatistik Kurumu. (2023). *Bitkisel üretim istatistikleri 2023*. <https://data.tuik.gov.tr>

Detection of Alzheimer's disease from magnetic resonance images with deep learning

 Aslıhan Güngör^{1*},  Necaattin Barışçı²

¹Department of Computer Science, Institute of Informatics, Gazi University, Ankara, Türkiye

²Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara, Türkiye

Cite this article as: Güngör, A. & Barışçı, N. (2026). Detection of Alzheimer's disease from magnetic resonance images with deep learning. *J Comp Electr Electron Eng Sci*, 4(1), 34-45.

Received: 27.01.2026

Accepted: 27.03.2026

Published: 25.04.2026

ABSTRACT

Alzheimer's disease (AD) is a neurodegenerative process that leads to irreversible cognitive impairment in the elderly population and constitutes a significant socio-economic burden on a global scale. The lack of a definitive cure for the disease has made early diagnosis strategies based on magnetic resonance imaging (MRI) data critical. This study offers an innovative perspective to the literature by systematically examining current deep learning approaches used in AD diagnosis. The review includes a comprehensive technical evaluation of innovative preprocessing techniques, specialized convolutional neural network (CNN) architectures, and different input representation strategies (2D, 3D, and stacked sections) used in the process from raw data to clinical prediction. Focusing on methodological gaps in the existing literature, this study discusses key obstacles threatening diagnostic validity, such as class imbalance and data leakage, and highlights application suggestions for overcoming these problems. Ultimately, the research gaps identified in light of the findings and the practical solutions offered aim to contribute to the design of next-generation diagnostic systems by providing researchers with a strategic roadmap in terms of model generalizability and clinical integration.

Keywords: Alzheimer's disease, deep learning, MRI, classification

INTRODUCTION

Alzheimer's disease (AD) is a slow neurological disease that directly affects the development of mental abilities and neurocognitive functionality, destroying the thought process and consciousness of a person (Srivastava et al, 2021). The exact cause of the disease is unknown, but it is more common in older people, and can be fatal, although it is more frequently seen in people with chronic diseases such as diabetes, cardio problems, and hypertension (Srivastava et al, 2021) in later life (Alzheimer Derneği, 2015; Alzheimer Derneği, 2016). It is the leading cause of dementia in the elderly due to damage to neurons related to human memory (Breijyeh & Karaman, 2020) and questioning and learning functions (Jack et al, 2008). The neurological disorder begins with a gradual deterioration, and symptoms increase day by day (Han & Kaushik, 2020). Memory problems are one of the first signs of Alzheimer's disease (Alzheimer Derneği, 2015), but these symptoms can vary from person to person (Alzheimer Derneği, 2016). Factors such as difficulty finding appropriate and correct words during speech, spatial problems, and loss of reasoning ability are also among the symptoms (Han & Kaushik, 2021). The appearance of the first symptoms of mild cognitive impairment is considered an early signal that a person may have AD (Üngar et al., 2014). Although there is no complete cure for AD, early detection can help in taking

preventive measures and improve AD symptoms (Breijyeh & Karaman, 2020).

Alzheimer's disease can be diagnosed and its progression monitored using clinical measurements, but identifying these symptoms requires expertise and is a very time-consuming process (Kundaram & Pathak, 2021; Nasir et al., 2021). Early diagnosis is very difficult unless symptoms become very pronounced. Early detection of AD can help reduce the risk of neuronal damage (Dadar et al., 2022; Aderghal et al., 2018). Early diagnosis raises awareness of the need for patients to take precautionary measures to reduce the risk of the disease progressing from mild cognitive impairment to Alzheimer's disease (Srivastava et al, 2021). Studies suggest different machine learning and deep learning methods for predicting the stages of AD through self-regulating analysis of magnetic resonance imaging (MRI) images, providing efficient and improved diagnostic results for AD (Aderghal et al., 2018; İbrahimi & Luo, 2021; Faruk et al., 2017). The main factors or parameters used by researchers are the cortical thickness of the human brain, gray matter (GM) density in the brain, ventricular enlargements, and brain crumples. Many research studies claim a relationship between gray matter reduction and some brain diseases such as Alzheimer's disease (Dadar et al., 2022). The hippocampus is the part of the brain affected

Corresponding Author: Aslıhan Güngör, aslihankaralok@gmail.com



This work is licensed under a Creative Commons Attribution 4.0 International License.

in the first stage of Alzheimer's disease. White matter (WM), GM, and cerebrospinal fluid are the main and most primitive tissues in images of the human brain. Researchers have discovered that of these three basic tissues of the brain, GM shrinkage corresponds more to physical decline in mild cognitive impairment (Hızır et al., 2015).

METHODS

Ethics

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data. All procedures were carried out in accordance with the ethical rules and principles.

Datasets

In recent years, many research centers around the world have been collecting and releasing a large amount of medical and visual data to the public. Publicly available data plays a significant role for researchers conducting research and development on AD. Online datasets generate biomarker information such as neuroimaging methods, genetic and blood information, clinical and cognitive assessments. Among the most widely used datasets are the Alzheimer's disease neuroimaging initiative (ADNI) (Jack et al., 2008), the Australian Imaging, Biomarker and Lifestyle Flagship Study of Aging (AIBL) (Ellis et al., 2009), the Open Access Imaging Studies Series (OASIS) (Marcus et al., 2010; Marcus et al., 2007; LaMontagne et al., 2019), and Minimal Interval Resonance Imaging in Alzheimer's Disease (MIRIAD) (Malone et al., 2013).

ADNI stands out as the most widely used, longitudinal, and multicenter study. The aim of ADNI is to investigate whether a combination of MRI, PET, other biomarkers, and clinical and neuropsychological assessments can measure the progression of mild cognitive impairment and early AD. ADNI-1, ADNI-GO, ADNI-2, and ADNI-3. The following collections are complementary and improved upon the previous ones. ADNI researchers collect various data types from patients, including clinical, genetic, MRI, PET images, and biological samples. ADNI-1 includes 200 NC, 400 MCI, and 200 AD. ADNI-GO adds 200 MCI to ADNI-1. ADNI-2 expands on ADNI-1 and ADNI-GO with 150 NC, 100 early MCI, 150 late MCI, and 150 AD. ADNI-3 expands on the existing ADNI-1, ADNI-GO, and ADNI-2 by adding 133 NC, 151 MCI, and 87 AD. It collects imaging and medical data from 211 individuals with AD, 133 individuals with MCI, and 768 healthy individuals without cognitive impairment. OASIS aims to share neuroimaging brain datasets with researchers in related fields. There are three versions of OASIS: OASIS-1 contains 434 MRI scans from 416 subjects. OASIS-2 contains 373 MRI scans from 150 subjects. OASIS-3 contains 2,168 MRI and 1,608 AIBL PET scans from 1,098 subjects.

The MIRIAD dataset contains 708 MRI scans from 46 AD patients and 23 NC volunteers.

Some research uses the above datasets in conjunction with their own datasets. For example, the 2016 study by Suk et al. (2016b) used images from ADNI-2 and its in-house dataset with 37 participants, consisting of 12 MCI subjects and 25 NC subjects. In the study conducted by Basaia et al. in 2019, 3D T1-weighted images were collected from 124 patients with AD

probability, 50 patients with HBB and 55 healthy controls, and the dataset was named "Milan" dataset.

Preprocessing

The size of the training and test datasets affects the classification performance. In the studies conducted, the data are used after passing through preprocessing steps. Some MRI software such as FreeSurfer (Fischl, 2012), computational anatomy toolbox (CAT12), FMRIB Software Library (FSL) (Jenkinson 2012), statistical parametric mapping (SPM), ANTS (Avants et al., 2009) are preprocessing libraries used. Recording, normalization, smoothing, segmentation, skull scraping, noise reduction, temporal filtering, and covariate extraction are among the most commonly used preprocessing techniques.

Registration: It is the spatial alignment of image scans to ensure anatomical consistency between individuals and studies. It is also used in multimodal tasks for common registration. MIN 305, Collin27 and MNI152 are among the most commonly used templates. Liu et al. (2016) reported higher performance in their 2016 study adopting multiple templates over a single template. By using multiple templates for feature extraction, it selects the most representative features of each template. By training multiple DVM classifiers, it combines the results of all classifiers. However, multiple templates lead to high computational costs, especially in image registration.

Normalization

Density normalization, also known as area correction or density inhomogeneity correction (Zhao et al., 2023), means rescaling the density of each pixel to a normalized density. During MR imaging, devices with different capabilities scan different subjects or the same subject at different time intervals, which can lead to significant density changes. Large density changes significantly affect the performance of preprocessing such as registration and segmentation.

Skull stripping: Brain scan involves removing extra-brain tissues such as skull, fat, eye, etc., and separating the remaining GM, WM, and cerebrospinal fluid (CSF) (Zhao et al., 2023).

Tissue segmentation: Tissue segmentation means dividing an image scan into segments corresponding to various tissues. Tissue volume is a frequently used measurement after tissue segmentation. GM probability maps are a popular input form in classification tasks. Preprocessing techniques such as density normalization and registration are usually required (Zhao et al., 2023).

Data enhancement: Data enhancement is one of the methods of enhancing a dataset by generating new data samples from existing data without engaging in new data collection, in order to overcome the limitation of the number of subjects in a dataset. Data enhancement techniques include clipping, mirroring, random translation, gamma correction, scaling, random rotation, elastic transformation, vertical flipping, horizontal flipping, and different types of blurring. In addition, new synthesis techniques such as autoencoders and generative adversarial networks are also used in data enhancement. However, synthesis techniques need further proof of the effectiveness of the images generated in AD-related classification and prediction tasks (Zhao et al., 2023).

Figure shows a flowchart illustrating the preprocessing process for MRI images used for Alzheimer's disease.

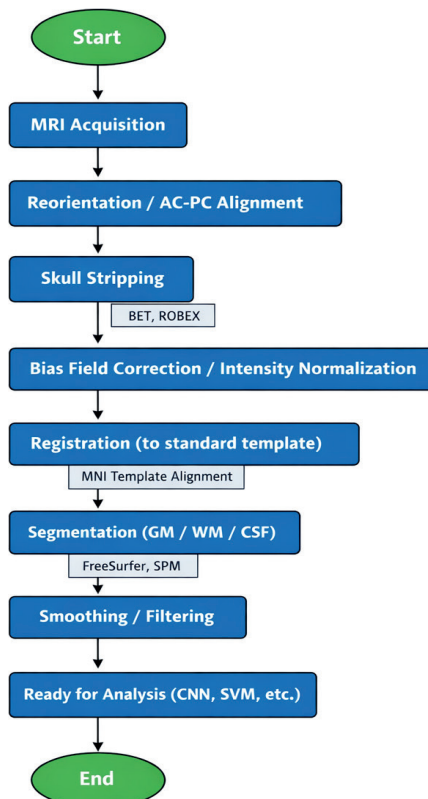


Figure. Standard MRI preprocessing pipeline for Alzheimer's studies
MRI: Magnetic resonance imaging

LITERATURE REVIEW

Traditional Machine Learning

Support vector machines (SVMs) are one of the supervised learning methods used to solve classification and regression problems. SVMs map the input to points in a multidimensional space to maximize the margin between hyperplanes of different data types. A kernel function, such as a Gaussian or polynomial function, maps the existing multidimensional space to a higher-dimensional space. SVMs can be used alone or in conjunction with other methods for traditional machine learning and deep learning methods. Since SVMs can achieve relatively good performance and their operating principles are clear and understandable, SVMs are widely applied in industrial and scientific fields. In a study by Suk et al. in 2014, a linear SVM is used to work with feature representations found by a Deep Boltzmann Machine (DBM) for hierarchical classifiers. In the study by Suk et al. in 2015, multi-core SVMs are used to classify integrated features from multi-mode inputs, while in the study by Suk et al. in 2016, a linear SVM classifier is used. In the study by Shi et al. in 2018, a model is proposed that takes stacked deep polynomial networks (DPN) as feature extractors and a linear kernel SVM as a classifier.

When comparing multi-core and single-core SVMs, multi-core algorithms are seen to offer greater flexibility. Although multi-core SVMs have performed well in most tasks, efficiency is one of the most important problems in developing multi-core SVMs. The computational complexity and difficulty of multi-core SVMs are much more significant than those of single-core SVMs. In terms of space, multi-core SVM algorithms need to calculate the core combination coefficients corresponding to each core matrix, therefore multi-core matrices need to be included in the processing. The need to store multi-core matrices in memory simultaneously is among the challenges encountered. If the number of samples

is very large, the size of the core matrix will be very large. If the number of cores is also very large, it will undoubtedly occupy an enormous amount of memory space. In terms of time, training multi-core SVMs is time-consuming. High time and space complexity are among the main reasons why multi-core SVM algorithms are not widely used (Suk et al., 2025, Shi et al., 2018).

Random forest (RF) is an ensemble algorithm consisting of decision trees. Multiple decision tree classifiers form the random forest. While the decision trees are trained in parallel, the Random Forest combines all classification voting results and ultimately selects the class with the most votes. Random Forest is a flexible and practical method, capable of processing inputs without dimensionality reduction and working well on large datasets. When working with multiple inputs, it can estimate the importance of different variables depending on the type of problem. Computing multiple decision trees and integrating their outputs can consume many computational resources. In 2015, Moradi et al. (2018) proposed a novel biomarker-based diagnostic for classifying different stages of MCI using a low-density separation classifier and a random forest classifier. Lebedev et al. in 2014 tested the random forest on the ADNI and AddNeuroMed datasets using a combination of MRI images and morphometric measurements with ApoE-genotype and demographic (age, sex, and education) MRI images. Bi et al. in 2020 proposed a clustering evolutionary random forest architecture to handle multimodal data from ADNI to detect abnormalities in the brain and pathogenic genes, aiming to overcome the small sample size problem.

Convolutional Neural Networks (CNN)

Deep learning is a machine learning technique where the learning process is carried out with a hierarchical structure. Interest in deep learning techniques has increased significantly in recent years and is widely used in various brain research. Convolutional neural networks are also among the successful deep learning methods. Convolutional Neural Networks (CNNs) are artificial neural networks that use convolutional operations to filter input data and extract useful features. Research on CNNs is rapidly advancing. They play a role in various stages of detection, classification, and segmentation problems in different fields, including medical imaging and natural language processing, achieving high-accuracy results. The success of CNNs in classification studies and the segmentation of realistic images is increasing the application of CNNs in the medical field. Recent studies show that CNNs perform well in segmentation and disease detection problems. The classic CNN structure consists of a series of convolutional layers, a pooling layer, an activation layer, and fully coupled layers. A SoftMax function is applied to classify the input image with probability values between zero and one. The convolution layer includes local receiver areas, shared weights, filters, step, and padding concepts. A filter contains unknown parameters to be learned during training. Convolution involves the process where a filter shifts from top left to bottom right across the entire image and convolutions with the input image to calculate the weighted sum. Step refers to the step size a filter moves per slice. However, the pixels at the edges are never at the center of a filter, and a filter cannot extend beyond the edge region. After each convolution between the input and the filter, only a portion of the pixels at the edge are detected, and information at the image boundary is lost. Padding is designed to overcome this problem.

Padding means filling some values along the input boundaries to increase the input dimension, usually the filled values are zero. Padding is necessary in applications where dimensions must remain constant before and after convolution to prevent information loss. The size of the filters determines the receiver area in the convolutional layers. Convolutional layers are the most suitable feature extraction tool for image datasets with high spatial redundancy due to their ability to resolve features of images with shared weights. By reducing spatial redundancy, the feature vector generated from the outputs of the convolutional layers represents the content of the image. The pooling layer is a dimensionality reduction process in feature maps that aims to reduce training time by reducing the number of parameters to be trained. Maximum pooling, average pooling, and global pooling are the most commonly used pooling layers. Maximum pooling gives the maximum value within the region of the feature map covered by the filter. Average pooling calculates and presents the average value of the elements presented in the feature map region covered by the filter. Global pooling reduces each channel in the input to a single value.

The activation layer provides a nonlinear mapping to the output of the convolution layer to improve the network's reasoning ability. The most commonly used activation functions include ReLU, Sigmoid, Tanh, etc.

The fully connected layer takes the inputs of the feature extractor and predicts the correct label along with probabilities.

CNN can be used as a feature extractor and classifier or as a feature extractor only. CNN is used to obtain target-level representations generated from sparse regression for clinical decision-making (Suk et al., 2017). In the study conducted by Feng et al. in 2020, 3D-CNN with MRI is applied to perform AD classification using MRI images. By using SoftMax as a classifier together with an SVM, this 3D-CNN-SVM model achieves better classification performance than 2D-CNN and 3D-CNN. When comparing traditional machine learning and deep learning methods in AD-related fields; in general, deep learning methods provide better performance than traditional machine learning methods. The appropriate size of training samples should not be <1,000. A dataset containing more than five thousand samples can be considered sufficient to train a deep learning model that provides high accuracy (Zhao et al., 2021).

CNN algorithms: Commonly used CNN algorithms in Alzheimer's disease include LeNet, AlexNet, VGG, GoogleNet, ResNet, and DenseNet.

LeNet: In 1998, LeCun et al. proposed LeNet, the first study to use CNN as a solution to a character recognition problem. By introducing the fundamental concepts of convolutional, pooled, and fully connected layers in a single architecture, they laid the groundwork for the idea of local receptive fields within the CNN, which is the basis of other deep learning modules. In a study by Yang and Liu in 2020, they proposed their models with LeNet-5 for classification and prediction. In the model, which was constructed using PET images of 350 subjects with MCI from ADNI, they achieved 91.02% and 77.63% accuracy and specificity in predicting MCI transformations (Yang & Liu, 2020).

AlexNet: In 2017, Krizhevsky et al. proposed AlexNet, an important architecture following LeNet. The rectified linear

unit (RELU) is used as the activation function, and a method for training networks using multiple GPUs is presented.

In a study conducted by Padmavathi et al. in 2023, a classifier is created to distinguish Alzheimer's disease using the AlexNet and ResNet50 methods. The model is fed with a dataset. Then, the classification and processing of the image is performed by comparing it with the key features obtained during preprocessing and feature extraction. The entire extraction process is performed before training the model in the convolution and pooling phases. The model achieves an accuracy of 54% with AlexNet and 56% with ResNet50 (Padmavathi et al., 2023).

VGG: In a study by Simonyan and Zisserman in 2015, VGG is proposed. A stack of 3×3 convolution filters is used to replace large convolution filters such as 5×5, 7×7, 9×9, or 11×11 convolution filters. A small stack of convolution filters yields better results than a single large convolution filter. By using small filters, deeper networks are obtained with fewer parameters, allowing for the training of a more complex model in a shorter time (Jain et al., 2019).

In a study by Jain et al. in 2019, a transfer learning approach is applied to create an AD classification model using VGG16, pre-trained on ImageNet, as a feature extractor. By converting 3D MRI images into 2D slices, they select 32 slices with the most informative features as a result of preprocessing steps and feed these selected slices as input to VGG16, then reach the result with fully linked layers. Although the datasets contain MRI images of 150 subjects from ADNI, the model achieves an accuracy of 99.14%, 99.30%, and 99.22% for the classifications of CN for AD, MCI for AD, and CN for MCI, respectively. Despite the high classification accuracy for the aforementioned tasks, the generality of the proposed model is quite doubtful due to the very small size of the dataset.

In a study conducted by Lim et al. in 2022, they tested a CNN, VGG-16, and ResNet-50 as feature extractors to distinguish NC, AD, and MCI using MRI images. While training the CNN from scratch, they pre-trained VGG-16 and ResNet-50 using the ImageNet database. VGG showed the best performance with 83.90% accuracy, 82.49% precision, 83.90% recall, and 83.19% F1 score.

GoogleNet: Instead of manually determining whether to use 3×3, 5×5, or 7×7 filters in the initial studies, a new model is proposed that allows the network to automatically learn how to reach the optimal structure. The batch normalization introduced in Inception v2 reduces the internal covariate shift created after convolution operations. Inception v3, while maintaining the consistency of the statistical properties of the data during the training phase, replaces the large convolution kernel with a small convolution kernel. By dividing an N×n convolution kernel into a batch or parallel form of 1×n and 1×n convolution kernels, the proposed general network design principles gradually reduce the information dimension to the desired extent (Szegedy et al., 2015; Szegedy et al., 2016; Szegedy et al., 2017).

In the study conducted by Ding et al. in 2019, using 2,109 PET images of 1,002 patients from ADNI as the dataset, they used Inception v3, which was pre-trained on ImageNet (Ding et al., 2019).

In 2016, He et al. proposed deep residual neural networks (ResNet) to handle vanishing and exploding gradients. Before

the introduction of ResNet, designing such a deep network presented challenges because the gradient vanished rapidly as the network deepened. As the number of network layers increased, more complex feature models could be extracted. Theoretically, better results should be obtained as the model deepens. As network depth increases, network accuracy becomes saturated or even decreases. ResNet overcomes this problem by adding shortcut connections that skip one or more layers. The accumulation layer will only perform identity mapping when the residual is zero, preventing network performance degradation, ensuring the residual is not zero, and allowing the accumulation layer to learn new features based on input features. Since the residual will generally be small, training the model will be easy.

In a study conducted in 2017 by Korolev et al., a CNN network similar to 3D ResNet and VGG was proposed to extract features necessary for 3D image classification of brain MRIs. Although both networks perform well in classifying AD and NC, they fail to separate AD and NC from MCI.

Islam and Zhang (2018) tested an architecture combining Inception v4 and ResNet to identify different stages of AD and achieved an accuracy of 93.18% in OASIS.

In the study by Abrol et al. in 2020, a 3D ResNet network is proposed for classification and prediction. First, the model is trained for MCI detection using 3D gray matter images as input, then the trained model is transferred to the NC and AD classification domain, utilizing the transfer learning method.

DenseNet: In 2017, Huang et al. proposed the DenseNet approach to fully utilize features across all layers. There are two main approaches to improving neural effects; going deeper and going broader. DenseNet directly connects all layers. In other words, the input of each layer is derived from the output of all previous layers. By doing this, DenseNet reduces the vanishing gradient and ensures optimal use of features to improve the effect. At the same time, the number of parameters is reduced to some extent.

In a study by Wang et al. in 2018, a combined 3D-DenseNets method is proposed for the diagnosis of AD and MCI. Due to the limited data problem, the DenseNet method is adopted, and several 3D-DenseNets are trained with varying hyperparameters. The final result is generated by the weighted sum of each basic 3D DenseNet, and the model achieves an accuracy value of 97.19%.

In the study by Wang et al. in 2019, a model is proposed in which 3D DenseNet is adopted as the backbone classifier, followed by fully connected layers and a SoftMax function. Each 3D DenseNet is initialized separately and trained on images of 833 subjects in the ADNI dataset. Probability scores generated by each independent classifier are voted on and integrated into the proposed model.

In the study by Liu et al. in 2020, multitasking deep CNN and DenseNet models are integrated together for hippocampal segmentation and AD classification. In detail, multitasking deep CNN is used to extract features during the segmentation and classification phase, and 3D DenseNet is trained using features for disease classification. Finally, the model combines features learned from multitasking CNN and DenseNet models to perform the classification (Wang et al., 2019).

Zhang et al. (2021) proposed a method using 3D DenseNet in 2021. Training a deep learning model like DenseNet with

such a small dataset often results in a high risk of overfitting. Voting strategy is used to try to compensate for this error. However, training multiple deep learning models from scratch is a time-consuming and inefficient practice. Transfer learning can be a good choice.

Input types: CNN is a powerful tool capable of processing features of varying sizes, and it is categorized into four main methodology categories based on four different input types: 2D slice-based, 3D patch-based, 3D region of interest (ROI-based), and 3D topic-level.

2D dimensional slice: 2D slice-based approaches extract 2D slices from 3D images to reduce the number of hyperparameters, based on the hypothesis that useful features for classification or prediction tasks can be extracted from 2D slices. A common method for extracting 2D slices from a 3D image involves projecting the entire brain scan onto sagittal, coronal, and axial planes. These planes are also referred to as the median, anterior, and horizontal planes. Images from the central part of the brain are more informative because their information entropy is greater than others. Therefore, not all slices will be used during training. Slices from sagittal, coronal, and axial images contain complementary information, and features extracted from these images are integrated and used. While it is easy to obtain a large number of samples when working with 2D slices, a deep learning model incorporating a 2D CNN generally requires fewer parameters and a shorter training time compared to a 3D model. The disadvantage of slice-based approaches is that 2D slices of a brain image lose spatial information between each other because each 2D slice is processed independently. Sarraf et al. (2017), Wang et al. (2018) and Jain et al. (2019) use 2D MRI slices as input type in their studies. In 2017, Sarraf et al. achieved an accuracy of 96.86% for AD and NC classification using LeNet-5 as the CNN method. Wang et al. (2018) established and trained their own 2D CNN model from scratch. In 2019, Jain et al. used 2D MRI slices as input type in their model, while using VGG-16, pre-trained on ImageNet, as a feature extractor. In their 2018 study, Lin et al. (2018) attempted to integrate PCA and Lasso methods with CNN to predict the conversion from MCI to AD. They used CNN as a feature extractor in the training phase to input 2.5D patches, while using PCA and Lasso to reduce dimensions and aiming to select the most informative features. The selected features fed overlearning in the classification phase. Furthermore, by testing features generated from FreeSurfer together with CNN-based features, it was revealed that using both features could provide better performance than using only CNN-based or FreeSurfer-based features.

3D patch: 3D patch-based approaches work similarly to 2D slice-based methods, but instead of sampling projections that cut across specific planes, the 3D brain is studied by scanning it as a series of stepped 3D patches as a hyperparameter. The sample size is larger after the cutting operations. 3D patch-based methods compensate for spatial information loss compared to 2D slice-based methods, but patches are generally used independently during the training phase. When a model is run on the same network for each patch, 3D patch-based methods require low memory. If you train an independent network separately for each patch and combine the results from previous independent networks to use a structure, the overall complexity of the network becomes high. One of the challenges in 3D patch-based methods is selecting informative

patches from the brain scan, while another challenge is selecting the most distinctive features. Qiu et al. (2020) and Zhang et al. (2021) use 3D patches as the input type.

Region of interest based (ROI based): 3D ROI-based methods represent a 3D image of a segmented brain region, paying attention to specific regions that have been clinically proven to be associated with AD. The selected regions are usually informative, such as gray matter volume, hippocampal volume, and cortical thickness. The use of an ROI-based method does not easily lead to overfitting, and the interpretability of the model is near perfect, as the contribution of each region in the model can be seen. The shortcoming of ROI-based methods is the prerequisite of selecting regions in AD. In their 2014 study, Liu et al. extracted features in clustered sparse ADs by taking 3D ROI-based input. In their 2015 study, Liu et al. adopted 3D ROI-based input and used an SVM classifier.

3-dimensional subject level: 3D subject-based methods capture the 3D brain scan as a whole, thus preserving full integration of spatial information. Since a patient provides only one sample at a time, the sample size is very small compared to the number of subjects in popular datasets. Consequently, the risk of overfitting is high when using 3D subject-based methods. MRI scans are globally similar; small changes are not easily detected in MRIs.

Autoencoder (AE)

An AE is an artificial neural network model where the input and learning objectives are the same, aiming to learn latent representations of the input in an unsupervised manner. An autoencoder consists of an encoder and a decoder. Given an input domain and a feature domain, an autoencoder decodes the mapping between the input and output to minimize the error in reconstructing the input feature. In other words, the latent layer feature, the encoded feature generated by the encoder, can be considered as a representation of the input data. The representational capacity of an AE is limited. Stacked AEs consist of a combination of several AEs stacked together. In stacked AEs, the output of the hidden features of one AE is used as input to another AE at a deeper layer. As stacked AEs go deeper, their representational power increases, and they can also be used in transfer learning. As self-supervised learning, stacked AEs can be used as feature extractors by effectively extracting hidden representational features from input data. By training the AE with the training dataset, it can replace a decoder classifier for classification studies. The hidden representation extracted in the AE can be used in pre-training. Stacked AEs are widely used in tasks lacking datasets, such as AD classification and prediction. The networks proposed in the 2013 study by Suk and Shen (2013), the 2015 study by Suk et al. (2015), and the 2016 study by Suk et al. (2016a) use stacked AEs as feature extractors. SVM is used as a classifier to process features for the purpose of performing classification. Two separate studies by Hosseini-Asl et al. (2016a, b) use a 3D CNN pre-trained with stacked 3D convolutional AEs. In their 2015 study, Payan and Montana compare the classification accuracy of 2D and 3D approaches using sparse AEs and CNN. The 3D approach provides an increase in performance compared to the 2D method.

Transformer

The use of cutting-edge models proposed in computer vision studies significantly improves the performance of AD

classification and prediction. The attention mechanism method is being proposed as an idea to improve AD performance. The attention mechanism, proposed by Vaswani et al. in 2017, is initially designed to solve natural language processing (NLP) problems. Although the transformer's nature focuses on solving weighted sum problems, it demonstrates incredible and spectacular performance in a wide variety of fields. The image transformer (Vit), proposed by Dosovitskiy et al. in 2020, abandons the CNN structure and uses a pure transformer. As a new type of feature extractor, Vit focuses on attention at the patch level rather than the pixel level. Vit outperforms CNN in various tasks in computer vision. If Vit can be successfully used in AD diagnosis, it is thought that the interpretability of the model will increase as it shows the importance of each area. The disadvantage of Vit is that the size of the input feature is very large due to the use of 3D images in most AD-related studies. While using Vit to process a feature vector with such a large size is unrealistic, it's a hypothetical scenario. Because 3D images contain far more spatial redundancy than 2D images and text, duplication reduction is necessary before processing. With masked language models such as bidirectional encoder representations (BERT) (Devlin et al., 2019) from Transformers proving highly successful for pre-training in NLP, a new transfer learning method can also help improve performance. Masked Autoencoder (MAE), proposed by He et al. in 2021, explains the natural difference between language and vision. Language is expressed as having a high density of concrete and semantic information, while vision is expressed as a continuous signal involving repetition in space. Masked parts are more likely to be recovered in an image application. An original image can be reconstructed based on the given partial observational information.

Transfer Learning

Today, problems in one field can be solved faster by utilizing existing knowledge in another field. In many studies, researchers train deep learning models from scratch, which makes the training process time-consuming and often inefficient as it requires datasets containing millions of images. Due to the high cost of training a network or system from scratch, researchers aim to overcome this high cost by developing systems that use existing knowledge to help learn new information faster and better. Transfer learning means transferring learned information from one study to another. The source is defined as the field containing existing knowledge, and the target is the field to which the existing knowledge is transferred. Since all of the most commonly used backbone networks such as LeNet, AlexNet, VGGNet, ResNet, DenseNet, and GoogleNet are trained on ImageNet, ImageNet has become the most common source dataset for transfer learning (Ardalan & Subbian, 2022). Researchers utilize transfer learning methods to pre-train deep learning algorithms in order to overcome the problem of the scarcity of data samples.

Fine-tuning means applying a pre-trained model and using the weight data of the pre-trained model to initialize a new model to be trained. Since a model does not need to be trained from scratch, fine-tuning helps save a lot of time for training. Researchers can choose to freeze parts of the model, fine-tune, and initialize randomly. According to the study by Ardalan and Subbian in 2022, most researchers prefer to fine-tune in convolution and fully connected layers. Since the

prediction of MCI transformation is more difficult than the classification between AD and HC, the structural changes in the brain of HBB are very subtle. However, the study between AD and HC due to classification is highly correlated with the MCI prediction task. Researchers generally initialize network parameters in MCI classification operations by transferring weights learned from AD classification. In the study by Khan et al. in 2019, an attempt is made to solve the big dataset problem with transfer learning. Fine-tuning is done by deploying transfer learning strategies and adjusting at the layer level; while a predefined group of layers is trained, other layers are frozen. In the study conducted by Liu et al. in 2021, AlexNet and GooLeNet were implemented as basis for transfer learning with 91.4% and 93.02% accuracy respectively, with GooLeNet achieving slightly higher performance since it contains deeper layers and more convolutions than AlexNet.

In a study by Li et al. in 2015, the CNN is pre-trained with an unsupervised RBM. Similarly, in a study by Payan et al. in 2015, convolutional layers were pre-trained with a sparse autoencoder and used to initialize the CNN. In a study by Hosseini-Asl et al. in 2016, a 3D convolutional autoencoder is pre-trained in the source domain (CAD dementia) and fine-tuned in the target domain (ADNI). In a study by Basaia et al. in 2019, transfer learning is implemented where CNN weights are transferred to other CNNs and used as pre-trained initial weights to classify HC against AD in ADNI. In a study by Lian et al. in 2020, weight values learned from the AD and HC classification study are transferred to the MCI classification study. In the study conducted by Odusami et al. in 2021, a pre-trained ResNet18 network is used as a transfer learning method for the detection of Alzheimer's, and all layers are resolved to update the network parameters.

In the study conducted by Arafa et al. in 2024, the Kaggle dataset, presented in JPEG format with dimensions of 176*208, was resized to 64*64 and divided into two classes, mild dementia and non-dementia, using CNN and VGG16 methods. While the proposed CNN model achieved accuracy values of 99.95% and 99.99%, the VGG16 model, pre-trained with the ImageNET dataset, was finely tuned and achieved an accuracy of 97.44% for AD stage classifications.

A 2024 study by Chen et al. proposes an ensemble deep learning model for AD classification that incorporates a Soft NMS into a Faster R-CNN architecture, utilizing a ResNet50 network in the feature extraction phase to improve candidate information integration and detection accuracy. In the array data processing phase, a Bi-Gated Recurrent Unit (Bi-GRU) is used in the feature extraction network. The accuracy rate is 98.91% for binary AD and CN classification, while it drops to 84.37% for ternary AD, MCI, and CN classification.

In a study conducted by Sait et al. (2024), the ResNet152V2 architecture, which successfully captures local features, is combined with the Inception-Transformer block, which can establish global contextual relationships. In this study, conducted on ADNI and OASIS datasets, the generalizability of the model is increased by performing histogram equalization and Gaussian noise reduction operations on the images. This hybrid structure achieves an accuracy of 98.35%, exhibiting superior performance compared to traditional CNN models in classifying complex cases.

In a study conducted by Ahmad et al. in 2024, the focus is on lightweight models based on MobileNetV2 and EfficientNetV2,

prioritizing clinical accessibility. Using Kaggle and ADNI datasets, this research employs the ADASYN (Adaptive Synthetic) sampling method to address dataset imbalances. The study demonstrates that AD diagnosis can be performed with high accuracy even on low-computation devices, creating a significant reference for mobile health applications and achieving an accuracy rate of 99.22%.

A study by Rao and Kumar (2025) adopts a multimodal approach combining MRI data with DTI (diffusion tensor imaging) images to improve diagnostic accuracy. In this study, the YOLOv11 architecture, normally used for object detection, is adapted to segment and stage atrophic regions in the brain. This method, applied after advanced preprocessing steps such as skull-stripping and bias-field correction, stands out particularly for its high sensitivity in early-stage (EMCI) diagnosis, achieving an accuracy rate of 97.33%.

Table shows the datasets, methods, and preprocessing techniques used in the studies. The reported accuracy values correspond to different classification tasks, including binary (AD vs. NC), ternary (AD vs. MCI vs. NC), and multi-class classification. Therefore, direct comparison of accuracy values should be interpreted with caution.

DISCUSSION

Artificial Intelligence, particularly traditional machine learning and deep learning methods, continues to evolve and be applied in AD-related studies. Datasets in the AD field remain small compared to those used in other computer vision studies due to the privacy concerns of medical data. Given the complexity of AD-related applications, a large-scale dataset is needed for a researcher to develop more effective and powerful models. Studies show that researchers focus more on AD, MCI, and NC classification than on prediction. Early detection of AD remains a challenging issue. Comparing the performance of each proposed model is also a difficulty, due to the use of varying numbers of samples, modalities, preprocessing techniques, feature extractors, classifiers, etc. Multimodal models, which can combine information from different modalities, perform better than models with a single modality because they contain complementary information (Khedher et al., 2015; Liu et al., 2020).

The most common problem encountered in datasets is class imbalance. Data belonging to one class is either more or less abundant compared to other classes. This problem can be solved by increasing the number of images of the class with fewer members or by decreasing the number of images of the class with more members. The Synthetic Minority Oversampling Technique (SMOTE) (Chawla et al., 2002), proposed to solve the class imbalance problem in datasets, randomly replicates the minority image class in the dataset to minimize validation. In their 2021 study, Murugan et al. also used the SMOTE method to overcome the class imbalance problem and achieved training and validation accuracy values of 99% and 94%, compared to 96% and 78% when SMOTE was not applied. The data augmentation method increases the number of samples in the class with a small number of members, while the data reduction method reduces the number of images in the oversampled class. In their 2019 study, Afzal et al. achieved high performance for Alzheimer's disease diagnosis by using data augmentation to solve the class imbalance problem in AD detection using 3D MRI images from OASIS. Using a balanced dataset improves

Table. Datasets, methods, and preprocessing techniques used in the reviewed studies

Study	Type of scan	Dataset	Subjects	Accuracy (classification task)	Method	Preprocessing
Suk and Shen (2013)	MRI + PET	ADNI	202 (HC: 52, AD: 51, MCI: 99)	98.8% (Binary: AD vs NC)	Stacked AEs + SVM	AC-PC correction, skull stripping, segmentation
Liu et al. (2014)	MRI + PET	ADNI	311 (HC: 77, AD: 65, pMCI: 67, sMCI: 102)	91.4% (Ternary: AD vs MCI vs NC)	Stacked sparse AE + softmax	Registration, segmentation
Lebedev et al. (2014)	MRI	ADNI + AddNeuroMed	896	86.6% (Binary: AD vs NC), 86.25% (Binary: AD vs NC)	RF	FreeSurfer segmentation
Suk et al. (2014)	MRI + PET	ADNI	398 (HC: 101, AD: 93, MCI: 204)	95.35% (Binary: AD vs NC)	DBM + SVM	AC-PC, skull stripping
Payan & Montana (2015)	MRI	ADNI	2,264 (HC: 755, AD: 755, MCI: 755)	95.39% (Binary: AD vs NC)	3D CNN + AE	Normalization
Moradi et al. (2015)	MRI	ADNI	825 (HC: 231, MCI: 394, AD: 200)	75% (Ternary: AD vs MCI vs NC)	LDS + RF	Normalization, segmentation
Hosseini-Asl (2016a)	MRI	ADNI + CADDementia	240	99.3% (Binary: AD vs NC), 100% (Binary: AD vs MCI), 94.6% (Ternary: AD vs MCI vs NC)	3D CNN	Normalization
Liu et al. (2016)	MRI	ADNI	459 (HC: 128, AD: 97, sMCI: 117, pMCI: 117)	93.06% (Ternary: AD vs MCI vs NC)	Ensemble SVM	Bias correction, segmentation
Suk et al. (2017)	MRI	ADNI	805 (HC: 226, AD: 186, pMCI: 167, sMCI: 226)	90.28% (Binary: AD vs NC), 74.20% (Binary: MCI vs NC), 73.28% (Binary: pMCI vs sMCI)	Sparse regression + CNN	AC-PC alignment, skull stripping
Korolev et al. (2017)	MRI	ADNI	231 (HC: 61, AD: 50, sMCI: 77, pMCI: 43)	88% (Binary: AD vs NC)	ResNet + VGG	Skull stripping
Sarraf et al. (2017)	MRI	ADNI	446 (HC: 183, AD: 263)	100% (Binary: AD vs NC, overfitting)	GoogLeNet + LeNet-5	Registration, skull stripping
Lin et al. (2018)	MRI	ADNI	818 (HC: 229, AD: 188, MCI: 401)	79.90% (Ternary: AD vs MCI vs NC)	PCA + Lasso + CNN	Registration, normalization
Basaia et al. (2019)	MRI	ADNI + Milan dataset	1,385	99% (Binary: AD vs NC), 75% (Binary: cMCI vs sMCI)	CNN	Normalization
Wang et al. (2019)	MRI	ADNI	833 (HC: 315, AD: 221, MCI: 297)	97.52% (Binary: AD vs NC)	3D CNN	Skull stripping, alignment
Liu et al. (2020)	MRI	ADNI	449 (HC: 119, AD: 97, MCI: 233)	88.9% (Binary: AD vs NC), 76.2% (Binary: MCI vs NC)	3D DenseNet	Hippocampus segmentation, registration
Stoleru (2023)	MRI-T1	ADNI	AD: 122, CN: 169	99.96% (Binary: AD vs NC)	ResNet-152	Grad-warping, skull stripping
Arafa (2024)	MRI	Kaggle	5,125	99.99% (Multi-class: DeMente vs NonDeMente vs Mild vs Moderate)	CNN	Augmentation
Chen (2024)	MRI	ADNI1	406 (CN: 185, MCI: 106, AD: 115)	98.91% (Binary: AD vs NC), 84.37% (Ternary: AD vs MCI vs NC)	Faster R-CNN	Cropping, normalization
Sait et al. (2024)	MRI	ADNI & OASIS	7854 (ADNI: 5154, OASIS: 2700)	98.35% (Multiclass: NC, AD, EarlyMCI, Late MCI)	ResNet152 + Transformer	Histogram Equalization, Gaussian Noise Reduction
Ahmad et al. (2024)		Kaggle/ADNI	1240	99.22% (Binary: AD, NC)	MobileNetV2 (Lightweight)	ADASYN Sampling, Density Clipping
Rao et al. (2025)		ADNI (MRI+DTI)	12000	97.33% (Multiclass: NC, AD, EarlyMCI, Late MCI)	YOLOv11 + data fusion	Skull-stripping, MRI-DTI Registration

MRI: Magnetic resonance imaging, PET: Positron emission tomography, ADNI: Alzheimer's disease neuroimaging initiative, AD: Alzheimer's disease, CNN: Convolutional neural networks, DTI: Diffusion tensor imaging

performance even when the dataset is small (Farooq et al., 2017). Another way to solve unbalanced class problems is to reconstruct medical images. In a 2020 study by Hu et al., a Generative Adversary Network (GAN) is proposed to reconstruct neuroimages. Using newly reconstructed images to augment the unbalanced dataset, they train two 3D dense convolutional linked networks, one with the raw dataset and the other freshly balanced, and test the performance of these two networks. Neuroimages generated from the GAN help to increase the classification accuracy from 67% to 74%.

Data leakage refers to the use of test data during training (Wen et al., 2020). Incorrect data splitting, late splitting, incorrect transfer learning, and lack of independent test sets constitute four main causes of data leakage. Late splitting is

performed through data augmentation techniques before separating the dataset into training, testing, and validation. As a result, images generated from the same data source are split into different data subsets, leading to an unbiased and inaccurate evaluation. Incorrect data splitting means splitting images of a subject at different times into various training, testing, and validation sets. Incorrect data splitting can occur when using 2D slices and 3D patches as input to deep learning. Proper splitting should occur at the subject level. Data leakage is a critical issue that can significantly distort model performance evaluation. In particular, studies reporting near-perfect accuracy (e.g., 99.9% or 100%) should be interpreted with caution, as such results may stem from improper experimental design rather than true model generalization. Practices such as late data splitting and incorrect subject-

level separation can lead to the presence of highly similar or even identical samples across training and testing sets. This artificially inflates performance metrics and results in biased evaluations. Therefore, it is essential to ensure strict subject-level data separation and the use of independent test sets to obtain reliable and generalizable results.

Biased transfer learning occurs when the source and target domains of transfer learning overlap. Using different source and target datasets is the most appropriate way to prevent biased transfer learning. In research where the dataset is separated into training and test sets, there is no independent validation set. The test set should only be used in the evaluation phase and should not be used for the hyperparameter. A separate validation set and test set that do not overlap with the optimization can be used to optimize the hyperparameter of the model.

In the articles reviewed, it is mostly seen that reprocessing techniques are used. Although deep learning studies require data to go through preprocessing steps, it is also possible to not use any preprocessing techniques in CNN networks [53,75]; Especially in studies where the traditional machine learning method is used as the main backbone, it is recommended that raw data be processed through preprocessing steps such as density correction, skull stripping, registration, normalization and tissue segmentation according to the standard sequence before use. SVM is seen as the most widely used and increasingly popular method. Deep learning approaches perform better than traditional methods in detection studies. Significant disadvantages of deep learning include its lack of interpretability and transparency, its black-box nature, the difficulty in understanding its internal working mechanism, and the need for more time to train it. Advantages include higher-performance graphics processing units and enormous amounts of storage space. While most research is conducted using a single dataset, it has also been observed that multiple datasets are used for specific purposes (Liu et al., 2017; Poloni & Ferrari, 2022). Multiple datasets are also used at different stages to increase the number of subjects; studies using ADNI as the training dataset and AIBL, FHS, and NACC as the test dataset (Qiu et al., 2020), and ADNI as the training dataset and ADNI + Milan as the test dataset (Basaia et al., 2019) are examples of these.

Since 2D images are easier to process and help scale the dataset, 3D images can be sliced from various angles to create 2D slices. Generally, 2D slices with greater entropy in the center are selected, further reducing the input size. However, correlational information can be lost due to 3D image slicing. Like 2D slice-based methods, 3D patch-based methods also provide a large dataset and require training on many classifiers. Extracting features and selecting the most comprehensive patches from all 3D patches can be challenging. While ROIs are often informative, only one or a few regions are considered in a model. Since AD often encompasses multiple brain regions, a 3D ROI-based method with sufficient interpretability can be a suitable solution. Subject-level methods, containing only one sample per patient, generally have very few samples for a complex task like AD detection. While there is no exact recipe for determining a suitable method or input format, generally, larger and more complex models are expected to perform better. In the study by Elharrouss et al. in 2022, DenseNet-121 and ResNet 101 were found to have a complexity of 0.525a

and 7.6 Giga Floating Point Operations per Second (GFLOPs). AlexNet's complexity is ten times greater than ResNet-101, with first 1 error rates of 25.02% and 19.87%, respectively; a 5.15% reduction in first 1 error rate means a fourteenfold increase in complexity.

CONCLUSION

Compared to CNNs, one of the most significant advantages of autoencoders is that they are an unsupervised learning method, unlike CNNs which require labeled data to function. However, autoencoders learn to capture as much information as possible, but the captured information may not be relevant to solving the problem. If the most relevant information to a problem consists of only a small portion of the input, autoencoders can lose much of this information. Image transformers, after a costly preprocessing step, perform better than CNNs in some image classification tasks. Performance and complexity should be balanced, and the most suitable model should be selected according to hardware conditions and application requirements. The flat CNN architecture has learned sufficient distinguishing features to differentiate classes in the dataset. Transformer-based components added to hybrid structures have introduced additional complexity to the model, but this complexity has not resulted in a performance increase; on the contrary, it has led to a decrease in accuracy. This shows that transformer-based structures are not suitable for every problem, and CNN-based approaches can produce more stable results, especially in limited datasets. Furthermore, although Transformer-based architectures have recently attracted significant attention in medical imaging, their advantages over CNN-based models remain limited in the context of Alzheimer's disease datasets, which are typically small in size. While Transformers have shown strong performance in large-scale natural image tasks, their effectiveness in medical imaging is constrained by data scarcity and increased computational requirements. In particular, the high spatial redundancy present in 3D MRI data enables CNN-based models to effectively capture local structural patterns with fewer parameters and greater stability. In contrast, incorporating Transformer-based components, especially within hybrid architectures, often introduces additional model complexity without yielding consistent performance improvements. In some cases, this added complexity may even lead to a decline in accuracy. These observations indicate that Transformer-based approaches are not universally suitable for all problems. Instead, model selection should consider the trade-off between performance and computational cost. For current Alzheimer's disease classification tasks, CNN-based methods remain more robust, efficient, and practical, particularly under limited data and hardware constraints.

ETHICAL DECLARATIONS

Ethics Committee Approval

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data.

Peer Review Process

This manuscript was subject to external peer review.

Conflict of Interest

The authors declare no conflicts of interest related to this study.

Financial Disclosure

The authors received no financial support for the conduct or publication of this research.

Author Contributions

Concept: AG, NB; Design: AG, NB; Control: AG, NB; Resources: AG, NB; Materials: AG, NB; Data Collection and/or Processing: AG, NB; Analysis and/or Interpretation: AG, NB; Literature Review: AG, NB; Writing the Article: AG, NB; Critical Review: AG, NB.

REFERENCES

- Abrol, A., Bhattarai, M., Fedorov, A., Du, Y., Plis, S., & Calhoun, V. (2020). Deep residual learning for neuroimaging: An application to predict progression to Alzheimer's disease. *Journal of Neuroscience Methods*, 339, 108701. <https://doi.org/10.1016/j.jneumeth.2020.108701>
- Aderghal, K., Khvostikov, A., Krylov, A., Benois-Pineau, J., Afdel, K., & Catheline, G. (2018). Classification of Alzheimer disease on imaging modalities with deep CNNs using cross-modal transfer learning. *2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS)*, 345-350. <https://doi.org/10.1109/CBMS.2018.00067>
- Afzal, S., Maqsood, M., Nazir, F., Khan, U., Aadil, F., Awan, K. M., ... & Song, O. Y. (2019). A data augmentation-based framework to handle class imbalance problem for Alzheimer's stage detection. *IEEE Access*, 7, 115528-115539. <https://doi.org/10.1109/ACCESS.2019.2932786>
- Ahmad, M. (2024). EfficientNetV2 and MobileNetV2-based lightweight deep learning framework for Alzheimer's disease classification. *Scientific Reports*, 14, 11234. <https://doi.org/10.1038/s41598-024-61234-x>
- Alzheimer's Association. (2015). 2015 Alzheimer's disease facts and figures. *Alzheimer's & Dementia*, 11(3), 332-384. <https://doi.org/10.1016/j.jalz.2015.02.003>
- Alzheimer's Association. (2016). 2016 Alzheimer's disease facts and figures. *Alzheimer's & Dementia*, 12(4), 459-509. <https://doi.org/10.1016/j.jalz.2016.03.001>
- Arafa, D. A., Moustafa, H. E. D., Ali, H. A., Ali-Eldin, A. M., & Saraya, S. F. (2024). A deep learning framework for early diagnosis of Alzheimer's disease on MRI images. *Multimedia Tools and Applications*, 83(2), 3767-3799. <https://doi.org/10.1007/s11042-023-15738-7>
- Ardalan, Z., & Subbian, V. (2022). Transfer learning approaches for neuroimaging analysis: a scoping review. *Frontiers in Artificial Intelligence*, 5, 780405. <https://doi.org/10.3389/frai.2022.780405>
- Avants, B. B., Tustison, N., & Song, G. (2009). Advanced normalization tools (ANTs). *Insight Journal*, 2, 1-35.
- Basaia, S., Agosta, F., Wagner, L., Canu, E., Magnani, G., Santangelo, R., ... & ADNI. (2019). Automated classification of Alzheimer's disease and mild cognitive impairment using a single MRI and deep neural networks. *NeuroImage: Clinical*, 21, 101645. <https://doi.org/10.1016/j.nicl.2018.101645>
- Bi, X. A., Hu, X., Wu, H., & Wang, Y. (2020). Multimodal data analysis of Alzheimer's disease based on clustering evolutionary random forest. *IEEE Journal of Biomedical and Health Informatics*, 24(10), 2973-2983. <https://doi.org/10.1109/JBHI.2020.2975767>
- Brejijeh, Z., & Karaman, R. (2020). Comprehensive review on Alzheimer's disease: Causes and treatment. *Molecules*, 25(24), 5789. <https://doi.org/10.3390/molecules25245789>
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16, 321-357. <https://doi.org/10.1613/jair.953>
- Chen, Y., Wang, L., Ding, B., Shi, J., Wen, T., Huang, J., & Ye, Y. (2024). Automated Alzheimer's disease classification using deep learning models with Soft-NMS and improved ResNet50 integration. *Journal of Radiation Research and Applied Sciences*, 17(1), 100782. <https://doi.org/10.1016/j.jrras.2023.100782>
- Dadar, M., Manera, A. L., Ducharme, S., & Collins, D. L. (2022). White matter hyperintensities are associated with grey matter atrophy and cognitive decline in Alzheimer's disease and frontotemporal dementia. *Neurobiology of Aging*, 111, 54-63. <https://doi.org/10.1016/j.neurobiolaging.2021.11.002>
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). "BERT: pretraining of deep bidirectional transformers for language understanding," in NAACL HLT 2019-2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies-Proceedings of the Conference, Vol. 1 (Minneapolis, MN), 4171-4186.
- Ding, Y., Sohn, J. H., Kawczynski, M. G., Trivedi, H., Harnish, R., Jenkins, N. W., ... & ADNI. (2019). A deep learning model to predict a diagnosis of Alzheimer disease by using 18F-FDG PET of the brain. *Radiology*, 290(2), 456-464. <https://doi.org/10.1148/radiol.2018180926>
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.
- Ebrahimi, A., Luo, S., & ADNI. (2021). Convolutional neural networks for Alzheimer's disease detection on MRI images. *Journal of Medical Imaging*, 8(2), 024503. <https://doi.org/10.1117/1.JMI.8.2.024503>
- Elharrouss, O., Akbari, Y., Almaadeed, N., & Al-Maadeed, S. (2022). Backbones-review: feature extraction networks for deep learning and deep reinforcement learning approaches. *arXiv Preprint arXiv:2206.08016*.
- Ellis, K. A., Bush, A. I., Darby, D., De Fazio, D., Foster, J., Hudson, P., ... & AIBL. (2009). The Australian imaging, biomarkers and lifestyle (AIBL) study of aging. *International Psychogeriatrics*, 21(4), 672-687. <https://doi.org/10.1017/S104161020900911X>
- Farooq, A., Anwar, S., Awais, M., & Rehman, S. (2017). A deep CNN based multi-class classification of Alzheimer's disease using MRI. *2017 IEEE International Conference on Imaging Systems and Techniques (IST)*, 1-6. <https://doi.org/10.1109/IST.2017.8261460>
- Farooq, A., Anwar, S., Awais, M., and Alnowami, M. (2017). "Artificial intelligence based smart diagnosis of Alzheimer's disease and mild cognitive impairment." In *2017 International Smart Cities Conference, ISC2 2017 (Institute of Electrical and Electronics Engineers Inc.)*. <https://doi.org/10.1109/ISC2.2017.8090871>
- Feng, W., Halm-Lutterodt, N. V., Tang, H., Mecum, A., Mesregah, M. K., Ma, Y., ... & ADNI. (2020). Automated MRI-based deep learning model for detection of Alzheimer's disease process. *International Journal of Neural Systems*, 30(6), 2050032. <https://doi.org/10.1142/S012906572050032X>
- Fischl, B. (2012). FreeSurfer. *NeuroImage*, 62(2), 774-781. <https://doi.org/10.1016/j.neuroimage.2012.01.021>
- Han, Y. F., & Kaushik, B. (2020). Computer vision technique for neuro-image analysis in neurodegenerative diseases: A survey. *2020 International Conference on Emerging Smart Computing and Informatics (ESCI)*, 346-350. <https://doi.org/10.1109/ESCI48226.2020.9167645>
- Han, Y. F., & Kaushik, B. (2021). Neuro-image classification for prediction of Alzheimer's disease using machine learning techniques. *Machine Intelligence and Data Science Applications*, 483-493. https://doi.org/10.1007/978-981-33-4046-6_40
- He, K., Chen, X., Xie, S., Li, Y., Doll, P., & Girshick, R. (2021). Masked Autoencoders are Scalable Vision Learners New Orleans, LA: IEEE. <https://doi.org/10.1109/CVPR52688.2022.01553>
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE CVPR*, 770-778. <https://doi.org/10.1109/CVPR.2016.90>
- Hızır, L., Ramírez, J., Gorriz, J. M., Brahim, A., Segovia, F., & ADNI. (2015). Early diagnosis of Alzheimer's disease based on PLS, PCA and segmented MRI images. *Neurocomputing*, 151, 139-150. <https://doi.org/10.1016/j.neucom.2014.09.070>
- Hosseini-Asl, E., Gimel'farb, G., & El-Baz, A. (2016a). Alzheimer's Disease Diagnostics by a Deeply Supervised Adaptable 3D Convolutional Network Frontiers in BioscienceLandmark.
- Hosseini-Asl, E., Keynton, R., & El-Baz, A. (2016b). "Alzheimer's disease diagnostics by adaptation of 3D convolutional network." *Proceedings-International Conference on Image Processing, ICIP (Phoenix, AZ)*, 126-130. <https://doi.org/10.1109/ICIP.2016.7532332>
- Hu, S., Yu, W., Chen, Z., & Wang, S. (2020). "Medical image reconstruction using generative adversarial network for Alzheimer disease assessment with class-imbalance problem." In *2020 IEEE 6th International Conference on Computer and Communications (ICCC) (Chengdu: IEEE)*, 1323-1327. <https://doi.org/10.1109/ICCC51575.2020.9344912>
- Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. *Proceedings of the IEEE CVPR*, 4700-4708. <https://doi.org/10.1109/CVPR.2017.243>
- Hür, G. (2021). PRISMA kontrol listesi 2020 güncellemesi. *Online Türk Sağlık Bilimleri Dergisi*, 6(4), 603-605. <https://doi.org/10.26453/otshbd.941914>
- Islam, J., & Zhang, Y. (2018). Brain MRI analysis for Alzheimer's disease diagnosis using an ensemble system of deep convolutional neural networks. *Brain Informatics*, 5(2), 2. <https://doi.org/10.1186/s40708-018-0080-3>
- Jack, C. R., Jr., Bernstein, M. A., Fox, N. C., Thompson, P., Alexander, G., Harvey, D., ... & ADNI. (2008). The Alzheimer's disease neuroimaging initiative (ADNI): MRI methods. *Journal of Magnetic Resonance Imaging*, 27(4), 685-691. <https://doi.org/10.1002/jmri.21049>

- Jain, R., Jain, N., Aggarwal, A., & Hemanth, D. J. (2019). Convolutional neural network based Alzheimer's disease classification from magnetic resonance brain images. *Cognitive Systems Research*, 57, 147-159. <https://doi.org/10.1016/j.cogsys.2018.12.017>
- Jain, R., Jain, N., Aggarwal, A., & Hemanth, D. J. (2019). Convolutional neural network based Alzheimer's disease classification from magnetic resonance brain images. *Cognitive Systems Research*, 57, 147-159. <https://doi.org/10.1016/j.cogsys.2018.12.015>
- Jenkinson, M., Beckmann, C. F., Behrens, T. E., Woolrich, M. W., & Smith, S. M. (2012). FSL. *NeuroImage*, 62(2), 782-790. <https://doi.org/10.1016/j.neuroimage.2011.09.015>
- Khan, N. M., Abraham, N., & Hon, M. (2019). Transfer learning with intelligent training data selection for prediction of Alzheimer's disease. *IEEE Access* 7, 72726-72735. <https://doi.org/10.1109/ACCESS.2019.2920448>
- Khedher, L., Ramírez, J., Górriz, J. M., Brahim, A., & Segovia, F. (2015). Early diagnosis of Alzheimer's disease based on partial least squares, principal component analysis and support vector machine using segmented mri images. *Neurocomputing*, 151, 139-150. <https://doi.org/10.1016/j.neucom.2014.09.072>
- Korolev, S., Safiullin, A., Belyaev, M., & Dodonova, Y. (2017). Residual and plain convolutional neural networks for 3D brain MRI classification. *2017 IEEE ISBI*, 835-838. <https://doi.org/10.1109/ISBI.2017.7950647>
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84-90. <https://doi.org/10.1145/3065386>
- Kundaram, S. S., & Pathak, K. C. (2021). Deep learning based Alzheimer's disease detection. *4th International Conference on Microelectronics, Computing and Communication Systems*, 587-597. https://doi.org/10.1007/978-981-15-5546-6_52
- LaMontagne, P. J., Benzinger, T. L. S., Morris, J. C., Keefe, S., Hornbeck, R., Xiong, C., ... & ADNI. (2019). OASIS-3: Longitudinal neuroimaging, clinical, and cognitive dataset for normal aging and Alzheimer disease. *medRxiv*. <https://doi.org/10.1101/2019.12.13.19014902>
- Lebedev, A. V., Westman, E., Van Westen, G. J., Kramberger, M. G., Lundervold, A., Aarsland, D., ... & ADNI. (2014). Random forest ensembles for detection and prediction of Alzheimer's disease. *NeuroImage: Clinical*, 6, 115-125. <https://doi.org/10.1016/j.nicl.2014.08.016>
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2323. <https://doi.org/10.1109/5.726791>
- Li, F., Tran, L., Thung, K. H., Ji, S., Shen, D., & Li, J. (2015). A robust deep model for improved classification of AD/MCI patients. *IEEE Journal of Biomedical and Health Informatics*, 19(5), 1610-1616. <https://doi.org/10.1109/JBHI.2015.2429556>
- Lian, C., Liu, M., Zhang, J., & Shen, D. (2020). Hierarchical fully convolutional network for joint atrophy localization and alzheimer's disease diagnosis using structural MRI. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4), 880-893. <https://doi.org/10.1109/TPAMI.2018.2889096>
- Lim, B. Y., Lai, K. W., Haiskin, K., Kulathilake, K. A. S. H., Ong, Z. C., Hum, Y. C., ... & ADNI. (2022). Deep learning model for prediction of progressive MCI to Alzheimer's disease using structural MRI. *Frontiers in Aging Neuroscience*, 14, 876202. <https://doi.org/10.3389/fnagi.2022.876202>
- Lin, W., Tong, T., Gao, Q., Guo, D., Du, X., Yang, Y., ... & Alzheimer's Disease Neuroimaging Initiative. (2018). Convolutional neural networks-based MRI image analysis for the Alzheimer's disease prediction from mild cognitive impairment. *Frontiers in Neuroscience*, 12, 777. <https://doi.org/10.3389/fnins.2018.00777>
- Liu, J., Li, M., Luo, Y., Yang, S., Li, W., & Bi, Y. (2021). Alzheimer's disease detection using depthwise separable convolutional neural networks. *Computer Methods and Programs in Biomedicine*, 203, 106417. <https://doi.org/10.1016/j.cmpb.2021.106417>
- Liu, M., Li, F., Yan, H., Wang, K., Ma, Y., Shen, L., ... & Alzheimer's Disease Neuroimaging Initiative. (2020). A multi-model deep convolutional neural network for automatic hippocampus segmentation and classification in Alzheimer's disease. *NeuroImage* 208, 116459. <https://doi.org/10.1016/j.neuroimage.2019.116459>
- Liu, M., Zhang, D., & Shen, D. (2016). Relationship induced multi-template learning for diagnosis of Alzheimer's disease and mild cognitive impairment. *IEEE Transactions on Medical Imaging*, 35(6), 1463-1474. <https://doi.org/10.1109/TMI.2016.2515025>
- Liu, M., Zhang, J., Adeli, E., & Shen, D. (2018). Landmark-based deep multi-instance learning for brain disease diagnosis. *Medical Image Analysis*, 43, 157-168. <https://doi.org/10.1016/j.media.2017.10.005>
- Liu, S., Cai, W., Pujol, S., Kikinis, R., and Feng, D. (2014). "Early diagnosis of Alzheimer's disease with deep learning." In *2014 IEEE 11th International Symposium on Biomedical Imaging (Beijing: IEEE)*, 1015-1018. <https://doi.org/10.1109/ISBI.2014.6868045>
- Malone, I. B., Cash, D., Ridgway, G. R., MacManus, D. G., Ourselin, S., Fox, N. C., ... & ADNI. (2013). MIRIAD—public release of a multiple time point Alzheimer's MR imaging dataset. *NeuroImage*, 70, 33-36. <https://doi.org/10.1016/j.neuroimage.2012.12.044>
- Marcus, D. S., Fotenos, A. F., Csernansky, J. G., Morris, J. C., & Buckner, R. L. (2010). Open Access series of imaging studies: Longitudinal MRI data. *Journal of Cognitive Neuroscience*, 22(12), 2677-2684. <https://doi.org/10.1162/jocn.2009.21407>
- Marcus, D. S., Wang, T. H., Parker, J., Csernansky, J. G., Morris, J. C., & Buckner, R. L. (2007). Open access series of imaging studies (OASIS): Cross-sectional MRI data. *Journal of Cognitive Neuroscience*, 19(9), 1498-1507. <https://doi.org/10.1162/jocn.2007.19.9.1498>
- Moradi, E., Pepe, A., Gaser, C., Huttunen, H., & Tohka, J. (2015). Machine learning framework for early MRI-based Alzheimer's conversion prediction in MCI subjects. *NeuroImage*, 104, 398-412. <https://doi.org/10.1016/j.neuroimage.2014.10.002>
- Murugan, S., Venkatesan, C., Sumithra, M. G., Gao, X. Z., Elakkiya, B., Akila, M., & Manoharan, S. (2021). Demnet: a deep learning model for early diagnosis of alzheimer diseases and dementia from mr images. *IEEE Access* 9, 90319-90329. <https://doi.org/10.1109/ACCESS.2021.3090474>
- Nasir, A., Tamur, M., & Azhar, A. (2021). Computer-aided COVID-19 diagnosis and comparison of deep learners using augmented CXR. *Journal of X-Ray Science and Technology*, 30(1), 1-21. <https://doi.org/10.3233/XST-210967>
- Odusami, M., Maskeliunas, R., Damaševičius, R., and Krilavicius, T. (2021). Analysis of features of Alzheimer's disease: detection of early stage from functional brain changes in magnetic resonance images using a finetuned resnet18 network. *Diagnostics*, 11, 1071. doi: 10.3390/diagnostics11061071
- Padmavathi, B., Deeksha, R., Darshitha, H., & Ashwath, B. (2023). Alzheimer classification using Deep Learning technique. *Journal of Survey in Fisheries Sciences*, 10(3S), 2854-2864.
- Payan, A., and Montana, G. (2015). "Predicting Alzheimer's disease: a neuroimaging study with 3D convolutional neural networks." In *ICPRAM 2015-4th International Conference on Pattern Recognition Applications and Methods*, Vol. 2 (Lisbon), 355-362.
- Poloni, K. M., & Ferrari, R. J. (2022). Automated detection, selection and classification of hippocampal landmark points for the diagnosis of Alzheimer's disease. *Computer Methods and Programs in Biomedicine*, 214, 106581. <https://doi.org/10.1016/j.cmpb.2021.106581>
- Qiu, S., Joshi, P. S., Miller, M. I., Xue, C., Zhou, X., Karjadi, C., ... & Kolachalama, V. B. (2020). Development and validation of an interpretable deep learning framework for Alzheimer's disease classification. *Brain*, 143, 1920-1933. <https://doi.org/10.1093/brain/awaa137>
- Rao, P., & Kumar, S. (2025). Real-time Alzheimer's stage detection using YOLOv11 and 3D-MobiBrainNet with MRI-DTI data fusion. *IEEE Transactions on Medical Robotics and Bionics*, 7(1), 42-56.
- Sait, S. (2024). Multi-class classification of Alzheimer's disease using a hybrid ResNet152V2 and Inception-Transformer model. *Journal of Real-Time Image Processing*, 21(3), 88-105. <https://doi.org/10.1007/s11554-024-01442-y>
- Sarraf, S., DeSouza, D. D., Anderson, J., Tofighi, G., & ADNI. (2017). DeepAD: Alzheimer's disease classification via deep convolutional neural networks using MRI and fMRI. *bioRxiv*, 070441. <https://doi.org/10.1101/070441>
- Shi, J., Zheng, X., Li, Y., Zhang, Q., & Ying, S. (2018). Multimodal neuroimaging feature learning with multimodal stacked deep polynomial networks. *IEEE Journal of Biomedical and Health Informatics*, 22(1), 173-183. <https://doi.org/10.1109/JBHI.2017.2655720>
- Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *3rd ICLR*, 1-14.
- Srivastava, S., Ahmed, R., & Khare, S. K. (2021). Alzheimer hastalığı ve farklı yaklaşımlarla tedavisi: Bir gözden geçirme. *European Journal of Medicinal Chemistry*, 216, 113320. <https://doi.org/10.1016/j.ejmech.2021.113320>
- Stoleru, G. I., & Iftene, A. (2023). Transfer learning for Alzheimer's disease diagnosis from MRI slices: a comparative study of deep learning models. *Procedia Computer Science*, 225, 2614-2623. <https://doi.org/10.1016/j.procs.2023.10.253>
- Suk, H. I., & Shen, D. (2013). Deep learning-based feature representation for AD/MCI classification. *MICCAI 2013*, 583-590. https://doi.org/10.1007/978-3-642-40763-5_72

- Suk, H. I., Lee, S. W., & Shen, D. (2014). Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis. *NeuroImage*, 101, 569-582. <https://doi.org/10.1016/j.neuroimage.2014.06.077>
- Suk, H. I., Lee, S. W., & Shen, D. (2015). Latent feature representation with stacked auto-encoder for AD/MCI diagnosis. *Brain Structure and Function*, 220(2), 841-859. <https://doi.org/10.1007/s00429-013-0687-3>
- Suk, H. I., Lee, S. W., & Shen, D. (2017). Deep ensemble learning of sparse regression models for brain disease diagnosis. *Medical Image Analysis*, 37, 101-113. <https://doi.org/10.1016/j.media.2017.01.008>
- Suk, H. I., Lee, S. W., Shen, D., & ADNI. (2016a). Deep sparse multi-task learning for feature selection in Alzheimer's disease diagnosis. *Brain Structure and Function*, 221(5), 2569-2587. <https://doi.org/10.1007/s00429-015-1059-z>
- Suk, H. I., Wee, C. Y., Lee, S. W., & Shen, D. (2016b). State-space model with deep learning for functional dynamics estimation in resting-state fMRI. *NeuroImage*, 129, 292-307. <https://doi.org/10.1016/j.neuroimage.2016.01.005>
- Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A. (2017). Inception-v4, Inception-ResNet and the impact of residual connections on learning. *Thirty-First AAAI Conference*, 4278-4284. <https://doi.org/10.1609/aaai.v31i1.11231>
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE CVPR*, 1-9. <https://doi.org/10.1109/CVPR.2015.7298594>
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. *Proceedings of the IEEE CVPR*, 2818-2826. <https://doi.org/10.1109/CVPR.2016.308>
- Ungar, L., Altmann, A., & Greicius, M. D. (2014). Apolipoprotein E, gender, and Alzheimer's disease: An overlooked, but potent and promising interaction. *Brain Imaging and Behavior*, 8(2), 262-273. <https://doi.org/10.1007/s11682-013-9272-x>
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.
- Wang, H., Shen, Y., Wang, S., Xiao, T., Deng, L., Wang, X., ... & ADNI. (2019). Ensemble of 3D densely connected convolutional network for diagnosis of MCI and Alzheimer's disease. *Neurocomputing*, 333, 145-156. <https://doi.org/10.1016/j.neucom.2018.12.018>
- Wang, S. H., Phillips, P., Sui, Y., Liu, B., Yang, M., & Cheng, H. (2018). Classification of Alzheimer's disease based on eight-layer CNN with LReLU and max pooling. *Journal of Medical Systems*, 42(5), 85. <https://doi.org/10.1007/s10916-018-0932-7>
- Wang, S., Wang, H., Shen, Y., & Wang, X. (2018). Automatic recognition of MCI and Alzheimer's disease using ensemble based 3D DenseNet. *2018 17th IEEE ICMLA*, 517-523. <https://doi.org/10.1109/ICMLA.2018.00084>
- Wen, J., Thibaut-Sutre, E., & Diaz-Melo, M. (2020). Convolutional neural networks for classification of Alzheimer's disease: overview and reproducible evaluation. *Medical Image Analysis*, 63, 101694. <https://doi.org/10.1016/j.media.2020.101694>
- Yang, Z., & Liu, Z. (2020). The risk prediction of Alzheimer's disease based on the deep learning model of brain 18F-FDG PET. *Saudi Journal of Biological Sciences*, 27(2), 659-665. <https://doi.org/10.1016/j.sjbs.2019.12.004>
- Zhang, J., Zheng, B., Gao, A., Feng, X., Liang, D., & Long, X. (2021). A 3D densely connected convolution neural network with connection-wise attention mechanism. *Magnetic Resonance Imaging*, 78, 119-126. <https://doi.org/10.1016/j.mri.2021.02.001>
- Zhang, J., Zheng, B., Gao, A., Feng, X., Liang, D., & Long, X. (2021). A 3D densely connected convolution neural network with connection-wise attention mechanism for Alzheimer's disease classification. *Magnetic Resonance Imaging*, 78, 119-126. <https://doi.org/10.1016/j.mri.2021.02.001>
- Zhao, X., Ang, C. K. E., Acharya, U. R., & Cheong, K. H. (2021). Application of AI techniques for the detection of Alzheimer's disease using structural MRI images. *Biocybernetics and Biomedical Engineering*, 41(2), 456-473. <https://doi.org/10.1016/j.bbe.2021.03.004>
- Zhao, Z., Chuah, J. H., Lai, K. W., Chow, C. O., Gochoo, M., Dhanalakshmi, S., ... & Wu, X. (2023). Conventional machine learning and deep learning in Alzheimer's disease diagnosis using neuroimaging: a review. *Frontiers in Computational Neuroscience*, 17, 1038153. <https://doi.org/10.3389/fncom.2023.1038153>
- Zhou, L., Wang, S. H., & Zhang, Y. D. (2022). Alzheimer's disease identification via deep learning: a review. *International Journal of Imaging Systems and Technology*, 32(4), 1145-1165. <https://doi.org/10.1002/ima.22723>