

# BİLGİSAYAR BİLİMLERİ VE MÜHENDİSLİĞİ ALANINDA GÜNCEL KONULAR VE AKADEMİK DEĞERLENDİRMELER

Editör: Dr.Öğr.Üyesi Mehmet Süleyman YILDIRIM



**Bilgisayar Bilimleri ve Mühendisliği Alanında  
Güncel Konular ve Akademik Değerlendirmeler**

**Editör**

Dr.Öğr.Üyesi Mehmet Süleyman YILDIRIM

**yaz**  
yayınları

2026

**Bilgisayar Bilimleri ve Mühendisliği  
Alanında Güncel Konular ve Akademik  
Değerlendirmeler**

Editör: Dr.Öğr.Üyesi Mehmet Süleyman YILDIRIM

---

**© YAZ Yayınları**

Bu kitabın her türlü yayın hakkı Yaz Yayınları'na aittir, tüm hakları saklıdır. Kitabın tamamı ya da bir kısmı 5846 sayılı Kanun'un hükümlerine göre, kitabı yayınlayan firmanın önceden izni alınmaksızın elektronik, mekanik, fotokopi ya da herhangi bir kayıt sistemiyle çoğaltılamaz, yayımlanamaz, depolanamaz.

---

E\_ISBN 978-625-8926-83-5

Ekim 2026 – Afyonkarahisar

Dizgi/Mizanpaj: YAZ Yayınları

Kapak Tasarım: YAZ Yayınları

YAZ Yayınları. Yayıncı Sertifika No: 73086

M.İhtisas OSB Mah. 4A Cad. No:3/3

İscehisar/AFYONKARAHİSAR

[www.yazyayinlari.com](http://www.yazyayinlari.com)

[yazyayinlari@gmail.com](mailto:yazyayinlari@gmail.com)

## İÇİNDEKİLER

|                                                                                                                                                          |            |
|----------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Machine Learning-Based Air Quality Assessment and Explainable Rule Extraction: A Soft-Sensor Framework and Urban Case Study .....</b>                 | <b>1</b>   |
| <i>Aleyna SEVİMLİ, Arzum KARATAŞ</i>                                                                                                                     |            |
| <b>Artificial Intelligence Applications in Multiparametric MRI for Prostate Cancer Detection and Grading .....</b>                                       | <b>18</b>  |
| <i>Mecit YÜZKAT</i>                                                                                                                                      |            |
| <b>Gıda Tedarik Zincirinde Veri Güvenilirliği ve Blokzincir Teknolojisinin Kullanımı .....</b>                                                           | <b>44</b>  |
| <i>Bora ASLAN, Füsün YAVUZER ASLAN</i>                                                                                                                   |            |
| <b>Discriminative Capability and Temporal Generalizability of Static Reverse Engineering Features in Windows X64 Malware Detection .....</b>             | <b>69</b>  |
| <i>Güney UĞURLU</i>                                                                                                                                      |            |
| <b>Towards Intelligent Clinical Decision Support: Extracting Keyphrases from Turkish Patient Texts Via Open-Source Pre-Trained Language Models .....</b> | <b>89</b>  |
| <i>Ezgi ZORARPACI, Zeynep DEMİRTAŞ, Pınar Sarısaray BÖLÜK</i>                                                                                            |            |
| <b>Güç Sistemlerinin Ekonomik Yük Dağıtımının Bir İncelenmesi.....</b>                                                                                   | <b>111</b> |
| <i>Enzel AYDIN, Abdullah Erhan AKKAYA</i>                                                                                                                |            |
| <b>The Effect of Graph Density and Thread Count on the Performance of Malatya Centrality-Based Maximal Independent Set Detection.....</b>                | <b>130</b> |
| <i>Furkan ÖZTEMİZ</i>                                                                                                                                    |            |

|                                                                                                                                                      |            |
|------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Students Experiences of Learning to Prompt<br/>Generative AI: A Meta-Synthesis of Qualitative<br/>Research in K-12 and Higher Education .....</b> | <b>153</b> |
|------------------------------------------------------------------------------------------------------------------------------------------------------|------------|

*Nihan ARSLAN NAMLI*

|                                                                                                                                  |            |
|----------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Deep Learning Versus Statistical Benchmarks in<br/>Forecasting the Global Prevalence of Mental<br/>Disorders To 2030.....</b> | <b>174</b> |
|----------------------------------------------------------------------------------------------------------------------------------|------------|

*Erol ÖZÇEKİÇ*

|                                                                                                                                           |            |
|-------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Akademik Dokümanların Otomatik Analizi: Metin<br/>Sınıflandırma, Bilgi Çıkarımı ve Otomatik Özetleme<br/>Üzerine Bir İnceleme.....</b> | <b>201</b> |
|-------------------------------------------------------------------------------------------------------------------------------------------|------------|

*Erhan AKAGÜNDÜZ, Cengiz HARK, Taner UÇKAN*

|                                                                                                                    |            |
|--------------------------------------------------------------------------------------------------------------------|------------|
| <b>Joint Evaluation of Solution Quality and<br/>Computational Resources in Independent Set<br/>Algorithms.....</b> | <b>212</b> |
|--------------------------------------------------------------------------------------------------------------------|------------|

*Furkan ÖZTEMİZ*

|                                                                                                                                        |            |
|----------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Comparative Evaluation of Malatya Centrality and<br/>Machine Learning Algorithms for Spotify Song<br/>Popularity Analysis .....</b> | <b>236</b> |
|----------------------------------------------------------------------------------------------------------------------------------------|------------|

*Selman YAKUT*

|                                                                                                 |            |
|-------------------------------------------------------------------------------------------------|------------|
| <b>Graph-Based Movie Recommendation Using<br/>Malatya Centrality: A Comparative Study .....</b> | <b>256</b> |
|-------------------------------------------------------------------------------------------------|------------|

*Selman YAKUT*

*"Bu kitapta yer alan bölümlerde kullanılan kaynakların, görüşlerin, bulguların, sonuçların, tablo, şekil, resim ve her türlü içeriğin sorumluluğu yazar veya yazarlarına ait olup ulusal ve uluslararası telif haklarına konu olabilecek mali ve hukuki sorumluluk da yazarlara aittir."*

# **MACHINE LEARNING-BASED AIR QUALITY ASSESSMENT AND EXPLAINABLE RULE EXTRACTION: A SOFT-SENSOR FRAMEWORK AND URBAN CASE STUDY**

**Aleyna SEVİMLİ<sup>1</sup>**

**Arzum KARATAŞ<sup>2</sup>**

## **1. INTRODUCTION**

Air quality classification is a pressing concern both globally and in Turkey, carrying direct implications for public health, environmental sustainability, and economic resilience. Elevated concentrations of pollutants such as particulate matter (PM<sub>10</sub>, PM<sub>2.5</sub>), nitrogen dioxide (NO<sub>2</sub>), ozone (O<sub>3</sub>), carbon monoxide (CO), and sulfur dioxide (SO<sub>2</sub>) are known to cause severe respiratory and cardiovascular diseases, posing serious risks to human life (Zencirci & Işıklı, 2017). Moreover, poor air quality can damage vegetation, pollute water resources, and disrupt ecosystems, ultimately reducing agricultural productivity and threatening biodiversity.

Ankara, the capital of Turkey, ranks among the most polluted cities in the country. The Ulus district, in particular, has emerged as a critical hotspot due to unchecked industrial emissions and a rapidly growing vehicle population. According to the World Air Quality Index, Ulus consistently records some of the highest pollution levels nationwide. A Greenpeace report

---

<sup>1</sup> MSc, Bandırma Onyedi Eylül University, Faculty of Engineering and Natural Sciences, Department of Computer Engineering, ORCID: 0009-0002-6562-2815.

<sup>2</sup> Asst. Prof., Bandırma Onyedi Eylül University, Faculty of Engineering and Natural Sciences, Department of Computer Engineering, ORCID: 0000-0001-6433-3355.

from 2023 states that residents of Ulus breathe polluted air for nearly one-third of the year (Greenpeace Turkey, 2024), while national media reports have compared its pollution levels unfavorably to those of heavily industrialized regions in India and China (Haber7, 2024).

In response to escalating environmental concerns, Turkey enacted the Regulation on the Protection of Air Quality in 1986 (Ministry of Justice of the Republic of Turkey, 2024), followed by a series of targeted policies including the Regulation on Control of Industrial Air Pollution, Gasoline and Diesel Quality, Heating-Related Air Pollution, and the Air Quality Assessment and Management By-law (Sümer, 2014). At the international level, Turkey is a signatory to several key agreements such as the Convention on Long-Range Transboundary Air Pollution, the Vienna Convention and Montreal Protocol, the Kyoto Protocol, and the Paris Agreement (Republic of Turkey Ministry of Environment, Urbanization and Climate Change, 2024a).

To support evidence-based environmental governance, the Ministry of Environment, Urbanization and Climate Change established the National Air Quality Monitoring Network in 2005. This network comprises continuous monitoring centers located in major cities and districts, including Ulus, and collects real-time data on pollutants such as NO<sub>2</sub>, O<sub>3</sub>, SO<sub>2</sub>, NH<sub>3</sub>, and PM<sub>10</sub>. The data are publicly accessible via the national monitoring portal and are categorized into six air quality levels: good, moderate, sensitive, unhealthy, hazardous, and poor (Republic of Turkey Ministry of Environment, Urbanization and Climate Change, 2024b).

Despite the growing body of research on air quality classification, most studies focus on national or metropolitan scales, leaving localized high-risk areas like Ulus underexplored. While the Air Quality Index (AQI) is typically calculated using

deterministic formulas defined by regulations, implementing Machine Learning (ML) models for this task offers distinct engineering advantages. First, ML models can act as "soft sensors," capable of classifying air quality even when data for specific pollutants is noisy or partially missing (imputation). Second, interpretable models like Decision Trees perform "rule extraction," visualizing the hierarchy of pollutants affecting a specific region. Therefore, this study does not aim to forecast future values but to establish a robust, data-driven classification framework that validates local pollution dynamics and supports rapid decision-making mechanisms for the Ulus district.

### **1.1. Soft Sensors and Explainable Machine Learning Frameworks**

While the traditional Air Quality Index (AQI) is calculated using deterministic formulas defined by regulatory thresholds, implementing Machine Learning (ML) models provides key engineering and operational advantages in smart city frameworks:

1. **Soft Sensing:** ML models function as "soft sensors," predicting or classifying air quality status even when specific hardware sensors experience temporary failure, calibration drift, or missing data streams.
2. **Rule Extraction & Explainability:** Interpretable models such as Decision Trees perform automated rule extraction, uncovering explicit hierarchical pollutant thresholds that drive local environmental warnings.

Therefore, this chapter establishes a robust, data-driven soft-sensor framework that validates local pollution dynamics and supports rapid decision-making mechanisms for high-risk urban hotspots.

The main contributions of this study to the literature and engineering practice are threefold:

1. **Localized Hotspot Analysis:** It provides a localized ML-based air quality classification specifically for the Ulus district of Ankara, a high-risk region requiring tailored decision support.
2. **Soft Sensor & Rule Extraction:** It utilizes decision tree structures to extract explicit, human-readable pollution rules ( $\text{\$NO}_2\text{\$}$  and  $\text{\$PM}_{10}\text{\$}$  thresholds), demonstrating an explainable soft-sensor approach for monitoring networks.
3. **Regulatory Compliance & Modeling:** Achieving 98.36% overall accuracy with Random Forest, the study demonstrates that ensemble learning models effectively mirror complex regulatory air quality categorization without reliance on static deterministic lookup tables.

The remainder of this chapter is structured as follows: Section 2 outlines the regulatory framework and health standards for air pollutants. Section 3 reviews related data-driven methodologies in the literature. Section 4 details the empirical workflow, soft-sensor architecture, and model parameters. Section 5 presents performance benchmarkings and rule extraction dynamics. Finally, Section 6 concludes with strategic recommendations for smart-city environmental monitoring.

## **2. AIR QUALITY CLASSES AND POLLUTANT CHARACTERISTICS**

Air quality assessment relies on evaluating physical concentrations of key atmospheric pollutants. Table 1 summarizes the primary pollutants, their anthropogenic/natural sources, and associated health impacts.

**Table 1. Sources of pollutants and their health effects (Zencirci & Işıklı, 2017)**

| <b>Pollutant</b>          | <b>Main Source</b>                                                                                                                                                   | <b>Potential Environmental and Health Impacts</b>                                                                                                                  |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Sulfur dioxide</b>     | <ul style="list-style-type: none"><li>• Coal-fired thermal power plant</li><li>• Oil refinery</li><li>• Solid, liquid fuels for heating purposes</li></ul>           | <ul style="list-style-type: none"><li>• Nasal irritation</li><li>• Respiratory tract spasms</li></ul>                                                              |
| <b>Nitrogen oxides</b>    | <ul style="list-style-type: none"><li>• Motor vehicles</li><li>• High temperature combustion processes</li></ul>                                                     | <ul style="list-style-type: none"><li>• Decreased oxygen level in the blood</li><li>• Respiratory tract irritation</li><li>• Susceptibility to infection</li></ul> |
| <b>Particulate Matter</b> | <ul style="list-style-type: none"><li>• Solid, liquid fuels</li><li>• Agriculture and secondary chemical reactions</li></ul>                                         | <ul style="list-style-type: none"><li>• Heart attack and sudden death</li><li>• Asthma attacks</li><li>• Respiratory tract irritation</li><li>• Cancer</li></ul>   |
| <b>Carbon Monoxide</b>    | <ul style="list-style-type: none"><li>• Vehicle emissions</li><li>• Metal processing industry</li><li>• Wood burning</li></ul>                                       | <ul style="list-style-type: none"><li>• Headache</li><li>• Drowning</li></ul>                                                                                      |
| <b>Ozone</b>              | <ul style="list-style-type: none"><li>• Photochemical processes from pollutants in anthropogenic sources such as power plants, refineries, chemical plants</li></ul> | <ul style="list-style-type: none"><li>• Impaired lung function</li><li>• Damage to lung structure in chronic exposure</li></ul>                                    |

National monitoring frameworks assign air quality status into six distinct categories. Table 2 outlines these health risk classifications as defined by official standards.

**Table 2. Air quality classes (T. R. Ministry of Environment, Urbanization and Climate Change, 2024)**

| <b>Quality Class</b> | <b>Description</b>                                                                                                              |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------|
| <b>Good</b>          | Air quality is healthy. Low or no risk of air pollution.                                                                        |
| <b>Moderate</b>      | Air quality is acceptable. However, some pollutants may pose a health risk to sensitive groups.                                 |
| <b>Sensitive</b>     | Health effects may occur for sensitive groups (e.g. children, the elderly and people with respiratory conditions).              |
| <b>Unhealthy</b>     | There may be adverse effects on the health of the general population. Serious health effects are possible for sensitive groups. |
| <b>Poor/Bad</b>      | For everyone, the health effects can be serious. Health warnings become common.                                                 |
| <b>Dangerous</b>     | Emergency health warnings. Health risks for the general population are very high.                                               |

### **3. RELATED WORK**

Several studies have applied decision-tree-based approaches to air quality classification. Kujaroentavon, Kiattisin & Leelasantitham (2014) used O<sub>3</sub>, NO<sub>2</sub>, CO, SO<sub>2</sub> and PM<sub>10</sub> as inputs for a decision tree model in Thailand, achieving 96.8% accuracy. Gore and Deshpande (2018) compared decision tree and Naive Bayes classifiers using NO<sub>2</sub>, SO<sub>2</sub>, CO and O<sub>3</sub>. They obtained about 92% accuracy in decision tree model and 86.7% accuracy in Naive Bayes model. Mahalingam et al. (2019) applied neural networks and support vector machines to PM<sub>10</sub>, PM<sub>2.5</sub>, NO<sub>2</sub>, O<sub>3</sub>, CO, SO<sub>2</sub>, NH<sub>3</sub> and Pb data for smart-city air quality classification. They obtained 97.3% accuracy in support vector machines and 91.62% accuracy in neural networks.

Several urban studies focused on Jakarta's air monitoring data. Evaluated. Putra & Sitanggang (2020) evaluated Decision Trees and Random Forests across multi-year pollutant datasets, finding Random Forest to yield 99.97% accuracy on 2017 records and Decision Trees achieving 98.28% on 2018 records. Hamami & Dahlan (2022) compared Logistic Regression, k-NN, Decision Trees, and Random Forests, noting that tuned Decision Trees reached 100% classification accuracy on selected environmental parameters. Sari and Purwaningsih (2022) applied Artificial Neural Networks to the same pollutant set, reporting 97.76% accuracy. Handayani, Soim, Agusdi & Husni (2021) used support vector machines on CO, CO<sub>2</sub>, HC and PM<sub>10</sub> pollutant values and temperature attributes, achieving 95.02% accuracy, while Sanmorino, Alie, Ariati & Wulanda (2022) applied k-nearest neighbor classification using CO, SO<sub>2</sub>, NO<sub>2</sub>, PM<sub>10</sub> and O<sub>3</sub> in Palembang, achieving 80% accuracy.

Recent studies have addressed data imbalance and missing value imputation. Jayadi, Lauro, Rusdi & Handhayani (2024) evaluated classification performance under class imbalance in

Jakarta, obtaining optimal results with Decision Trees (99% accuracy, 98% F1-score). Krisbiantoro, Waluyo, Hasanah, Pratama & Sarmini (2024) demonstrated that Random Forest and stacking ensembles provided high robustness when managing missing values and severe outliers. Hybrid deep learning approaches, such as ARIMA-ANN models proposed by Al-Jawahry, Ramyateja, Reddy, Ramadan & Maharajan (2023), achieved 93% accuracy in time-series index forecasting. Sasmita et al. (2024) compared Decision Trees against k-NN algorithms in Indonesia, demonstrating Decision Tree superiority (90.67% vs. 84.91%).

While existing literature largely focuses on country-wide or broad metropolitan scales, localized analysis of high-density pollution corridors remains essential. This study fills this gap by deploying an explainable soft-sensor framework specifically tailored to the historical air profile of the Ulus district in Ankara.

## **4. METHODOLOGY AND EXPERIMENTAL SETUP**

### **4.1. System Configuration and Workflow**

Experiments were performed on an Intel Core i7 (9th Gen) computational node equipped with 16 GB RAM and SSD storage under Windows 10. Data processing, imputation, imbalance handling, and model evaluations were executed in Python using pandas, numpy, scikit-learn, imbalanced-learn, matplotlib, and seaborn libraries.

Figure 1 outlines the methodological execution flow:

1. Continuous data ingestion from the national monitoring network.
2. Quality inspection and missing value handling via k-Nearest Neighbors Imputation (k-NN Imputer, k=5).

3. Temporal split into independent training (2023–2024) and evaluation (2024–2025) sets.
4. Synthetic Minority Over-sampling Technique (SMOTE) applied strictly to the training set to prevent data leakage.
5. Classifier training, hyperparameter optimization, and evaluation.

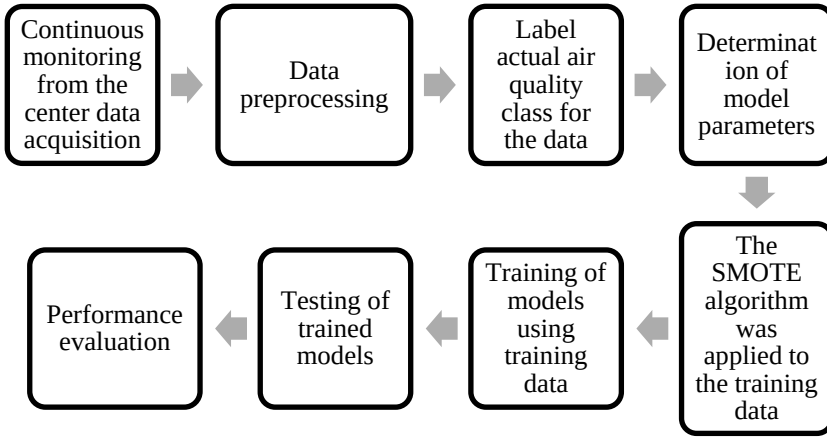


**Figure 1. Proposed Methodology and System Execution Flow**

#### **4.2. Dataset Description and Preprocessing**

The dataset comprises continuous station records from the Ulus monitoring station between July 1, 2023, and July 1, 2025 (Republic of Turkey Ministry of Environment, Urbanization and Climate Change, 2024c). To ensure temporal evaluation independence, records from 2023–2024 were assigned to training, whereas records from 2024–2025 were reserved exclusively for testing.

Table 3 summarizes sample distributions across target air quality classes. Classes such as *Unhealthy*, *Poor*, and *Dangerous* contained zero records during the observed timeframe due to station data availability limits and regional air dynamics; hence,

model evaluation focuses on observed classes (*Good, Moderate, Sensitive*).

**Table 3. Class Sample Distribution Across Temporal Splitting Sets**

| Air quality class | Train number of instances | Test number of instances |
|-------------------|---------------------------|--------------------------|
| <b>Good.</b>      | 211                       | 185                      |
| <b>Middle</b>     | 134                       | 152                      |
| <b>Sensitive</b>  | 22                        | 29                       |
| <b>Total</b>      | 367                       | 366                      |

Table 4 specifies input features and operational measurement ranges utilized by the classifiers.

**Table 4. Feature Attributes and Range Characteristics**

| # | Attribute | Description        | Value Range | Explanation                                |
|---|-----------|--------------------|-------------|--------------------------------------------|
| 1 | PM10      | Particulate Matter | 0-520       | Values >520 trigger Dangerous thresholds   |
| 2 | SO2       | Sulfur dioxide     | 0 -1100     | Values >1100 trigger Dangerous thresholds  |
| 3 | NO2       | Nitrogen dioxide   | 0-2000      | Values >2000 trigger Dangerous thresholds  |
| 4 | O3        | Ozone              | 0-700       | Values >700 trigger Dangerous thresholds   |
| 5 | CO        | Carbon Monoxide    | 0-32000     | Values >32000 trigger Dangerous thresholds |

### **4.3. Machine Learning Classifiers & Tuning Parameters**

Four supervised algorithms were selected and optimized:

- **k-Nearest Neighbors (k-NN):** Non-parametric lazy learning model. Distance metric evaluations across  $k \in [1, 30]$  revealed optimal performance at  $k = 1$  using Euclidean distance.
- **Support Vector Machines (SVM):** Maximum-margin classifier. Parameter search using logarithmic grids ( $C \in$

$[10^{-2}, 10^{10}]$ ,  $\gamma \in [10^{-9}, 10^3]$ ) identified optimal hyperparameters at  $C = 10.0$  and  $\gamma = 0.0001$  with RBF kernel.

- **Decision Tree (DT):** Explainable glass-box model. Tree depth limits (None, 10, 20, 30, 40, 50) and leaf split thresholds (2, 10, 20) yielded best performance at unconstrained depth (None) and  $min\_samples\_split=2$ .
- **Random Forest (RF):** Ensemble bootstrap aggregation model. Parameter sweeps on estimator count ( $n\_estimators=200$ ) with automatic feature sub-sampling produced maximum ensemble stability.

#### 4.4. Evaluation Metrics

Performance was evaluated using standard classification metrics derived from the Confusion Matrix (True Positives TP, True Negatives TN, False Positives FP, False Negatives FN):

**Table 5. TP, TN, FP and FN descriptions**

|           |                                                                         |
|-----------|-------------------------------------------------------------------------|
| <b>TP</b> | Examples where the model correctly predicts a positive                  |
| <b>TN</b> | Examples where the model correctly predicts a negative                  |
| <b>FP</b> | Instances that the model predicts as positive but are actually negative |
| <b>FN</b> | Instances where the model predicts negative but in reality are positive |

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

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (2)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (3)$$

$$\text{F1 Score} = 2 * \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

## 5. RESULTS AND DISCUSSION

### 5.1. Classifier Performance Comparison

Table 5 outlines class-wise performance and overall classification metrics across models on the independent test dataset.

**Table 6. Performance Comparison of Machine Learning Models  
(Class-wise Accuracy and Overall Metrics)**

| Metric / Class                | k-NN   | SVM    | Decision Tree | Random Forest |
|-------------------------------|--------|--------|---------------|---------------|
| <b>Good (Class Acc.)</b>      | 0.9027 | 0.8054 | 0.9675        | <b>0.9945</b> |
| <b>Moderate (Class Acc.)</b>  | 0.7631 | 0.8815 | 0.9736        | <b>0.9802</b> |
| <b>Sensitive (Class Acc.)</b> | 0.5172 | 0.4137 | 0.4482        | <b>0.9310</b> |
| <b>Overall Accuracy</b>       | 0.8142 | 0.8060 | 0.9290        | <b>0.9836</b> |
| <b>Overall F1-Score</b>       | 0.8102 | 0.8029 | 0.9233        | <b>0.9835</b> |

\*Note: Classes with zero instances (Unhealthy, Bad, and Hazardous) are excluded. F1-Score is the weighted average.

Tree-based ensemble learning achieved superior performance. Random Forest obtained an overall classification accuracy of 98.36% and a weighted F1-Score of 0.9835. Crucially, while distance-based models (k-NN and SVM) struggled with minority class recognition (*Sensitive* class accuracy of 51.72% and 41.37%, respectively), Random Forest correctly classified 93.10% of sensitive pollution events. This performance highlights the capability of ensemble tree decision boundaries to resolve non-linear, overlapping atmospheric feature spaces.

## 5.2. Soft-Sensor Rule Extraction

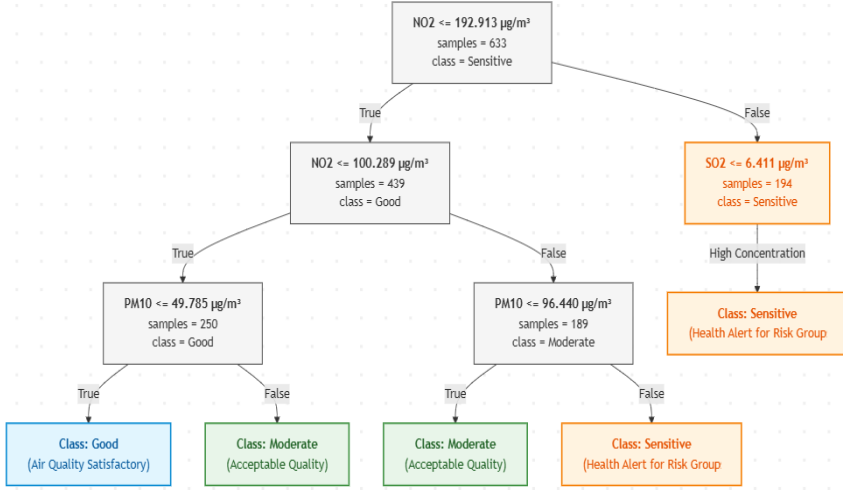


**Figure 2. Decision tree visualization used in air quality classification**

The automated extraction yields clear, human-understandable environmental rules without manual AQI coding:

• **Rule 1 (Good Air Quality):** If  $NO_2 \leq 100.289$  and  $PM_{10} \leq 49.785$ , the air quality  $\rightarrow$  "Good".

• **Rule 2 (Moderate Air Quality):** If  $NO_2 \leq 100.289$  and  $PM_{10}$  is in the range  $(49.785, 113.24]$ , the air quality  $\rightarrow$  "Moderate".

• **Rule 3 (Sensitive/Unhealthy):** Higher concentrations, specifically where  $NO_2 > 192.913$ , the air quality  $\rightarrow$  "Sensitive".

These extracted split boundaries match regulatory breakpoints established in national air management guidelines. This empirical alignment proves that soft-sensor machine learning models can continuously monitor, audit, and validate urban air status even under physical sensor telemetry loss.

## 6. CONCLUSION

This chapter presented a data-driven soft-sensor framework for air quality classification in high-risk urban hotspots, using the Ulus district of Ankara as a representative case study. By leveraging supervised machine learning models on continuous monitoring data from 2023 to 2025, the study verified that ensemble decision models accurately categorize air health risk without reliance on static formula calculations.

The Random Forest model achieved high effectiveness (98.36% accuracy, 0.9835 F1-Score), outperforming distance-based classifiers in identifying minority sensitive pollution events. Furthermore, decision tree rule extraction verified  $NO_2$  and  $PM_{10}$  as the dominant regional drivers. Historical observations over the two-year period indicated that air quality predominantly fluctuated between *Good* and *Moderate* levels, interspersed with manageable Sensitive spikes.

Future work will expand this framework by incorporating meteorological variables (wind direction, dispersion velocity, ambient temperature, relative humidity) and integrating the models into IoT-driven smart city dashboards for real-time public health advisories.

## REFERENCES

- Al-Jawahry, H. M., Ramyateja, O., Reddy, R. A., Ramadan, G. M., & Maharajan, K. (2023, November). Classification of Air Quality Index Using Autoregressive Integrated Moving Average (ARIMA) and Artificial Neural Networks Models. In 2023 International Conference on Ambient Intelligence, Knowledge Informatics and Industrial Electronics (AIKIIE) (pp. 1-6). IEEE.
- Gore, R. W., & Deshpande, D. S. (2017, October). An approach for classification of health risks based on air quality levels. In 2017 1st International Conference on Intelligent Systems and Information Management (ICISIM) (pp. 58-61). IEEE.
- Greenpeace Turkey. (2024). 20 regions with the most polluted air in 2023 [Press release]. Retrieved from <https://www.greenpeace.org/turkey/basin-bultenleri/2023-yilinin-havasi-en-kirli-20-bolgesi-belli-oldu>
- Haber7. (2024). Turkey detail in air pollution report: Ankara's that neighborhood is the most risky in the world. Retrieved from <https://www.haber7.com/guncel/haber/3274998>
- Hamami, F., & Dahlan, I. A. (2022, February). Air quality classification in urban environment using machine learning approach. In IOP Conference Series: Earth and Environmental Science (Vol. 986, No. 1, p. 012004). IOP Publishing.
- Handayani, A. S., Soim, S., Agusdi, T. E., & Husni, N. L. (2021). Air quality classification using support Vector machine. Computer Engineering and Applications, 10(1).

- Jayadi, B. V., Lauro, M. D., Rusdi, Z., & Handhayani, T. (2024). Klasifikasi indeks standar pencemaran udara pada data tidak seimbang. *Sistemasi: Jurnal Sistem Informasi*, 13(3), 951–958.
- Krisbiantoro, D., Waluyo, R., Hasanah, U., Pratama, I., & Sarmini, S. (2024). Classification of air pollutant index on data with outliers and imbalance class problem. *JOIV: International Journal on Informatics Visualization*, 8(3), 1049-1054.
- Kujaroentavon, K., Kiattisin, S., Leelasantham, A., & Thammaboosadee, S. (2014, November). Air quality classification in Thailand based on decision tree. In *The 7th 2014 biomedical engineering international conference* (pp. 1-5). IEEE.
- Mahalingam, U., Elangovan, K., Dobhal, H., Valliappa, C., Shrestha, S., & Kedam, G. (2019, March). A machine learning model for air quality prediction for smart cities. In *2019 International conference on wireless communications signal processing and networking (WiSPNET)* (pp. 452-457). IEEE.
- Ministry of Justice of the Republic of Turkey. (2024). Turkish code of obligations. Legislation Information System. Retrieved from <https://www.mevzuat.gov.tr/mevzuat?MevzuatNo=12188>
- Putra, F. M., & Sitanggang, I. S. (2020, July). Classification model of air quality in Jakarta using decision tree algorithm based on air pollutant standard index. In *IOP Conference Series: Earth and Environmental Science* (Vol. 528, No. 1, p. 012053). IOP Publishing.

- Republic of Turkey Ministry of Environment, Urbanization and Climate Change. (2024a). Conventions. Directorate General for European Union and Foreign Relations. Retrieved from <https://ab.csb.gov.tr/sozlesmeler-i-98916>
- Republic of Turkey Ministry of Environment, Urbanization and Climate Change. (2024b). Air quality index (AQI). SIM. Retrieved from <https://sim.csb.gov.tr/Home/HKI>
- Republic of Turkey Ministry of Environment, Urbanization and Climate Change. (2024c). Station data download. Retrieved from [https://sim.csb.gov.tr/STN/STN\\_Report/StationDataDownloadNew](https://sim.csb.gov.tr/STN/STN_Report/StationDataDownloadNew)
- Sanmorino, A., Alie, J., Ariati, N., & Wulanda, S. V. (2022). K-NN Based Air Classification as Indicator of the Index of Air Quality in Palembang. *Sinkron: jurnal dan penelitian teknik informatika*, 6(3), 853-859.
- Sari, E. N., & Purwaningsih, E. (2022). Air Quality Index Classification Using Neural Network Algorithms: Klasifikasi Index Kualitas Udara Menggunakan Algoritma Neural Network. *Systematics*, 4(3), 473-781.
- Sarkhosh, M., Najafpoor, A. A., Alidadi, H., Shamsara, J., Amiri, H., Andrea, T., & Kariminejad, F. (2021). Indoor Air Quality associations with sick building syndrome: An application of decision tree technology. *Building and Environment*, 188, 107446.
- Sasmita, N. R., Ramadeska, S., Kesuma, Z. M., Noviandy, T. R., Maulana, A9., Khairul, M., & Suhendra, R. (2024). Decision tree versus k-nn: A performance comparison for air quality classification in indonesia. *Infolitika Journal of Data Science*, 2(1), 9-16.

- Sümer, G. Ç. (2014). Air pollution control: An examination of legal regulations and organizations to prevent air pollution in Turkey. *International Journal of Economic and Administrative Review*, 13, 37–56.
- T. R. Ministry of Environment, Urbanization and Climate Change. (2024). Air quality index. Aegean Clean Air Center. Retrieved from <https://egethm.csb.gov.tr/hava-kalitesi-indeksi-i-87403>
- Zencirci, S. A., & Işıklı, B. (2017). Hava kirliliği. *ESTÜDAM Halk Sağlığı Dergisi*, 2(2), 24–36.

# **ARTIFICIAL INTELLIGENCE APPLICATIONS IN MULTIPARAMETRIC MRI FOR PROSTATE CANCER DETECTION AND GRADING**

**Mecit YÜZKAT<sup>1</sup>**

## **1. INTRODUCTION**

In recent years, cancer has become an important public health problem worldwide. According to GLOBOCAN data, prostate cancer (PCa) was among the most commonly diagnosed cancers in men in 2022 (International Agency for Research on Cancer, 2024a). In Türkiye, more than 17,000 men were diagnosed with prostate cancer in 2022, making it the second most common cancer in men after lung cancer (International Agency for Research on Cancer, 2024b). Therefore, early and accurate diagnosis is important for effective treatment. Current diagnostic methods include Prostate-Specific Antigen (PSA) testing, Digital Rectal Examination (DRE), Magnetic Resonance Imaging (MRI), Transrectal Ultrasonography (TRUS), and histopathological examination of biopsy samples. However, biopsy evaluation may vary between pathologists due to differences in experience and interpretation. Recent developments in Artificial Intelligence (AI) offer new approaches to these challenges. In particular, AI-based methods have been increasingly used for prostate cancer detection from multiparametric MRI (mpMRI) and for Prostate Imaging Reporting and Data System (PI-RADS) assessment. This section

---

<sup>1</sup> Dr, Muş Alparslan University, Faculty of Engineering and Architecture, Department of Software Engineering & Health Institutes of Türkiye, ORCID: 0000-0003-4808-5181.

focuses on these applications and the current challenges in this field.

## **2. ARTIFICIAL INTELLIGENCE AND HEALTH**

AI refers to systems that can learn and perform tasks by processing information in ways similar to human intelligence (Transformation of Technology in Healthcare, 2024). Healthcare systems face increasing challenges due to rising costs, aging populations, disability, and disease-related mortality (Al Kuwaiti et al., 2023). In this context, AI is increasingly being used in healthcare applications. The digitalization of healthcare data, advances in machine learning, and improvements in computing infrastructure have expanded the use of AI in medical applications. The increasing volume and complexity of healthcare data have expanded the use of AI in areas such as medical imaging, diagnosis, patient care, and biomedical research. AI-based technologies may also help reduce inequalities in access to diagnosis and treatment (Abràmoff et al., 2023). AI has become particularly useful in medical image analysis and radiology. In prostate cancer, AI-based methods are being investigated for the analysis of mpMRI, including prostate cancer detection and PI-RADS assessment. These methods may support radiologists in diagnosis and help improve the efficiency of the diagnostic process.

### **2.1. Artificial Intelligence-Assisted Diagnostic And Treatment Systems**

AI-based tools are being explored in several stages of healthcare, including diagnosis, treatment planning, and patient management. These models can process large clinical datasets and provide information that may support clinical decision-making. Their use is particularly relevant in fields such as

radiology, pathology, and genetics, where large and complex datasets are routinely generated.

In medical imaging, AI models can analyze MRI and Computed Tomography (CT) data to support the detection and characterization of diseases such as cancer. AI can also contribute to personalized medicine by combining clinical, genetic, and other patient-specific data to support treatment decisions. Beyond diagnosis and treatment, AI is used in areas such as robotic surgery, virtual reality, virtual health assistants, and medical record management, as shown in Figure 1.

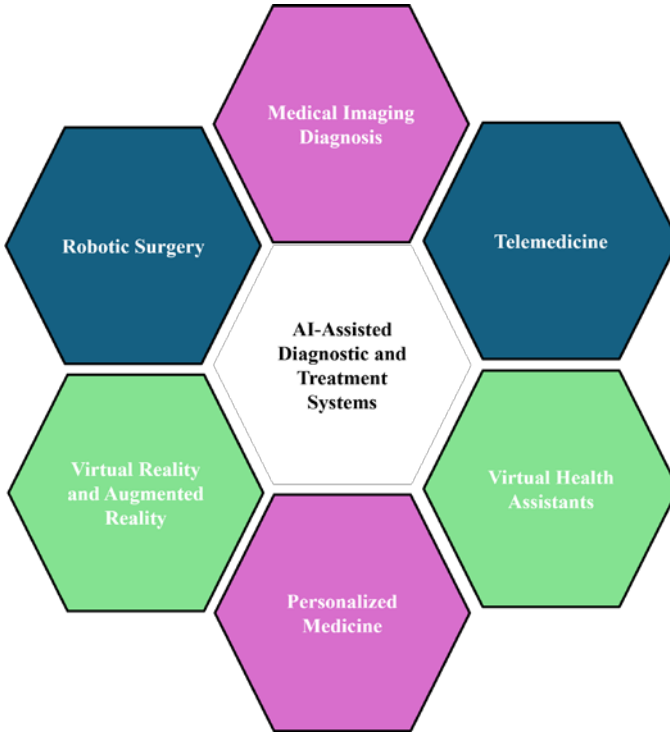


**Figure 1. AI-Assisted Diagnosis and Treatment Systems**

## **2.2. The Role of Artificial Intelligence In Medical Imaging**

AI is increasingly used for medical image analysis and processing. Research has focused particularly on clinical fields such as radiology and pathology (Wang et al., 2020; Wang et al., 2021; Yüzkat et al., 2021). AI is now widely used for common medical imaging tasks, including classification (Cai et al., 2020; Kim et al., 2022), segmentation (Ramesh et al., 2021; Siddique et al., 2021; Wang et al., 2022), and detection (Gaur et al., 2023; Kaur & Garg, 2023; Zhou et al., 2023). The safe and effective use of these systems in clinical practice also requires appropriate clinical expertise. The main medical imaging modalities where AI is applied include:

### **Magnetic Resonance Imaging**

MRI uses a magnetic field and radiofrequency signals to generate detailed images of internal tissues. AI can support image interpretation, including lesion detection and characterization. In prostate cancer, these methods are increasingly studied in combination with mpMRI.

### **Computed Tomography**

CT uses X-rays to generate cross-sectional images. AI-based analysis can assist in identifying abnormalities in organs and tissues and can support faster image interpretation.

### **Ultrasonography**

Ultrasonography produces images using high-frequency sound waves. AI can assist in identifying abnormalities and has been investigated in several organ systems, including the liver, breast, and thyroid.

## **Radiography**

Radiography uses X-rays to produce two-dimensional images and is widely used because of its speed and accessibility. AI-based analysis can support the detection of abnormalities in the lungs, bones, and heart.

## **Digital Pathology Imaging**

Digital pathology involves the analysis of digitized microscopic images of tissue samples. AI can assist pathologists in detecting cancer cells, classifying tumors, and assessing disease characteristics.

### **3. PROSTATE CANCER EPIDEMIOLOGY AND DIAGNOSTIC METHODS**

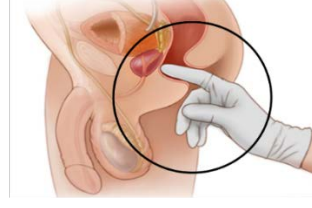
Prostate cancer incidence is influenced by age, genetic background, ethnicity, and geographic location (Rawla, 2019). BPH becomes increasingly common with age and may cause urinary symptoms (Langan, 2019). Incidence is higher among African-American men and lower in many Asian populations. Family history is also an established risk factor. Other factors, including obesity, diet, hormonal changes, and smoking, may contribute to disease risk. Higher reported incidence in developed countries is partly related to the wider use of PSA testing and the resulting detection of otherwise undiagnosed disease (Mohler et al., 2016).

Prostate cancer may not cause clear symptoms in its early stages. Therefore, PSA testing, DRE, MRI, and TRUS are used in the diagnostic process, as shown in Figure 2.

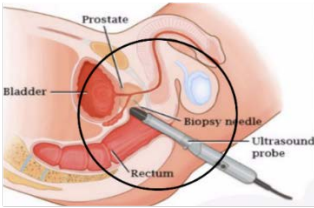
## Prostate Cancer Diagnosis Methods



Prostate-Specific Antigen (PSA) Test



Digital Rectal Exam (DRE)



Trans Rectal Ultrasound (TRUS)



Magnetic Resonance Imaging (MRI)

### Figure 2 Prostate Cancer Diagnostic Methods

#### Prostate-Specific Antigen Test

PSA is a protein produced by prostate cells and can be measured through a blood test. It has been used in prostate cancer screening since the 1980s and became widely adopted during the 1990s (Catalona et al., 1993). An elevated PSA level may indicate prostate cancer, but it can also result from benign enlargement or inflammation. Conversely, prostate cancer may occur in men with relatively low PSA levels (Mohler, 2010; Balk et al., 2003). For this reason, PSA results should be interpreted together with other clinical and imaging findings rather than used as a stand-alone diagnostic measure.

#### Digital Rectal Examination

DRE involves physical examination of the prostate through the rectum to assess its size, shape, and consistency. It may help identify abnormalities and determine whether further

evaluation or biopsy is required. However, its ability to detect prostate cancer is limited, and it is generally used together with other diagnostic methods (Smith & Catalona, 1995).

### **Transrectal Ultrasound**

TRUS uses sound waves to evaluate the size and structure of the prostate and is commonly used to guide prostate biopsy. Although it provides useful information about prostate volume and biopsy localization, its sensitivity and specificity for prostate cancer detection and staging are limited (Harvey et al., 2012).

### **Magnetic Resonance Imaging**

MRI provides detailed images of prostate tissue and is an important imaging method in the diagnosis and assessment of prostate cancer. MRI systems commonly operate at 1.5 or 3.0 Tesla, and the signal generated by hydrogen protons in the body is used to produce images (Berger, 2002).

Two commonly used MRI approaches for prostate imaging are bpMRI and mpMRI (Kuhl et al., 2017; Ghai & Haider, 2015; Kumar et al., 2018; Turkbey et al., 2009). bpMRI generally includes T2-weighted and diffusion-weighted imaging (DWI), while mpMRI additionally includes dynamic contrast-enhanced imaging (DCE). mpMRI provides information from multiple imaging sequences and can improve the detection and characterization of suspicious prostate lesions. It also plays an important role in risk assessment using the PI-RADS system.

### **Biopsy**

Biopsy is performed when clinical, PSA, or imaging findings indicate a suspicion of prostate cancer. Tissue samples are collected from the prostate and examined histopathologically. TRUS has traditionally been used to guide biopsy, while MRI has increasingly been used to target suspicious lesions (Nguyen-Nielsen & Borre, 2016; Mottet et al., 2021).

MRI-targeted biopsy allows suspicious areas identified on MRI to be sampled more precisely. In MR-fusion biopsy, MRI findings are combined with real-time ultrasound images to guide the biopsy needle toward suspicious lesions (Hadaschik et al., 2011; Pinto et al., 2011). This approach can improve the targeting of lesions that may be missed by systematic biopsy.

#### **4. MULTIPARAMETRIC MRI IN PROSTATE CANCER**

The use of MRI for prostate cancer diagnosis began with T1- and T2-weighted sequences in the 1980s (Giganti et al., 2019). T1-weighted imaging has limited value for detecting prostate cancer because cancerous tissue usually does not produce significant changes on T1-weighted images (Stabile et al., 2020). In contrast, mpMRI combines different imaging sequences to provide structural and functional information about prostate tissue. The main sequences used in mpMRI are T2-weighted imaging, DWI, and DCE. These sequences provide complementary information for prostate cancer detection and assessment (Johnson et al., 2014).

Magnetic resonance spectroscopy imaging (MRSI) can provide additional metabolic information, but its routine clinical use remains limited because of technical requirements, cost, and limited availability. Therefore, current prostate mpMRI protocols mainly rely on T2-weighted imaging, DWI, and DCE. Figure 3 shows an example of a prostate lesion detected using mpMRI.

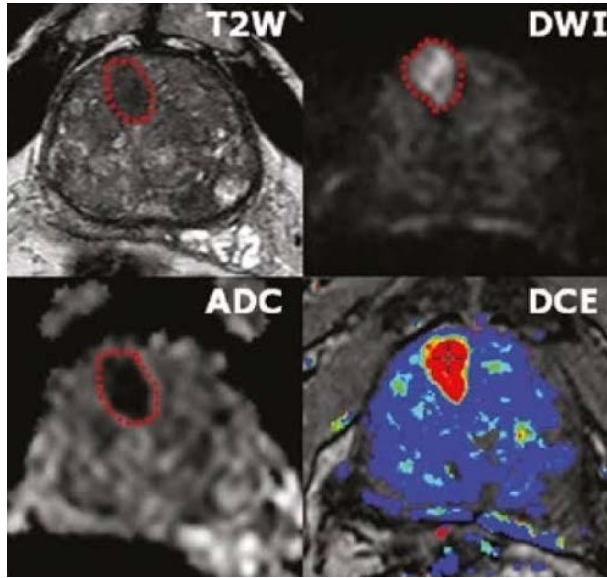


**Figure 3 Example of Lesion with mp-MRI (Bjurlin et al., 2016)**

#### **4.1. T2 Weighted Imaging**

T2-weighted imaging is mainly used to visualize the anatomical zones of the prostate and is an essential part of prostate MRI (Hricak et al., 1987). In a normal prostate, the peripheral zone usually has high signal intensity because of its relatively high water content. Prostate cancer often appears as a region of lower signal intensity. However, this finding is not specific to cancer, as inflammation, fibrosis, and post-biopsy hemorrhage may produce similar appearances. Lesion assessment is also more difficult in the transition zone because of its heterogeneous appearance and the frequent presence of benign prostatic hyperplasia.

#### **4.2. Diffusion Weighted Imaging And Apparent Diffusion Coefficient Maps**

DWI provides information about water diffusion within prostate tissue and is useful for distinguishing malignant lesions from normal tissue (Somford et al., 2008). Because of their high

cellular density, prostate tumors usually restrict water movement and therefore show higher signal intensity on high b-value DWI. DWI is particularly useful for identifying and localizing clinically significant prostate cancer and may also provide information about tumor aggressiveness and local extent (Scheenen et al., 2015; Schieda et al., 2021).

The ADC value is derived from DWI and reflects the degree of water diffusion within tissue. Prostate cancer generally shows lower ADC values than normal prostate tissue. ADC maps are calculated from images acquired with different b values. The combination of DWI and T2-weighted imaging can improve lesion assessment, particularly in the transition zone where benign and malignant findings may have similar appearances (Wu et al., 2012; Jung et al., 2013).

#### **4.3. Dynamic Contrast Imaging**

DCE evaluates the uptake and washout of intravenously administered contrast material in prostate tissue. It provides information about tissue vascularity and can help identify areas with abnormal enhancement associated with tumor angiogenesis (Hara et al., 2005).

DCE can provide additional information when findings from T2-weighted imaging and DWI are unclear or affected by artifacts. However, its routine use has been debated because it requires gadolinium-based contrast material and can increase examination time. Studies have also shown that bpMRI based on T2-weighted imaging and DWI can provide useful diagnostic information without DCE (Verma et al., 2012; Boesen et al., 2018). Therefore, DCE is mainly used as a complementary sequence in current prostate mpMRI protocols.

## **5. PROSTATE CANCER RISK GRADING**

Histopathological examination is used to determine the grade and aggressiveness of prostate cancer. The Gleason grading system evaluates the architectural patterns of prostate cancer cells on a scale from 1 to 5. The Gleason score is calculated by combining the two most common patterns, with higher scores generally indicating more aggressive disease (Berney et al., 2016). Following revisions by the International Society of Urological Pathology (ISUP), Grade Groups 1–5 are used together with the Gleason score to provide a more clinically meaningful classification (Epstein et al., 2016). For example, Gleason score 3+3 corresponds to ISUP Grade Group 1, while 3+4 and 4+3 correspond to Grade Groups 2 and 3, respectively. Higher-grade patterns are associated with more aggressive disease.

MRI is now an important part of the diagnostic pathway for prostate cancer. In biopsy-naïve men, its use can help avoid unnecessary biopsies while improving the detection of clinically significant disease (Ahmed et al., 2017). MRI-targeted biopsy can further improve the identification of clinically significant tumors compared with systematic biopsy (Kasivisvanathan et al., 2018). PI-RADS was introduced to provide a common framework for the interpretation and reporting of prostate MRI. The first guidelines were published by the European Society of Urogenital Radiology (ESUR) in 2012 and were later updated as PI-RADS v2.1 in 2019 (Barentsz et al., 2012; Turkbey et al., 2019).

The PI-RADS system uses a five-point scale to describe the likelihood of clinically significant prostate cancer. The final score is mainly determined from T2-weighted and diffusion-weighted images, while DCE may provide additional information when required. Scores of 1 and 5 represent the lowest and highest likelihoods, respectively.

## **6. SEGMENTATION AND GRADING OF PROSTATE CANCER WITH ARTIFICIAL INTELLIGENCE**

AI methods have been increasingly used for prostate cancer detection and classification from mpMRI data (Suarez-Ibarrola et al., 2022). Machine learning (ML) and deep learning (DL) are the main approaches used to develop computer-aided systems for medical image analysis (Zhou, 2021). ML methods generally rely on predefined features or statistical representations, whereas DL models learn relevant features directly from the input data. In prostate cancer, these methods can support lesion detection, segmentation, and classification, including the assessment of clinically significant disease.

### **6.1. Models Used For Segmentation Of Prostate, Prostate Zones, And Lesions**

DL, particularly convolutional neural networks, has become widely used for medical image segmentation. Early approaches used patch-based networks to classify individual image regions (Ciresan et al., 2012). Later, U-Net introduced an encoder-decoder architecture specifically designed for image segmentation (Ronneberger et al., 2015). U-Net and its variants have since become widely used in medical image segmentation, including prostate MRI.

For prostate MRI, segmentation is generally performed at three levels: the prostate gland, anatomical zones, and suspicious lesions.

#### **Prostate Segmentation**

Prostate segmentation defines the prostate region for subsequent image analysis. U-Net and its variants have been applied to both 2D and 3D prostate MRI segmentation (Vu et al., 2018; Li et al., 2023).

## **Prostate Region Segmentation**

Segmentation of the prostate zones can provide additional anatomical information for lesion analysis. CNN-based methods have been used to automatically segment regions such as the central zone and peripheral zone (Jensen et al., 2019; Rundo et al., 2020).

## **Prostate Lesion Segmentation**

Lesion segmentation is more challenging because lesions can vary in size, shape, location, and appearance. Methods based on architectures such as U-Net, DeepLabv3+, and other specialized networks have been investigated for prostate lesion segmentation (Duran et al., 2022; Hambarde et al., 2020). Lesion segmentation also provides information about lesion location and extent for subsequent classification.

## **6.2. Models Used For Lesion Classification**

AI-based approaches for prostate cancer classification can generally be divided into two groups. The first focuses directly on estimating the Gleason score or grade of prostate cancer lesions using models such as CNNs (Abraham & Nair, 2018, 2019). These models use MRI features to predict lesion grade.

The second approach combines segmentation and classification. In these systems, the prostate and lesions are first identified, followed by classification or Gleason grade group prediction using an integrated deep learning model (Cao et al., 2019; De Vente et al., 2020). This approach combines lesion localization and grading within the same framework.

### **6.3. Evaluation Metrics**

AI models are evaluated using different metrics depending on the task.

#### **Segmentation Performance Metrics**

The Dice coefficient is widely used to assess the quality of medical image segmentation. It measures the degree of overlap between the predicted region and the reference annotation, with values between 0 and 1. A score of 1 represents complete agreement between the two regions, whereas a score close to 0 indicates little or no overlap (Milletari et al., 2016).

#### **Classification Performance Metrics**

Classification models can be assessed using several measures, including accuracy, precision, recall, F1-score, and AUC. Among these, AUC is commonly used to summarize model performance over different decision thresholds (Hanley & McNeil, 1982).

#### **Lesion Detection Performance Metrics**

Lesion detection is commonly evaluated using true positives (TP), false positives (FP), and intersection over union (IoU). A predicted lesion can be considered a true positive when it overlaps sufficiently with the ground-truth lesion. In some prostate cancer detection studies, a three-dimensional IoU threshold of 10% has been used for this purpose (Baumgartner et al., 2021; Bosma et al., 2023; Saha et al., 2021). Predictions that do not meet the required overlap threshold are considered false positives.

## **7. CONCLUSIONS AND FUTURE PERSPECTIVES**

AI-based methods have been increasingly investigated for prostate cancer detection, segmentation, and grading using mpMRI. These approaches can provide additional information during MRI interpretation and assist in the assessment of suspicious lesions.

Before routine clinical use, AI-based prostate cancer systems require further validation. Multicenter and prospective studies are needed to assess their performance across different patient groups and clinical settings. Long-term studies should also examine whether these systems provide measurable benefits in patient management.

The use of AI in healthcare also raises issues related to data privacy, model transparency, and diagnostic errors. These systems should therefore be evaluated within appropriate ethical and regulatory frameworks. Training healthcare professionals will also be necessary for effective clinical use.

Federated learning may provide another approach for developing AI models using data from multiple institutions while keeping patient data within the respective institutions. Using data from multiple centers may improve model generalizability compared with models trained on single-center datasets. Future studies combining mpMRI, clinical data, and AI-based methods across multiple institutions may contribute to the development of more robust prostate cancer diagnostic systems.

## REFERENCES

- Abraham, B., & Nair, M. S. (2018). Computer-aided classification of prostate cancer grade groups from MRI images using texture features and stacked sparse autoencoder. *Computerized Medical Imaging and Graphics*, 69, 60-68.
- Abraham, B., & Nair, M. S. (2019). Automated grading of prostate cancer using convolutional neural network and ordinal class classifier. *Informatics in Medicine Unlocked*, 17, 100256.
- Abràmoff, M. D., Tarver, M. E., Loyo-Berrios, N., Trujillo, S., Char, D., Obermeyer, Z., ... & Maisel, W. H. (2023). Considerations for addressing bias in artificial intelligence for health equity. *NPJ Digital Medicine*, 6(1), 170.
- Ahmed, H. U., Bosaily, A. E. S., Brown, L. C., Gabe, R., Kaplan, R., Parmar, M. K., ... & Emberton, M. (2017). Diagnostic accuracy of multi-parametric MRI and TRUS biopsy in prostate cancer (PROMIS): a paired validating confirmatory study. *The Lancet*, 389(10071), 815-822.
- Al Kuwaiti, A., Nazer, K., Al-Reedy, A., Al-Shehri, S., Al-Muhanna, A., Subbarayalu, A. V., ... & Al-Muhanna, F. A. (2023). A review of the role of artificial intelligence in healthcare. *Journal of Personalized Medicine*, 13(6), 951.
- Balk, S. P., Ko, Y. J., & Bubley, G. J. (2003). Biology of prostate-specific antigen. *Journal of Clinical Oncology*, 21(2), 383-391.
- Barentsz, J. O., Richenberg, J., Clements, R., Choyke, P., Verma, S., Villeirs, G., ... & Fütterer, J. J. (2012). ESUR prostate MR guidelines 2012. *European Radiology*, 22, 746-757.
- Baumgartner, M., Jäger, P. F., Isensee, F., & Maier-Hein, K. H. (2021). nnDetection: a self-configuring method for

- medical object detection. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2021* (pp. 530-539). Springer International Publishing.
- Berger, A. (2002). How does it work?: Magnetic resonance imaging. *BMJ: British Medical Journal*, 324(7328), 35.
- Berney, D. M., Beltran, L., Fisher, G., North, B. V., Greenberg, D., Møller, H., ... & Cuzick, J. (2016). Validation of a contemporary prostate cancer grading system using prostate cancer death as outcome. *British Journal of Cancer*, 114(10), 1078-1083.
- Bjurlin, M. A., Mendhiratta, N., Wysock, J. S., & Taneja, S. S. (2016). Multiparametric MRI and targeted prostate biopsy: Improvements in cancer detection, localization, and risk assessment. *Central European Journal of Urology*, 69(1), 9–18.
- Boesen, L., Nørgaard, N., Løgager, V., Balslev, I., Bisbjerg, R., Thestrup, K. C., ... & Thomsen, H. S. (2018). Assessment of the diagnostic accuracy of biparametric magnetic resonance imaging for prostate cancer in biopsy-naïve men: the biparametric MRI for detection of prostate cancer (BIDOC) study. *JAMA Network Open*, 1(2), e180219-e180219.
- Bosma, J. S., Saha, A., Hosseinzadeh, M., Slootweg, I., de Rooij, M., & Huisman, H. (2023). Semisupervised learning with report-guided pseudo labels for deep learning-based prostate cancer detection using biparametric MRI. *Radiology: Artificial Intelligence*, 5(5), e230031.
- Cai, L., Gao, J., & Zhao, D. (2020). A review of the application of deep learning in medical image classification and segmentation. *Annals of Translational Medicine*, 8(11).

- Cao, R., Bajgirani, A. M., Mirak, S. A., Shakeri, S., Zhong, X., Enzmann, D., ... & Sung, K. (2019). Joint prostate cancer detection and Gleason score prediction in mp-MRI via FocalNet. *IEEE Transactions on Medical Imaging*, 38(11), 2496-2506.
- Catalona, W. J., Smith, D. S., Ratliff, T. L., & Basler, J. W. (1993). Detection of organ-confined prostate cancer is increased through prostate-specific antigen—based screening. *JAMA*, 270(8), 948-954.
- Chen, M., Zheng, H., Lu, C., Tu, E., Yang, J., & Kasabov, N. (2018). A spatio-temporal fully convolutional network for breast lesion segmentation in DCE-MRI. In *Neural Information Processing: 25th International Conference, ICONIP 2018, Siem Reap, Cambodia, December 13–16, 2018, Proceedings, Part VII* 25 (pp. 358-368). Springer International Publishing.
- Ciresan, D., Giusti, A., Gambardella, L., & Schmidhuber, J. (2012). Deep neural networks segment neuronal membranes in electron microscopy images. *Advances in Neural Information Processing Systems*, 25.
- De Vente, C., Vos, P., Hosseinzadeh, M., Pluim, J., & Veta, M. (2020). Deep learning regression for prostate cancer detection and grading in bi-parametric MRI. *IEEE Transactions on Biomedical Engineering*, 68(2), 374-383.
- Duran, A., Dussert, G., Rouvière, O., Jaouen, T., Jodoin, P. M., & Lartizien, C. (2022). ProstAttention-Net: A deep attention model for prostate cancer segmentation by aggressiveness in MRI scans. *Medical Image Analysis*, 77, 102347.
- Epstein, J. I., Egevad, L., Amin, M. B., Delahunt, B., Srigley, J. R., Humphrey, P. A., & Grading Committee. (2016). The

- 2014 International Society of Urological Pathology (ISUP) consensus conference on Gleason grading of prostatic carcinoma: definition of grading patterns and proposal for a new grading system. *The American Journal of Surgical Pathology*, 40(2), 244-252.
- Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88, 303-338.
- Gaur, L., Bhatia, U., Jhanjhi, N. Z., Muhammad, G., & Masud, M. (2023). Medical image-based detection of COVID-19 using deep convolution neural networks. *Multimedia Systems*, 29(3), 1729-1738.
- Ghai, S., & Haider, M. A. (2015). Multiparametric-MRI in diagnosis of prostate cancer. *Indian Journal of Urology*, 31(3), 194-201.
- Giganti, F., Rosenkrantz, A. B., Villeirs, G., Panebianco, V., Stabile, A., Emberton, M., & Moore, C. M. (2019). The evolution of MRI of the prostate: the past, the present, and the future. *American Journal of Roentgenology*, 213(2), 384-396.
- Gleason, D. F. (1966). Classification of prostatic carcinomas. *Cancer Chemother. Rep.*, 50, 125-128.
- Hadaschik, B. A., Kuru, T. H., Tulea, C., Rieker, P., Popeneciu, I. V., Simpfendörfer, T., ... & Hohenfellner, M. (2011). A novel stereotactic prostate biopsy system integrating pre-interventional magnetic resonance imaging and live ultrasound fusion. *The Journal of Urology*, 186(6), 2214-2220.
- Hambarde, P., Talbar, S., Mahajan, A., Chavan, S., Thakur, M., & Sable, N. (2020). Prostate lesion segmentation in MR

images using radiomics based deeply supervised U-Net. *Biocybernetics and Biomedical Engineering*, 40(4), 1421-1435.

Hanley, J. A., & McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1), 29-36.

Hara, N., Okuizumi, M., Koike, H., Kawaguchi, M., & Bilim, V. (2005). Dynamic contrast-enhanced magnetic resonance imaging (DCE-MRI) is a useful modality for the precise detection and staging of early prostate cancer. *The Prostate*, 62(2), 140-147.

Harvey, C. J., Pilcher, J., Richenberg, J., Patel, U., & Frauscher, F. (2012). Applications of transrectal ultrasound in prostate cancer. *The British Journal of Radiology*, 85(special\_issue\_1), S3-S17.

Hricak, H., Doms, G. C., McNeal, J. E., Mark, A. S., Marotti, M., Avallone, A., ... & Tanagho, E. A. (1987). MR imaging of the prostate gland: normal anatomy. *American Journal of Roentgenology*, 148(1), 51-58.

International Agency for Research on Cancer. (2024a). *Rate per 100 000 incidence males in 2022 continents-cancer statistics world*. Accessed on: December 5, 2024, [https://gco.iarc.fr/today/en/dataviz/bars?mode=cancer&group\\_populations=1&sexes=1](https://gco.iarc.fr/today/en/dataviz/bars?mode=cancer&group_populations=1&sexes=1)

International Agency for Research on Cancer. (2024b). *Absolute numbers, incidence males in 2022 cancer statistics Türkiye*. Accessed on: December 12, 2024, [https://gco.iarc.fr/today/en/dataviz/bars?mode=cancer&group\\_populations=1&sexes=1&populations=792&key=total](https://gco.iarc.fr/today/en/dataviz/bars?mode=cancer&group_populations=1&sexes=1&populations=792&key=total)

- Jensen, C., Sørensen, K. S., Jørgensen, C. K., Nielsen, C. W., Høy, P. C., Langkilde, N. C., & Østergaard, L. R. (2019). Prostate zonal segmentation in 1.5 T and 3T T2W MRI using a convolutional neural network. *Journal of Medical Imaging*, 6(1), 014501-014501.
- Johnson, L. M., Turkbey, B., Figg, W. D., & Choyke, P. L. (2014). Multiparametric MRI in prostate cancer management. *Nature Reviews Clinical Oncology*, 11(6), 346-353.
- Jung, S. I., Donati, O. F., Vargas, H. A., Goldman, D., Hricak, H., & Akin, O. (2013). Transition zone prostate cancer: incremental value of diffusion-weighted endorectal MR imaging in tumor detection and assessment of aggressiveness. *Radiology*, 269(2), 493-503.
- Kasivisvanathan, V., Rannikko, A. S., Borghi, M., Panebianco, V., Mynderse, L. A., Vaarala, M. H., ... & Moore, C. M. (2018). MRI-targeted or standard biopsy for prostate-cancer diagnosis. *New England Journal of Medicine*, 378(19), 1767-1777.
- Kaur, C., & Garg, U. (2023). Artificial intelligence techniques for cancer detection in medical image processing: A review. *Materials Today: Proceedings*, 81, 806-809.
- Kim, H. E., Cosa-Linan, A., Santhanam, N., Jannesari, M., Maros, M. E., & Ganslandt, T. (2022). Transfer learning for medical image classification: a literature review. *BMC Medical Imaging*, 22(1), 69.
- Kuhl, C. K., Bruhn, R., Krämer, N., Nebelung, S., Heidenreich, A., & Schrading, S. (2017). Abbreviated biparametric prostate MR imaging in men with elevated prostate-specific antigen. *Radiology*, 285(2), 493-505.

- Kumar, V., Bora, G. S., Kumar, R., & Jagannathan, N. R. (2018). Multiparametric (mp) MRI of prostate cancer. *Progress in Nuclear Magnetic Resonance Spectroscopy*, 105, 23-40.
- Langan, R. C. (2019). Benign prostatic hyperplasia. *Primary Care: Clinics in Office Practice*, 46(2), 223-232.
- Li, Y., Lin, C., Zhang, Y., Feng, S., Huang, M., & Bai, Z. (2023). Automatic segmentation of prostate MRI based on 3D pyramid pooling Unet. *Medical Physics*, 50(2), 906-921.
- Milletari, F., Navab, N., & Ahmadi, S. A. (2016). V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)* (pp. 565-571). IEEE.
- Mohler, J. L. (2010). The 2010 NCCN clinical practice guidelines in oncology on prostate cancer. *Journal of the National Comprehensive Cancer Network*, 8(2), 145-145.
- Mohler, J. L., Armstrong, A. J., Bahnson, R. R., D'Amico, A. V., Davis, B. J., Eastham, J. A., ... & Freedman-Cass, D. A. (2016). Prostate cancer, version 1.2016. *Journal of the National Comprehensive Cancer Network*, 14(1), 19-30.
- Mottet, N., van den Bergh, R. C., Briers, E., Van den Broeck, T., Cumberbatch, M. G., De Santis, M., ... & Cornford, P. (2021). EAU-EANM-ESTRO-ESUR-SIOG guidelines on prostate cancer—2020 update. Part 1: screening, diagnosis, and local treatment with curative intent. *European Urology*, 79(2), 243-262.
- Nguyen-Nielsen, M., & Borre, M. (2016). Diagnostic and therapeutic strategies for prostate cancer. *Seminars in Nuclear Medicine*, 46(6), 484-490. WB Saunders.
- Pinto, P. A., Chung, P. H., Rastinehad, A. R., Baccala, A. A., Kruecker, J., Benjamin, C. J., ... & Wood, B. J. (2011). Magnetic resonance imaging/ultrasound fusion guided

prostate biopsy improves cancer detection following transrectal ultrasound biopsy and correlates with multiparametric magnetic resonance imaging. *The Journal of Urology*, 186(4), 1281-1285.

Ramesh, K. K. D., Kumar, G. K., Swapna, K., Datta, D., & Rajest, S. S. (2021). A review of medical image segmentation algorithms. *EAI Endorsed Transactions on Pervasive Health and Technology*, 7(27), e6-e6.

Rawla, P. (2019). Epidemiology of prostate cancer. *World Journal of Oncology*, 10(2), 63.

Redmon, J. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015* (pp. 234-241). Springer International Publishing.

Rundo, L., Han, C., Zhang, J., Hataya, R., Nagano, Y., Militello, C., ... & Mauri, G. (2020). CNN-based prostate zonal segmentation on T2-weighted MR images: a cross-dataset study. In *Neural Approaches to Dynamics of Signal Exchanges* (pp. 269-280).

Saha, A., Hosseinzadeh, M., & Huisman, H. (2021). End-to-end prostate cancer detection in bpMRI via 3D CNNs: effects of attention mechanisms, clinical priori and decoupled false positive reduction. *Medical Image Analysis*, 73, 102155.

Scheenen, T. W., Rosenkrantz, A. B., Haider, M. A., & Fütterer, J. J. (2015). Multiparametric magnetic resonance imaging in prostate cancer management: current status and future perspectives. *Investigative Radiology*, 50(9), 594-600.

- Schieda, N., Lim, C. S., Zabihollahy, F., Abreu-Gomez, J., Krishna, S., Woo, S., ... & Turkbey, B. (2021). Quantitative prostate MRI. *Journal of Magnetic Resonance Imaging*, 53(6), 1632-1645.
- Siddique, N., Paheding, S., Elkin, C. P., & Devabhaktuni, V. (2021). U-net and its variants for medical image segmentation: A review of theory and applications. *IEEE Access*, 9, 82031-82057.
- Smith, D. S., & Catalona, W. J. (1995). Interexaminer variability of digital rectal examination in detecting prostate cancer. *Urology*, 45(1), 70-74.
- Somford, D. M., Fütterer, J. J., Hambrock, T., & Barentsz, J. O. (2008). Diffusion and perfusion MR imaging of the prostate. *Magnetic Resonance Imaging Clinics of North America*, 16(4), 685-695.
- Stabile, A., Giganti, F., Rosenkrantz, A. B., Taneja, S. S., Villeirs, G., Gill, I. S., ... & Kasivisvanathan, V. (2020). Multiparametric MRI for prostate cancer diagnosis: current status and future directions. *Nature Reviews Urology*, 17(1), 41-61.
- Suarez-Ibarrola, R., Sigle, A., Eklund, M., Eberli, D., Miernik, A., Benndorf, M., ... & Gratzke, C. (2022). Artificial intelligence in magnetic resonance imaging-based prostate cancer diagnosis: Where do we stand in 2021?. *European Urology Focus*, 8(2), 409-417.
- Transformation of technology in healthcare [Sağlıkta Teknoloji Dönüşümü]. (2024). Ankara Nobel Tıp Kitapevi.
- Turkbey, B., Albert, P. S., Kurdziel, K., & Choyke, P. L. (2009). Imaging localized prostate cancer: current approaches and new developments. *American Journal of Roentgenology*, 192(6), 1471-1480.

- Turkbey, B., Rosenkrantz, A. B., Haider, M. A., Padhani, A. R., Villeirs, G., Macura, K. J., ... & Weinreb, J. C. (2019). Prostate imaging reporting and data system version 2.1: 2019 update of prostate imaging reporting and data system version 2. *European Urology*, 76(3), 340-351.
- Verma, S., Turkbey, B., Muradyan, N., Rajesh, A., Cornud, F., Haider, M. A., ... & Harisinghani, M. (2012). Overview of dynamic contrast-enhanced MRI in prostate cancer diagnosis and management. *American Journal of Roentgenology*, 198(6), 1277-1288.
- Vu, C. C., Zamdborg, L., Siddiqui, Z. A., Gustafson, G. S., Krauss, D. J., & Guerrero, T. M. (2018). Automatic segmentation using convolutional neural networks in prostate cancer. *International Journal of Radiation Oncology, Biology, Physics*, 102(3), S61.
- Wang, M., Zhang, Q., Lam, S., Cai, J., & Yang, R. (2020). A review on application of deep learning algorithms in external beam radiotherapy automated treatment planning. *Frontiers in Oncology*, 10, 580919.
- Wang, R., Lei, T., Cui, R., Zhang, B., Meng, H., & Nandi, A. K. (2022). Medical image segmentation using deep learning: A survey. *IET Image Processing*, 16(5), 1243-1267.
- Wang, T., Lei, Y., Fu, Y., Wynne, J. F., Curran, W. J., Liu, T., & Yang, X. (2021). A review on medical imaging synthesis using deep learning and its clinical applications. *Journal of Applied Clinical Medical Physics*, 22(1), 11-36.
- Wu, L. M., Xu, J. R., Ye, Y. Q., Lu, Q., & Hu, J. N. (2012). The clinical value of diffusion-weighted imaging in combination with T2-weighted imaging in diagnosing prostate carcinoma: a systematic review and meta-

analysis. *American Journal of Roentgenology*, 199(1), 103-110.

Yüzkat, M., İlhan, H. O., & Aydın, N. (2021). Multi-model CNN fusion for sperm morphology analysis. *Computers in Biology and Medicine*, 137, 104790.

Zhou, S. K., Greenspan, H., & Shen, D. (Eds.). (2023). *Deep learning for medical image analysis*. Academic Press.

Zhou, Z. H. (2021). *Machine learning*. Springer Nature.

# **GIDA TEDARİK ZİNCİRLERİNDE VERİ GÜVENİLİRLİĞİ VE BLOKZİNCİR TEKNOLOJİSİNİN KULLANIMI<sup>1</sup>**

**Bora ASLAN<sup>2</sup>**

**Füsun YAVUZER ASLAN<sup>3</sup>**

## **1. GİRİŞ**

Gıda tedarik zinciri, tarımsal üretimden işleme, depolama, taşıma, dağıtım, perakende ve son tüketiciye kadar uzanan; kamu kurumları, çiftçiler, kooperatifler, işleyiciler, lojistik işletmeleri, laboratuvarlar, sertifikasyon kuruluşları ve perakendeciler gibi çok sayıda aktörün bir arada bulunduğu ağ yapısıdır. Bu yapı içerisinde ürünün yalnızca fiziksel olarak hareket etmesi yeterli değildir. Ürüne ilişkin kimlik, menşe, parti, miktar, üretim zamanı, işleme koşulları, sıcaklık, taşıma rotası, kalite kontrol sonuçları ve teslimat olaylarının da güvenilir biçimde aktarılması gerekir. Nitekim güncel düzenleyici yaklaşımlar giderek ürün hareketlerinin olay ve veri ögeleri üzerinden izlenmesine yönelmektedir. ABD Gıda ve İlaç Dairesinin Food Traceability Rule düzenlemesi, belirli gıdalar için Kritik İzleme Olayları ile bunlara bağlı “Anahtar Veri Ögelerinin” kayıt altına alınmasını

---

<sup>1</sup> Bu kitap bölümü, TÜBİTAK 1833 SAYEM Yeşil Dönüşüm Programı kapsamında desteklenen “TARIM2030 – Gıdada Arz Güvenliği, Sürdürülebilirlik, Yenilik ve Şeffaflık İçin Tarımda Dijitalleşme Programı” bünyesindeki Ürünleştirme Projesi 1: Blokzincir Tabanlı İzlenebilirlik Platformunun Geliştirilmesi çalışmaları kapsamında hazırlanmıştır. <https://tarim2030program.org>

<sup>2</sup> Doç. Dr., Kırklareli Üniversitesi, Mühendislik Fakültesi, Yazılım Mühendisliği Bölümü, ORCID: 0000-0002-8069-8204.

<sup>3</sup> Dr. Öğr. Üyesi, Kırklareli Üniversitesi, Mühendislik Fakültesi, Yazılım Mühendisliği Bölümü, ORCID: 0000-0001-7096-3425.

ve gerektiğinde düzenleyici otoriteye hızlı biçimde sunulmasını öngörmektedir. (FDA, 2026)

Gıda tedarik zincirinde verinin önemi, gıda güvenliğinin halk sağlığı üzerindeki etkisi düşünüldüğünde daha belirgin hâle gelmektedir. Dünya Sağlık Örgütü, 2026 yılında güncellediği değerlendirmesinde gıda güvenliği, beslenme ve gıda güvencesinin birbirinden ayrı düşünülmemeyeceğini; kontamine gıdalara bağlı hastalıkların küresel sağlık ve ekonomi üzerinde önemli bir yük oluşturduğunu vurgulamaktadır. WHO'nun güncel tahminlerine göre yılda yaklaşık 866 milyon insan kontamine gıda tüketiminin ardından hastalanmakta ve 1,52 milyon kişi yaşamını yitirmektedir. Kuruluş ayrıca gıda güvenliğinin üreticiden tüketiciye kadar paylaşılan bir sorumluluk olduğunu belirtmektedir. (WHO, 2026)

Bu noktada Türkçe terminolojide sıklıkla birbirinin yerine kullanılan iki kavramın ayrılması gerekir. Bu çalışmada **gıda güvenliği**, İngilizcedeki *food safety* karşılığı olarak gıdanın biyolojik, kimyasal ve fiziksel tehlikeler bakımından tüketici sağlığına zarar vermemesini; **gıda güvencesi** veya daha dar kapsamda **gıda arz güvenliği** ise *food security* kavramının arz ve süreklilik boyutlarını ifade etmektedir. FAO'nun klasik çerçevesi gıda güvencesini bulunabilirlik, ekonomik ve fiziksel erişim, kullanım ve zaman içinde istikrar olmak üzere dört boyutta ele almaktadır. Buna göre arz güvenliği esas olarak bulunabilirlik ve istikrar boyutlarıyla ilişkilidir; ancak yeterli miktarda gıda bulunması, tek başına hanelerin gıdaya erişebildiği veya gıdanın güvenli olduğu anlamına gelmez. (FAO, 2008)

Gıda tedarik zincirinin dijitalleşmesi, fiziksel ürün ile onun dijital temsilinin birbirine bağlandığı yeni bir yönetim alanı doğurmuştur. Bir paletin RFID etiketiyle okunması, soğuk zincir sıcaklığının sensörle kaydedilmesi, bir ürünün QR kodla tüketiciye açılması, laboratuvar analizinin parti numarasıyla

ilişkilendirilmesi veya sevkiyatın işletme bilgi sisteminde kaydedilmesi, zincirin farklı noktalarında veri üretir. Sorun, bu verilerin yalnızca var olması değil, doğru ürünle ilişkilendirilmiş, anlamı korunmuş, değiştirilmemiş, zamanında paylaşılmış ve gerektiğinde denetlenebilir olmasıdır. Gualo ve arkadaşlarının (2021) veri kalitesi üzerine gerçekleştirdiği endüstriyel çalışmalar da örgütsel kararların dayandıkları verinin kalitesinden daha iyi olamayacağını ve veri güvenilirliğinin sistematik kalite yönetimi gerektirdiğini göstermektedir.

Blokszincir teknolojisinin gıda sektöründe ilgi görmesinin temelinde bu güven sorunu bulunmaktadır. Blokszincir, dağıtık bir ağda doğrulanan işlemlerin kriptografik olarak birbirine bağlandığı, sonradan yapılan değişikliklerin tespit edilmesini sağlayan ortak bir kayıt yapısı sunmaktadır. NIST'in teknik tanımında blokszincirler “tamper-evident” ve “tamper-resistant” dağıtık dijital kayıt sistemleri olarak nitelenmektedir; normal çalışma koşullarında yayımlanmış bir işlemin geriye dönük değiştirilmesi engellenmekte veya son derece güçleştirilmektedir (Yaga et al., 2018).

Bununla birlikte, blokszincirin değiştirilemezliği ile verinin doğruluğu aynı şey değildir. Zincir, “10 °C” olarak kaydedilmiş bir değerin daha sonra “4 °C” olarak değiştirilmesini engelleyebilir; ancak bozuk veya yanlış kalibre edilmiş bir sensörün gerçekte 14 °C olan ortamı 10 °C olarak ölçmesini kendiliğinden fark edemez. Benzer biçimde, çalışan yanlış parti numarası girerse veya üretici menşe bilgisini kasıtlı olarak yanlış beyan ederse, blokszincir yanlış bilginin değiştirilmemiş bir kaydını tutabilir. Leteane ve Ayalew'in (2024) gıda izlenebilirliği üzerine yaptığı çalışma bu problemi açık biçimde GIGO (garbage in, garbage out) sorunu altında tartışmakta ve blokszincirin kaynağında hatalı ya da kötü niyetli olarak oluşturulmuş veriyi kendiliğinden düzeltmediğini göstermektedir.

Dolayısıyla blokzincirin gıda sektöründeki değeri “blokzincire kaydedilen her veri güvenilirdir” önermesinde değil; güvenilir biçimde elde edilmiş verinin daha sonra değiştirilmesini güçleştiren, farklı taraflar arasında ortak kayıt oluşturabilen ve denetim izini kuvvetlendirebilen bir altyapı olmasında aranmalıdır. Kamilaris, Fonts ve Prenafeta-Boldú (2019), tarım ve gıda alanındaki erken dönem blokzincir uygulamalarını incelerken teknolojinin özellikle şeffaflık ve izlenebilirlik bakımından olanak sunduğunu, ancak teknik, örgütsel, düzenleyici ve eğitimle ilgili engeller nedeniyle uygulama olgunluğunun dikkatle değerlendirilmesi gerektiğini belirtmektedir. Benzer biçimde Garrard ve Fielke'nin (2020) Avustralya karides sektörü üzerine çalışması, bazı durumlarda beklenen faydanın blokzincirin kendisinden çok, blokzincire geçiş için gerekli olan temel dijitalleşmeden kaynaklanabileceğini ortaya koymaktadır.

Bu çalışmanın temel yaklaşımı bu ayırmadan hareket etmektedir. Amaç blokzinciri gıda tedarik zincirlerinin bütün veri sorunlarını çözen bağımsız bir teknoloji olarak sunmak değil; hangi veri güvenilirliği sorunlarına doğrudan müdahale edebildiğini, hangi sorunlarda başka teknolojilere ve kurumsal mekanizmalara bağımlı olduğunu ve hangi koşullarda kullanımının ekonomik ve örgütsel açıdan anlamlı hâle geldiğini değerlendirmektir. Bu doğrultuda veri güvenilirliği kavramları, blokzincirin teknik işlevleri, IoT ve standartlarla entegrasyonu, gerçek uygulamalar ve teknolojinin temel sınırlılıkları birlikte ele alınmaktadır.

## **2. GIDA TEDARİK ZİNCİRLERİNDE VERİ GÜVENİLİRLİĞİ**

### **2.1. Gıda Tedarik Zincirlerinde Veri Üretimi ve Paylaşımı**

Geleneksel gıda izlenebilirliği, büyük ölçüde “bir adım geriye–bir adım ileriye” anlayışıyla işletmelerin kendi tedarikçileri ve müşterileri hakkında kayıt tutmasına dayanmaktaydı. Dijital tedarik zincirlerinde ise yalnızca ticari ilişkinin taraflarını bilmek yeterli olmamakta; belirli bir ürün partisinin ne zaman, nerede, hangi işlemde geçtiğini ve hangi koşullara maruz kaldığını belirleyebilmek gerekmektedir. FDA'nın mevcut yaklaşımında hasat, ilk soğutma, ilk paketlenme, sevkiyat, teslim alma ve dönüşüm gibi aşamalar Kritik İzleme Olayı olarak tanımlanmakta ve olaylara ilişkin lot kodu ile diğer veri öğelerinin birbirine bağlanması beklenmektedir. (FDA, 2026)

Benzer bir olay merkezli yaklaşım GS1 EPCIS standardında görülmektedir. EPCIS, farklı işletmelerdeki uygulamaların görünürlük olay verisi oluşturması ve paylaşması için standartlaştırılmış bir yapı sunmaktadır. EPCIS 2.0'da bir olayın anlamı temel olarak “ne”, “ne zaman”, “nerede”, “neden” ve “nasıl” boyutlarıyla ifade edilmekte; JSON/JSON-LD, REST bağlayıcıları ve sensör bilgilerinin taşınmasına yönelik SensorElement gibi özellikler desteklenmektedir. Böylece ürün kimliğinin yanı sıra konum, zaman, iş süreci ve bazı durumlarda ölçüm koşulları birlikte ifade edilebilmektedir. (GS1, 2022)

Bu bağlamda veri güvenilirliliği tek boyutlu bir özellik değildir. Doğruluk, kaydedilen değer gerçek dünyadaki olguyu ne ölçüde doğru temsil ettiğini ifade eder. Bir sıcaklık sensörünün 4,1 °C ölçmesi ve gerçek sıcaklığın buna yakın olması doğrulukla ilgilidir. Bütünlük, verinin yetkisiz şekilde değiştirilmemiş veya bozulmamış olmasıyla ilgilidir. Veri kalitesi, doğruluk yanında

eksiksizlik, güncellik, tutarlılık, uygunluk ve kullanılabilirlik gibi daha geniş özellikleri içine alan üst bir kavramdır. Gualo ve arkadaşlarının (2021) ISO/IEC 25012 tabanlı endüstriyel deneyimleri, veri kalitesinin yalnızca teknolojik koruma değil, ölçme, değerlendirme ve yönetim süreçleriyle yönetilmesi gereken kurumsal bir konu olduğunu göstermektedir.

Özgünlük, verinin veya iddianın gerçekten iddia edilen kaynaktan geldiğine dair güveni; veri kökeni/provenance, verinin kim tarafından, hangi süreçte ve hangi dönüşümlerden geçerek üretildiğine ilişkin geçmişi ifade eder. İzlenebilirlik, bir ürünün veya partinin tedarik zinciri boyunca geçmiş ve gerektiğinde ileri yönlü hareketlerinin ilişkilendirilerek bulunabilmesidir. Şeffaflık ise ilgili tarafların gerekli bilgiyi görebilme düzeyidir. Bu nedenle izlenebilir bir sistemin tamamen şeffaf olması gerekmediği gibi, çok fazla verinin görünür olması da verinin doğru veya güvenilir olduğunu kanıtlamaz. Garrard ve Fielke'nin (2020) çalışması, güvenilir provenance üretiminin yalnızca kayıtların sonradan değiştirilmemesinden ibaret olmadığını; fiziksel ürüne ilişkin başlangıç bilgisinin de güvenilir olması gerektiğini göstermektedir.

Denetlenebilirlik, bir kayıt veya kararın oluşum sürecinin daha sonra yetkili kişiler tarafından incelenebilmesine karşılık gelir. Blokzincirin işlem geçmişini geriye dönük görünür tutması bu özelliği güçlendirebilir. Veri güvenliği ise gizlilik, bütünlük, erişilebilirlik, kimlik doğrulama ve yetkilendirme gibi daha geniş güvenlik hedeflerini kapsar. Dolayısıyla “veri güvenliği” ile “veri bütünlüğü” eş anlamlı değildir; bütünlük güveniğin yalnızca bir boyutudur. Benzer biçimde bütünlüğü korunmuş veri yanlış olabilir: bir kayıt oluşturulduğu andan itibaren değiştirilmeden saklanmış olsa dahi başlangıçta yanlış girilmiş olabilir. Bu ayrım, blokzincirin gıda sektöründeki katkısını doğru değerlendirmek için temel önemdedir. (Leteane & Ayalew, 2024; Yaga et al., 2018)

## **2.2. Veri Güvenilirliğini Etkileyen Başlıca Sorunlar**

İlk sorun tedarik zincirinin parçalı bilgi altyapısıdır. Büyük perakendeciler gelişmiş ERP, depo yönetimi ve izlenebilirlik sistemlerine sahipken küçük üreticiler kâğıt kayıtlar, elektronik tablolar veya birbirleriyle konuşmayan yerel uygulamalar kullanabilmektedir. Sonuç olarak aynı ürün için farklı biçimlerde tanımlanmış alanlar, farklı parti kodları veya standartlaştırılmamış lokasyon bilgileri oluşabilmektedir. 2026 tarihli gerçek bir meyve tedarik zinciri pilotunu inceleyen Romão ve arkadaşları, blokzincir uygulamalarında temel engellerden birinin mevcut kurumsal sistemlerle entegrasyon olduğunu; API, dosya aktarımı, veri dönüştürme, mesajlaşma ara katmanı ve blokzincir yükleyicisi gibi adaptörlere ihtiyaç duyulduğunu göstermektedir. Çalışma üç işletmeyle yürütülen bir pilot niteliğinde olduğundan sonuçlar genellenirken ihtiyat gerektirir. (Romão et al., 2026)

İkinci sorun insan kaynaklı veri hatalarıdır. Manuel veri girişi, yanlış kod okutma, aynı olayın birden fazla kez kaydedilmesi, eksik kayıt, hatalı zaman veya lokasyon bilgisi gibi sorunlara açıktır. Kasıtlı davranışlar söz konusu olduğunda sorun daha da büyür: üretim yerinin yanlış beyan edilmesi, sertifikasyon bilgisinin usulsüz kullanılması veya düşük kaliteli ürünün farklı bir partiyle ilişkilendirilmesi mümkündür. Blokzincir işlem kaydının sonradan değiştirilmesini zorlaştırırsa da yanlış beyanın fiziksel dünyada doğru olup olmadığını doğrudan doğrulayamaz. (Caldarelli, 2022; Leteane & Ayalew, 2024)

Üçüncü sorun sensör ve IoT verisinin güvenilirliğidir. Otomasyon, insan kaynaklı kayıt hatalarını azaltabilir; fakat otomatik kayıt doğru kayıtlarla eş anlamlı değildir. Sensörün kalibrasyonu bozulabilir, cihaz yanlış konumlandırılabilir, batarya zayıflayabilir, ağ iletişimi kesilebilir veya cihaz fiziksel ve siber müdahaleye maruz kalabilir. Leteane ve Ayalew (2024),

IoT verisinin güvenilirliğini değerlendirirken cihaz kalibrasyonu, enerji düzeyi ve zaman içindeki ölçümlerin tutarlılığı gibi değişkenleri ayrıca ele almakta; dolayısıyla blokzincirin sensörden gelen ölçümün fiziksel doğruluğunu tek başına garanti edemediğini göstermektedir.

Dördüncü sorun fiziksel ürün ile dijital kayıt arasındaki bağıdır. Dijital sistem belirli QR kodlu paketin organik üretimden geldiğini gösterebilir; ancak etiket başka ürüne aktarılmışsa doğru bir dijital geçmiş yanlış fiziksel ürüne bağlanabilir. Bu nedenle güvenilir izlenebilirlik; seri veya lot kimliği, kurcalamaya karşı korumalı ambalajlar, RFID/QR altyapısı, gerektiğinde laboratuvar analizleri, sertifikasyon ve fiziksel gözetim mekanizmalarıyla desteklenmelidir. Galvez, Mejuto ve Simal-Gandara (2018), blokzincirin gıda izlenebilirliğindeki potansiyelini değerlendirirken fiziksel ve analitik doğrulama ile dijital kayıt arasındaki ilişkinin kritik olduğunu vurgulayan erken dönem çalışmalardan biridir.

Bütün bu unsurlar birlikte düşünüldüğünde veri güvenilirliğini bir zincir şeklinde ele almak daha doğru görünmektedir. Bu bölümün literatür sentezinden hareketle önerdiği çerçevede güvenilir gıda verisi en az beş katmana dayanır: güvenilir fiziksel kimlik, güvenilir veri toplama, ortak semantik ve standartlar, güvenilir kayıt ve aktarım altyapısı ve hesap verebilir yönetim. Blokzincir bu yapının özellikle kayıt bütünlüğü ve çok taraflı denetim katmanında güçlüdür; diğer katmanların yerine geçmez. Bu, mevcut literatürdeki GIGO, oracle (kapalı bir yapıya sahip olan blok zinciri ağlarının dış dünyadaki verilere doğrudan ve güvenli bir şekilde ulaşamaması), veri standardizasyonu ve organizasyonel yönetim bulgularından çıkarılan bir sentezdir. (Caldarelli, 2022; Garrard & Fielke, 2020; GS1, 2022; Leteane & Ayalew, 2024)

### **3. BLOKZİNCİR TEKNOLOJİSİNİN TEMEL ÖZELLİKLERİ VE VERİ GÜVENİLİRLİĞİYLE İLİŞKİSİ**

Blokszincir, işlemlerin mutabakat mekanizmalarıyla doğrulandığı ve kriptografik bağlantılar aracılığıyla birbirine bağlanan bloklarda saklandığı dağıtık bir kayıt sistemidir. Kriptografik hash işlevleri, kayıtlardaki değişikliklerin tespit edilmesini sağlar. Örneğin bir belgenin hash değerinin blokszincire kaydedilmesi, belgenin daha sonra değiştirilip değiştirilmediğinin kontrol edilmesine imkan verir. Ancak hash, belgenin içeriğinin doğru olduğunu değil, kayıtlı özetle karşılaştırılan belgenin aynı olduğunu gösterir. Bu nedenle blokszincirin değiştirilemezliği mutlak bir özellik olarak değil, değişikliklerin tespit edilmesini sağlayan ve mutabakat kuralları altında değiştirilmesi güç bir yapı olarak değerlendirilmelidir. (Yaga et al., 2018)

Gıda tedarik zincirlerinde ağ mimarisi, veri gizliliği ve paydaşlar arasındaki güven ilişkisine göre belirlenmelidir. İzinli ağlarda katılımcıların kimlikleri ve erişim yetkileri kontrol edilirken, izinsiz ağlar daha geniş kamusal doğrulama imkanı sunar. Hyperledger Fabric gibi izinli platformlar, ticari açıdan hassas verilerin paylaşımını sınırlandırabilen kurumsal uygulamalara uygundur. Hibrit modeller ise operasyonel verileri izinli veya zincir dışı sistemlerde tutarken belirli doğrulama kanıtlarını kamusal zincire kaydedebilir. Bununla birlikte hiçbir mimari bütün tedarik zincirleri için tek başına üstün değildir; seçim gizlilik, performans, yönetim ve mevzuat gereksinimlerine bağlıdır. (Wang, Li, Liu, Zhang, & Pan, 2023)

Akıllı sözleşmeler, önceden tanımlanmış koşullar gerçekleştiğinde işlemleri otomatik olarak yürüten yazılımlardır. Gıda tedarik zincirinde sıcaklık sınırının aşılması, eksik veri girişi veya sertifika süresinin dolması gibi durumlarda uyarı oluşturmak

ya da işlemleri sınırlandırmak için kullanılabilir. Ancak bu sözleşmeler yalnızca kendilerine iletilen veriler ve programlanan kurallar doğrultusunda çalışır. Sensörler, laboratuvarlar veya işletme sistemleri gibi dış veri kaynaklarının hatalı bilgi sağlaması durumunda, blokzincir yanlış bilginin değiştirilmeksizin kaydedilmesini engelleyemez. Bu durum, fiziksel dünyadan gelen verilerin güvenilirliğine ilişkin oracle problemini ortaya çıkarır. (Caldarelli, 2022)

Dolayısıyla blokzincirin veri güvenilirliğine katkısı kayıt öncesi güvenilirlik, kayıt anındaki doğrulama ve kayıt sonrası bütünlük olmak üzere üç düzeyde ele alınmalıdır. Teknoloji özellikle kayıt sonrası bütünlüğün korunmasında güçlüdür. Verinin başlangıçtaki doğruluğu ise sensör kalibrasyonu, dijital kimlik, laboratuvar analizi, çoklu kaynaktan doğrulama ve kurumsal denetim gibi tamamlayıcı mekanizmalara bağlıdır. Bu nedenle blokzincir, gıda tedarik zincirlerinde veri güvenilirliğini tek başına garanti eden bir çözüm değil, güvenilir veri üretimi ve doğrulama süreçlerini destekleyen bir altyapı olarak değerlendirilmelidir. (Caldarelli, 2022; Yaga et al., 2018)

#### **4. BLOKZİNCİR TEKNOLOJİSİNİN GIDA TEDARİK ZİNCİRLERİNDEKİ UYGULAMA ALANLARI**

##### **4.1. Ürün Kökeni ve İzlenebilirlik**

Blokzincirin gıda sektöründeki en sık tartışılan kullanım alanı ürünün üretim kaynağından tüketiciye kadar izlenmesidir. Üretici kimliği, parsel, hasat zamanı, işleme tesisi, lojistik işlemler ve perakende noktası birbirine bağlandığında, tek bir partinin geçmişi daha hızlı sorgulanabilir. Bu işlev özellikle çok taraflı zincirlerde, aynı ürün geçmişinin farklı işletmelerin kendi kayıtlarından sonradan birleştirilmesi yerine ortak ve denetlenebilir olay kaydı üzerinden oluşturulması bakımından

anlamlıdır. Kamilaris ve arkadaşlarının (2019) literatür değerlendirmesi ve Casino ve arkadaşlarının (2021) süt tedarik zinciri vaka çalışması, blokzincirin gıda alanındaki kullanımının ağırlıklı olarak bu izlenebilirlik ve provenance ekseninde geliştiğini göstermektedir.

#### **4.2. Geri Çağırma ve Gıda Güvenliği**

Gıda kaynaklı bir salgında sorunun hangi lotlardan kaynaklandığını ve ürünlerin nereye dağıtıldığını hızlı belirlemek, geri çağırmanın kapsamını daraltabilir. FDA'nın izlenebilirlik kuralının açık hedeflerinden biri potansiyel olarak kontamine gıdanın daha hızlı tespit edilip piyasadan çıkarılmasıdır. Düzenleme blokzinciri zorunlu kılmamakta; aksine belirli veri olayları ve veri öğelerinin tutulmasını öngörmektedir. Bu ayrım önemlidir. Zira hızlı geri çağırmanın temel gereksinimi güvenilir ve birlikte çalışabilir izlenebilirlik verisidir. Blokzincir bu amaca hizmet edebilecek olası altyapılardan yalnızca biridir. (FDA, 2026)

#### **4.3. Kalite Kontrol ve Soğuk Zincir**

Et, süt, balık, taze meyve-sebze ve dondurulmuş ürünlerde sıcaklık ve taşıma süresi kalite ve güvenlikle doğrudan ilişkilidir. Spitaleri ve arkadaşlarının (2023) BioTrak platformu, dönüşüm ve taşıma süreçlerini, özellikle soğuk zincirde zaman ve sıcaklık bilgilerini izlemek üzere süreç modelleme ile blokzinciri birleştiren bir prototip sunmaktadır. Bu tür sistemler sıcaklık ihlallerinin ne zaman gerçekleştiğine ilişkin denetim izi sağlayabilir; ancak gıdanın mikrobiyolojik olarak güvenli olup olmadığının nihai değerlendirilmesi gerektiğinde analitik testler ve gıda bilimi kriterleri yine gereklidir. (WHO, 2026)

#### **4.4. Gıda Sahteciliği ve Özgünlük**

Blokzincir menşe, sertifika ve işlem geçmişinin sonradan değiştirilmesini zorlaştırarak sahtecilikle mücadeleye katkıda

bulunabilir. Bununla birlikte blokzincir ürünün biyolojik veya kimyasal kimliğini ölçmez. Örneğin “%100 zeytinyağı”, “organik”, belirli coğrafi köken veya belirli balık türü iddialarının bilimsel doğrulanması için kimi durumlarda laboratuvar analizi, sertifikasyon veya bağımsız denetim gerekir. Dijital kayıt ile fiziksel ürünün doğrulanması birbirini tamamlamadığı sürece sahtecilik yalnızca kayıt sisteminin dışına taşınabilir. Galvez ve arkadaşlarının (2018) gıda izlenebilirliği tartışması ile GIGO literatürü bu sınırı açık biçimde ortaya koymaktadır. (Leteane & Ayalew, 2024)

Gerçek uygulamalar incelendiğinde ise blokzincirin “başarı öyküsü” olarak sunulan örnekleriyle bağımsız bilimsel kanıtların düzeyi arasında önemli farklar bulunmaktadır.

**Walmart–IBM:** Çin’de domuz eti ve Amerika’da mango tedarik zincirlerinde IBM/Hyperledger tabanlı izlenebilirlik uygulamaları gerçekleştirilmiştir. Mango kaynağının bulunma süresinin günlerden saniyelere indirildiği raporlanmıştır. Bununla birlikte literatürde sıkça aktarılan “7 gün–2,2 saniye” bulgusu esas olarak Walmart/IBM pilotuna dayanmaktadır. Kamath’ın çalışmasındaki endüstri finansmanı bağlantısı nedeniyle bu sonuç tamamen bağımsız bir etki kanıtı olarak değerlendirilmemelidir. (Kamath, 2018)

**Carrefour:** Carrefour Quality Line kapsamındaki tavuk ve diğer gıda ürünlerinde, ürün geçmişini tüketiciye açan blokzincir tabanlı izlenebilirlik uygulanmıştır. Bu yaklaşım menşee ve üretim süreçlerinin görünürlüğünü artırmayı amaçlamaktadır. Uygulamanın varlığı kurumsal açıklamalarda ve bilimsel literatürde belgelenmekle birlikte, satışlar, gıda güvenliği veya hata oranları üzerindeki nedensel etkisine ilişkin bağımsız uzun dönem kanıt sınırlıdır. (Carrefour, 2018)

**TE-FOOD:** Çeşitli tarım ve gıda tedarik zincirlerinde farm-to-table izlenebilirlik hizmeti sunan platform, hedefli geri

çağırma, operasyonel görünürlük ve tüketici şeffaflığını desteklemeyi amaçlamaktadır. Platform, 2026 itibarıyla aktif kurumsal hizmet sunduğunu ve 6.000'den fazla şirkete görünürlük sağladığını bildirmektedir. Ancak bu ölçek ve performans iddiaları ağırlıklı olarak sağlayıcının kendi beyanlarına dayanmaktadır. (TE-FOOD, t.y.)

**OriginTrail / AB projeleri:** Süt, kümes hayvanları ve diğer tarımsal ürünlere yönelik pilotlarda blokzincir, bilgi grafikleri ve merkeziyetsiz bilgi altyapısı kullanılarak farklı paydaşların verilerinin bütünleştirilmesi ve provenance oluşturulması hedeflenmiştir. H2020 SmartAgriHubs ve DEMETER ekosistemlerinde gerçek pilot bağlamları bulunmakla birlikte, güncel OriginTrail platformu daha geniş DKG/AI kullanım alanlarına yönelmiştir. Gıda sektörüne özgü uzun dönem ekonomik etkilere ilişkin bağımsız kanıt sınırlıdır. (DEMETER, t.y.; OriginTrail, t.y.; SmartAgriHubs, t.y.)

**Avustralya karides yetiştiriciliği:** Karides tedarik zincirinde blokzincir tabanlı provenance yaklaşımının potansiyeli bağımsız bir akademik vaka çalışmasıyla değerlendirilmiştir. Araştırma, blokzincirin alternatif çözümlere kıyasla büyük ek kazanımlar yaratmayabileceğini ve önemli faydaların temel dijitalleşme süreçlerinden kaynaklanabileceğini göstermektedir. (Garrard & Fielke, 2020)

**Organik gıda vaka çalışmaları:** Organik ve fair-trade tedarik zincirlerinde blokzincir tabanlı farm-to-fork izlenebilirlik, ürün iddialarının ve paydaşlar arası bilgi akışının takip edilmesi amacıyla incelenmiştir. Hakemli vaka temelli araştırma, teknolojik olanakların yanı sıra örgütsel koşulların ve uygulama sınırlarının da belirleyici olduğunu ortaya koymaktadır. (van Hilten et al., 2020)

**Brezilya tahıl kalite zinciri:** Tahıl ve GDO'suz soya kalite güvencesine yönelik çalışmada, blokzincir iş ağı

aracılığıyla kalite ölçümlerinin ve sertifikasyon geçmişinin izlenmesi ele alınmıştır. Gerçek iş senaryosuna dayalı akademik araştırma, yaklaşımın potansiyelini göstermekle birlikte ön değerlendirme niteliğindedir ve büyük ölçekli, uzun dönemli saha etkisini kanıtlamamaktadır. (Lucena, Binotto, Momo, & Kim, 2018)

Walmart–IBM örneği, blokzincir literatüründeki kanıt sorununun iyi bir göstergesidir. Mango pilotunda iz sürme süresinin yaklaşık yedi günden 2,2 saniyeye düşürüldüğü sıkça aktarılmaktadır. Bununla birlikte bu sonuç kontrollü bir saha deneyiyle ölçülmüş blokzincirin bağımsız nedensel etkisi değil, belirli bir pilotta dijital izlenebilirlik sisteminin sorgu hızına ilişkin raporlanmış sonuçtur. Ayrıca eski manuel süreçle bütün dijital altyapıyı karşılaştırdığı için gözlenen farkın ne kadarının “blokzincir”den, ne kadarının standartlaştırılmış dijital veri toplama ve bütünleştirmeden kaynaklandığını ayırmak zordur. Kamath'ın (2018) vaka çalışması bu pilot için temel bilimsel kaynaklardan biridir; ancak çalışmanın finansman bağlamı bağımsızlık değerlendirilirken dikkate alınmalıdır.

Carrefour örneğinde de tüketicinin ürünün üretim ve tedarik geçmişine erişmesi, şeffaflık açısından anlamlı bir kullanım biçimidir. Ancak QR kodla daha fazla bilgi göstermek, gösterilen bilginin bağımsız olarak doğru olduğunun kanıtı değildir. Dolayısıyla böyle bir sistemde blokzincirin katkısı, kaydedilen tedarik geçmişinin bütünlüğünü ve taraflar arası paylaşımını desteklemektir; ürünün başlangıçtaki menşe beyanı yine tedarikçi doğrulaması, sertifikasyon ve diğer kontrollerle güvence altına alınmalıdır. Carrefour'un erken dönem uygulamalarının varlığı kurumsal ve akademik kaynaklarda belgelenmekle birlikte, 2026 itibarıyla programın her ürün grubu için bağımsız biçimde ölçülmüş uzun dönem performansına ilişkin güçlü kanıt belirlemek güçtür. (Carrefour, 2018; Leteane & Ayalew, 2024)

TE-FOOD örneği farklı bir kanıt kategorisindedir. Şirketin güncel sitesi, blokzincir destekli uçtan uca gıda izlenebilirliği, daha hedefli geri çağırma ve binlerce işletmede görünürlük sağlama gibi faydalar bildirmektedir. Bunlar platformun devam eden ticari varlığını göstermesi açısından değerlidir; fakat sağlayıcının kendi performans beyanları bağımsız hakemli etki analiziyle aynı kanıt düzeyine sahip değildir. Bu nedenle bilimsel bir kitap bölümünde “TE-FOOD geri çağırma süresini azaltır” yerine “TE-FOOD sağlayıcısı sistemin hedefli ve daha hızlı geri çağırmayı desteklediğini bildirmektedir” biçimindeki ifade daha doğrudur. (TE-FOOD, t.y.)

OriginTrail açısından mevcut durum daha da öğreticidir. AB destekli SmartAgriHubs ve DEMETER gibi projeler tarımsal IoT ve paydaşlar arası veri birlikte çalışabilirliğini saha pilotlarıyla sınamıştır. DEMETER, 20 pilot, 18 ülke, 6.000 çiftçi ve on binlerce cihaz/sensör ölçeğinde geniş bir dijital tarım projesi olarak yürütülmüştür; ancak bu sayıların tamamı OriginTrail veya blokzincir etkisinin ölçüsü değildir. Güncel OriginTrail platformu ise 2026'da kendisini merkeziyetsiz bilgi grafiği ve doğrulanabilir AI/veri provenance altyapısı olarak daha geniş bir kullanım alanında konumlandırmaktadır. Dolayısıyla teknoloji sağlayıcısının günümüzde aktif olması, eski gıda pilotlarının bütün hedeflerinin uzun dönemli ticari başarıya dönüştüğünü tek başına kanıtlamaz. (DEMETER, t.y.; OriginTrail, t.y.; SmartAgriHubs, t.y.)

En önemli karşılaştırmalı bulgu, Garrard ve Fielke'nin (2020) Avustralya karides sektöründe ulaştığı sonuçtur. Görüşmelere dayanan çalışmada araştırmacılar, blokzincir tabanlı provenance'ın sektöre alternatif dijital çözümlerden çok daha büyük bir katkı sağlama ihtimalinin sınırlı olduğunu; önemli kazanımların önemli ölçüde tedarik zincirinin önce dijitalleşmesinden kaynaklanacağını değerlendirmiştir. Bu sonuç,

blokzincirin her durumda en uygun çözüm olmadığı yönündeki eleştirel literatür için güçlü bir dayanak oluşturmaktadır.

## **5. BLOKZİNCİR TABANLI SİSTEMLERİN VERİ GÜVENİLİRLİĞİ AÇISINDAN DEĞERLENDİRİLMESİ**

Blokzincirin veri güvenilirliğine en önemli katkısı, kayıt bütünlüğünü ve taraflar arasında ortak bir denetim izini güçlendirmesidir. Dağıtık kayıt yapısı, işlemlerin kim tarafından ve ne zaman gerçekleştirildiğinin izlenmesini kolaylaştırırken kriptografik kimlikler ve onay politikaları, belirli işlemlerin yetkili taraflarca doğrulanmasını sağlayabilir. Bununla birlikte değiştirilemezlik, hatalı kayıtların düzeltilemeyeceği anlamına gelmemelidir. Gerçek izlenebilirlik sistemlerinde düzeltmelerin geçmişte koruyacak biçimde kaydedilmesi ve denetlenebilir hata düzeltme prosedürlerinin bulunması gerekir. Blokzincir ayrıca güveni tamamen ortadan kaldırmaz; güvenin bir bölümünü merkezi veri tabanı yöneticisinden protokol, kimlik yönetimi ve konsorsiyum yönetişimine aktarır.

Veri güvenilirliğinin temel sınırlılığı, yanlış veya eksik verilerin teknik olarak geçerli biçimde zincire kaydedilebilmesidir. GIGO problemi olarak bilinen bu durum, blokzincirin kayıt bütünlüğünü korumasının fiziksel gerçekliğin doğruluğunu garanti etmediğini göstermektedir. Bu nedenle zorunlu alan kontrolleri, dijital imzalar, sensör kalibrasyonu, laboratuvar doğrulanması ve çoklu kaynaktan veri kontrolü gibi mekanizmalar gereklidir. Örneğin bir sevkiyatın ERP, GPS ve RFID verileriyle doğrulanması tek kaynağa bağımlılığı azaltabilir. Ancak kaynakların bağımsızlığı ve güvenilirliği de ayrıca değerlendirilmelidir. Böylece blokzincir, güvenilir veri üretiminin yerine geçmekten çok, veri doğrulama süreçlerini

destekleyen bir altyapı olarak işlev görür. (Caldarelli, 2022; Leteane & Ayalew, 2024)

Gizlilik ve erişim kontrolü, özellikle ticari açıdan hassas bilgilerin bulunduğu gıda tedarik zincirlerinde önemlidir. Üretim miktarları, fiyatlar, sözleşmeler ve müşteri bilgileri gibi verilerin bütün katılımcılara açılması uygun olmayabilir. İzinli ağlar erişimi sınırlandırabilirken, kişisel ve hassas verilerin zincir dışında saklanması ve yalnızca gerekli bütünlük kanıtlarının zincire kaydedilmesi daha ihtiyatlı bir yaklaşım sunabilir. Bu tasarım, veri minimizasyonu ve performans açısından da avantaj sağlayabilir. (Romão et al., 2026)

Ölçeklenebilirlik ve maliyet, sistemlerin uygulanabilirliğini belirleyen diğer önemli unsurlardır. Yüksek hacimli sensör ve izlenebilirlik verilerinin tamamının blokzincirde tutulması depolama, sorgulama ve işlem maliyetlerini artırabilir. Bu nedenle verilerin bir bölümünün zincir dışında saklanması ve IoT cihazları, ağ geçitleri ile blokzincir düğümleri arasında görev paylaşımı yapılması işlevsel bir seçenek olabilir. Toplam maliyet ise yalnızca işlem ücretlerini değil; sensör altyapısı, kalibrasyon, ERP entegrasyonu, veri temizliği, eğitim ve teknik desteği de kapsar. Özellikle küçük üreticiler açısından mevcut iş süreçlerinin dijitalleştirilmesi ve entegrasyon maliyetleri önemli engeller oluşturabilir. (Wu, Jiang, & Cao, 2022; Garrard & Fielke, 2020; Romão et al., 2026)

Son olarak blokzincir sistemlerinin başarısı teknik özellikler kadar yönetim düzenlemelerine de bağlıdır. Katılımcıların yetkilendirilmesi, akıllı sözleşmelerin güncellenmesi, hatalı kayıtların düzeltilmesi, anahtar güvenliği ve uyuşmazlıkların çözümü gibi konular açık kurallarla belirlenmelidir. Bu nedenle blokzincir, güvenilir veri üretimini veya kurumsal sorumluluğu tek başına sağlayan bir çözüm değildir. Daha doğru bir değerlendirmeye teknoloji, uygun

biçimde doğrulanmış verilerin çok taraflı kaydının bütünlüğünü ve denetlenebilirliğini güçlendirebilir; ancak veri kalitesi yönetimi, fiziksel doğrulama ve kurumsal yönetim mekanizmalarıyla birlikte kullanılmalıdır. (Caldarelli, 2022; Yaga et al., 2018)

## **6. SONUÇ**

Bu çalışmada gerçekleştirilen literatür ve uygulama incelemesi, blokzincirin gıda tedarik zincirlerinde veri güvenilirliği sorununu tek başına çözen bir teknoloji olmadığını, ancak belirli güven problemlerinde değerli bir altyapı bileşeni olabileceğini göstermektedir. Blokzincir, farklı tarafların işlemlerini ortak bir kayıt üzerinde tutarak veri bütünlüğünü, provenance ve denetlenebilirliği güçlendirebilir. Bununla birlikte yanlış girilen veriler, hatalı sensör ölçümleri veya gerçeğe aykırı beyanlar blokzincire kaydedildiğinde sistem bunların doğruluğunu garanti etmez. Bu nedenle veri güvenilirliği; kayıt öncesi doğrulama, veri kalitesi, fiziksel-dijital eşleşme ve kayıt sonrası bütünlüğü kapsayan çok katmanlı bir yapı olarak değerlendirilmelidir. (Yaga et al., 2018; Leteane & Ayalew, 2024)

Gıda tedarik zincirlerinde mimari seçim, tarafların güven ilişkisi, gizlilik, performans ve yönetim gereksinimlerine bağlıdır. İzinli ağlar kurumsal erişim kontrolü açısından uygun olabilirken, kamusal veya hibrit modeller belirli durumlarda açık doğrulama avantajı sağlayabilir. Yüksek hacimli sensör verileri ve belgelerin zincir dışında saklanması, kritik olaylar ile bütünlük kanıtlarının zincire kaydedilmesi daha ölçeklenebilir bir yaklaşım sunabilir. Bu doğrultuda GS1 EPCIS gibi birlikte çalışabilir veri standartları, teknolojinin operasyonel değerini destekleyen önemli bileşenlerdir. (GS1, 2022)

İncelenen uygulamalar, blokzincirin izlenebilirlik süreçlerini geliştirme potansiyelini ortaya koymakla birlikte,

ticari başarı iddiaları ile bağımsız kanıt düzeyinin ayrılmasını gerektirmektedir. Walmart–IBM gibi pilotlarda önemli iz sürme iyileşmeleri raporlanmış olsa da bu kazanımların ne kadarının blokzincirden, ne kadarının genel dijitalleşmeden kaynaklandığını belirlemek güçtür. Eleştirel saha araştırmaları da bazı tedarik zincirlerinde blokzincirin ilave değerinin sınırlı olabileceğini göstermektedir. Bu nedenle teknoloji, özellikle taraflar arasında ortak bir kayıt sahibinin kabul edilmediği, kayıtların değiştirilmesine ilişkin anlamlı risklerin bulunduğu ve çok taraflı denetlenebilirliğin ekonomik değer yarattığı durumlarda daha güçlü bir gerekçeye sahiptir. Diğer durumlarda merkezi veya federatif veri tabanları daha basit ve ekonomik seçenekler olabilir. (Kamath, 2018; Garrard & Fielke, 2020)

Gıda güvenliği açısından blokzincirin temel katkısı, tehlikeli ürünlerin daha hızlı tespit edilmesini ve geri çağırma süreçlerinin iyileştirilmesini destekleyebilecek güvenilir kayıt altyapısıdır. Ancak bu katkı, doğru veri modeli, kayıt disiplini ve tedarik zinciri paydaşlarının katılımına bağlıdır. Gıda arz güvenliği üzerindeki etkisi ise daha dolaylıdır; görünürlük, kayıpların azaltılması ve karar alma süreçlerinin iyileştirilmesi yoluyla arz istikrarına katkıda bulunabilir, ancak üretim kapasitesi, yoksulluk veya iklim kaynaklı sorunları tek başına çözemez. (FDA, 2026; FAO, 2008)

Sonuç olarak gıda izlenebilirliğinde belirleyici olan, blokzincirin yaygınlaştırılmasından çok güvenilir veri ekosisteminin oluşturulmasıdır. Fiziksel kimlik doğrulama, kalibre edilmiş sensörler, laboratuvar ve sertifikasyon verileri, birlikte çalışabilir standartlar, uygun veri yönetimi ve bağımsız denetim mekanizmaları birlikte uygulandığında blokzincir anlamlı değer yaratabilir. Bu unsurlar bulunmadığında ise teknoloji, yanlış verilerin yalnızca bütünlüğü korunmuş kayıtlarını üretme riski taşımaktadır. (Caldarelli, 2022; GS1, 2022; Leteane & Ayalew, 2024)

## TEŞEKKÜR

Bu kitap bölümü, TÜBİTAK 1833 – Sanayi Yenilik Ağ Mekanizması (SAYEM) Yeşil Dönüşüm Programı kapsamında desteklenen “TARIM2030-Gıdada Arz Güvenliği, Sürdürülebilirlik, Yenilik ve Şeffaflık İçin Tarımda Dijitalleşme Programı” bünyesinde yürütülen Ürünleştirme Projesi 1: Blokzincir Tabanlı İzlenebilirlik Platformunun Geliştirilmesi çalışmaları kapsamında hazırlanmıştır.

Migros Ticaret A.Ş.’nin koordinatör kuruluş olarak yer aldığı TARIM2030 Programı, farklı disiplin ve sektörlerden 17 kuruluşu ortak hedefler doğrultusunda bir araya getirerek tarım ve gıda sektöründe dijital ve sürdürülebilir bir dönüşümün desteklenmesini amaçlamaktadır. Program; paydaşlar arasında yönetim, bilgi paylaşımı ve birlikte uygulamaya yönelik koordinasyonu güçlendirirken, gıda arz güvenliğini bütüncül bir tedarik zinciri yönetimi yaklaşımıyla ele almaktadır. Üretimden tüketime kadar uzanan süreçlerde planlama, verimlilik, güvenilirlik, şeffaflık ve izlenebilirliğin geliştirilmesi; su, gübre ve enerji gibi kaynakların etkin kullanılması, çevresel ayak izinin azaltılması ve sürdürülebilir tarımsal üretim sistemlerinin yaygınlaştırılması programın temel hedefleri arasında yer almaktadır.

Program kapsamında geliştirilen dijital çözümlerle, tarımsal değer zincirinde yer alan farklı paydaşların ortak bir dijital ekosistem içerisinde buluşturulması; veriye dayalı karar verme, izlenebilirlik ve kaynak optimizasyonu olanaklarının geliştirilmesi hedeflenmektedir. Özellikle küçük işletmeler de dâhil olmak üzere farklı ölçekteki üreticilerin dijital dönüşüm sürecine katılabilmesi ve akıllı ve hassas tarım uygulamalarının gıda arz güvenliği, gıda kalitesi ve sürdürülebilir kalkınma hedefleri doğrultusunda yaygınlaştırılması önem taşımaktadır.

Bu bölümün hazırlanmasına olanak sağlayan desteklerinden dolayı Türkiye Bilimsel ve Teknolojik Araştırma Kurumuna (TÜBİTAK), programın koordinasyonunu yürüten Migros Ticaret A.Ş.'ye, TARIM2030 Programı kapsamında araştırma, geliştirme ve uygulama faaliyetlerine katkı sunan program ortaklarına, araştırmacılara ve tüm paydaşlara teşekkür ederiz.

## KAYNAKÇA

- Caldarelli, G. (2022). *Formalizing oracle trust models for blockchain-based business applications: An example from the supply chain sector*. arXiv preprint arXiv:2202.13930. Retrieved from <https://arxiv.org/abs/2202.13930>
- Carrefour. (2018). *Carrefour launches Europe's first food blockchain*. Retrieved from <https://www.carrefour.com/en/news/carrefour-launches-europes-first-food-blockchain>
- Casino, F., Kanakaris, V., Dasaklis, T. K., Moschuris, S., Stachtiaris, S., Pagoni, M., & Rachaniotis, N. P. (2021). Blockchain-based food supply chain traceability: A case study in the dairy sector. *International Journal of Production Research*, 59(19), 5758–5770. doi:10.1080/00207543.2020.1789238
- DEMETER. (t.y.). *About DEMETER*. Retrieved from <https://h2020-demeter.eu/about-demeter/>
- Food and Agriculture Organization of the United Nations. (2008). *An introduction to the basic concepts of food security*. EC–FAO Food Security Programme.
- Galvez, J. F., Mejuto, J. C., & Simal-Gandara, J. (2018). Future challenges on the use of blockchain for food traceability analysis. *TrAC Trends in Analytical Chemistry*, 107, 222–232. doi:10.1016/j.trac.2018.08.011
- Garrard, R., & Fielke, S. (2020). Blockchain for trustworthy provenances: A case study in the Australian aquaculture industry. *Technology in Society*, 62, 101298. doi:10.1016/j.techsoc.2020.101298
- Ge, L., Brewster, C., Spek, J., Smeenk, A., & Top, J. (2017). *Blockchain for agriculture and food: Findings from the*

*pilot study.* Wageningen Economic Research.  
doi:10.18174/426747

George, R. V., Harsh, H. O., Ray, P., & Babu, A. K. (2019). Food quality traceability prototype for restaurants using blockchain and food quality data index. *Journal of Cleaner Production*, 240, 118021. doi:10.1016/j.jclepro.2019.118021

GS1. (2022). *EPCIS standard: Release 2.0*. GS1 AISBL. Retrieved from <https://ref.gs1.org/standards/epcis/2.0.0/>

Gualo, F., Rodríguez, M., Verdugo, J., Caballero, I., & Piattini, M. (2021). Data quality certification using ISO/IEC 25012: Industrial experiences. *Journal of Systems and Software*, 176, 110938. doi:10.1016/j.jss.2021.110938

Iftekhar, A., Cui, X., Hassan, M., & Afzal, W. (2020). *Application of blockchain and Internet of Things to ensure tamper-proof data availability for food safety*. arXiv preprint arXiv:2006.01307. Retrieved from <https://arxiv.org/abs/2006.01307>

Kamath, R. (2018). Food traceability on blockchain: Walmart's pork and mango pilots with IBM. *The Journal of the British Blockchain Association*, 1(1). doi:10.31585/jbba-1-1-(10)2018

Kamilaris, A., Fonts, A., & Prenafeta-Boldú, F. X. (2019). The rise of blockchain technology in agriculture and food supply chains. *Trends in Food Science & Technology*, 91, 640–652. doi:10.1016/j.tifs.2019.07.034

Leteane, O., & Ayalew, Y. (2024). Improving the trustworthiness of traceability data in food supply chain using blockchain and trust model. *The Journal of the British Blockchain Association*, 7(1).

- Lucena, P., Binotto, A. P. D., Momo, F. da S., & Kim, H. (2018). *A case study for grain quality assurance tracking based on a blockchain business network*. arXiv preprint arXiv:1803.07877. Retrieved from <https://arxiv.org/abs/1803.07877>
- Menon, S., & Jain, K. (2024). Blockchain technology for transparency in agri-food supply chain: Use cases, limitations, and future directions. *IEEE Transactions on Engineering Management*, 71, 106–120. doi:10.1109/TEM.2021.3110903
- OriginTrail. (t.y.). *OriginTrail*. Retrieved from <https://origintrail.io/>
- Romão, A., Faria, F., Matos, J. R., Nunes, E., Eisa, S., & Pardal, M. L. (2026). *Integration adapter architecture for food traceability blockchain*. arXiv preprint arXiv:2603.26306. Retrieved from <https://arxiv.org/abs/2603.26306>
- SmartAgriHubs. (t.y.). *SmartAgriHubs*. Retrieved from <https://smartagrihubs.eu/>
- Spitalleri, A., Kavasidis, I., Cartelli, V., Mineo, R., Rundo, F., Palazzo, S., Spampinato, C., & Giordano, D. (2023). *BioTrak: A blockchain-based platform for food chain logistics traceability*. arXiv preprint arXiv:2304.09601. Retrieved from <https://arxiv.org/abs/2304.09601>
- TE-FOOD. (t.y.). *Farm-to-table food traceability solution using blockchain technology*. Retrieved from <https://te-food.com/>
- FDA. (2026). *FSMA final rule on requirements for additional traceability records for certain foods*. Retrieved from <https://www.fda.gov/food/food-safety-modernization-act-fsma/fsma-final-rule-requirements-additional-traceability-records-certain-foods>

- van Hilten, M., Ongena, G., & Ravesteijn, P. (2020). Blockchain for organic food traceability: Case studies on drivers and challenges. *Frontiers in Blockchain*, 3, 567175. doi:10.3389/fbloc.2020.567175
- Wang, Y., Li, S., Liu, H., Zhang, H., & Pan, B. (2023). *A reference architecture for blockchain-based traceability systems using domain-driven design and microservices*. arXiv preprint arXiv:2302.06184. Retrieved from <https://arxiv.org/abs/2302.06184>
- World Health Organization. (2026, June 4). *Food safety*. Retrieved from <https://www.who.int/news-room/fact-sheets/detail/food-safety>
- Wu, H., Jiang, S., & Cao, J. (2022). *High-efficiency blockchain-based supply chain traceability*. arXiv preprint arXiv:2210.09202. Retrieved from <https://arxiv.org/abs/2210.09202>
- Yaga, D., Mell, P., Roby, N., & Scarfone, K. (2018). *Blockchain technology overview* (NISTIR 8202). National Institute of Standards and Technology. doi:10.6028/NIST.IR.8202

# **DISCRIMINATIVE CAPABILITY AND TEMPORAL GENERALIZABILITY OF STATIC REVERSE ENGINEERING FEATURES IN WINDOWS X64 MALWARE DETECTION**

**Güney UĞURLU<sup>1</sup>**

## **1. INTRODUCTION**

Malware detection for Windows executables remains a classification problem in which predictive performance must be considered together with generalizability, data leakage, and interpretability. Static machine-learning approaches are attractive because they can evaluate a file without executing it, allowing large collections of Portable Executable (PE) files to be screened at relatively low operational cost. The original EMBER benchmark established a reproducible basis for this line of work by releasing large-scale PE feature vectors and showing that gradient-boosted decision trees could provide a strong static baseline (Anderson & Roth, 2018). Later datasets such as SOREL-20M expanded the scale of malicious PE research, while BODMAS explicitly emphasized timestamped samples and temporal analysis (Harang & Rudd, 2020; Yang, Ciptadi, Laziuk, Ahmadzadeh, & Wang, 2021).

High scores obtained under random train-test splits do not necessarily represent deployment performance. Malware ecosystems evolve, toolchains change, packers and families rise or disappear, and benign software distributions also shift. Pendlebury, Pierazzi, Jordaney, Kinder, and Cavallaro (2019)

---

<sup>1</sup> Lecturer, Ankara University, Nallıhan Vocational School, Electricity Generation, Transmission and Distribution Program, ORCID: 0000-0002-1410-0243.

demonstrated that temporal and spatial experimental biases can materially inflate malware-classification results. A model intended for practical use should therefore be evaluated on samples that occur after the samples used for training. This requirement is particularly relevant for static PE classifiers, whose features include file structure, section characteristics, strings, imported APIs, byte distributions, and signing metadata that may themselves change over time.

EMBER2024 provides a current benchmark designed around this problem. It contains more than 3.2 million files across six file formats and introduces EMBER feature version 3, which expands the available metadata and static feature families. The benchmark is temporally organized: the first 52 weeks form the training period and the final 12 weeks form the test period, enabling evaluation on newer malware rather than on a random mixture of historical and future files (Joyce et al., 2025). This chapter uses two temporally separated Windows x64 weekly subsets from EMBER2024 to examine not only whether malware can be detected accurately, but also which static reverse-engineering feature groups carry the greatest discriminative information and whether the observed performance survives removal of time-related PE metadata.

The study addresses three questions. First, how much discriminative capability is provided independently by PE header/general information, section structure, strings, imports, byte histograms, byte-entropy representations, and Authenticode attributes? Second, how do Logistic Regression, Random Forest, and LightGBM compare when all selected static features are combined? Third, does the strongest model preserve its performance when three potentially time-revealing binary metadata fields are removed? The contribution is therefore an ablation-based and temporally separated evaluation rather than a simple accuracy comparison. The analysis also interprets the most

frequently used LightGBM features from a reverse-engineering perspective while explicitly avoiding causal claims from feature-importance rankings.

## **2. BACKGROUND AND RELATED WORK**

### **2.1. Static Analysis of Windows PE Files**

Windows executables use the PE/COFF structure, which includes an MS-DOS stub, PE signature, COFF header, optional header, section table, and loader-relevant structures such as imports, exports, resources, relocations, and certificates. The PE32+ form is used for 64-bit address spaces. Header fields describe characteristics such as machine type, section count, image base, linker version, code and initialized-data sizes, entry point, subsystem, checksum, and memory layout. Section headers further specify sizes, virtual addresses, and flags such as whether a section is executable (Microsoft, n.d.). These properties are routinely inspected during static reverse engineering because they expose how the binary is organized before any instruction is executed.

Static malware analysis extends beyond header parsing. String statistics can reveal URLs, protocol markers, command-related words, cryptographic terms, or unusually encoded text. Import tables expose external functionality requested from DLLs and can contain APIs associated with memory allocation, process manipulation, networking, cryptography, or persistence, although the same APIs are also common in benign software. Byte histograms and byte-entropy representations capture distributional properties of the raw file, while section-level entropy can reflect compression, encryption, embedded resources, or packed content. These indicators are not individually equivalent to malicious behavior; their value

emerges from combinations and distributions learned across large corpora.

## **2.2. Machine Learning for Static Malware Detection**

Feature-engineered PE classifiers and end-to-end byte models represent two major approaches to static detection. Raff et al. (2018) introduced MalConv to learn directly from whole executable byte sequences, demonstrating that raw-byte learning is feasible but computationally demanding. EMBER instead emphasized reproducible engineered features and showed that a gradient-boosted tree baseline could outperform the reported MalConv baseline on the original benchmark (Anderson & Roth, 2018). The continued use of gradient-boosted trees in large PE benchmarks reflects their suitability for heterogeneous, sparse, and nonlinearly interacting static features.

LightGBM is an efficient gradient-boosting decision-tree implementation designed for large and high-dimensional datasets. Its training strategy includes Gradient-based One-Side Sampling and Exclusive Feature Bundling to improve efficiency without substantially sacrificing predictive quality (Ke et al., 2017). Random Forest provides a useful bagging-based tree baseline (Breiman, 2001), while Logistic Regression provides a linear reference point. Comparing these model classes helps distinguish gains attributable to nonlinear feature interactions from gains obtainable with a simpler linear boundary.

## **2.3. Temporal Generalization and Experimental Bias**

A central weakness of many malware-classification experiments is the assumption that samples are independently and identically distributed across training and test partitions. In real deployment, the classifier is trained on the past and applied to the future. TESSERACT formalized this concern by showing that inappropriate temporal splitting can create unrealistic evaluation conditions and inflated F1 scores (Pendlebury et al., 2019).

BODMAS likewise enabled study of PE malware evolution using timestamped samples collected over multiple months (Yang et al., 2021). EMBER2024 adopts temporal partitioning at dataset scale and therefore provides an appropriate basis for a future-facing evaluation (Joyce et al., 2025).

Temporal separation alone does not eliminate every shortcut. Static files may contain timestamps or signing-time attributes that correlate with collection periods. Such fields can be legitimate static evidence, yet a model might exploit them as proxies for time rather than learning structural characteristics that generalize across periods. This motivates the robustness experiment in the present study: the strongest model is retrained after removing the COFF timestamp and two Authenticode signing-time fields. The comparison provides a direct test of whether the reported performance depends materially on time-revealing PE metadata.

### **3. MATERIAL AND METHOD**

#### **3.1. Dataset and Temporal Subsets**

The experiments use Windows x64 records derived from EMBER2024 (Joyce et al., 2025). The local training JSONL file corresponds to week\_id 51 and contains first-submission dates from September 15 to September 21, 2024. The local test JSONL file corresponds to week\_id 63 and contains first-submission dates from December 7 to December 14, 2024. Thus, the evaluation moves from an earlier weekly population to a population twelve dataset weeks later. The raw local files contained 20,000 rows each, but SHA-256 inspection showed only 10,000 unique hashes in each file. Duplicate rows were removed before feature extraction. No sample had conflicting labels within a file, and the intersection of unique SHA-256 hashes between training and test sets was zero.

After deduplication, each split contained exactly 5,000 benign and 5,000 malicious files. This balanced design makes accuracy directly interpretable while still permitting precision, recall, F1, and ROC-AUC to capture different error characteristics. The malware-family field was not used as a predictor. It nevertheless indicated distributional change: the training and test subsets contained hundreds of distinct family labels, and the most prevalent family names differed between periods. This reinforces the use of a temporal holdout rather than a random reshuffling of the two weeks.

**Table 1. Composition and temporal separation of the experimental subsets**

| Property                    | Training subset | Test subset   |
|-----------------------------|-----------------|---------------|
| EMBER2024 week_id           | 51              | 63            |
| First-submission interval   | 15-21 Sep 2024  | 7-14 Dec 2024 |
| Raw JSONL rows              | 20,000          | 20,000        |
| Unique SHA-256 samples      | 10,000          | 10,000        |
| Benign / malware            | 5,000 / 5,000   | 5,000 / 5,000 |
| Unique family labels        | 382             | 352           |
| Cross-split SHA-256 overlap | 0               | 0             |

### **3.2. Leakage Control and Preprocessing**

Fields that would directly or indirectly disclose the target were excluded before model construction. Hashes (MD5, SHA-1, SHA-256, and TLSH), the binary label, antivirus detection\_ratio, malware family and family\_confidence, submission and analysis timestamps, week\_id, and higher-level annotations such as behavior, capabilities, ATT&CK techniques, Malware Behavior Catalog tags, groups, exploits, and packer tags were not used as predictors. In particular, detection\_ratio was excluded because it summarizes antivirus detections and would create an obvious target-leakage channel. The training vocabulary and all vectorizers were fitted only on the training subset and then applied to the future test subset.

For duplicated hashes, only one record was retained. Numeric histogram and byte-entropy vectors were truncated or

padded to 256 dimensions and transformed with  $\log(1+x)$ . Import information was represented as binary tokens for imported DLLs and APIs; the 3,000 most frequent import tokens were selected from the training subset only. PE header/general, section, string, and Authenticode dictionaries were converted to sparse numeric/categorical representations using a training-fitted dictionary vectorizer. Section dictionaries were summarized with count statistics and aggregate descriptors such as mean, standard deviation, minimum, maximum, and sum for numeric section properties, together with categorical tokens for names and flags.

### 3.3. Static Feature Groups

Seven feature groups were evaluated independently and jointly. The resulting dimensions are shown in Table 2. The section group is relatively high-dimensional because it includes both aggregated numeric descriptors and categorical section tokens. The import group is capped at 3,000 training-derived tokens. The complete baseline representation contains 5,708 features.

**Table 2. Static feature groups used in the experiments**

| Feature group       | Dimension | Representative information                                         |
|---------------------|-----------|--------------------------------------------------------------------|
| PE Header + General | 57        | Image size, entropy, linker/header fields, entry point, image base |
| Sections            | 2,047     | Section counts, sizes, ratios, entropy, names, memory flags        |
| Strings             | 84        | Counts, average length, entropy, URLs and lexical indicators       |
| Imports             | 3,000     | Training-derived DLL and API presence tokens                       |
| Byte Histogram      | 256       | Log-transformed byte-frequency representation                      |
| Byte Entropy        | 256       | Log-transformed byte/entropy distribution                          |
| Authenticode        | 8         | Certificate/signing structure attributes                           |
| All static features | 5,708     | Concatenation of all groups above                                  |

### 3.4. Models and Evaluation Metrics

Three supervised classifiers were trained on the full static representation. Logistic Regression used sparse standardization, the liblinear solver, and a maximum of 1,500 iterations. Random Forest used 300 trees, square-root feature sampling, balanced

class weights, and a fixed random seed. LightGBM used 500 boosting iterations, a learning rate of 0.05, 31 leaves, 0.9 subsampling, and 0.9 column sampling. The feature-group ablation used the same LightGBM family with 400 estimators so that each group could be assessed under a nonlinear model capable of exploiting within-group interactions.

Performance was measured with accuracy, precision, recall, F1 score, and area under the receiver operating characteristic curve (ROC-AUC). Because the test subset is balanced, accuracy summarizes the overall error rate, while precision quantifies the proportion of predicted malware that is truly malicious and recall measures the proportion of malicious files detected. F1 balances precision and recall, and ROC-AUC evaluates ranking quality across classification thresholds. Confusion-matrix counts were also retained for direct inspection of false positives and false negatives.

### **3.5. Temporal-Metadata Robustness Experiment**

The initial full LightGBM model ranked three time-related static fields among its influential features: `authenticode.latest_signing_time`, `authenticode.signing_time_diff`, and `header.coff.timestamp`. These values originate from the binary or its signing metadata rather than from the external collection timestamps that were already excluded. Nevertheless, because the training and test subsets are temporally separated, these fields could potentially act as period proxies. A second LightGBM model was therefore trained with the same configuration after removing all three fields. The resulting representation contained 5,705 features. The robustness test compares the original and reduced models on the identical future test subset.

## 4. RESULTS

### 4.1. Feature-Group Ablation

Table 3 shows that combining heterogeneous static features produced the strongest result: 0.9656 F1 and 0.9940 ROC-AUC. Among individual groups, strings were strongest by F1 (0.9403), followed closely by PE Header + General (0.9359) and Sections (0.9350). Byte Entropy and Imports produced intermediate F1 scores of 0.9193 and 0.9178, respectively, while Byte Histogram reached 0.9141. Authenticode alone behaved differently from the other groups: recall was very high at 0.9800, but precision was only 0.6827, producing an F1 of 0.8048. Thus, certificate/signing structure alone was insufficient for precise discrimination in this temporal test.

**Table 3. Feature-group ablation results on the temporally later test subset**

| Feature group       | Accuracy | Precision | Recall | F1     | ROC-AUC |
|---------------------|----------|-----------|--------|--------|---------|
| All static features | 0.9657   | 0.9679    | 0.9634 | 0.9656 | 0.9940  |
| Strings             | 0.9409   | 0.9500    | 0.9308 | 0.9403 | 0.9843  |
| PE Header + General | 0.9358   | 0.9349    | 0.9368 | 0.9359 | 0.9829  |
| Sections            | 0.9356   | 0.9438    | 0.9264 | 0.9350 | 0.9842  |
| Byte Entropy        | 0.9205   | 0.9331    | 0.9060 | 0.9193 | 0.9760  |
| Imports             | 0.9159   | 0.8977    | 0.9388 | 0.9178 | 0.9764  |
| Byte Histogram      | 0.9152   | 0.9258    | 0.9028 | 0.9141 | 0.9621  |
| Authenticode        | 0.7623   | 0.6827    | 0.9800 | 0.8048 | 0.7660  |

The difference between the complete feature set and the strongest single group is approximately 2.53 F1 percentage points. This is meaningful because the groups capture complementary views of the same executable: structural layout, lexical evidence, imported functionality, byte statistics, and signing information. The results therefore support feature fusion rather than reliance on a single static evidence family.

### 4.2. Model Comparison

LightGBM was the strongest of the three full-feature classifiers, with 0.9657 accuracy, 0.9679 precision, 0.9634 recall, 0.9656 F1, and 0.9940 ROC-AUC. Random Forest ranked second

with 0.9481 F1, while Logistic Regression produced 0.9380 F1. The gap between Logistic Regression and both tree ensembles indicates that nonlinear interactions among the heterogeneous static features contribute materially to future-period classification. The ROC curves in Figure 1 show the same ordering across thresholds.

**Table 4. Model performance using all static features**

| Model               | Accuracy | Precision | Recall | F1     | ROC-AUC |
|---------------------|----------|-----------|--------|--------|---------|
| LightGBM            | 0.9657   | 0.9679    | 0.9634 | 0.9656 | 0.9940  |
| Random Forest       | 0.9489   | 0.9623    | 0.9344 | 0.9481 | 0.9896  |
| Logistic Regression | 0.9380   | 0.9382    | 0.9378 | 0.9380 | 0.9773  |

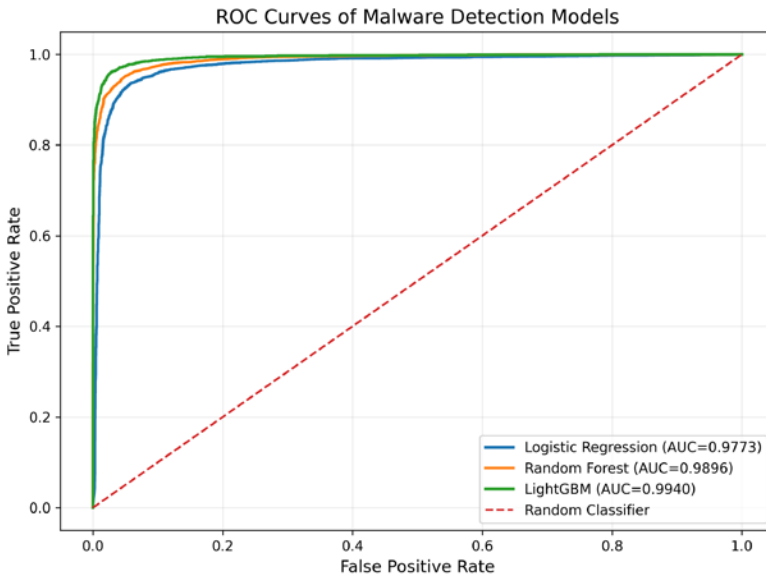


**Figure 1. ROC curves for Logistic Regression, Random Forest, and LightGBM on the future test subset.**

#### **4.3. Error Profile of the Baseline LightGBM Model**

On 10,000 unique test files, the baseline LightGBM model correctly classified 4,840 benign files and 4,817 malicious files. It produced 160 false positives and 183 false negatives, for 343

total errors. The relatively similar false-positive and false-negative counts are consistent with the balanced test set and the close precision-recall values. In operational settings, however, the relative cost of these two error types may differ substantially; the present experiment uses the default probability decision rule and does not optimize a deployment-specific threshold.

#### 4.4. Feature Importance After Temporal-Metadata Removal

The robustness model provides a cleaner view of static indicators because the three explicitly time-related metadata fields were removed. Figure 2 presents the top 20 LightGBM features using the model's default split-count importance. The highest-ranked feature was Authenticode chain\_max\_depth, followed by the count of http:// strings, PE checksum, average string length, string entropy, minor linker version, maximum section entropy, image base, the presence of an executable-section memory flag, and overlay size. Other prominent features included initialized-data size, DOS-message string count, overlay entropy, major image version, the imported Kernel32 VirtualAlloc API, e\_lfanew, stack-reserve size, general entropy, and crypt-related string counts.

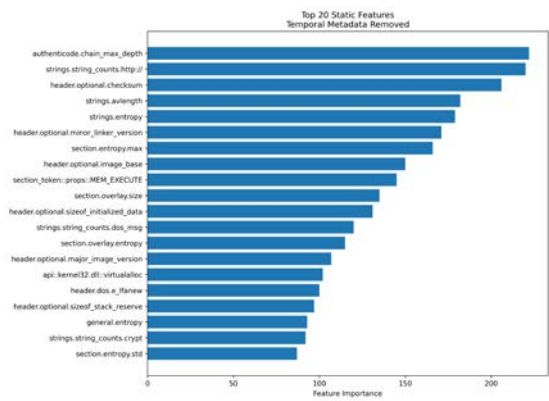


**Figure 2. Top 20 LightGBM split-count feature importances after removal of temporal PE metadata.**

These rankings should be interpreted as model usage rather than causal evidence. For example, VirtualAlloc is common in both benign and malicious programs, high entropy can be produced by legitimate compression, and executable sections are fundamental to normal PE files. Their importance indicates that specific values and interactions helped partition this dataset; it does not imply that the presence of any one feature is sufficient to classify a binary. This distinction is important when translating machine-learning outputs into reverse-engineering hypotheses.

#### **4.5. Robustness to Removal of Time-Related PE Metadata**

Removing `authenticcode.latest_signing_time`, `authenticcode.signing_time_diff`, and `header.coff.timestamp` caused only a small change in performance. Accuracy decreased from 0.9657 to 0.9649, F1 from 0.9656 to 0.9648, and ROC-AUC from 0.9940 to 0.9937. Precision increased marginally, while recall decreased from 0.9634 to 0.9616. The reduced model made 351 errors: 159 false positives and 192 false negatives, only eight more errors than the baseline. The results therefore indicate that the principal performance is not dependent on these three time-revealing static fields.

**Table 5. Baseline and temporal-metadata robustness comparison**

| Experiment                | Accuracy | Precision | Recall  | F1      | ROC-AUC |
|---------------------------|----------|-----------|---------|---------|---------|
| Original LightGBM         | 0.9657   | 0.9679    | 0.9634  | 0.9656  | 0.9940  |
| Temporal metadata removed | 0.9649   | 0.9680    | 0.9616  | 0.9648  | 0.9937  |
| Difference                | -0.0008  | +0.0001   | -0.0018 | -0.0008 | -0.0003 |

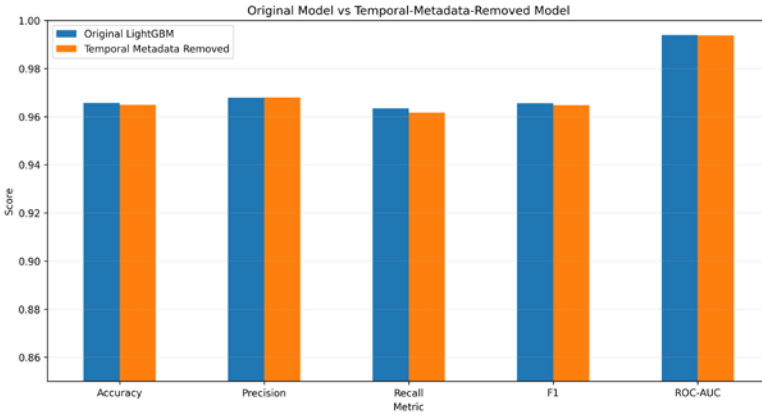


**Figure 3. Comparison of the original LightGBM model and the model trained without the three temporal PE metadata features.**

## 5. DISCUSSION

### 5.1. Discriminative Power of Static Feature Families

The ablation study demonstrates that no single static feature family fully explains the performance of the combined model. Strings provided the strongest individual F1, which suggests that lexical and statistical characteristics remain highly informative even when the test data come from a later period. PE Header + General and Sections were almost equally effective, showing that structural properties of Windows binaries carry substantial discriminative information. This finding is consistent with the rationale behind EMBER-style feature engineering, where multiple low-cost static views are combined rather than relying on a single signature or semantic detector (Anderson & Roth, 2018; Joyce et al., 2025).

Imports deserve particular attention. The import representation contained 3,000 dimensions, far more than the 84 string features or 57 header/general features, yet its standalone F1 was lower. More dimensions therefore did not translate directly

into more discriminative power. Imported APIs describe potential capabilities, but they are context dependent and are affected by static linking, dynamic resolution, compiler frameworks, and library ecosystems. A malicious program can also resolve sensitive functions at runtime, while complex benign applications may import large sets of APIs. Imports become more useful when interpreted jointly with structure, entropy, and strings.

The weak precision of Authenticode as a standalone group also cautions against binary trust assumptions. Signing metadata can be informative, but the presence or structure of a certificate is not synonymous with benignness, and unsigned files are not automatically malicious. In the combined model, signing-related information interacts with other structural evidence. The robustness model's ranking of `chain_max_depth` shows that certificate structure still contributed after signing-time fields were removed, but its role should be viewed as one statistical signal among many.

## **5.2. Reverse-Engineering Interpretation of Dominant Features**

The post-robustness feature ranking is useful because many top variables correspond to artifacts that a reverse engineer would examine during triage. Maximum section entropy and overall entropy can direct attention toward compressed, encrypted, or densely encoded regions. Overlay size and overlay entropy can highlight data appended beyond the mapped PE image. Linker version, checksum, image base, initialized-data size, stack reserve, entry-point information, and `e_lfanew` describe how the executable was built and laid out. Executable-section flags identify regions permitted to contain executable code, while string statistics expose lexical characteristics that may reflect network indicators, embedded configuration, or encoding practices.

The appearance of Kernel32!VirtualAlloc among the top features is also interpretable but must be handled carefully. Dynamic memory allocation is ubiquitous in legitimate software and cannot be treated as a malware signature. In malicious workflows it may participate in unpacking, code staging, or injection, but those semantics cannot be established from an import token alone. The model is therefore best viewed as a prioritization mechanism: it can identify structural combinations that deserve analyst attention, while reverse engineering remains necessary to determine actual control flow and behavior.

### **5.3. Temporal Generalizability**

The strongest methodological result is the preservation of high performance across a twelve-week dataset separation. The training subset covers mid-September 2024, whereas the test subset covers early-to-mid December 2024. The absence of SHA-256 overlap rules out exact-file reuse between splits. This design is more realistic than a random partition because the model cannot train on files from the later test week. The approach follows the broader recommendation that malware evaluation respect chronological deployment order (Pendlebury et al., 2019) and is aligned with the temporal design goals of BODMAS and EMBER2024 (Yang et al., 2021; Joyce et al., 2025).

The metadata-removal experiment strengthens this conclusion. The baseline ranking initially placed a signing-time field first and the COFF timestamp among the top features. If the high test score were substantially driven by these fields, removing them should have produced a marked degradation. Instead, F1 declined by only 0.0008 and ROC-AUC by only 0.0003. This does not prove immunity to every form of temporal leakage, but it shows that the three most obvious time-related PE attributes were not essential to the model's discrimination on this pair of weeks.

## **5.4. Limitations**

Several limitations define the scope of the findings. First, the experiment uses one Windows x64 training week and one later test week rather than the entire EMBER2024 training and test periods. It therefore measures generalization across a specific twelve-week separation, not long-term performance over every future week. Second, the data are balanced by design. Production prevalence can be highly imbalanced, so precision at deployment will depend on the actual base rate and on threshold selection. Third, the work uses pre-extracted static features and does not perform instruction-level disassembly, control-flow recovery, decompilation, or dynamic execution. The phrase static reverse-engineering features refers here to structural and lexical indicators commonly consumed during static binary triage, not to a complete semantic reconstruction of each executable.

Fourth, the LightGBM feature-importance values are split counts, which are model dependent and can favor variables that offer many useful split points. They should not be interpreted as causal explanations or as universal malware indicators. Fifth, family labels and packer tags were intentionally excluded from the predictors, so the analysis does not quantify family-level generalization or explicitly separate packed from unpacked samples. Finally, although hash-based deduplication removed exact duplicates and the temporal-metadata robustness test addressed three important shortcut features, near-duplicate binaries, shared build systems, and broader ecosystem correlations may still affect performance. Static malware classifiers are also vulnerable to deliberate manipulation and subterfuge, an issue documented in prior work (Fleshman, Raff, Zak, McLean, & Nicholas, 2018).

## 6. CONCLUSION

This chapter evaluated the discriminative capability and temporal generalizability of static PE features for Windows x64 malware detection using temporally separated EMBER2024 subsets. After SHA-256 deduplication, the study used 10,000 unique training files and 10,000 unique future test files, each balanced between benign and malicious classes, with no cross-split hash overlap. Seven static feature families were evaluated independently and jointly, and three machine-learning models were compared on the complete representation.

The results show that heterogeneous feature fusion is more effective than any individual static evidence family. Strings were the strongest standalone group with 0.9403 F1, while the complete representation reached 0.9656 F1 and 0.9940 ROC-AUC. LightGBM outperformed Random Forest and Logistic Regression, indicating that nonlinear interactions among structural, lexical, import, byte-distribution, and signing features are important. The post-robustness feature ranking further showed that certificate structure, HTTP-related string counts, PE checksum, string statistics, linker and image-layout fields, section entropy, executable-section flags, overlay properties, and selected API imports contributed to the model's decisions.

Most importantly, removal of three time-related static metadata fields reduced F1 by only 0.0008 and ROC-AUC by only 0.0003. The future-period performance therefore remained essentially intact without the most obvious temporal shortcuts. Within the limits of a two-week temporal study, the findings support the use of combined static reverse-engineering indicators as a fast and generalizable basis for Windows x64 malware triage. Future work should extend the same leakage-controlled pipeline across all EMBER2024 weeks, examine performance decay and family-level drift continuously, and complement split-count

importance with local explanation methods and direct binary-level validation.

## REFERENCES

- Anderson, H. S., & Roth, P. (2018). EMBER: An open dataset for training static PE malware machine learning models. arXiv preprint arXiv:1804.04637.
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5-32. doi:10.1023/A:1010933404324
- Fleshman, W., Raff, E., Zak, R., McLean, M., & Nicholas, C. (2018). Static malware detection & subterfuge: Quantifying the robustness of machine learning and current anti-virus. In 2018 13th International Conference on Malicious and Unwanted Software (MALWARE) (pp. 1-10). IEEE. doi:10.1109/MALWARE.2018.8659360
- Harang, R., & Rudd, E. M. (2020). SOREL-20M: A large scale benchmark dataset for malicious PE detection. arXiv preprint arXiv:2012.07634.
- Joyce, R. J., Miller, G., Roth, P., Zak, R., Zaresky-Williams, E., Anderson, H., ... Holt, J. (2025). EMBER2024 - A benchmark dataset for holistic evaluation of malware classifiers. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2* (pp. 5516-5526). Association for Computing Machinery. doi:10.1145/3711896.3737431
- Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems* 30 (pp. 3146-3154).
- Microsoft. (n.d.). PE format. Microsoft Learn. Retrieved August 12, 2026, from <https://learn.microsoft.com/en-us/windows/win32/debug/pe-format>
- Pendlebury, F., Pierazzi, F., Jordaney, R., Kinder, J., & Cavallaro, L. (2019). TESSERACT: Eliminating experimental bias

in malware classification across space and time. In 28th USENIX Security Symposium (USENIX Security 19) (pp. 729-746). USENIX Association.

Raff, E., Barker, J., Sylvester, J., Brandon, R., Catanzaro, B., & Nicholas, C. K. (2018). Malware detection by eating a whole EXE. In Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence: Artificial Intelligence for Cyber Security (pp. 268-276). AAAI Press.

Yang, L., Ciptadi, A., Laziuk, I., Ahmadzadeh, A., & Wang, G. (2021). BODMAS: An open dataset for learning based temporal analysis of PE malware. In 2021 IEEE Security and Privacy Workshops (SPW) (pp. 78-84). IEEE. doi:10.1109/SPW53761.2021.00020

# **TOWARDS INTELLIGENT CLINICAL DECISION SUPPORT: EXTRACTING KEYPHRASES FROM TURKISH PATIENT TEXTS VIA OPEN-SOURCE PRE-TRAINED LANGUAGE MODELS**

**Ezgi ZORARPACI<sup>1</sup>**

**Zeynep DEMİRTAŞ<sup>2</sup>**

**Pınar Sarısaray BÖLÜK<sup>3</sup>**

## **1. INTRODUCTION**

The use of technology in healthcare increases with the development of telemedicine applications and remote consultation services (Ozyegen, Cevik & Basar, 2024). The widespread adoption of these technologies in healthcare brings new challenges that need innovative solutions. Some of them are addressed through clinical decision support system approaches. In general terms, clinical decision support systems are the systems that assist those working in clinical environments in making decisions. These systems can enable healthcare professionals to work faster, while also reducing the number of errors made in the treatments administered.

---

<sup>1</sup> PhD, Istanbul University, Faculty of Computers & Information Technologies, Department of Computer Engineering, ORCID: 0000-0003-0974-7584.

<sup>2</sup> MSc Student, Istanbul University, Institute of Graduate Studies in Sciences, Department of Computer Engineering.

<sup>3</sup> Assoc. Prof., Istanbul University, Faculty of Computers & Information Technologies, Department of Artificial Intelligence & Data Engineering, ORCID: 0000-0001-8274-8423.

The use of electronic health record (EHR) systems, which are helpful for diagnosis and treatment, is causing doctors to spend too much time on patient documents (Khan & Hajiarbabi, 2024). The rapid increase in digital data within the healthcare domain, particularly patient-related documents, has led to the emergence of several new challenges. In response to these challenges, a variety of computational approaches have been developed, including the extraction of medical diagnoses from patient complaints (Li et al., 2019), the generation of domain-specific medical literature summaries (Hu et al., 2024), the transcription of doctor–patient dialogues via voice-to-text systems, the classification of textual inputs into different fields of electronic health records (EHR) (Khan & Hajiarbabi, 2024), the analysis of patient–doctor communication (Bulut & Diri, 2025), and the highlighting of salient information in medical texts (Ozyegen, Cevik & Basar, 2024).

Language models successfully perform natural language processing (NLP) tasks such as text translation, text generation, text summarization, and similar tasks. For example, keyphrases represent salient expressions that capture the core content of a text. Keyphrases highlighted in the text help us outline the themes and topics of the text (Yoo & Lee, 2023). The application areas of the keyphrase extraction task include generating informative labels that can support downstream processes such as classification, prediction systems, information retrieval, and question answering. To give an example from a real-life scenario: producing answers for a healthcare system chatbot or accelerating the healthcare professional's diagnosis and treatment by focusing on the right points in the patient's complaints and making a diagnosis.

Keyphrase extraction is a critical task—whether employed during the pre-training phase of large-scale models, utilized as an intermediate representation for downstream

systems, or applied directly in standalone applications. Furthermore, in the context of modeling patient language during symptom description, Turkish healthcare data presents a particularly valuable yet underexplored research direction, offering significant opportunities for future investigation. Therefore, our study focuses on the context of the problems of these potential use cases in the field of health. In this study, keyphrases are extracted from natural language texts using pre-trained open-source language models. We have chosen Turkish patient complaint texts that can be considered representative of the healthcare context.

Our main objective in this study is to apply keyphrase extraction to patient complaints in the healthcare domain, using Turkish patient texts as the source. The annotation process is performed using KeyBERT model and the resulting keyphrases are incorporated into the dataset as reference labels. This study aims to compare the performance of three transformer-based models—BERTurk, mT5, and Semantic-MPNet—on the keyphrase extraction task, leveraging the dataset enriched with reference keyphrases generated by KeyBERT.

## **2. RELATED WORK**

There are some studies that provide a comprehensive, hierarchical assessment of the keyphrase finding field in different languages (Xie et al., 2023). In one of the studies focusing on text data, 167 articles were systematically analyzed with a focus on the keyphrase prediction task, considering both automatic keyphrase generation, which produces absent keyphrases not present in the source text, and automatic keyphrase extraction, which identifies words directly appearing in the text.

Türkmen et al. (2023) have emphasized that keyphrase extraction methods are often insufficiently effective for Turkish due to its morphological structure, where various meanings are derived by attaching suffixes to word roots, and the high density of verbs, which complicates the extraction process when words are reduced to their stems.

In the literature, it has been observed that domain-specific NLP studies have been carried out using transformer models. Although there are studies that use statistical-based approaches such as TF-IDF to perform the keyphrase extraction task (Chen, 2024), high accuracy results seen in transformer-based studies have been shown in various studies in the literature (Ayan et al., 2021; Türkmen et al., 2023). There are also studies in Turkish on the application of transformer-based solutions in the medical field of NLP (Ozyegen, Cevik & Basar, 2024), where BERTurk, TurkRadBERT-task v1, and BERT-Clinical models were observed to have high success rates.

Unlike the costly and expert-level manual labeling process, alternative approaches to data labeling have been observed in the literature, including weak supervision, pseudo-labeling, and silver-standard data. One such example is seen in a research article (Chang, Ahn & Park, 2024), which focuses on how to train an information retrieval model using data that only contains documentation and no problem queries; this data bottleneck problem was attempted to be solved by generating pseudo-queries with keyphrase extraction and using weak supervision, which obtained performance close to ground truth data.

In another article using the weak supervision approach (Yoon et al., 2023), for biomedical relation extraction tasks where data is unlabeled, the knowledge base refinement was proposed, where datasets generated with weak supervision are

filtered using a knowledge base and transformer-based models are trained with this data, performing better than standard supervised models by enabling learning more relation patterns.

There are studies that examined how the use of automatically generated labeled silver standard data affects model performance in zero-shot information extraction tasks (Wang, Wang & Zeng, 2024); silver-standard data has also been observed to be used in studies aiming for cross-domain generalization for keyphrase generation (Boudin & Aizawa, 2024), and automatically extracting important information has been studied (Guo et al., 2024). With all these studies, it is demonstrated that using automatically generated labels is an approach studied in NLP.

Based on the literature reviewed above, a notable gap has been identified in the analysis of health-related texts in Turkish, particularly in the context of keyphrase extraction from patient-generated content. While previous studies have explored keyphrase extraction in various domains and languages, limited attention has been given to Turkish healthcare data, especially patient speech-based texts. Accordingly, our study focuses on keyphrase extraction using a Turkish dataset derived from patient complaints, aiming to address this gap and contribute to domain-specific NLP research in Turkish healthcare contexts.

### **3. BACKGROUND**

This section presents the methodological background of the models selected to evaluate the performance of keyphrase extraction from Turkish medical texts. It also provides an overview of KeyBERT model, which is employed in this study to generate reference keyphrases.

### **3.1. BERTurk**

BERTurk is a pre-trained language model specifically developed for Turkish and trained on a large-scale corpus comprising 44,049,766,662 tokens collected from diverse Turkish sources. Thanks to this extensive and heterogeneous training data, BERTurk provides robust contextual representations tailored to the morphological and syntactic characteristics of Turkish language. This specialization is particularly important for keyphrase extraction tasks, as Turkish is an agglutinative language in which suffixation significantly alters word forms and meanings.

### **3.2. mT5**

mT5 model supports over 100 natural languages and therefore has a large vocabulary. T5 model is text-to-text and, while cited as a suitable model for NLP tasks such as translation and text summarization, it can also be used for classification tasks where we expect text output instead of class indexes. mT5 model, like T5, has an encoder-decoder transformer architecture. Unlike encoder-based models like BERTurk, mT5 uses a complete encoder-decoder transformer architecture, enabling it to consider all NLP tasks as text-to-text problems. This design makes mT5 particularly suitable for generative tasks such as keyphrase generation, where the model can produce both present keyphrases (words found in the source text) and absent keyphrases (words not explicitly in the text but semantically relevant).

### **3.3. KeyBERT**

KeyBERT is a language model that is a practical implementation and customized for keyphrase extraction task. It is based on the original BERT architecture and it uses BERT embeddings to perform keyphrase extraction task. For keyphrase extraction, KeyBERT model implements these steps in

sequence: getting document embeddings from BERT, extracting word embeddings, calculation of cosine similarity between words and document vector representation, giving the most similar phrases as a result for unsupervised texts.

### **3.4. Semantic-MPNet**

MPNet is a transformer-based architecture introduced by Microsoft researchers and academicians. Unlike BERT, which relies solely on masked language modeling (MLM), MPNet employs a permuted language modeling strategy to better capture relationships between tokens. In this approach, previously predicted tokens are incorporated along with unmasked tokens to provide additional context, while masking is applied only to tokens that have not yet been predicted. This mechanism allows the model to leverage both the sequential and bidirectional context of tokens, combining the strengths of MLM and the permutation-based approach used in XLNet. MPNet's hybrid pre-training strategy enhances its ability to learn rich contextual representations, which is particularly beneficial for complex tasks such as keyphrase extraction.

## **4. THE PROPOSED METHOD**

A multi-layered evaluation approach is used to compare the performance of the open-source language models for keyphrase extraction.

The proposed method consists of three main components: generating reference keyphrases, open-source language based keyphrase extraction, and hybrid lexical-semantic evaluation. The image illustrating our proposed method is presented in Figure 1. In evaluating different transformer-based language models in terms of keyphrase extraction, BERTurk, mT5, and Semantic-MPNet represent the

contextual language understanding, generative sequence-to-sequence, and semantic similarity-based sentence embedding paradigms, respectively.

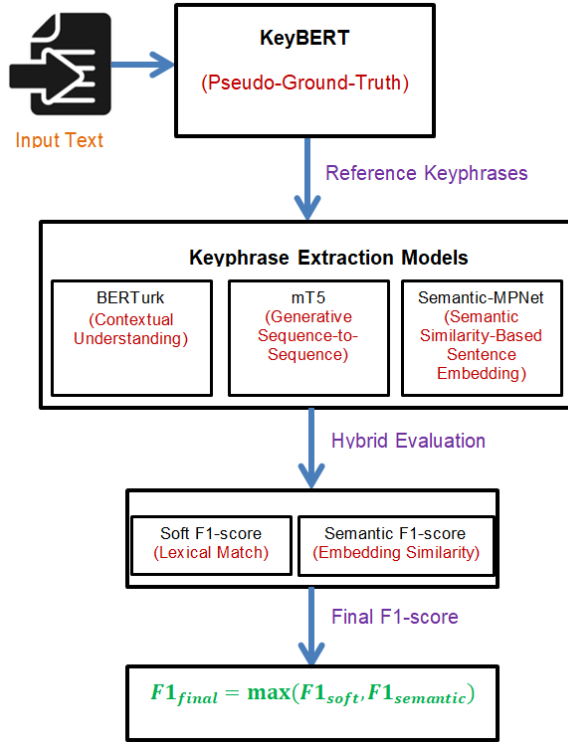


**Figure 1. The proposed performance comparison method for keyphrase extraction models.**

#### **4.1. Generating Reference Keyphrases**

The dataset used in this study does not contain actual keyphrases. Therefore, it is necessary to determine reference keyphrases to compare the performance of the language models. Reference keyphrases are determined using KeyBERT language model.

## **4.2. Open-source Language-based Keyphrase Extraction**

Three language models with different characteristics are selected for keyphrase extraction performance comparison. These methods are monolingual Turkish BERT (BERTurk), multilingual generative model (mT5), and sentence-embedding model (Semantic-MPNet). BERTurk model represents a language-specific contextual encoder-based keyphrase extraction paradigm. mT5 considers keyphrase extraction as a text generation problem. Semantic-MPNet is a sentence embedding-based semantic language model. This model approaches keyphrase extraction as a problem of semantic similarity. It is strong at capturing terminological variations.

In this study, lexical and semantic focused models are evaluated for keyphrase extraction from Turkish patient texts. Thus, this study can determine which of the following is more effective in extracting keyphrases from Turkish patient texts: contextual encoders, generative models, and semantic embedding models.

## **4.3. Hybrid Evaluation**

To eliminate potential bias for the models, a dual-metric evaluation approach is adopted in this study. This metric includes soft lexical and semantic F1-scores. Soft F1-score measures the lexical matching between keyphrases generated by the models and KeyBERT model. On the other hand, semantic F1-score gauges the semantic similarity between the generated keyphrases and reference keyphrases. Since the keyphrase extraction problem involves both selecting the right words and capturing the correct meaning, methods that rely on a single measurement metric may fail.

In lexical matching, the models are evaluated to see if they can generate the correct keyphrases. This method is

generally known as the classic evaluation approach for keyphrase extraction. However, it may fail to capture terminological variations because it penalizes semantically similar expressions.

In semantic matching, the semantic similarity of keyphrases is measured using cosine similarity via embedding vectors. Therefore, semantic matching can capture similarities in meaning, paraphrase words can be evaluated similarly, and it can provide a more flexible assessment approach for terminological variations in patient texts. However, superficially different, very general statements can be evaluated as false positives. Therefore, with the hybrid evaluation method given in Equation (1), the models are rewarded for generating the correct keyphrases, while also considering models that present keyphrases with the same meaning in different ways.

$$F1_{final} = \max(F1_{lexical}, F1_{semantic}) \quad (1)$$

Soft lexical matching is the calculation of partial overlaps between predicted (Pred) and reference (Gold) labels (i.e., keyphrases). A predicted keyphrase  $p$  is checked for a match with a reference keyphrase  $g$  using token-level Jaccard similarity. In Equation (2) where  $Tokens(x)$  represents the token set of the keyphrases and  $\tau$  is matching threshold and determined as 0.5 in this study. Thus, the soft precision, recall and F1-score can be computed given as in Equations (3), (4) and (5) respectively.

$$\frac{Tokens(p) \cap Tokens(g)}{Tokens(p) \cup Tokens(g)} \geq \tau \quad (2)$$

$$Precision = \frac{|Soft\_predictions\_keyphrases|}{|predicted\_keyphrases|} \quad (3)$$

$$Recall = \frac{|Soft\_predictions\_keyphrases|}{|gold\_keyphrases|} \quad (4)$$

$$F1_{soft} = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (5)$$

Soft matching allows for tolerance not only of minor spelling differences between tokens but also of word order changes. Semantic F1-score considers the semantic similarity between the predicted and reference (gold) keyphrases. For this, the cosine similarity between the sentence embedding vectors is calculated. Each keyphrase is represented by a vector space as in Equation (6).

$$v_i = E(keyphrase_i) \quad (6)$$

where  $v_i \in \mathbb{R}^d$  and represents the embedding vector of the keyphrase  $i$ . Firstly, the cosine similarity between each predicted and reference keyphrase pair is computed as in Equation (7).

$$cosine_{sim}(p, g) = \frac{v_p \cdot v_g}{||v_p|| \cdot ||v_g||} \quad (7)$$

If the cosine similarity value between a predicted keyphrase  $p$  and a reference keyphrase  $g$  exceeds a predefined threshold, then they are considered semantically matched. For this, Equation (8) is used.

$$cosine_{sim}(p, g) \geq \delta \quad (8)$$

where  $\delta$  represents semantic threshold value and is determined as 0.7 in our study. Simultaneously, the semantic F1-score is computed according to Equation (9), while semantic precision and recall are calculated based on Equation (10).

$$F1_{semantic} = \frac{2 \times Precision_{sem} \times Recall_{sem}}{Precision_{sem} + Recall_{sem}} \quad (9)$$

$$Precision_{sem} = \frac{TP_{sem}}{|Pred|}, \quad Recall_{sem} = \frac{TP_{sem}}{|Gold|} \quad (10)$$

where  $TP_{sem}$  stands for semantic true positives and is given as in Equation (11).

$$TP_{sem} = |\{p \in Pred: \exists g \in Gold \text{ s.t. } cosine_{sim}(p, g) \geq \delta\}|$$

$$cosine_{sim}(p, g) \geq \delta \} | \quad (11)$$

The semantic F1-score can find synonymous keyphrases and tolerate medical terminology. Consequently, if the model generates the correct keyphrases in the reference keyphrase set, the lexical F1-score will be high. On the other hand, if the model can generate different keyphrases with the same meaning in the reference keyphrase set, the semantic F1-score will be high. Therefore, to compare models in terms of keyphrase extraction performance, the final F1-score can be given as in Equation (12).

$$F1_{final} = \max(F1_{soft}, F1_{semantic}) \quad (12)$$

## 5. DATASET

The performance of open-source pre-trained language models is evaluated using the Patient-Doctor Q&A TR 19,583 dataset (Bulut, 2025), which comprises a total of 19,583 doctor-patient question-answer pairs. This dataset is divided into two distinct subsets. The first subset consists of translations of authentic patient questions and doctor responses from the iCliniq platform from English to Turkish using GPT-3.5-turbo. The second subset contains Turkish translations of the same texts generated by Google Translate (Bulut & Diri, 2025). Accordingly, the effectiveness of open-source language models in keyphrase extraction is assessed separately on each subset. Additionally, the influence of the two translation methods—GPT-3.5-turbo and Google Translate—on model performance in keyphrase extraction is systematically compared.

## **6. RESULTS AND DISCUSSION**

A light stemming process and a surface-form normalization are applied to Turkish patient texts before keyphrase extraction. These consist of pre-processing steps such as lowercasing, punctuation cleaning, whitespace correction, and suffix stripping to reduce superficial variations that may be caused by suffixes in Turkish. A heuristic suffix stripping is used in our study. Nominal and possessive suffixes such as “-leri, -ları, -lerin, -ların, -si, -sı, -nin, -nın, -nun, -nün” are removed. For example, “hastanın → hasta”, “ağrısı → ağrı”, “kemiğin → kemik”, etc. This pre-processing step aims to diminish lexical sparsity and increase matching between semantically identical but morphologically different keyphrases.

To ensure a fair comparison of the models, a two-stage keyphrase extraction framework is used in this study, because the language models used cannot naturally determine which words or phrases will be keyphrases. Therefore, this study ensures the pre-determination of keyphrase candidates for the models. On the other hand, if this candidate generation process is not performed before keyphrase selection, very general words may be extracted as keyphrases, long sentence fragments may be selected, and noise and computational costs may increase significantly. Therefore, expressions that seem plausible as keyphrase candidates are presented to the language models, and the language models are then expected to choose from among them. This two-stage keyphrase extraction framework narrows the search space, significantly reducing noise by eliminating stopwords, numbers, and nonsensical words. It is also faster because there is no need to generate embeddings for each word in the vocabulary. The first stage performs surface-level filtering using n-gram statistics, while the second stage applies deep contextual embeddings for semantic ranking.

Based on our two-stage keyphrase extraction framework, potential keyphrases are produced utilizing the CountVectorizer function configured with an n-gram range (i.e., (1,3)) to enable the extraction of unigrams, bigrams, and trigrams. For our study, the vocabulary size is limited to the most frequent n-grams (i.e., max\_features=40) in the documents.

The performances of the open-source pre-trained language models are compared in terms of keyphrase extraction from Turkish patient texts translated by GPT 3.5-turbo and Google’s neural machine translation system separately. Therefore, the 5-fold cross-validation results for each translated dataset are presented under a separate heading. The best results are indicated in bold. Mean F1-scores are presented in tables for performance comparison while the results for 5-fold are given via violin plots.

### **6.1. The Performance Comparison Using GPT Translated Patient Comments**

In Table 1, the average results of the models for keyphrase extraction from Turkish patient texts translated by GPT 3.5-turbo are given.

**Table 1. Average F1-scores of the Keyphrase Extraction Models  
Across 5-fold Cross-validation on GPT Translated Patient  
Comments**

| <b>Model</b>   | <b>Mean F1-score</b> |
|----------------|----------------------|
| Semantic-MPNet | <b>0.5638</b>        |
| BERTurk        | 0.3433               |
| mT5            | 0.2686               |

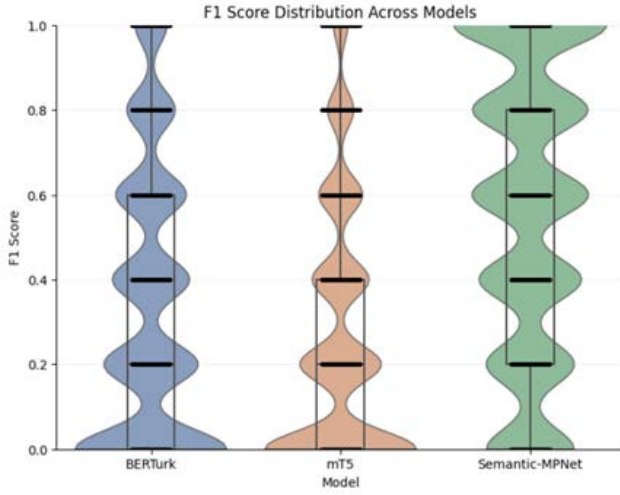


**Figure 2. The violin plots of the F1-score distribution across models for 5-fold cross-validation on GPT translated patient comments.**

From the observed results in Table 1, it can be concluded that Semantic-MPNet has the highest and stable performance for keyphrase extraction compared to BERTurk and mT5 models. On the other hand, mT5, which is a generative model, lags behind other models in terms of keyphrase extraction performance.

Examining the violin plot given in Figure 2, it is observed that most of the F1-scores for Semantic-MPNet are clustered in the high range, and it can be seen that even the lowest performance value of Semantic-MPNet is at an acceptable level in terms of keyphrase extraction. An analysis of BERTurk's violin plot reveals a distribution of F1-scores in both the upper and lower regions. This suggests that the keyphrase extraction performance is sensitive to data splitting for BERTurk. When mT5 violin plot is examined, it is observed that F1-scores are mostly clustered in sub-regions. This suggests that mT5 language model shows weaker and more inconsistent

performance in keyphrase extraction from Turkish patient texts compared to BERTurk and Semantic-MPNet language models.

## **6.2. The Performance Comparison Using Google Translated Patient Comments**

This section compares the performance of language models for keyphrase extraction using patient texts translated with the Google neural machine translation system. For this purpose, Table 2 presents the average F1-scores of the language models after 5-fold cross-validation.

**Table 2. Average F1-scores of the Keyphrase Extraction Models Across 5-fold Cross-validation on Google Translated Patient Comments**

| <b>Model</b>   | <b>Mean F1-score</b> |
|----------------|----------------------|
| Semantic-MPNet | <b>0.5587</b>        |
| BERTurk        | 0.3398               |
| mT5            | 0.2655               |

From the observed results in Table 2, it is apparent that Semantic-MPNet has the highest mean performance (i.e.,  $\approx 56\%$ ) for keyphrase extraction from Turkish patient texts. On the other hand, BERTurk has the second-highest average F1-score performance value. mT5, however, shows lower performance than Semantic-MPNet and BERTurk in terms of average F1-score.

The average F1-scores for the models given in Table 1 and Table 2 are quite close to each other. For example, in Table 1, the keyphrase extraction performance of Semantic-MPNet from patient texts translated with GPT 3.5-turbo is 56.38% successful, while its keyphrase extraction performance from patient texts translated with Google Translate is 55.87% successful. The difference between these two values is negligible. This situation also holds for BERTurk and mT5 language models shown in Tables 1 and 2.

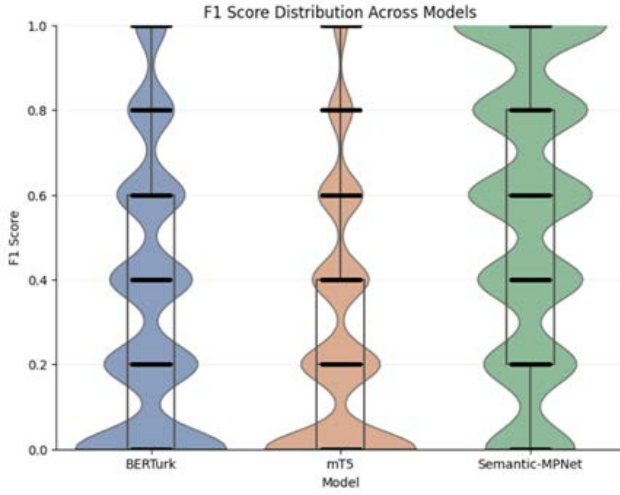


**Figure 3. The violin plots of the F1-score distribution across models for 5-fold cross-validation on Google translated patient comments.**

Furthermore, when examining the violin plot performances given for the language models in Figure 2, it can be easily seen that the F1-score spreads given in Figure 3 are the same as those presented in Figure 2.

When the violin plots in Figures 2-3 and results in Tables 1-2 presented above are evaluated as a whole, the following conclusions can be drawn:

- Translating patient texts using GPT 3.5-turbo or Google Neural Machine Translation did not have any effect on the performance of language models in the keyphrase extraction task.
- The insensitivity of the proposed keyphrase extraction methodology to the translation source of the patient texts clearly demonstrates that the proposed method considers not only lexical word matches but also semantic representation.

- Our proposed keyphrase extraction pipeline consists of candidate generation → embedding similarity → ranking. This method works better with encoder-based models that have high contextual and semantic representational ability.
- BERTurk aims to enhance token-level contextual representations. In contrast, Semantic-MPNet is optimized for sentence-level semantic similarity tasks. Consequently, Semantic-MPNet can produce more discriminative sentence-embeddings. Therefore, it performs better than BERTurk in the similarity-based ranking stage of the proposed keyphrase extraction pipeline.

## 7. CONCLUSION

This study proposes a two-stage keyphrase extraction framework designed to enhance clinical decision support systems by effectively extracting key information from Turkish patient texts. The proposed method is evaluated in terms of performance using open-source pre-trained language models, including BERTurk, mT5, and Semantic-MPNet, on Patient-Doctor Q&A TR 19,583 dataset. The language models have different NLP paradigms; BERTurk is a contextual encoder-based language model trained for token-level representations, Semantic-MPNet is a sentence-embedding language model specialized for semantic similarity, and mT5 is a sequence-to-sequence generative model in which keyphrase extraction can be considered a text generation task. Therefore, comparing these different architectures provides us to examine how model paradigm changes keyphrase extraction performance from Turkish clinical data, offering a paradigm-level analysis of language models in terms of keyphrase extraction from Turkish

clinical texts beyond simply comparing models in a classical way.

Our experimental results have shown that Semantic-MPNet exhibits strong model performance for extraction-based tasks. In conclusion, this study demonstrates that embedding and semantic similarity-based language models can be used as an effective method for keyphrase extraction in morphologically rich and unstructured Turkish clinical raw texts.

## REFERENCES

- Ayan, E. T., Arslan, R., Zengin, M. S., Duru, H. A., Salman, S., & Bardak, B. (2021). Turkish keyphrase extraction from web pages with BERT. In *Proceedings of the 29th Signal Processing and Communications Applications Conference (SIU)* (pp. 1–4).
- Boudin, F., & Aizawa, A. (2024). Unsupervised domain adaptation for keyphrase generation using citation contexts. In *Findings of the Association for Computational Linguistics: EMNLP 2024* (pp. 598–614).
- Bulut, M. K. (2025). *Patient Doctor Q&A TR 19583* [Data set]. Kaggle.  
<https://www.kaggle.com/datasets/kaayra2000/patient-doctor-qa-dataset-tr>
- Bulut, M. K., & Diri, B. (2025). Türkçe sağlık danışmanlığında büyük dil modellerinin hasta-doktor iletişimde kullanım potansiyeli. *KSU Journal of Engineering Sciences*, 28(2), 802–822.
- Chen, L. C. (2024). An extended TF-IDF method for improving keyword extraction in traditional corpus-based research: An example of a climate change corpus. *Data & Knowledge Engineering*, 153, 102322.
- Chang, S., Ahn, G. J., & Park, S. (2024). Improving performance of neural IR models by using a keyword-extraction-based weak-supervision method. *IEEE Access*, 12, 46851–46863.
- Guo, P., Song, Y., Wang, B., Liu, J., & Zhang, Q. (2024). PLBR: A semi-supervised document key information extraction via pseudo-labeling bias rectification. *IEEE Transactions on Knowledge and Data Engineering*, 36(12), 9025–9036.

- Hu, J., Cang, Y., Liu, G., Wang, M., He, W., & Bao, R. (2024). Deep learning for medical text processing: BERT model fine-tuning and comparative study. In *Proceedings of the 3rd International Symposium on Sensor Technology and Control (ISSTC)* (pp. 302–306).
- Khan, F. A., & Hajiabadi, M. (2024). BERT-driven automation in electronic health record management system. In *Proceedings of IEEE SoutheastCon 2024* (pp. 398–404). Atlanta, GA.
- Li, G., Song, H., Liang, H. N., Qu, Y., Liu, L., & Bai, X. (2019). Medical diagnosis by complaints of patients and machine learning. In *Proceedings of the 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)* (pp. 1–5).
- Ozyegen, L., Cevik, M., & Basar, A. (2024). Transformer-based text highlighting for medical terms. In *Proceedings of the 34th International Conference on Collaborative Advances in Software and Computing (CASCON)* (pp. 1–10).
- Türkmen, H., Dikenelli, O., Eraslan, C., Calli, M., & Ozbek, S. (2023). Harnessing the power of BERT in the Turkish clinical domain: Pretraining approaches for limited data scenarios. In *Proceedings of the 5th Clinical Natural Language Processing Workshop (ClinicalNLP)* (pp. 161–170).
- Xie, B., Song, J., Shao, L., Wu, S., Wei, X., Yang, B., et al. (2023). From statistical methods to deep learning, automatic keyphrase prediction: A survey. *Information Processing & Management*, 60(4), 103382.
- Wang, J., Wang, T., & Zeng, Z. (2024). On the use of silver standard data for zero-shot classification tasks in

information extraction. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING)* (pp. 12423–12434).

Yoo, C., & Lee, H. (2023). Improving abstractive dialogue summarization using keyword extraction. *Applied Sciences*, 13(17), 9771.

Yoon, W., Yi, S., Jackson, R., Kim, H., Kim, S., & Kang, J. (2023). Biomedical relation extraction with knowledge base-refined weak supervision. *Database*, 2023, baad054.

# **GÜÇ SİSTEMLERİNİN EKONOMİK YÜK DAĞITIMININ BİR İNCELENMESİ<sup>1</sup>**

**Enzel AYDIN<sup>2</sup>**

**Abdullah Erhan AKKAYA<sup>3</sup>**

## **1. GİRİŞ**

Elektrik üretiminde işletme maliyetinin azaltılması, mevcut üretim birimlerinin talebi hangi güç düzeylerinde karşılayacağını belirlenmesini gerektirir. Aynı toplam güç, santraller arasında farklı biçimlerde paylaştırılabilir; ancak bu dağılımların toplam maliyetleri eşit değildir. Ekonomik yük dağıtımını (EYD), üretim birimlerinin maliyet özelliklerini ve işletme koşullarını birlikte değerlendirerek talebi en düşük toplam maliyetle karşılayan üretim bileşimini arar. Bu nedenle problem, yalnızca üretim kapasitesinin yeterliliğiyle değil, mevcut kapasitenin ekonomik biçimde kullanılmasıyla da ilgilidir (Saadat, 1999).

Türkiye'deki bölgesel güç sistemleri, farklı yakıt maliyetlerine ve üretim kapasitelerine sahip santralleri birlikte içerir. Bu sistemlerde ekonomik dağıtımın incelenmesi, santrallerin ayrı ayrı değerlendirilmesinin ötesinde, toplam talebi karşılayan ortak bir işletme düzeninin kurulmasına dayanır.

---

<sup>1</sup> Bu bölüm, Enzel Aydın'ın Dr. Öğr. Üyesi A. Erhan Akkaya danışmanlığında İnönü Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalında 2025 yılında hazırladığı "Türkiye'deki Bölgesel Güç Sistemlerinin Ekonomik Yük Dağıtımının Farklı Algoritmalarla Optimizasyonu" başlıklı yüksek lisans tezinden üretilmiştir.

<sup>2</sup> İnönü Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, ORCID: 0009-0003-0919-6847.

<sup>3</sup> Dr. Öğretim Üyesi, İnönü Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, ORCID: 0000-0001-6193-5166.

Başaran'ın (2004) Türkiye'deki 380 kV enterkonnekte güç sistemine ilişkin çalışması, bölgesel sistemler üzerinde güç akışı ve ekonomik dağıtım analizlerinin uygulanmasına yönelik bir temel sunmaktadır.

EYD problemlerinin çözümünde matematiksel programlama ve metasezgisel arama yöntemlerinden yararlanılır. Matematiksel programlama yöntemleri, amaç fonksiyonu ve kısıtların yapısını kullanarak çözümü iyileştirir. Metasezgisel yöntemler ise çözüm uzayında farklı arama mekanizmalarıyla aday üretim dağılımları oluşturur. Bir yöntemin başarısı, ulaşılan maliyet kadar çözüm süresi ve tekrarlanan çalıştırmalardaki değişkenlikle de değerlendirilmelidir. Özellikle dağıtımın sık güncellenmesinin amaçlandığı uygulamalarda, hesaplamaların zamanında tamamlanabilmesi ayrı bir değerlendirme ölçütüdür.

Bu bölümde, sekiz generatörlü ve 22 baralı sistem ile altı generatörlü ve 14 baralı sistem için kayıpsız EYD sonuçları incelenmektedir. Sıralı En Küçük Kareler Programlama (SLSQP), Genetik Algoritma (GA), Yapay Arı Kolonisi (YAK), Tabu Arama (TA) ve Tavlama Benzetimi (TB) yöntemleri, uygulanmış oldukları sistemlerde karşılaştırılmaktadır. Sekiz generatörlü sistemde GA, YAK ve SLSQP; altı generatörlü sistemde GA, TB, TA ve SLSQP sonuçları bulunmaktadır. İncelenen örneklerde SLSQP, diğer yöntemlerle yakın maliyetlere daha kısa raporlanan çözüm süreleriyle ulaşmıştır. Bu bulgu, kullanılan modeller ve deney koşulları çerçevesinde ele alınmaktadır.

## 2. EKONOMİK YÜK DAĞITIMININ KAVRAMSAL VE MATEMATİKSEL TEMELLERİ

### 2.1. Üretim Maliyeti ve Güç Dengesi

EYD’de karar değişkenleri, generatörlerin aktif güç üretimleridir. Bir üretim biriminin maliyeti, aktif güç çıkışına bağlı ikinci dereceden bir fonksiyonla yaklaşık olarak modellenebilir. Bu gösterimde sabit terim ile doğrusal ve ikinci dereceden katsayılar, ilgili üretim biriminin maliyet karakteristiğini tanımlar. Her santral için farklı katsayıların kullanılması, aynı miktarda güç artışının bütün santrallerde aynı maliyet artışına yol açmamasını sağlar (Saadat, 1999).

$$C_i(P_i) = a_i + b_i P_i + c_i P_i^2 \quad (1)$$

Burada  $C_i$ ,  $i$ . generatörün saatlik maliyetini;  $P_i$ , MW cinsinden aktif güç üretimini;  $a_i$ ,  $b_i$  ve  $c_i$  ise maliyet katsayılarını göstermektedir. Sistemin toplam maliyeti, devredeki  $n$  generatörün maliyetlerinin toplamıdır. Optimizasyonun amacı bu toplamı en aza indirmektir.

$$\min C_T = \sum_{i=1}^n C_i(P_i) \quad (2)$$

İletim kayıplarının ihmal edildiği modelde toplam üretim, talep edilen aktif güce eşit olmalıdır. Üretim sınırları modele dâhil edildiğinde her generatörün çıkışı alt ve üst işletme limitleri arasında kalır. Temel kısıtlar aşağıdaki gibidir:

$$\sum_{i=1}^n P_i = P_D \quad (3)$$

$$P_{i,\min} \leq P_i \leq P_{i,\max}, \quad i = 1, \dots, n \quad (4)$$

$P_D$  toplam güç talebini,  $P_{i,\min}$  ve  $P_{i,\max}$  ise ilgili birimin alt ve üst üretim sınırlarını belirtir. Maliyet azaltımı, talep karşılanmadan elde edilen bir sonuç olarak yorumlanamaz. Bu nedenle maliyet değerinin yanında güç dengesi ve uygulanabilirlik koşullarının da değerlendirilmesi gerekir. Aynı

şekilde, bir santralin düşük maliyetli olması bütün talebin o santrale aktarılabilceği anlamına gelmez; üretim kapasitesi bu aktarımı sınırlar.

## **2.2. Artan Maliyet ve Kayıpların Etkisi**

Üretim maliyetinin aktif güce göre türevi, üretimdeki küçük bir artışın maliyet üzerindeki etkisini ifade eden artan maliyettir. İkinci dereceden maliyet modeli için bu değer, doğrusal bir fonksiyonla gösterilir:

$$\frac{dC_i}{dP_i} = b_i + 2c_iP_i \quad (5)$$

Kayıpların olmadığı ve üretim sınırlarının etkin olmadığı bir iç çözümde, ekonomik dağıtım koşulu generatörlerin artan maliyetlerinin eşitlenmesidir. Bir birimin artan maliyeti diğerinden yüksekse, uygun sınırlar içinde üretimin daha düşük artan maliyetli birime kaydırılması toplam maliyeti azaltabilir. Üretim sınırına ulaşmış birimler için değerlendirme, sınır kısıtları dikkate alınarak yapılır (Saadat, 1999).

İletim kayıpları hesaba katıldığında ise toplam üretimin, talebin yanında kayıpları da karşılaması gerekir. Bu durumda güç dengesi, toplam üretimin talep ile kayıp gücün toplamına eşit olması biçiminde değişir. Kayıpların üretim dağılımına bağlı olması, maliyet minimizasyonu ile şebeke yapısı arasındaki ilişkiyi güçlendirir. Dolayısıyla kayıpsız bir modelde elde edilen dağılım, kayıpların bulunduğu modelin çözümüyle aynı olmak zorunda değildir.

Bu bölümdeki uygulama sonuçları kayıpsız durumlara aittir. Generatör limitleri ve maliyet katsayıları veri setinde sunulmakla birlikte, gerçek şebekenin bütün işletme ve güvenlik koşullarını içeren bir optimal güç akışı analizi yapılmamaktadır. Bara sayıları sistemlerin tanımlanmasını sağlar; kayıpsız EYD'nin karar değişkenleri ise generatörlerin aktif güçleridir. Bu

ayrım, sonuçların kapsamını ve farklı sistemlere aktarılabilirliğini değerlendirirken korunmalıdır.

### **2.3. İlgili Çalışmaların Ortak Çerçevesi**

EYD literatüründeki yöntemler, amaç fonksiyonunun ve işletme kısıtlarının değişmesine bağlı olarak çeşitlenmektedir. Sinha, Chakrabarti ve Chattopadhyay (2003), ekonomik yük dağıtımında evrimsel programlama tekniklerini ele almıştır. Subathra, Easter Selvan, Aruldoss Albert Victoire, Hepzibah Christinal ve Amato (2015) ise çapraz entropi yöntemi ile sıralı kuadratik programlamayı birleştiren hibrit bir çözüm yaklaşımı sunmuştur. Bu çalışmalar, çözüm uzayını araştıran yöntemler ile matematiksel yapıyı kullanan yöntemlerin birlikte değerlendirilebildiğini göstermektedir.

Türkiye'deki güç sistemlerine ilişkin karşılaştırmalar bakımından Arslan'ın (2015) ekonomik ve çevresel ekonomik yük dağıtımı çalışması ile Duman, Döşoğlu, Öztürk ve Erdoğan'ın (2010) GA, TB ve TA uygulaması, burada incelenen sistemler için karşılaştırma bağlamı oluşturmaktadır. Literatürden derlenen karşılaştırma değerleri ile bu çalışma kapsamında Python ortamında elde ettiğimiz simülasyon sonuçlarının ayrı olarak sunulması, yöntemlerin bağımsız platformlardaki başarısını nesnel biçimde değerlendirmek açısından önemlidir.

Literatürde bir yöntemin iyi sonuç vermesi, bütün EYD biçimleri için aynı performans sıralamasının oluşacağı anlamına gelmez. Maliyet eğrisinin biçimi, iletim kayıpları, üretim sınırları ve algoritma ayarları değiştikçe çözümün niteliği de değişebilir. Bu bölümde karşılaştırmamızın odağı, iki belirli kayıpsız sistemde maliyet, süre ve tekrarlar arasındaki değişkenliğin birlikte değerlendirilmesidir.

### **3. ÇÖZÜMDE KULLANILAN OPTİMİZASYON YÖNTEMLERİ**

#### **3.1. Sıralı En Küçük Kareler Programlama**

SLSQP, kısıtlı optimizasyon problemlerinin çözümünde kullanılan ve sıralı kuadratik programlama yaklaşımıyla ilişkili bir yöntemdir. Temel fikir, mevcut çözüm çevresinde amaç fonksiyonu ile kısıtların yerel bir modelini kurmak ve bu modelden hareketle yeni bir arama yönü belirlemektir. Çözüm, ardışık alt problemler ve uygun adım uzunlukları üzerinden güncellenir. Böylece amaç fonksiyonunun azaltılması ile kısıtların sağlanması birlikte gözetilir (Nocedal ve Wright, 2006).

EYD’de her aday çözüm, generatör güçlerinden oluşan bir vektördür. Toplam maliyet amaç fonksiyonunu, talep dengesi eşitlik kısıtını ve üretim limitleri değişken sınırlarını oluşturur. Düzgün ve türevlenebilir maliyet fonksiyonları, yerel türev bilgisinden yararlanan bu yaklaşım için elverişli bir yapı sağlar. Bununla birlikte, yöntemin adı tek başına bütün problem sınıflarında küresel optimuma ulaşılacağını garanti etmez; sonuçlar problem yapısı ve durdurma koşullarıyla birlikte değerlendirilmelidir.

Bu çalışmada SLSQP uygulaması, Python ortamında SciPy kütüphanesinin minimize fonksiyonu aracılığıyla gerçekleştirilmiştir. Başlangıç üretim vektörünün oluşturulması, toplam maliyetin tanımlanması, talep eşitliğinin belirtilmesi ve çözümün raporlanması uygulamanın temel aşamalarıdır. Algoritmaya ilişkin Python kodlarının adresi tezde ayrıca verilmiştir (Aydın, 2025). Burada sunulan süreler, bu uygulama ortamında raporlanan hesaplama süreleridir.

#### **3.2. Genetik Algoritma**

GA, aday çözümlerden oluşan bir popülasyonu seçim, çaprazlama ve mutasyon işlemleriyle geliştiren evrimsel bir

yöntemdir. Her adayın uygunluğu problemdeki amaç fonksiyonuna göre değerlendirilir. Daha iyi adayların sonraki nesillere katkısı artırılırken, çaprazlama farklı çözümlerin özelliklerini birleştirir; mutasyon ise aramaya çeşitlilik kazandırır (Alhijawi ve Awajan, 2024).

EYD açısından bir birey, bütün generatörlerin üretim değerlerini içeren bir dağılıma karşılık gelir. Üretilen adayın toplam maliyetinin düşük olması kadar talep ve üretim koşullarına uygunluğu da önemlidir. Kodlama ve kısıt ele alma tercihi, algoritmanın uygulanışını etkiler. Bu nedenle GA'nın genel çalışma ilkesi ile belirli bir uygulamanın parametreleri birbirinden ayrılmalıdır.

GA'nın popülasyonla çalışması, çözüm uzayında birden fazla adayın eş zamanlı olarak değerlendirilmesini sağlar. Buna karşılık çok sayıda uygunluk hesabı yapılması süreyi artırabilir. Bu çalışma kapsamında GA her iki sisteme uygulanmış; sonuçları maliyetin yanı sıra tekrar deneylerindeki değişkenlik bakımından da değerlendirilmiştir.

### **3.3. Yapay Arı Kolonisi**

YAK, arıların besin arama davranışından esinlenen bir optimizasyon yöntemidir. Besin kaynakları aday çözümleri; kaynakların kalitesi ise uygunluk değerini temsil eder. İşçi arılar mevcut kaynakların çevresinde arama yapar, gözcü arılar uygunluk bilgisine göre kaynak seçer, kâşif arılar ise yeterince iyileşmeyen kaynakların yerine yeni adaylar üretir (Karaboga, 2005; Karaboga ve Basturk, 2008).

EYD uygulamasında bir besin kaynağı, tek bir santral yerine sistemin bütün üretim dağılımını temsil edecek biçimde tanımlanabilir. Komşuluk araması bu dağılımı değiştirerek daha düşük maliyetli adaylar araştırır. Mevcut iyi çözümlerin çevresini inceleme ile yeni bölgeleri keşfetme arasındaki denge, algoritmanın arama davranışını belirler. Gerçek değerli

optimizasyon için geliştirilen YAK düzenlemeleri de bu arama mekanizmalarını ele almaktadır (Akay ve Karaboga, 2012).

Bu çalışmada YAK sonuçları sekiz generatörlü, 22 baralı sistem için sunulmaktadır. Yöntemin performansı, aynı sistemde GA ve SLSQP ile karşılaştırılmıştır. Altı generatörlü sistem için YAK sonucu bulunmadığından, bu sistemdeki yöntem sıralamasına YAK dâhil edilmemektedir.

### **3.4. Tabu Arama**

TA, arama geçmişini kullanarak belirli hareketlerin kısa bir süre için yeniden yapılmasını engelleyen bellek temelli bir yöntemdir. Mevcut çözümün komşuluğundaki adaylar değerlendirilir ve seçilen hareketin özellikleri tabu listesinde tutulur. Böylece aramanın aynı çözümler arasında tekrar etmesi sınırlandırılır. Aspirasyon ölçütü, yeterince iyi bir çözüm sağlayan tabu hareketine istisna tanınabilmesini sağlar (Glover, 1986).

EYD probleminde komşu çözüm, mevcut üretim dağılımında yapılan değişikliklerle oluşturulur. Hareketlerin nasıl tanımlandığı, tabu süresi ve komşuluk büyüklüğü, hem hesaplama yükünü hem de çözüm kalitesini etkiler. Bu nedenle TA'nın bir uygulamadaki maliyet değişkenliği, bütün tabu arama yaklaşımlarına ait değişmez bir özellik olarak yorumlanmamalıdır. Burada TA, altı generatörlü sistemde GA, TB ve SLSQP ile karşılaştırılmaktadır.

### **3.5. Tavlama Benzetimi**

TB, kontrollü soğutma sürecinden esinlenen bir arama yöntemidir. Mevcut çözümden üretilen komşu çözüm daha iyi olduğunda kabul edilir. Daha yüksek maliyetli bir aday ise sıcaklık parametresine bağlı bir olasılıkla kabul edilebilir. Bu mekanizma, aramanın başlangıçta daha geniş bir bölgeyi araştırmasına ve yalnızca iyileştirici hareketlerle sınırlı

kalmamasına yardımcı olur (Kirkpatrick, Gelatt ve Vecchi, 1983).

Sıcaklık düştükçe maliyeti artıran hareketlerin kabul edilme olasılığı azalır. Başlangıç sıcaklığı, soğutma programı, komşuluk üretimi ve durdurma koşulları yöntemin davranışını belirler. EYD uygulamasında değerlendirilen büyüklük, üretim dağılımının toplam maliyetidir. Çalışmada TB, altı generatörlü sistem üzerinde uygulanmıştır; tekrar sonuçları, bazı çalıştırmalarda yakın maliyetler elde edilse de daha yüksek maliyetli sonuçların oluşabildiğini göstermektedir.

## **4. UYGULAMA SİSTEMLERİ VE DEĞERLENDİRME DÜZENİ**

### **4.1. Bölgesel Güç Sistemleri**

Birinci sistem, Türkiye'deki 380 kV seviyesindeki sekiz generatör ve 22 baradan oluşan bölgesel sistemdir. Üretim birimleri Hamitabat, Ambarlı, Bursa Doğalgaz, Seyitömer, Soma B, Yeniköy, Kemerköy ve Yatağan santralleridir. Toplam talep 4000 MW olarak alınmıştır. Sistem, sekiz üretim ve 14 yük barasıyla tanımlanmaktadır (Başaran, 2004).

İkinci sistem, altı generatörlü ve 14 baralı bölgesel sistemdir. Bursa Doğalgaz, Seyitömer, Soma B, Yeniköy, Kemerköy ve Yatağan üretim birimlerini içerir; toplam talep 2734,9 MW'tır. Bu sistem, altı üretim ve sekiz yük barasından oluşmaktadır. Çalışmadaki sistem tanımı ve literatür karşılaştırması Duman ve arkadaşlarının (2010) çalışmasına dayandırılmaktadır. Her iki sistemin çözümünde iletim kayıpları ihmal edilmiştir.

Generatörlerin maliyet katsayıları ile alt ve üst güç limitleri Tablo 1'de gösterilmiştir. Altı generatörlü sistemde tablonun ilk altı santrali kullanılmaktadır. Katsayılar, maliyet

fonksiyonundaki sabit, doğrusal ve ikinci dereceden terimlerin sırasını izlemektedir.

**Tablo 1. Generatörlerin üretim sınırları ve maliyet katsayıları**

| Santral   | Alt sınır<br>(MW) | Üst sınır<br>(MW) | a      | b      | c      |
|-----------|-------------------|-------------------|--------|--------|--------|
| Bursa DG  | 318               | 1432              | 6780,5 | 5,682  | 0,0106 |
| Seyitömer | 150               | 600               | 1564,4 | 3,1288 | 0,0139 |
| Soma B    | 210               | 990               | 5134,1 | 6,2232 | 0,0168 |
| Yeniköy   | 110               | 420               | 1159,5 | 3,3128 | 0,021  |
| Kemerköy  | 140               | 630               | 1697   | 3,2324 | 0,0137 |
| Yatağan   | 140               | 630               | 1822,8 | 3,472  | 0,0147 |
| Hamitabat | 190               | 1120              | 6595,5 | 7,0663 | 0,0168 |
| Ambarlı   | 245               | 1350              | 7290,6 | 7,2592 | 0,0127 |

#### **4.2. Hesaplama Ortamı ve Karşılaştırma Ölçütleri**

Bu çalışma kapsamındaki hesaplamalar, Intel Core i9-14900HX işlemci ve 32 GB DDR5 belleğe sahip bir bilgisayarda Python ile gerçekleştirilmiştir. Değerlendirmede üç ölçüt kullanılmıştır: elde edilen toplam maliyet, raporlanan çözüm süresi ve metasezgisel yöntemlerin tekrarlanan çalıştırmalarındaki değişkenlik. Maliyetler dolar/saat (\$/h), üretimler MW ve süreler saniye olarak ifade edilmiştir.

Metasezgisel yöntemler için 20 çalıştırmaya ait maliyet ve süre kayıtları bulunmaktadır. Buna karşılık SLSQP için sonuç tablolarında verilen sürelerin aynı biçimde bir tekrar dağılımı sunulmamıştır. Bu nedenle SLSQP'nin tekil süre değerleri ile diğer yöntemlerin 20 çalıştırma ortalamaları aynı tür istatistikler olarak değerlendirilmemelidir. Bölümde tekil sonuç tabloları ile tekrar özetleri ayrı olarak gösterilmektedir.

Elde edilen bulgular, belirlenen parametre setleri ve ele alınan iki test sistemi kapsamındaki deneysel performansı

yansıtmaktadır. Algoritmaların farklı problem boyutları ve kısıt yapıları altındaki ölçeklenebilirliği, ileriki çalışmalarda parametre duyarlılık analizleriyle desteklenebilir.

## **5. KARŞILAŞTIRMALI BULGULAR VE TARTIŞMA**

### **5.1. Sekiz Generatörlü Sistemin Sonuçları**

Sekiz generatörlü sistemin üretim dağılımları Tablo 2’de verilmiştir. GA, YAK ve SLSQP’nin santral bazındaki üretimleri birbirine yakındır. SLSQP’de en yüksek üretim Bursa Doğalgaz santralinde 649,6943 MW, en düşük üretim ise Hamitabat’ta 368,7548 MW olarak raporlanmıştır. Yakın toplam maliyetlere, santraller arasında küçük üretim farkları içeren dağılımlarla ulaşılabilmektedir.

**Tablo 2. Sekiz generatörlü sistemde üretim dağılımları**

| <b>Santral</b> | <b>GA<br/>(MW)</b> | <b>YAK<br/>(MW)</b> | <b>SLSQP<br/>(MW)</b> |
|----------------|--------------------|---------------------|-----------------------|
| Hamitabat      | 366,583            | 367,435             | 368,7548              |
| Ambarlı        | 479,371            | 480,295             | 480,1974              |
| Bursa DG       | 651,504            | 650,617             | 649,6943              |
| Seyitömer      | 587,308            | 587,724             | 587,3105              |
| Soma B         | 394,131            | 393,592             | 393,8555              |
| Yeniköy        | 387,252            | 385,278             | 384,375               |
| Kemerköy       | 591,856            | 592,367             | 592,111               |
| Yatağan        | 542,267            | 542,784             | 543,701               |

Maliyet ve süre sonuçları Tablo 3’te birlikte sunulmuştur. Elde edilen sonuçlarda GA maliyeti 80530 \$/h, YAK maliyeti 80530,59 \$/h ve SLSQP maliyeti 80530,333 \$/h olarak verilmiştir. Bu değerler farklı ondalık hassasiyetlerle raporlandığı için küçük maliyet farkları üzerinden kesin bir sıralama yapılması

uygun değildir. Bulguların temel özelliği, yöntemlerin yaklaşık aynı maliyet düzeyine ulaşmasıdır.

**Tablo 3. Sekiz generatörlü sistemde maliyet ve çözüm süreleri**

| Ölçüt          | GA      | YAK      | SLSQP     |
|----------------|---------|----------|-----------|
| Maliyet (\$/h) | 80530   | 80530,59 | 80530,333 |
| Süre (s)       | 112,908 | 27,53    | 0,00914   |

Arslan'dan (2015) bu çalışmaya aktarılan GA ve YAK karşılaştırma değerleri de 80530 \$/h düzeyindedir. Bu yakınlık, aynı sistem için benzer maliyetlerin elde edildiğini göstermektedir. Ancak bu literatür değerleri, tezdeki Python uygulamalarının tekrarları olarak değerlendirilmemelidir.

Raporlanan sürelerde belirgin fark vardır. GA için 112,908 saniye, YAK için 27,53 saniye ve SLSQP için 0,00914 saniye verilmiştir. Elde edilen bulgular, SLSQP'nin GA'ya göre yaklaşık 12000 kat ve YAK'a göre yaklaşık 3000 kat daha kısa sürede sonuç ürettiğini göstermektedir. Bu oranlar, belirli ölçümlerin karşılaştırılmasıdır; farklı donanım, durdurma ölçütü veya uygulamalar için aynı oranların oluşacağı sonucunu vermez.

## **5.2. Altı Generatörlü Sistemin Sonuçları**

Altı generatörlü sistemin üretim dağılımları Tablo 4'te gösterilmiştir. SLSQP çözümünde Bursa Doğalgaz 555,002 MW, Seyitömer 515,072 MW, Soma B 334,073 MW, Yeniköy 336,559 MW, Kemerköy 518,819 MW ve Yatağan 475,372 MW üretmektedir. GA ve TB dağılımları SLSQP'ye yakinken, TA'da özellikle Kemerköy ve Yeniköy üretimlerinde daha belirgin ayrışma görülmektedir.

**Tablo 4. Altı generatörlü sistemde üretim dağılımları**

| Santral   | GA<br>(MW) | TB<br>(MW) | TA<br>(MW) | SLSQP<br>(MW) |
|-----------|------------|------------|------------|---------------|
| Bursa DG  | 557,269    | 555,480    | 562,116    | 555,002       |
| Seyitömer | 516,859    | 511,396    | 537,369    | 515,072       |
| Soma B    | 331,623    | 335,243    | 340,720    | 334,073       |
| Yeniköy   | 334,035    | 336,629    | 358,286    | 336,559       |
| Kemerköy  | 517,798    | 520,791    | 459,218    | 518,819       |
| Yatağan   | 477,564    | 475,387    | 477,245    | 475,372       |

Tablo 5'te SLSQP maliyeti 47660 \$/h, GA ve TB maliyetleri 47661 \$/h, TA maliyeti ise 47727 \$/h olarak raporlanmıştır. Böylece bu sonuç tablosunda en düşük maliyet SLSQP ile elde edilmiştir. Bununla birlikte GA ve TB ile farkın 1 \$/h olduğu ve maliyetlerin tam sayı düzeyinde verildiği dikkate alınmalıdır.

**Tablo 5. Altı generatörlü sistemde maliyet ve çözüm süreleri**

| Ölçüt          | GA    | TB    | TA    | SLSQP   |
|----------------|-------|-------|-------|---------|
| Maliyet (\$/h) | 47661 | 47661 | 47727 | 47660   |
| Süre (s)       | 26,24 | 7,79  | 33,77 | 0,00706 |

Çalışmada Duman ve arkadaşlarından (2010) aktarılan karşılaştırma maliyetleri GA için 47679 \$/h, TB için 47876 \$/h ve TA için 47678 \$/h'tır. Çalışma kapsamındaki GA ve TB uygulamaları bu değerlerden daha düşük, TA uygulaması ise daha yüksek maliyet vermiştir. Bu durum, aynı yöntem ailesinin farklı uygulamalarının aynı sonuca ulaşmak zorunda olmadığını göstermektedir.

Süre tablosunda SLSQP için 0,00706 saniye raporlanırken, GA için 26,24 saniye, TB için 7,79 saniye ve TA için 33,77 saniye verilmiştir. İncelenen bu ölçümlerde SLSQP en kısa sürede sonuç üretmiştir. Ancak TB'nin Tablo 5'te yer alan

7,79 saniyelik değeri algoritmanın Python ortamındaki ilk çalıştırma/önbellek yükleme süresini yansıtmakta olup, takip eden çalıştırmalardaki kararlı ortalama süre Tablo 7'de görüldüğü üzere 2,00 saniye düzeyindedir.

### **5.3. Tekrar Deneylerinde Maliyet ve Süre Değişkenliği**

Metasezgisel yöntemlerin 20 çalıştırmasına ait maliyet özetleri Tablo 6'da sunulmaktadır. Altı generatörlü sistemde GA'nın ortalama maliyeti 47662,97 \$/h ve standart sapması 1,38'dir. TB'de ortalama 47709,34 \$/h, standart sapma 134,95; TA'da ortalama 48148,80 \$/h, standart sapma 238,21 olarak raporlanmıştır. Bu kayıtlar içinde GA'nın maliyet değişkenliği daha düşüktür.

**Tablo 6. Yirmi çalıştırmada raporlanan maliyet istatistikleri**

| <b>Sistem / yöntem</b> | <b>Ortalama (\$/h)</b> | <b>Standart sapma</b> | <b>Varyans</b> |
|------------------------|------------------------|-----------------------|----------------|
| 14 bara / GA           | 47662,97               | 1,38                  | 1,91           |
| 14 bara / TB           | 47709,34               | 134,95                | 18211,81       |
| 14 bara / TA           | 48148,80               | 238,21                | 56743,92       |
| 22 bara / GA           | 80533,94               | 2,06                  | 4,25           |
| 22 bara / YAK          | 80554,04               | 18,08                 | 326,98         |

Sekiz generatörlü sistemde GA'nın ortalama maliyeti 80533,94 \$/h, YAK'ın ortalama maliyeti 80554,04 \$/h'tır. Standart sapmalar sırasıyla 2,06 ve 18,08 olarak verilmiştir. Dolayısıyla bu sistemde de GA'nın maliyet sonuçları, raporlanan tekrarlar içinde daha dar bir dağılım göstermektedir. Bu bulgu, tek bir iyi çözümün yanı sıra tekrarlanabilirliğin de değerlendirmeye katılmasının önemini ortaya koymaktadır.

Sürelerin tekrar özetleri Tablo 7'de verilmiştir. Altı generatörlü sistemde TB'nin ortalama süresi 2,00 saniye ile GA ve TA'dan kısadır. Sekiz generatörlü sistemde YAK'ın ortalama

süresi 27,85 saniye, GA'nınki 104,59 saniyedir. Böylece maliyet değişkenliği daha düşük olan yöntemin her zaman daha kısa sürede çalışmadığı görülmektedir.

**Tablo 7. Yirmi çalıştırmada raporlanan süre istatistikleri**

| Sistem / yöntem | Ortalama (s) | Standart sapma | Varyans |
|-----------------|--------------|----------------|---------|
| 14 bara / GA    | 22,34        | 2,00           | 4,01    |
| 14 bara / TB    | 2,00         | 0,029          | 0,00085 |
| 14 bara / TA    | 33,46        | 0,45           | 0,20    |
| 22 bara / GA    | 104,59       | 7,96           | 63,39   |
| 22 bara / YAK   | 27,85        | 0,44           | 0,19    |

#### **5.4. Bulguların Yorumlanması ve Uygulama Sınırları**

İki sistem birlikte ele alındığında SLSQP'nin dikkat çeken özelliği, yakın maliyet düzeylerine kısa raporlanan sürelerle ulaşmasıdır. Bu sonuç, kullanılan ikinci dereceden maliyet fonksiyonlarının ve kayıpsız modelin yapısıyla birlikte değerlendirilmelidir. Düzgün maliyet fonksiyonları üzerinde türev bilgisinden yararlanabilen bir yöntemin performansı, daha karmaşık veya düzgün olmayan bir amaç fonksiyonundaki performansını doğrudan belirlemez.

Deney sonuçları ayrıca maliyet, süre ve değişkenliğin ayrı ölçütler olduğunu göstermektedir. GA, tekrar deneylerinde düşük maliyet değişkenliği sağlarken daha uzun sürelerle ihtiyaç duymuştur. YAK ve TB belirli karşılaştırmalarda daha kısa süreler sunmuş, ancak maliyet dağılımları daha geniş olmuştur. TA'nın altı generatörlü sistemdeki maliyet değişkenliği, bu uygulamadaki komşuluk ve parametre tercihlerinin ayrıca incelenmesini gerektiren bir bulgudur.

Sistemlerin yalnızca iki boyutta karşılaştırılması, büyük ölçekli sistemler için bir ölçeklenebilirlik sonucu çıkarmaya yeterli değildir. Benzer biçimde, optimizasyonun kısa sürmesi tek

başına çevrim içi işletimin doğrulandığı anlamına gelmez. Çevrim içi kullanımda veri toplama, talep güncelleme, iletişim, uygulanabilirlik kontrolü ve işletme güvenliği de toplam karar sürecinin parçalarıdır. Bulgular bu tür uygulamalar için bir başlangıç gerekçesi sunmakta, tamamlanmış saha doğrulaması oluşturmamaktadır.

Çalışmada generatörlerin aktif güç üretim sınırları ( $P_{i,min}, P_{i,max}$ ) ile sistem güç dengesi eşitliği temel kısıtlar olarak dikkate alınmıştır. Buna karşılık, iletim hatlarının termik kapasite limitleri, bara gerilim sınırları ve şebeke kayıpları gibi şebeke düzeyindeki operasyonel kısıtlar kayıpsız EYD modeli kapsamında analiz dışı bırakılmıştır.

## 6. SONUÇ

Bu bölümde Türkiye'deki sekiz generatörlü, 22 baralı ve altı generatörlü, 14 baralı bölgesel güç sistemlerinin kayıpsız ekonomik yük dağıtım sonuçları değerlendirilmiştir. Matematiksel modelin ardından SLSQP, GA, YAK, TA ve TB yöntemlerinin temel çalışma ilkeleri açıklanmış; uygulanmış oldukları sistemlerde üretim dağılımları, maliyetler, çözüm süreleri ve tekrar istatistikleri karşılaştırılmıştır.

Sekiz generatörlü sistemde SLSQP ile 80530,333 \$/h maliyet ve 0,00914 saniye çözüm süresi elde edilmiştir. GA ve YAK aynı sistemde yakın maliyetler vermiş, raporlanan süreleri daha uzun olmuştur. Altı generatörlü sistemde SLSQP'nin maliyeti 47660 \$/h ve süresi 0,00706 saniyedir. GA ve TB'nin 47661 \$/h değerleri bu maliyete yakınken, TA için 47727 \$/h raporlanmıştır.

Tekrar deneyleri, değerlendirmeye farklı bir boyut eklemektedir. GA'nın her iki sistemdeki maliyet standart sapmaları, karşılaştırıldığı metasezgisel yöntemlerden düşüktür.

Buna karşılık süre bakımından YAK ve TB'nin kendi sistemlerinde GA'dan daha kısa ortalamalara sahip olduğu görülmüştür. Bu nedenle yöntem seçimi, yalnızca tek bir çalıştırmanın en düşük maliyetine dayandırılmamalıdır.

Sonuçlar, incelenen kayıpsız modellerde SLSQP'nin kısa sürede ekonomik dağıtım üretebildiğini göstermektedir. Daha kapsamlı değerlendirmelerde iletim kayıpları ve işletme kısıtlarının açık biçimde modellenmesi, farklı talep düzeylerinin incelenmesi, ortak durdurma ölçütlerinin kullanılması ve SLSQP dâhil bütün yöntemler için tekrarlı süre ölçümlerinin sunulması yararlı olacaktır. Böyle bir değerlendirme, gözlenen hesaplama avantajının hangi koşullarda sürdüğünü ve çevrim içi uygulamalara ne ölçüde aktarılabildiğini ortaya koyabilir.

## KAYNAKÇA

- Akay, B., & Karaboga, D. (2012). A modified artificial bee colony algorithm for real-parameter optimization. *Information Sciences*, 192, 120–142. doi:10.1016/j.ins.2010.07.015
- Alhijawi, B., & Awajan, A. (2024). Genetic algorithms: Theory, genetic operators, solutions, and applications. *Evolutionary Intelligence*, 17(3), 1245–1256. doi:10.1007/s12065-023-00822-6
- Arslan, Ö. P. (2015). *Güç sistemlerinin ekonomik ve çevresel ekonomik yük dağıtımında yeni optimizasyon tekniklerinin kullanımı* (Yüksek lisans tezi). Kırıkkale Üniversitesi, Kırıkkale.
- Aydın, E. (2025). *ELD optimization with SLSQP* [Bilgisayar yazılımı]. <https://github.com/EnzelAydin/ELD-optimization-with-SLSQP> adresinden erişildi.
- Başaran, Ü. (2004). *Türkiye’deki 380 kV’luk enterkonnekte güç sisteminde çeşitli güç akışı ve ekonomik dağıtım analizleri* (Yüksek lisans tezi). Anadolu Üniversitesi, Eskişehir.
- Duman, S., Döşoğlu, M. K., Öztürk, A., & Erdoğmuş, P. (2010). Türkiye’deki güç sisteminde tavlama benzetimi, genetik algoritma ve tabu araştırma algoritmaları kullanılarak ekonomik dağıtım. *Engineering Sciences*, 5(1), 64–78.
- Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13(5), 533–549. doi:10.1016/0305-0548(86)90048-1
- Karaboga, D. (2005). *An idea based on honey bee swarm for numerical optimization* (Teknik Rapor TR06). Erciyes Üniversitesi, Kayseri.

- Karaboga, D., & Basturk, B. (2008). On the performance of artificial bee colony (ABC) algorithm. *Applied Soft Computing*, 8(1), 687–697. doi:10.1016/j.asoc.2007.05.007
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598), 671–680.
- Nocedal, J., & Wright, S. J. (2006). *Numerical optimization* (2. bs.). New York, NY: Springer.
- Saadat, H. (1999). *Power system analysis*. Boston, MA: McGraw-Hill.
- Sinha, N., Chakrabarti, R., & Chattopadhyay, P. K. (2003). Evolutionary programming techniques for economic load dispatch. *IEEE Transactions on Evolutionary Computation*, 7(1), 83–94. doi:10.1109/TEVC.2002.806788
- Subathra, M. S. P., Easter Selvan, S., Aruldoss Albert Victoire, T., Hepzibah Christinal, A., & Amato, U. (2015). A hybrid with cross-entropy method and sequential quadratic programming to solve economic load dispatch problem. *IEEE Systems Journal*, 9(3), 1031–1044. doi:10.1109/JSYST.2013.2297471

# **THE EFFECT OF GRAPH DENSITY AND THREAD COUNT ON THE PERFORMANCE OF MALATYA CENTRALITY-BASED MAXIMAL INDEPENDENT SET DETECTION**

**Furkan ÖZTEMİZ<sup>1</sup>**

## **1. INTRODUCTION**

Graphs enable many real-world systems, such as communication networks, transportation systems, social interaction structures, biological networks, text representations, and scheduling problems, to be modeled through vertex and edge relationships. In these models, an independent set is a fundamental combinatorial structure that requires selecting vertices that are pairwise nonadjacent. Whereas the maximum independent set problem seeks an independent set of highest cardinality, a maximal independent set is one to which no new vertex can be added. This distinction is especially critical in the evaluation of greedy methods, because a greedy process can always produce a maximal output but does not guarantee maximum cardinality on general graphs. Exact algorithms developed for the maximum independent set problem retain exponential worst-case complexity, and techniques that are sensitive to problem structure are needed for large-scale instances (Xiao & Nagamochi, 2017).

The computational cost of the independent set problem is affected not only by the selection rule, but also by graph size, density, data structure, and the parallel execution capacity of the

---

<sup>1</sup> İnönü University, Faculty of Engineering, Department of Software Engineering,  
ORCID: 0000-0001-5425-3474.

hardware. In particular, for graphs containing hundreds of thousands of vertices and tens of millions of edges, scanning adjacency lists, updating vertex scores, and making a global selection at each iteration make memory-access and synchronization costs pronounced. Therefore, the way in which the same logic scales on a multicore system should be examined scientifically just as much as the theoretical selection logic of the algorithm. The parallel maximal independent set literature ranges from the classical randomized approaches of Luby (1986) and Alon, Babai, and Itai (1986) to more recent studies that parallelize greedy orderings while preserving their dependency structure (Blelloch, Fineman, & Shun, 2012).

The method examined in this chapter is a deterministic maximal independent set process that uses the Malatya Centrality (MC) value as the greedy selection priority (Yakut 2025b). Malatya Centrality is a centrality approach that measures the local structural position of a vertex by relating its degree to the degrees of its neighbors (Yakut, Öztemiz, & Karci, 2023), (Karaca, C., & Yakut, S. 2025). Subsequent studies compared MC rankings with well-known centrality measures (Yakut, Öztemiz, & Karci, 2025) and showed that derivative variants can be used in independent set selection (Öztemiz, 2025). The experimental question here is not whether the algorithm finds the maximum independent set, but rather the extent to which the same deterministic greedy selection can benefit from an increasing number of threads at different graph densities.

The experimental basis of the study consists of five  $G(n,p)$  graphs with  $n=100,000$  vertices and edge probabilities  $p=\{0.01; 0.03; 0.05; 0.07; 0.10\}$ . The  $G(n,p)$  model is the classical random graph model in which each possible edge is present independently with probability  $p$  (Gilbert, 1959). At each density, 22 thread configurations were run 21 times, yielding a total of 2,310 full-solution runs and 782,166 independent-set member detection

events. The primary objective is to jointly evaluate the interaction between thread count and graph density in terms of absolute latency, speedup, parallel efficiency, number of algorithmic rounds, and phase-based runtime.

The contribution of the chapter can be summarized in three points. First, it is shown that as density increases, the volume of the neighborhood scanned in an iteration grows, while the total number of iterations decreases because the closed neighborhood of the selected vertex also grows. Second, increasing the thread count up to the nominal minimum time is not always appropriate from the perspective of resource efficiency; on a system with 14 logical processors, the range of 13-20 threads provides a strong latency-resource balance at most densities. Third, threats to validity such as the use of a single seed and nested graphs, JVM warm-up, oversubscription, and measurement scope are reported explicitly, thereby defining the limits within which the results should be interpreted.

## **2. THEORETICAL FRAMEWORK AND RELATED WORK**

### **2.1. Independent Set, Maximality, and Maximum Cardinality**

In an undirected graph  $G=(V,E)$ , a set  $S \subseteq V$  is an independent set if no two vertices in  $S$  are connected by an edge. If  $S$  cannot be extended to a larger independent set by adding a single vertex, it is a maximal independent set; if  $|S|$  is the largest among all independent sets, it is a maximum independent set. Maximality is a local completeness property, whereas maximum cardinality is a global optimality property. Therefore, every maximum independent set is maximal, but not every maximal independent set is maximum. Throughout this study, the

measured  $|IS|$  values are deterministic outputs of the MC-based greedy method and are not interpreted as evidence of optimality.

The maximal independent set problem has a special place in parallel and distributed algorithms. Luby (1986) presented one of the foundational examples of fast parallel maximal independent set algorithms, while Alon et al. (1986) examined the runtime of the randomized parallel approach under PRAM models. Blelloch et al. (2012) showed that the dependency depth of the sequential greedy maximal independent set algorithm under random vertex orderings permits parallel implementations that preserve greedy behavior. This literature emphasizes that the maximality guarantee is compatible with parallel computation, but that the parallelization strategy must be designed according to data dependencies.

## **2.2. Malatya Centrality and the Greedy Selection Rule**

Centrality measures are used to quantify the structural positions of vertices in a graph. In the classical centrality literature, measures such as degree, closeness, and betweenness represent different structural meanings (Freeman, 1978). Malatya Centrality, by contrast, is a local ratio measure that relates the degree of a vertex to the degrees of its neighbors. According to the definition of Yakut et al. (2023), for a vertex  $v$  with  $d_t(v) > 0$  in the current induced graph  $G_t$ , the value used is as follows: (Yakut 2025a)

$$MC_t(v) = \sum_{u \in N_t(v)} \frac{d_t(v)}{d_t(u)} = d_t(v) \sum_{u \in N_t(v)} \frac{1}{d_t(u)}$$

This expression evaluates a high-degree vertex not only by its own degree but also in the context of the degrees of its neighbors. The Malatya centrality value of a vertex is obtained by dividing the degree of that vertex separately by the degrees of its

neighboring vertices and summing these values. In the greedy process examined, the vertex with the smallest MC value is selected at each iteration; in the event of a tie, the smaller vertex ID is used as the deterministic tie-breaker. The selected vertex  $v^*$  is added to the independent set, and  $v^*$  together with its closed neighborhood  $N_t[v^*]$  is removed from the active graph. Thus, independence is preserved between successive selections. Vertices with degree zero in the current graph are added directly to the set.

$$v^* = \arg \min_v (MC_{t(v)}, id(v)), S \leftarrow S \cup \{v^*\}, V_{t+1} \leftarrow V_t \setminus N_t[v^*]$$

Although the application of the MC approach to the independent set problem was presented in early studies in the context of the maximum independent set (Yakut et al., 2023), the safe and directly stated outcome of greedy selection for general graphs is maximality. The Differential Malatya Independent Set Algorithm study also reports strong experimental results for various graph families, while clearly showing that the result on random and social graphs need not be optimal in general (Öztemiz, 2025). This chapter conducts the performance analysis within this cautious maximality framework.

### **2.3. Parallelization Strategy**

In the implementation examined, the iterations themselves are sequential, because the active vertex set, degrees, and neighborhoods at time  $t+1$  depend on the vertex selected at time  $t$ . Parallelism is applied within the MC scan of each iteration. The coordinator thread partitions the active vertex-ID space into  $Q=\min(n,4P)$  task blocks, where  $P$  is the number of worker threads. In the same read-only iteration state, each worker computes the MC values in its own block and the task-local minimum. After all tasks are completed, a barrier is formed, the global minimum is selected deterministically from the local minima, and the active graph is updated by a single coordinator.

This design aims to keep the greedy selection order independent of the thread count. Rather than changing the decision logic of the algorithm, parallelization shares the score-computation cost of the same decision. A natural consequence is that the relative share of isolated-vertex scanning, minimum reduction, graph updating, and barrier operations, which either cannot be parallelized or require coordination, increases in the total runtime as the thread count increases. The serial fraction imposing an upper bound at a fixed problem size is consistent with Amdahl's (1967) classical scalability argument. Gustafson's (1988) scaled-problem perspective, in turn, reminds us that more processors create a different speedup regime only when the workload also scales; because  $n$  is fixed in these experiments, the results primarily represent strong-scaling behavior.

### **3. EXPERIMENTAL DESIGN AND METHOD**

#### **3.1. Research Questions**

The experimental study focuses on three research questions: (a) to what extent does increasing the thread count reduce the time required to detect all independent-set members at each density; (b) how does graph density change the saturation point of the speedup and parallel-efficiency curves; and (c) how can the observed saturation be explained through the number of algorithmic rounds and phase-based runtimes? Rather than merely identifying the “fastest” thread count, these questions also aim to determine a resource-efficient operating point.

#### **3.2. Graph Generation and Experimental Matrix**

Five experimental graphs were generated with  $n=100,000$  vertices and target edge probabilities  $p=0.01; 0.03; 0.05; 0.07; 0.10$ . Because all densities were generated using seed=42 and the same underlying vertex-pair random-number sequence, the

graphs are not independent samples; higher  $p$  values form a controlled, nested graph family in which more edges are activated over the same underlying randomness. Although this common-random-number approach facilitates examining the increase in density over the same underlying structure, it cannot be interpreted as providing five independent replications from the  $G(n,p)$  population.

**Table 1. Experimental design and execution environment**

| Component                  | Value                                                      |
|----------------------------|------------------------------------------------------------|
| Number of vertices ( $n$ ) | 100,000                                                    |
| Thread counts              | 1-14; 20; 40; 50; 60; 70; 80; 90; 100                      |
| Repetitions / cell         | 21                                                         |
| Total full-solution runs   | 2,310                                                      |
| CPU-Memory                 | Apple M4 Pro, 14 physical / 14 logical processors - 24 GiB |
| Operating system           | macOS 26.5.2, arm64                                        |
| Java                       | Oracle Java HotSpot 23.0.2                                 |
| Graph library              | JGraphT 1.5.2                                              |

**Note:** JGraphT is a library developed for general-purpose Java graph data structures and algorithms (Michail, Kinable, Naveh, & Sichi, 2020).

Each density was run in a separate JVM process in a fixed increasing order of  $p$ . Each density/thread cell was measured 21 times; the thread order was applied in the forward direction in odd-numbered trials and in the reverse direction in even-numbered trials. The small-scale  $n \leq 300$  warm-up run was excluded from the main measurement. Graph generation, independence/maximality validation, and file writing were not included in the primary timer. Because JIT compilation, garbage collection, thread scheduling, and runtime variability can affect Java performance measurements, repetition, warm-up, and statistical reporting are important (Georges, Buytaert, & Eeckhout, 2007).

### 3.3. Metrics and Statistical Evaluation

The primary endpoint is end to end detection ms, defined as the elapsed time from the start of the solution call until the

identity of the final independent-set member is determined. This metric includes state preparation, creation of the worker pool, and preceding iterations; it excludes the graph update after the final detection and pool shutdown. Parallel mc phase ms is the sum of task creation, submit, worker computations, and barrier waiting; therefore, it should not be interpreted solely as arithmetic MC computation time. The median and interquartile range (IQR) were reported for each cell. Medians and speedups relative to the 1-thread reference at the same density were converted into 95% confidence intervals using a percentile bootstrap with 20,000 resamples, blocking by trial ID. The graph-generation random seed value is 20260810. Speedup and parallel efficiency were calculated as follows:

$$\text{Speedup}(P) = T_1 / T_P, \text{Efficiency}(P) = \text{Speedup}(P) / P$$

These intervals reflect runtime variability on the fixed seed-42 graph; they do not reflect uncertainty across different graph realizations. Therefore, the confidence intervals were interpreted not as sampling intervals generalizable to the graph population, but as an indicator of runtime stability on the same experimental object. In addition, the “5% plateau” was defined as the smallest tested thread count whose median time remained no more than 5% above the lowest median time observed at each density. This threshold is a descriptive decision criterion from a resource/efficiency perspective; it is not a statistical equivalence test.

### **3.4. Accuracy and Reproducibility Checks**

Before the analysis, the raw measurement, summary, and member-trace files were rescanned; the  $5 \times 22 \times 21 = 2,310$  experimental cell/trial grid, 782,166 member events, active-vertex transitions, MC scores, monotonic times, and phase totals were checked. Independence, maximality, and the invariance of the selection sequence across different thread/repetition runs

within the same density were also verified. All 31 checks were successful. This finding supports that the performance comparisons were made not between different solutions, but between executions producing the same deterministic member sequence.

## 4. RESULTS

### 4.1. Realized Graphs and Algorithmic Workload

**Table 2. Topological and algorithmic properties of the realized graphs**

| <b>p</b> | <b>Edge count</b> | <b>Avg. degree</b> | <b> IS </b> | <b>Rounds</b> |
|----------|-------------------|--------------------|-------------|---------------|
| 0.01     | 49,997,548        | 999.95             | 889         | 889           |
| 0.03     | 149,989,446       | 2,999.79           | 334         | 332           |
| 0.05     | 249,986,983       | 4,999.74           | 209         | 209           |
| 0.07     | 349,982,357       | 6,999.65           | 155         | 153           |
| 0.10     | 499,982,710       | 9,999.65           | 106         | 106           |

**Note:** |IS| is the deterministic maximal-independent-set output size of the MC-greedy method; it is not claimed to be the maximum independent set cardinality.

When moving from  $p=0.01$  to  $p=0.10$ , the number of edges increased by approximately tenfold, while the average degree rose from about 1,000 to 10,000. In contrast, the independent set size decreased from 889 to 106, and the number of sequential algorithmic rounds also decreased from 889 to 106.

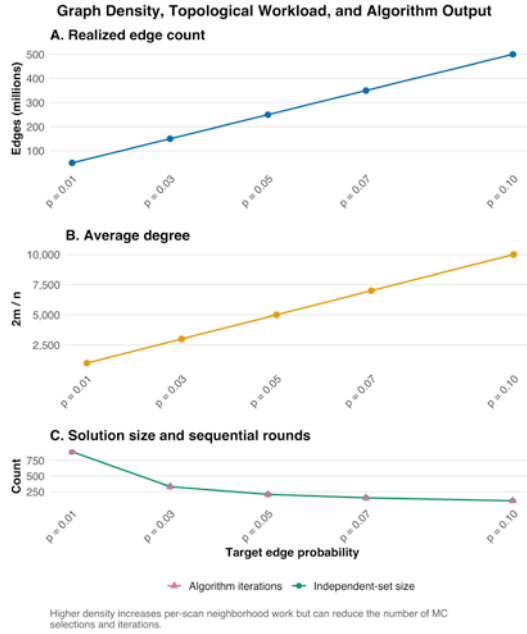


**Figure 1. Increase in edge/average-degree workload and decrease in MC-greedy output size and sequential round count as graph density increases.**

**Source:** Created from the experimental outputs in the benchmark report.

Thus, increasing density produces two opposing effects: longer adjacency lists for each active vertex require more memory reads and degree checks, whereas the larger closed neighborhood of a selected vertex allows the active graph to shrink more rapidly. The fact that total runtime does not increase monotonically with density can be explained by this opposition.

#### **4.2. Detection Time by Thread Count**

Across all densities, increasing the thread count from 1 to 8 produced a marked reduction in latency. Thereafter, the marginal gain diminished and the curves flattened after approximately 13-20 threads. This pattern is consistent with the fact that isolated-vertex scanning, task preparation, global-

minimum reduction, graph updating, and iteration barriers do not disappear completely even though the MC scan is parallelized.

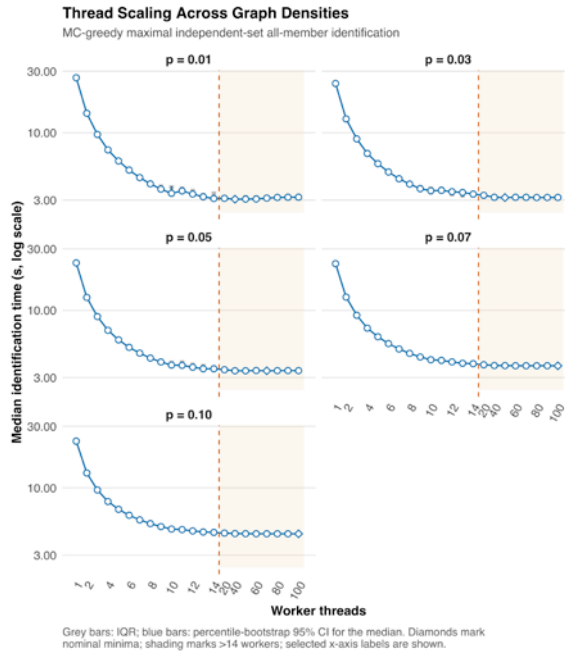


**Figure 2. Time required to identify all maximal independent set members as a function of thread count across five graph densities.**

Note: Gray bars show the IQR, blue bars show the bootstrap 95% confidence interval for the median, and diamonds indicate the nominal minimum among the tested configurations. The dashed line marks the limit of 14 available processors.

Beyond the 14 logical processors, the 20-100-thread settings are in the oversubscription region; therefore, additional threads do not imply additional physical execution resources.

### 4.3. Speedup, Efficiency, and the Resource-Efficient Operating Point

**Table 3. Key performance indicators by density**

| p    | 1T median (s) | Best T | Best (s) | Speedup | Eff.   | 5% T |
|------|---------------|--------|----------|---------|--------|------|
| 0.01 | 26.754        | 40     | 3.043    | 8.79×   | 21.98% | 13   |
| 0.03 | 24.153        | 50     | 3.134    | 7.71×   | 15.42% | 20   |
| 0.05 | 23.351        | 70     | 3.388    | 6.89×   | 9.85%  | 13   |
| 0.07 | 23.039        | 100    | 3.704    | 6.22×   | 6.22%  | 13   |
| 0.10 | 23.065        | 100    | 4.393    | 5.25×   | 5.25%  | 13   |

**Note:** “Best T” indicates only the lowest median among the 22 tested configurations. 5% T is the smallest tested thread count that remains no more than 5% above the minimum.

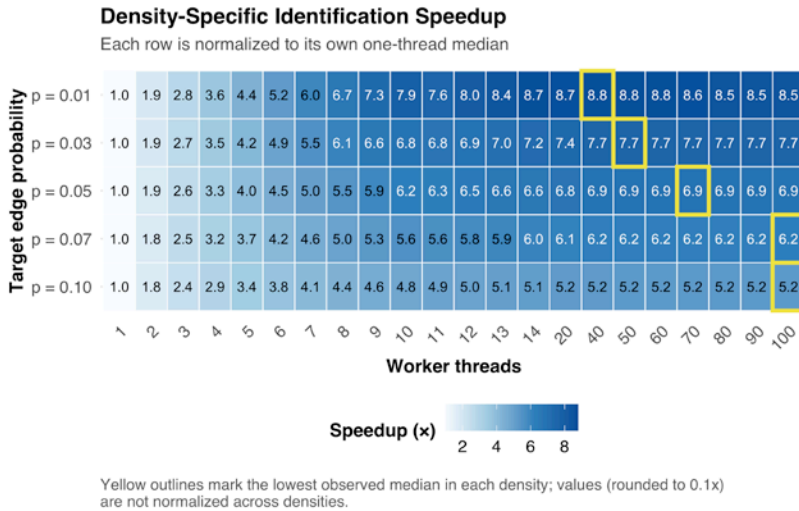


**Figure 3. Thread speedup gains by graph density.**

At the sparse end,  $p=0.01$ , 10 threads provided a  $7.91\times$  speedup, 14 threads provided  $8.69\times$ , and 40 threads, which yielded the nominal minimum, provided  $8.79\times$ . At  $p=0.10$ , the corresponding values decreased to  $4.82\times$ ,  $5.12\times$ , and  $5.25\times$  at 100 threads. Thus, the best observed speedup decreased by approximately 40.3% between the two ends of the density range. This decrease is consistent with the parallel scan reaching saturation earlier in denser graphs in the face of shared-memory

and coordination costs; however, because memory bandwidth was not measured directly using hardware counters, this explanation should be treated as a mechanistic interpretation.

The increase in the thread count yielding the nominal minimum from 40 to 100 may initially suggest that more threads are required. However, these points were obtained with very low parallel efficiency: 21.98% at  $p=0.01$  and only 5.25% at  $p=0.10$ . In contrast, the smallest setting within 5% of the minimum is 20 threads for  $p=0.03$  and 13 threads for the other four densities. For example, at  $p=0.10$ , 13 threads are only 3.33% slower than the nominal minimum, while imposing much less scheduler and task-coordination pressure than 100 threads. Therefore, in production systems, “lowest measured latency” and “a rational operating point per unit of resource” should be treated as separate decision variables.

#### **4.4. Effect of Density on Absolute Latency**

As density increased, the single-thread time decreased from 26.754 seconds to 23.065 seconds, or by approximately 13.8%. This result shows that the simple expectation that more edges will always produce a longer total runtime is insufficient. Although the number of edges and average degree increased by approximately tenfold, the roughly 88% reduction in the number of rounds compensates for the additional neighborhood cost in serial execution. On the other hand, the best observed parallel time increased from 3.043 seconds to 4.393 seconds, or by approximately 44.4%. Although the number of rounds continues to decrease in the parallel regime, the large neighborhood volume and coordination burden of each round limit the benefit obtained from additional threads.

This opposing behavior is an important performance-engineering result: graph density changes not only the total “amount of work,” but also the distribution of that work across

iterations. In the sparse case there are more iterations but cheaper scans, whereas in the dense case there are fewer iterations but more expensive scans. As the thread count increases, the latter structure amplifies the relative effect of memory access and synchronization, causing the strong-scaling curve to plateau earlier.

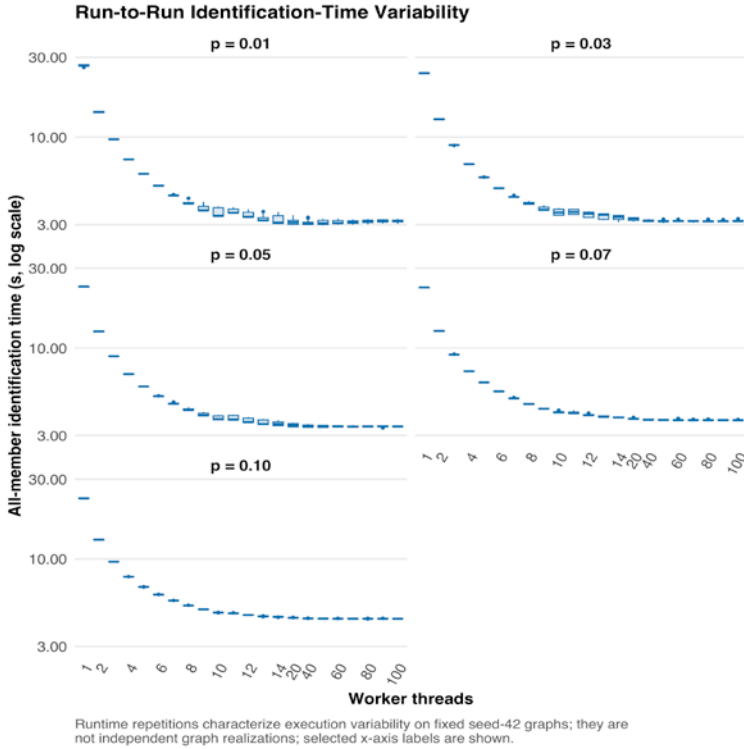


**Figure 4. Distribution of the 21 full-member detection times in each density/thread cell. These are runtime repetitions on the fixed graph, not independent graph realizations.**

#### **4.5. Phase Decomposition and the Amdahl Limit**

Phase decomposition shows that, with 1 thread, the dominant portion of total kernel time comes from the MC computation phase; as the thread count increases, this phase decreases sharply. In contrast, the graph-update stage remains

largely serial, while coordination costs such as isolated-vertex scanning and reduction/barrier operations do not disappear completely. At  $p=0.10$  and 100 threads, creating 400 task chunks in each iteration makes task-scheduling and join costs relatively more visible as parallel computation decreases.

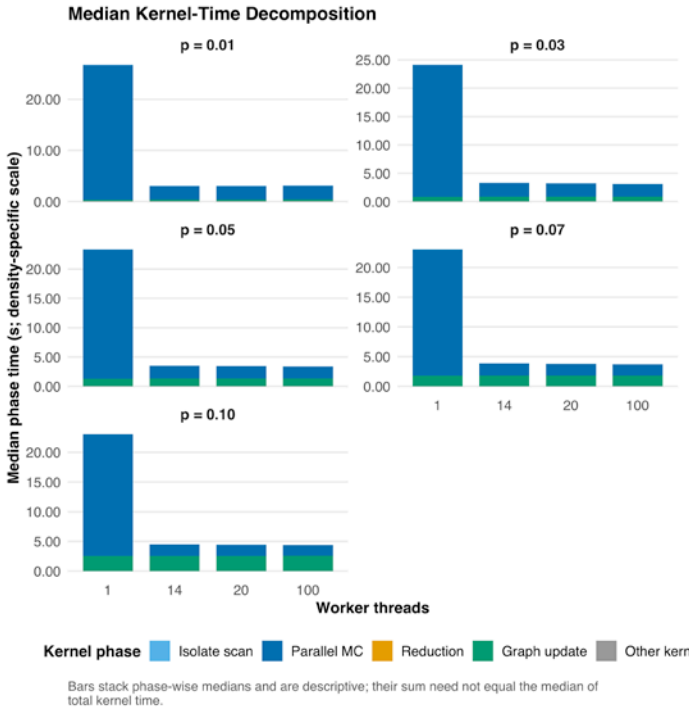


**Figure 5. Phase-based median time decomposition of the solver kernel at selected thread counts.**

Note: The bars stack the phase medians; because the median of each phase is calculated separately, the stacked total is not necessarily equal to the median of the total kernel time.

This result is the practical counterpart of Amdahl's law: once the parallelized MC phase becomes sufficiently small, serial and coordination-heavy components begin to determine the total runtime. Therefore, adding more threads does not increase total speedup at the same rate even if it provides a small gain in the parallel phase. For this reason, the sharp drop in the efficiency

curve after 14 threads should be explained not only by hardware oversubscription, but also by the decreasing share of the accelerated component in the total runtime.

## **5. DISCUSSION**

### **5.1. Meaning of the Density-Scalability Interaction**

The experiments show that the effect of graph density on parallel performance is not one-dimensional. On the one hand, as density increased, the number of independent-set members and the number of greedy iterations decreased dramatically, which reduced the total single-thread runtime. On the other hand, the neighborhood scan and degree-based MC computation in each iteration touched a larger volume of data, making the memory-access characteristics of the parallel phase more demanding. As a result, the serial run became slightly faster, while the best parallel run became slower and the maximum observed speedup decreased. This distinction shows that scalability analysis in graph algorithms should consider not only  $n$  or  $|E|$ , but also how the algorithm shrinks the active graph.

MC-based selection is particularly interesting here because the scores must be recomputed from the current active degrees at every iteration. This property keeps selection quality responsive to the current structure while creating strong data dependencies between iterations. Thus, the current implementation differs from Luby-type MIS algorithms that make many independent selections within the same round; parallelism occurs not between decisions, but in the score scan preceding a single decision. This architecture makes it easier to preserve the deterministic output order, but it also limits the maximum degree of parallelism.

## **5.2. Operating Point from the Perspective of Resource Efficiency**

When converting a benchmark result into a production setting, tracking only the smallest runtime can be misleading. On a 14-processor system, the range of 40-100 threads does not increase physical parallelism; instead, it presents more runnable tasks to the operating system and JVM scheduler. The fact that nominal minima occur in this region may lead very small, runtime-noise-sensitive differences in the tail of the curve to be called “optimal.” When metrics such as resource consumption, energy, impact on co-running processes, and tail latency are taken into account, the 5% plateau approach is useful for selecting a more balanced default.

In this data set, 13 threads remained within 5% of the best median at four densities, while 20 threads did so at  $p=0.03$ . This result is not a universal, hardware-independent recommendation; it was obtained in the context of the Apple M4 Pro, Java 23, G1GC, and the data structures used. Different NUMA architectures, servers with more cores, different heap/GC settings, or adjacency representations may change the saturation point. Nevertheless, the methodological lesson is general: the “fastest tested setting” and a “sufficiently fast and more efficient setting” should be reported together.

## **5.3. Benchmark Quality and Scientific Reproducibility**

One of the strengths of the study is the use of 21 repetitions per cell, forward/reverse balancing of order direction, median/IQR, bootstrap intervals, and comprehensive validation checks. In managed runtimes such as Java, performance results based on a single run or only the best measurement are known to be potentially misleading (Georges et al., 2007). However, the current design could be improved with respect to full-size warm-up/fork repetitions, randomization of density order, and the use of

multiple graph seeds. In particular, if generalization to the graph population is intended, multiple independent seeds should be used for each  $p$ , and within-seed runtime repetitions and between-seed structural variability should be separated hierarchically.

For reproducibility, it is important to specify the source code, fixed parameters, and the JGraphT version used. JGraphT provides a widely used Java infrastructure for data structures and algorithms in academic/applied graph studies (Michail et al., 2020). Nevertheless, implementing the same algorithm with different adjacency representations, primitive collections, or cache-local data structures can change absolute runtimes. Therefore, the numbers reported here are valuable for methodological comparison; when transferring the method to another software stack, the shapes of the curves and the saturation mechanisms should be considered more fundamental than the absolute time values in seconds.

## **6. LIMITATIONS AND THREATS TO VALIDITY**

The first limitation is that the experiments are based on a nested graph family generated from a single seed=42. The 21 repetitions do not generate new graph instances; they measure runtime variability on the same graph. Therefore, independent seed replications are required to generalize the density effect to the  $G(n,p)$  population. Second, the output of the algorithm is maximal; maximum independent set cardinality or approximation ratio is not the subject of this benchmark. Third, all settings beyond 14 threads are oversubscribed, and the optimum/saturation point may vary with the CPU, memory system, JVM, and operating system.

Fourth, the thread grid is sparse after 14; because the ranges 15-19 and 21-39 were not scanned, 40, 50, 70, or 100 threads cannot be called a “global optimum.” The scientifically

accurate statement is the lowest median among the tested configurations. Fifth, the forward/reverse order is neither full randomization nor a Latin square, and the positions are not perfectly balanced because of the 21 repetitions. JIT, GC, thermal state, and long-term system-load drift may remain. Sixth, warm-up is limited to  $n \leq 300$ ; the  $n=100,000$  execution path was not warmed with a separate full-size steady-state protocol.

Seventh, graph generation, validation, and file writing are not included in the main runtime. This approach is appropriate for isolating the solver kernel, but it does not represent one-off end-to-end application latency. Eighth, member-trace instrumentation is enabled in all runs and may introduce greater overhead than the uninstrumented solver. Ninth, ms/member or ms/iteration values do not represent an equal amount of work across densities because the neighborhood volume scanned in each iteration varies with  $p$ . Finally, the five densities were run in separate JVMs in a fixed increasing order of  $p$ ; this may confound the density effect with long-term changes in thermal conditions or system load.

## **7. CONCLUSION**

This chapter examined the thread scalability of the MC-based deterministic greedy maximal independent set algorithm and its interaction with graph density on five  $G(n,p)$  graphs with  $n=100,000$  vertices. At all densities, increasing the thread count substantially reduced the full-member detection time, but linear speedup was not observed. The best observed speedup was  $8.79\times$  at  $p=0.01$  and decreased to  $5.25\times$  at  $p=0.10$ ; as density increased, the parallel gain reached saturation earlier.

While the increase in density raised the number of edges by approximately tenfold, it reduced the greedy output size from 889 to 106 and the number of rounds from 889 to 106. Consequently, the single-thread time decreased by approximately

13.8% from  $p=0.01$  to  $p=0.10$ . In contrast, the best parallel time increased from 3.043 seconds to 4.393 seconds. The findings show that in dense graphs consisting of fewer but more expensive iterations, neighborhood-scanning and coordination costs constrain the scaling of the parallel phase earlier.

From the perspective of resource use, the most important practical result is the distinction between the nominal minimum runtime and the efficient operating point. On the test system with 14 logical processors, the range of 13-20 threads produced runtimes very close to the lowest medians, whereas the range of 40-100 threads provided only small additional latency gains with a marked loss of efficiency. Future studies should directly measure the effects of memory bandwidth, cache misses, and scheduling using multiple seeds, randomized density order, a full-size JVM warm-up/fork design, and hardware performance counters. In addition, repeating the same methodology with different data structures and high-core-count architectures will reveal the cross-hardware scalability profile of the MC-based greedy approach more reliably.

## REFERENCES

- Alon, N., Babai, L., & Itai, A. (1986). A fast and simple randomized parallel algorithm for the maximal independent set problem. *Journal of Algorithms*, 7(4), 567–583. [https://doi.org/10.1016/0196-6774\(86\)90019-2](https://doi.org/10.1016/0196-6774(86)90019-2)
- Amdahl, G. M. (1967). Validity of the single processor approach to achieving large scale computing capabilities. In *AFIPS Conference Proceedings*, 30, 483–485. <https://doi.org/10.1145/1465482.1465560>
- Blelloch, G. E., Fineman, J. T., & Shun, J. (2012). Greedy sequential maximal independent set and matching are parallel on average. In *Proceedings of the 24th ACM Symposium on Parallelism in Algorithms and Architectures* (pp. 308–317). <https://doi.org/10.1145/2312005.2312058>
- Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social Networks*, 1(3), 215–239. [https://doi.org/10.1016/0378-8733\(78\)90021-7](https://doi.org/10.1016/0378-8733(78)90021-7)
- Georges, A., Buytaert, D., & Eeckhout, L. (2007). Statistically rigorous Java performance evaluation. *ACM SIGPLAN Notices*, 42(10), 57–76. <https://doi.org/10.1145/1297027.1297033>
- Gilbert, E. N. (1959). Random graphs. *The Annals of Mathematical Statistics*, 30(4), 1141–1144. <https://doi.org/10.1214/aoms/1177706098>
- Gustafson, J. L. (1988). Reevaluating Amdahl's law. *Communications of the ACM*, 31(5), 532–533. <https://doi.org/10.1145/42411.42415>
- Karaca, C., & Yakut, S. (2025). Graph-Based Course Scheduling Using the Malatya Vertex Coloring Algorithm for Constraint Optimization. *Turkish Journal of Nature and*

- Science, 14(3), 46-56.  
<https://doi.org/10.46810/tdfd.1629184>
- Luby, M. (1986). A simple parallel algorithm for the maximal independent set problem. *SIAM Journal on Computing*, 15(4), 1036–1053. <https://doi.org/10.1137/0215074>
- Michail, D., Kinable, J., Naveh, B., & Sichi, J. V. (2020). JGraphT—A Java library for graph data structures and algorithms. *ACM Transactions on Mathematical Software*, 46(2), Article 16. <https://doi.org/10.1145/3381449>
- Öztemiz, F. (2025). A greedy approach to solve maximum independent set problem: Differential Malatya independent set algorithm. *Engineering Science and Technology, an International Journal*, 63, 101995. <https://doi.org/10.1016/j.jestch.2025.101995>
- S. Yakut (2025b), A robust and efficient algorithm for graph coloring problem based on Malatya centrality and sequent independent sets, *Egyptian Informatics Journal*, Volume 30, 2025, <https://doi.org/10.1016/j.eij.2025.100676>.
- S. Yakut, (2025a) "Efficient and Robust Graph Coloring Algorithm for Graph Coloring Problem: Malatya Vertex Coloring Algorithm," in *IEEE Access*, vol. 13, pp. 168512-168526, 2025, doi: 10.1109/ACCESS.2025.3614296
- Uçkan, T., & Karcı, A. (2020). Extractive multi-document text summarization based on graph independent sets. *Egyptian Informatics Journal*, 21(3), 145–157. <https://doi.org/10.1016/j.eij.2019.12.002>
- Xiao, M., & Nagamochi, H. (2017). Exact algorithms for maximum independent set. *Information and Computation*, 255, 126–146. <https://doi.org/10.1016/j.ic.2017.06.001>

- Yakut, S., Öztemiz, F., & Karci, A. (2023). A new approach based on centrality value in solving the maximum independent set problem: Malatya Centrality Algorithm. *Computer Science*, 8(1), 16–23. <https://doi.org/10.53070/bbd.1224520>
- Yakut, S., Öztemiz, F., & Karci, A. (2025). Kendall rank correlation analysis of Malatya Centrality Algorithm with well-known centrality measures. *Sigma Journal of Engineering and Natural Sciences*, 43(4), 1339–1354. <https://doi.org/10.14744/sigma.2025.00124>

# **STUDENTS' EXPERIENCES OF LEARNING TO PROMPT GENERATIVE AI: A META- SYNTHESIS OF QUALITATIVE RESEARCH IN K-12 AND HIGHER EDUCATION**

**Nihan ARSLAN NAMLI<sup>1</sup>**

## **1. INTRODUCTION**

When ChatGPT became publicly available in November 2022, many students soon discovered that the quality of what came back depended heavily on what they typed in. A one-line request produced a generic paragraph; one with a purpose, an audience and constraints produced something usable. The ability to formulate, test and revise such inputs, usually called prompt engineering, has quickly come to be treated as a component of AI literacy that students are expected to develop (Knoth, Tolzin, Janson, & Leimeister, 2024).

Reviews have catalogued prompting techniques in higher education (Lee & Palmer, 2025) but say little about the learner's side of the screen: what it is like to learn to prompt, and when students begin to treat prompt writing as a thinking task rather than a technical trick.

Qualitative studies addressing these questions are small and scattered across disciplines. Two recent meta-syntheses of nursing students' experiences with generative AI illustrate the gap: in both, prompting surfaced only as a subordinate finding, as limited AI literacy (Wang et al., 2026) or as demanding prompt

---

<sup>1</sup> Assist. Prof. Dr., İskenderun Technical University, Erzin Organized Industrial Zone Vocational School, Department of Computer Technologies, ORCID: 0000-0002-5425-1468.

formulation within a category on insufficient digital literacy (Du et al., 2026), and both covered a single profession. No qualitative evidence synthesis was identified that focused on how learners experience learning to prompt across school and university levels.

This chapter therefore reports a meta-synthesis of qualitative studies on students' experiences of learning to prompt generative AI in K-12 and higher education. Two questions guided the review:

- (1) How do students experience the process of learning to prompt generative AI?
- (2) How do these experiences differ between school-level and university-level learners?

## **2. CONCEPTUAL BACKGROUND**

### **2.1. From Prompt Engineering to Prompt Literacy**

In education the meaning of prompt engineering has widened beyond the systematic manipulation of model inputs. Hwang, Lee and Shin (2023) defined prompt literacy as the ability to write precise prompts, interpret outputs and refine prompts iteratively, treating prompting as a communicative and evaluative practice. Knoth et al. (2024) found that the quality of university students' prompts predicted the quality of their outputs. The phrase learning to prompt is used here to cover both senses.

### **2.2. Prompting as a Self-Regulated and Metacognitive Activity**

In Zimmerman's (2002) cyclical model of self-regulated learning, learners move through forethought (setting goals and planning), performance (executing and monitoring strategies) and self-reflection (evaluating and adjusting); a useful prompt seems to require all three. Yet generative AI can also weaken regulation:

learners supported by ChatGPT improved their essays while offloading self-regulatory effort, termed metacognitive laziness (Fan et al., 2025), and efficiency gains are not gains in metacognition or epistemic agency (L. Yan, Pammer-Schindler, Mills, Nguyen, & Gašević, 2025). Whether students experience prompting as regulation is empirical, so the lens was applied only at the interpretive stage.

### **3. METHOD**

#### **3.1. Design and Reporting Standards**

The review was designed as a qualitative evidence synthesis using thematic synthesis (Thomas & Harden, 2008), preferred to meta-aggregation because the aim was to generate an analytical theme, and to meta-ethnography because most eligible studies were expected to be descriptive (Flemming & Noyes, 2021). Reporting follows ENTREQ (Tong, Flemming, McInnes, Oliver, & Craig, 2012) and the PRISMA 2020 flow diagram (Page et al., 2021). The review questions, eligibility criteria and analysis plan were defined in an internal protocol before screening began.

#### **3.2. Research Team and Reflexivity**

The author, an academic in computer education and instructional technology, brings a prior expectation that prompting is educationally valuable; a second coder, an experienced qualitative researcher not involved in designing the review, therefore independently assessed all 30 full-text eligibility decisions (Cohen's  $\kappa = 0.92$ ). The single disagreement concerned a borderline study and was resolved through discussion by applying the decision rule in Section 3.4. The second coder also coded 30% of the extracted findings, and the themes were refined in two consensus meetings.

### **3.3. Search Strategy**

Because subscription databases were not accessible, an iterative, purposive search strategy was adopted, accepted when the aim is conceptual saturation rather than exhaustive retrieval (Flemming & Noyes, 2021). Searches were conducted in October 2026 through ERIC, DergiPark, publisher platforms and preprint servers, the latter used only to locate peer-reviewed versions. Queries combined four concept blocks (prompting, generative AI, learner population and qualitative inquiry) in English and Turkish. Successive rounds targeted higher education, K-12, teacher education, ERIC-indexed and Turkish-language studies, and searching stopped when further rounds yielded no new eligible studies, used here as a practical proxy for conceptual saturation; one report identified earlier but assessed last added a further concept (Section 5.3). As several platforms reported ranked results without reliable totals, records are reported cumulatively: the first round yielded 19 candidate records and later rounds raised the total to 45. Every record was verified against the publisher's page.

### **3.4. Eligibility Criteria**

Eligibility was defined with the SPIDER tool (Cooke, Smith, & Booth, 2012). Studies were included if they (a) involved K-12 or higher education students, including pre-service teachers; (b) addressed experiences of writing, testing or revising prompts for generative AI; (c) used a qualitative design or a distinct qualitative strand; (d) contained participant voice (interviews, reflections, journals or posts); and (e) were peer-reviewed articles or full conference papers in English or Turkish (2021–2026) with accessible full text. Criterion (b) was met when at least one reported finding supported by participant data concerned prompting; passing mentions were not sufficient. Studies of

prompt logs or outputs without participant accounts were drawn on only in the discussion.

### **3.5. Data Extraction**

Following Thomas and Harden (2008), all text reported as findings, including participant quotations and authors' interpretations, was treated as data; only qualitative strands and student data were extracted. Within mixed-methods studies, authors' interpretations of students' prompts or interaction logs were retained when they formed part of the qualitative strand; outcome measures from quantitative strands were not extracted.

### **3.6. Quality Appraisal**

The author appraised all 21 studies with the ten-item Critical Appraisal Skills Programme (CASP) checklist for qualitative research (Critical Appraisal Skills Programme, 2018), and the second coder independently appraised a random sample of six studies (Studies 3, 7, 10, 16, 17 and 19 in Table 1). Item-level agreement was 95%; the three disagreements, concerning recruitment, ethics and analysis, were resolved through discussion. Appraisal informed the confidence assessment rather than exclusion. Aims and findings were generally clear, whereas the researcher-participant relationship was rarely examined, analytic procedures were often briefly described, and in one study participant numbering was inconsistent with the reported sample (Jahan & Akbar, 2026).

### **3.7. Data Synthesis and Confidence Assessment**

Synthesis proceeded in three stages: line-by-line inductive coding of extracted findings in a spreadsheet matrix, grouping of codes into descriptive themes, and interrogation of these themes against the review questions to develop an analytical theme. The five themes in Sections 4.2–4.6 are descriptive; the analytical theme, developed at this stage through the self-regulated learning

lens, is presented in Section 5.1. Confidence in each finding was assessed with GRADE-CERQual (Lewin et al., 2018), which considers methodological limitations, coherence, adequacy of data and relevance.

## 4. FINDINGS

### 4.1. Study Selection and Characteristics

The iterative searches identified 45 records; 15 were excluded at title and abstract screening and 30 were assessed in detail. Nine were excluded: five contained no participant voice on prompting (e.g., Korkmaz & Tarım, 2025), two did not address prompting experience, one was available only as a preprint, and one had an insufficient qualitative strand (Figure 1). Twenty-one studies were included, involving 735 students in the 20 studies reporting sample size; 17 were conducted in higher education and four in K-12. Seven were conducted in the United States and fourteen elsewhere (Table 1).

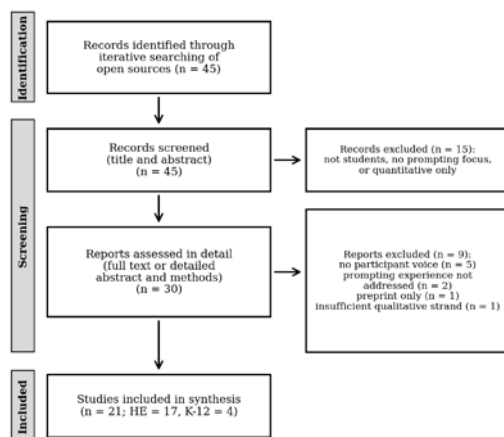


**Figure 1. PRISMA 2020 flow diagram of study selection**

**Table 1. Characteristics of the Included Studies**

| Study                     | Country     | Level                          | n    | Design / data                             |
|---------------------------|-------------|--------------------------------|------|-------------------------------------------|
| 1. Bae et al. (2024)      | USA         | HE, pre-service teachers       | 54   | Discussion posts                          |
| 2. Çiftci et al. (2025)   | Türkiye     | HE, pre-service teachers       | 32   | Chat logs, opinion forms                  |
| 3. Hawk et al. (2025)     | USA         | HE, nursing                    | 19   | Written reflections                       |
| 4. Hwang et al. (2025)    | South Korea | HE, EFL                        | n.r. | Artworks, reflective papers               |
| 5. Jahan & Akbar (2026)   | Oman        | HE, EFL                        | 30   | MM; interviews, focus groups, reflections |
| 6. Nguyen et al. (2024)   | USA         | HE, programming                | 120  | MM; prompt histories, interviews          |
| 7. Aruleba et al. (2025)  | UK          | HE, computer science           | 21   | Journals, interviews, artefacts           |
| 8. Shen et al. (2025)     | China       | HE, nursing                    | 17   | Semi-structured interviews                |
| 9. W. Yan et al. (2025)   | Japan       | HE, humanities                 | 16   | Dialogue logs, reflections, interviews    |
| 10. Woo et al. (2026)     | Hong Kong   | HE, history                    | 27   | MM; reflections, prompt libraries         |
| 11. Kim et al. (2026)     | China       | HE, preparatory                | 19   | MM; chat histories, interviews            |
| 12. Tümen Akyıldız (2026) | Türkiye     | HE, pre-service EFL teachers   | 10   | Reflexive TA; interviews                  |
| 13. Wen & Wen (2024)      | USA         | HE, pre-service teachers       | 16   | Reflections, recordings, focus group      |
| 14. M. Yang et al. (2025) | USA         | HE, nontraditional             | 73   | MM; focus groups, reflections, logs       |
| 15. Andrewi et al. (2025) | Indonesia   | HE, EFL                        | 30   | MM; interviews, focus group, logs         |
| 16. H. Yang (2025)        | South Korea | HE, EFL                        | 36   | Chat logs, open-ended questionnaire       |
| 17. Posavec (2026)        | Croatia     | HE, education & rehabilitation | 57   | MM; open-ended survey, artefacts          |
| 18. Kwon et al. (2026)    | USA         | K-12, secondary                | 55   | MM; written reflections, interaction logs |
| 19. Song et al. (2026)    | USA         | K-12, primary                  | 10   | Interviews, surveys, observation          |
| 20. Jalon et al. (2024)   | Philippines | K-12, secondary                | 26   | MM; open-ended survey                     |
| 21. Woo et al. (2025)     | Hong Kong   | K-12, secondary                | 67   | Stories, written reflections              |

Note. HE = higher education; EFL = English as a foreign language; MM = mixed methods; TA = thematic analysis; n.r. = not reported in accessible sources. For mixed-methods studies, n refers to participants who contributed qualitative data where this differed from the total sample.

#### **4.2. Theme 1: “Certain Ways You Have to Phrase Things”: Discovering the Missing Context**

Across most studies, students described a similar first encounter: they typed a request as they would type a search query or speak to a classmate, and received an answer that was fluent but generic or wrong. Beginning programmers struggled to write prompts even for problems at their own level (Nguyen et al., 2024). Some pre-service teachers addressed the system as a person while others wrote poorly articulated requests (Bae, Hur, Park, Choi, & Moon, 2024); one admitted, “I had no clue with the prompts” (Wen & Wen, 2024). Students with lower AI literacy relied on short requests to explain or define (Kim, Yu, Lee, & Detrick, 2026), mathematics teacher candidates pasted whole tasks into the chat window (Çiftci, Öz, & Çiftci, 2025), and high school students pasted drafts with little explanation (Kwon, Yang, Kale, Kim, & Park, 2026). Eleventh graders concluded that “AI needs specific instructions to work as intended” (Jalon, Chua, & Torres, 2024).

What changed over time was students’ model of the interlocutor. Humanities students came to explain that a human listener can fill gaps from shared experience whereas the model could not, so background, purpose and expected output had to be stated explicitly (W. Yan, Nakajima, & Sawada, 2025). Computer science students’ journals traced a move from generic one-shot prompts to deliberate dialogue (Aruleba, Sanusi, Obaido, Ogbuokiri, & Mienye, 2025), nursing students saw prompt design as shaping the quality of case analysis (Shen, Shen, Liu, & Jin, 2025), and pre-service EFL teachers moved from vague requests to level-specific instructions (Tümen Akyıldız, 2026), whereas nontraditional students kept questioning their skills even with a user guide (M. Yang et al., 2025). Indonesian undergraduates moved from commands such as “Fix this” to requests naming topic and tone: “When I added the topic and told it what I wanted,

the answer became much more useful” (Andewi et al., 2025). For these second-language learners the missing context also had a linguistic layer: most switched from prompting in Bahasa Indonesia to prompting in English by mid-course, without being instructed to, because English prompts produced more useful feedback. A tenth grader summed up the lesson: “There are certain ways you have to phrase things when speaking to AI” (Kwon et al., 2026). The theme concerns revising an assumption rather than acquiring a technique: that the machine shares the learner’s context.

#### **4.3. Theme 2: “Organize Your Thoughts Properly”: Prompting as Externalised Thinking**

In studies with sustained engagement, students reported that writing a precise request forced them to decide what they actually wanted. One student in W. Yan et al. (2025) put it simply: “It is important to organize your thoughts properly and write a quality prompt”; others described seeing their own reasoning from the outside and asking whether a disappointing output reflected the prompt or their own understanding. Korean undergraduates generating images to learn vocabulary monitored outputs and varied contextual details until the intended meaning emerged, a cycle of planning, monitoring and evaluation (Hwang, Lee, & Shin, 2025). Students with higher AI literacy re-read their prompts for ambiguity (Kim et al., 2026), and pre-service EFL teachers came to see prompt writing as lesson design (Tümen Akyıldız, 2026). Conversely, Korean English majors without a clear goal reported asking formulaic questions and running out of ideas for what to ask (H. Yang, 2025): the difficulty lay less in phrasing a request than in knowing what to request. Hong Kong secondary students who prompted their writing tools without a clear purpose offer a similar negative case at school level (Woo, Guo, & Susanto, 2025).

#### **4.4. Theme 3: Calibrating Trust Through Iteration**

Most students described moving from treating outputs as answers to treating them as drafts to be checked, although high school students ranged from questioning AI suggestions to accepting them uncritically (Kwon et al., 2026). Nursing students saw AI responses as summaries lacking depth that required external validation (Hawk, Coriasco, & Jones, 2025), mathematics teacher candidates noted inconsistencies (Çiftci et al., 2025), and pre-service teachers judged outputs on accuracy, relevance and specificity (Wen & Wen, 2024) or cross-checked them (Bae et al., 2024). Even primary pupils concluded that “AI is not going to be perfect” (Song et al., 2026), and a secondary student used it only as a guide (Jalon et al., 2024). Checking took two forms. Some students consulted external sources, as when a Croatian student corrected outdated statistics (Posavec, 2026); others asked the same model again, for example “Are you sure this structure is formal enough?” (Andewi et al., 2025), which tests the model’s consistency rather than its accuracy. A better prompt usually produced a better response, but not a trustworthy one.

#### **4.5. Theme 4: Drawing the Line: Delegation, Authorship and the Fear of Losing Skills**

As students became more capable prompters, they also became more conscious of what they were handing over. Students in W. Yan et al. (2025) built personal rules about delegation, accepting help with proofreading but refusing to let the model write a report meant to show their own learning. Others combined growing prompt fluency with ethical reflection (Aruleba et al., 2025), kept ownership by fact-checking (M. Yang et al., 2025), or wondered whether ideas that began with ChatGPT were still their own (Tümen Akyıldız, 2026). In Oman, acceptance of ChatGPT as a tutor coexisted with concerns about overreliance

and ethical ambiguity (Jahan & Akbar, 2026). An Indonesian student admitted, “I stopped trying to remember grammar rules. I just paste the sentence and let ChatGPT fix it”, while peers were unsure where assistance ended and cheating began (Andewi et al., 2025); this boundary was also the difficulty Croatian students reported most often (Posavec, 2026).

#### **4.6. Theme 5: Scaffolds That Make a Difference**

Explicit instruction appeared to matter. Undergraduates who attended a 100-minute prompt engineering workshop produced more effective prompts (Woo, Wang, Yung, & Guo, 2026), and Omani learners were trained in prompt generation before using ChatGPT in class (Jahan & Akbar, 2026). Practice, peer comparison and instructor modelling also helped (Bae et al., 2024; Tümen Akyıldız, 2026; W. Yan et al., 2025). By contrast, a written guide alone did not move most nontraditional students beyond low-level prompting (M. Yang et al., 2025), and after a single lecture relatively few Croatian students refined their prompts repeatedly; one asked for classes in which students are “guided to ask the right questions and adjust the answers” (Posavec, 2026). Task constraints could also scaffold: secondary students barred from untaught programming constructs had to state them in prompts and check compliance (Jalon et al., 2024).

#### **4.7. Differences Between School and University Learners**

Only four K-12 studies met the inclusion criteria, so comparisons are tentative. In Hong Kong, secondary students’ prompting purposes were associated with school achievement level (Woo et al., 2025). The other three K-12 studies (Jalon et al., 2024; Kwon et al., 2026; Song et al., 2026) support Themes 1, 3 and 5, but none offers clear evidence for Theme 2: younger learners described what they did with prompts, rarely how prompting changed their thinking, and Woo et al. (2025) provided

a negative case (Section 4.3). Whether this reflects development or the shorter, teacher-led activities involved cannot yet be determined.

**Table 2. Summary of Review Findings and GRADE-CERQual Assessments**

| Review finding                           | Studies (Table 1)                   | Limitations                                                      | Coherence                                                     | Adequacy                                 | Relevance                                                       | Overall                                        |
|------------------------------------------|-------------------------------------|------------------------------------------------------------------|---------------------------------------------------------------|------------------------------------------|-----------------------------------------------------------------|------------------------------------------------|
| 1. Discovering the missing context       | 1, 2, 6–9, 11–15, 18, 20 (n = 13)   | Minor: reflexivity rarely reported; brief strands                | No/very minor: consistent across levels and countries         | None: rich data                          | Minor: mostly HE                                                | Moderate (limitations, relevance)              |
| 2. Prompting as externalised thinking    | 4, 9, 11, 12, 16*, 21* (n = 6)      | Minor                                                            | Minor: two negative cases*                                    | Moderate: few studies; relies on study 9 | Moderate: no supporting K-12 data; mainly multi-week HE courses | Moderate (adequacy, relevance); borderline low |
| 3. Calibrating trust through iteration   | 1, 2, 3, 13, 15, 17–20 (n = 9)      | Minor                                                            | Minor: two forms of checking; some uncritical acceptance (18) | None                                     | No/very minor: includes K-12                                    | Moderate (limitations, coherence)              |
| 4. Delegation, authorship and skill loss | 5, 7, 9, 12, 14, 15, 17 (n = 7)     | Minor: sample inconsistency (5)                                  | Minor: ethics not always tied to prompting                    | Minor: some thin data                    | Moderate: no K-12 data                                          | Moderate (relevance)                           |
| 5. Scaffolds that make a difference      | 1, 5, 9, 10, 12, 14, 17, 20 (n = 8) | Moderate: self-evaluated interventions; sample inconsistency (5) | Minor: effects varied by type of support                      | None                                     | Minor                                                           | Moderate (limitations)                         |

Note. Numbers refer to Table 1; HE = higher education; \*negative case. Findings start at high confidence and are downgraded through a holistic judgement across the four components (Lewin et al., 2018); brackets list the components that drove downgrading; ratings were agreed by the author and the second coder.

## 5. DISCUSSION

### 5.1. Learning to Prompt as a Regulation Cycle

Interpreted through the self-regulated learning lens, learning to prompt, as students described it, resembles a regulation cycle rather than an accumulation of techniques.

Clarifying what one wants (Theme 2) corresponds to goal setting in the forethought phase of Zimmerman's (2002) model, and supplying context and revising requests (Theme 1) to strategy use and self-monitoring in the performance phase. Checking outputs against standards of accuracy and relevance (Theme 3) corresponds to self-evaluation in the self-reflection phase, and the personal rules about delegation that students built from experience (Theme 4) resemble adaptive reactions that feed back into forethought for the next task. Andewi et al. (2025) described a related scaffolding inversion, in which the student takes over the macro-level direction of the task; the gap between re-asking the model and consulting an independent source (Theme 3) suggests that students take over direction before evaluation.

Even if prompting skills matter less as models improve (see Lee & Palmer, 2025), the educational value of prompting lies in articulating goals, evaluating outputs and drawing boundaries; a model that infers what the learner wants removes the friction that prompted articulation, a possible route to metacognitive laziness (Fan et al., 2025) and lost epistemic agency (L. Yan et al., 2025).

## **5.2. Implications for Instructional Design and Teacher Education**

First, prompting instruction should follow the regulation cycle rather than lists of techniques; goal statements, revision logs and brief evaluations of outputs make the cycle teachable. A product-only study points the same way: prompting instruction improved the form of pre-service teachers' STEM activities more readily than their depth (Korkmaz & Tarım, 2025). Second, verification should be taught as part of prompting, including the difference between asking the model again and consulting an independent source. Third, extending Theme 5, teacher educators could model prompting aloud, including failed attempts, although

this was not tested directly. In K-12 settings, pupils may need help deciding why they are prompting before learning how (Woo et al., 2025).

### **5.3. Limitations**

The search was purposive rather than exhaustive, so paywalled studies are likely under-represented, and the platforms' ranking algorithms may have shaped which records surfaced first. Title and abstract screening and most data extraction were carried out by a single reviewer, and the protocol was not prospectively registered. Three included studies contributed mainly brief open-ended responses (Jalon et al., 2024; Posavec, 2026; H. Yang, 2025); none generated a theme on its own. A study assessed in the final round added a concept absent from earlier studies, the language of prompting among second-language learners (Andewi et al., 2025), so conceptual saturation was approached rather than fully reached for this group. Only English and Turkish studies were considered, and K-12 evidence rests on four studies.

## **6. CONCLUSION**

The synthesis of twenty-one qualitative studies suggests that, from the learner's perspective, learning to prompt is closer to learning to regulate one's own thinking than to learning a technical skill: students discovered that the system lacked their context, found that precise requests forced them to clarify their goals, learned to treat outputs as drafts to be checked, and set limits on what they would delegate. This applies most clearly to university students in multi-week courses, since Theme 2, central to the claim, has moderate confidence and no supporting K-12 evidence. Teacher modelling and structure helped.

Longitudinal studies are needed, particularly at school level. For educators, the aim is not efficient users of a tool that will change within a year, but learners who know what to ask, can judge what they receive and remain the authors of their own learning.

## REFERENCES

- Andewi, W., Waziana, W., Wibisono, D., Putra, K. A., Hastomo, T., & Oktarin, I. B. (2025). From prompting to proficiency: A mixed-methods analysis of prompting with ChatGPT versus lecturer interaction in an EFL classroom. *Journal of Studies in the English Language*, 20(2), 210–238. doi:10.64731/jsel.v20i2.282318
- Aruleba, K., Sanusi, I. T., Obaido, G., Ogbuokiri, B., & Mienye, I. D. (2025). Beyond the prompt: Student strategies, ethical reflections, and learning with ChatGPT in computer science. *Technology, Knowledge and Learning*. Advance online publication. doi:10.1007/s10758-025-09899-7
- Bae, H., Hur, J., Park, J., Choi, G. W., & Moon, J. (2024). Pre-service teachers' dual perspectives on generative AI: Benefits, challenges, and integration into their teaching and learning. *Online Learning*, 28(3), 131–156. doi:10.24059/olj.v28i3.4543
- Çiftci, Z., Öz, T., & Çiftci, O. (2025). ChatGPT'nin matematik görevlerinde kullanımı ve öğretmen adaylarının deneyimleri [The use of ChatGPT in mathematics tasks and pre-service teachers' experiences]. *İZÜ Eğitim Dergisi*. Retrieved from <https://dergipark.org.tr/en/pub/izujed/article/1793325>
- Cooke, A., Smith, D., & Booth, A. (2012). Beyond PICO: The SPIDER tool for qualitative evidence synthesis. *Qualitative Health Research*, 22(10), 1435–1443. doi:10.1177/1049732312452938
- Critical Appraisal Skills Programme. (2018). *CASP qualitative checklist*. Retrieved from <https://casp-uk.net/casp-tools-checklists/>

- Du, S., Wang, S., Yan, F., Gong, D., Jiang, Q., & Lin, J. (2026). A meta-synthesis of nursing students' experiences with generative artificial intelligence-assisted learning. *Frontiers in Medicine*, 13, 1874158. doi:10.3389/fmed.2026.1874158
- Fan, Y., Tang, L., Le, H., Shen, K., Tan, S., Zhao, Y., . . . Gašević, D. (2025). Beware of metacognitive laziness: Effects of generative artificial intelligence on learning motivation, processes, and performance. *British Journal of Educational Technology*, 56(2), 489–530. doi:10.1111/bjet.13544
- Flemming, K., & Noyes, J. (2021). Qualitative evidence synthesis: Where are we at? *International Journal of Qualitative Methods*, 20, 1–13. doi:10.1177/1609406921993276
- Hawk, H., Coriasco, M., & Jones, J. R. (2025). Generative artificial intelligence: A reflexive thematic analysis of nursing students' perceptions following a guided learning activity. *Nurse Educator*, 50(1), 18–22. doi:10.1097/NNE.0000000000001736
- Hwang, Y., Lee, J. H., & Shin, D. (2023). *What is prompt literacy? An exploratory study of language learners' development of new literacy skill using generative AI* (arXiv:2311.05373). Retrieved from <https://arxiv.org/abs/2311.05373>
- Hwang, Y., Lee, J. H., & Shin, D. (2025). Exploring prompt iteration process through generative AI image creation in EFL vocabulary learning. *International Review of Applied Linguistics in Language Teaching*. Advance online publication. doi:10.1515/iral-2025-0089

- Jahan, F., & Akbar, H. (2026). Exploring ChatGPT in EFL classrooms: Insights into engagement, proficiency and ethics. *Discover Education*, 5, 174. doi:10.1007/s44217-026-01158-w
- Jalon, J. B., Jr., Chua, G. A., & Torres, M. L. (2024). ChatGPT as a learning assistant: Its impact on students learning and experiences. *International Journal of Education in Mathematics, Science, and Technology*, 12(6), 1603–1619. doi:10.46328/ijemst.4471
- Kim, J., Yu, S., Lee, S.-S., & Detrick, R. (2026). Students' prompt patterns and its effects in AI-assisted academic writing: Focusing on students' level of AI literacy. *Journal of Research on Technology in Education*, 58(3), 638–655. doi:10.1080/15391523.2025.2456043
- Knoth, N., Tolzin, A., Janson, A., & Leimeister, J. M. (2024). AI literacy and its implications for prompt engineering strategies. *Computers and Education: Artificial Intelligence*, 6, 100225. doi:10.1016/j.caeai.2024.100225
- Korkmaz, N., & Tarım, K. (2025). Investigating the effects of prompt engineering in STEM activities designed with ChatGPT-4o. *Çukurova Üniversitesi Eğitim Fakültesi Dergisi*, 54(3), 1247–1298. doi:10.14812/cuefd.1673406
- Kwon, K., Yang, S., Kale, U., Kim, K., & Park, J. (2026). Generative AI in high school English career preparation units: Student interactions, perceptions, and ethical concerns. *Computers and Education: Artificial Intelligence*, 10, 100588. doi:10.1016/j.caeai.2026.100588
- Lee, D., & Palmer, E. (2025). Prompt engineering in higher education: A systematic review to help inform curricula. *International Journal of Educational Technology in*

*Higher Education*, 22(1). doi:10.1186/s41239-025-00503-7

- Lewin, S., Booth, A., Glenton, C., Munthe-Kaas, H., Rashidian, A., Wainwright, M., . . . Noyes, J. (2018). Applying GRADE-CERQual to qualitative evidence synthesis findings: Introduction to the series. *Implementation Science*, 13(Suppl. 1), 2. doi:10.1186/s13012-017-0688-3
- Nguyen, S., Babe, H. M., Zi, Y., Guha, A., Anderson, C. J., & Feldman, M. Q. (2024). How beginning programmers and code LLMs (mis)read each other. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24)*. New York, NY: ACM. doi:10.1145/3613904.3642706
- Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., . . . Moher, D. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ*, 372, n71. doi:10.1136/bmj.n71
- Posavec, K. (2026). Leveraging generative AI in support of self-directed learning: Insights from student prompting practices. *International Journal of Instruction*, 19(2), 281–298. doi:10.29333/iji.2026.19215a
- Shen, M., Shen, Y., Liu, F., & Jin, J. (2025). Prompts, privacy, and personalized learning: Integrating AI into nursing education – a qualitative study. *BMC Nursing*, 24(1), 470. doi:10.1186/s12912-025-03115-8
- Song, Y., Kim, J., Xing, W., Liu, Z., Li, C., & Oh, H. (2026). Elementary school students' and teachers' perceptions toward creative mathematical writing with generative AI. *Journal of Research on Technology in Education*, 58(3), 615–637. doi:10.1080/15391523.2025.2455057

- Thomas, J., & Harden, A. (2008). Methods for the thematic synthesis of qualitative research in systematic reviews. *BMC Medical Research Methodology*, 8, 45. doi:10.1186/1471-2288-8-45
- Tong, A., Flemming, K., McInnes, E., Oliver, S., & Craig, J. (2012). Enhancing transparency in reporting the synthesis of qualitative research: ENTREQ. *BMC Medical Research Methodology*, 12, 181. doi:10.1186/1471-2288-12-181
- Tümen Akyıldız, S. (2026). Prompt engineering and creative pedagogical thinking: Insights from a qualitative study with teacher candidates. *Thinking Skills and Creativity*, 60, 102105. doi:10.1016/j.tsc.2025.102105
- Wang, S., Niu, S., Zhang, X., Cao, Q., Li, Q., Yu, J., . . . Zeng, L. (2026). The experience of nursing students using generative artificial intelligence: A qualitative meta-synthesis. *BMC Nursing*, 25, 443. doi:10.1186/s12912-026-04663-3
- Wen, W., & Wen, H. (2024). Bridging generative AI technology and teacher education: Understanding preservice teachers' processes of unit design with ChatGPT. *Contemporary Issues in Technology and Teacher Education*, 24(4). Retrieved from <https://citejournal.org>
- Woo, D. J., Guo, K., & Susanto, H. (2025). Exploring EFL students' prompt engineering in human-AI story writing: An activity theory perspective. *Interactive Learning Environments*, 33(1), 863–882. doi:10.1080/10494820.2024.2361381
- Woo, D. J., Wang, D., Yung, T., & Guo, K. (2026). Effects of a prompt engineering intervention on undergraduate students' AI self-efficacy, AI knowledge and prompt

- engineering ability: A mixed methods study. *British Educational Research Journal*. Advance online publication. doi:10.1002/berj.70087
- Yan, L., Pammer-Schindler, V., Mills, C., Nguyen, A., & Gašević, D. (2025). Beyond efficiency: Empirical insights on generative AI's impact on cognition, metacognition and epistemic agency in learning. *British Journal of Educational Technology*, 56(5), 1675–1685. doi:10.1111/bjet.70000
- Yan, W., Nakajima, T., & Sawada, R. (2025). Beyond tool use: Tracking the evolution of generative AI literacy among university students through a process-oriented investigation. *Computers and Education: Artificial Intelligence*, 9, 100465. doi:10.1016/j.caeai.2025.100465
- Yang, H. (2025). EFL student engagement with ChatGPT in college reading classes via prompts and perceptions. *Korean Journal of English Language and Linguistics*, 25, 686–706. doi:10.15738/kjell.25..202505.686
- Yang, M., Jiang, S., Li, B., Herman, K., Luo, T., Moots, S. C., & Lovett, N. (2025). Analysing nontraditional students' ChatGPT interaction, engagement, self-efficacy and performance: A mixed-methods approach. *British Journal of Educational Technology*, 56, 1973–2000. doi:10.1111/bjet.13588
- Zimmerman, B. J. (2002). Becoming a self-regulated learner: An overview. *Theory Into Practice*, 41(2), 64–70. doi:10.1207/s15430421tip4102\_2

# **DEEP LEARNING VERSUS STATISTICAL BENCHMARKS IN FORECASTING THE GLOBAL PREVALENCE OF MENTAL DISORDERS TO 2030**

**Erol ÖZÇEKİ<sup>1</sup>**

## **1. INTRODUCTION**

Mental disorders are among the leading causes of health loss worldwide. According to the Global Burden of Disease (GBD) 2019 study, they remained among the ten leading causes of burden. There was no evidence of a global reduction since 1990 (GBD 2019 Mental Disorders Collaborators, 2022). Depressive and anxiety disorders are the two most common of these disorders. Both rose further after the COVID-19 pandemic (COVID-19 Mental Disorders Collaborators, 2021). The World Health Organization's Comprehensive Mental Health Action Plan 2013-2030 sets service coverage targets for the end of this decade (World Health Organization [WHO], 2021, 2022). Planning for these targets requires an expectation of how many people will live with a mental disorder in the coming years. Forecasting prevalence is therefore a practical public-health question and not only a statistical exercise.

Deep-learning models are increasingly used to answer this kind of question. Long short-term memory (LSTM) networks, gated recurrent units (GRU), convolutional-recurrent hybrids and attention-based architectures have been applied to epidemiological time series, often with very high reported

---

<sup>1</sup> Asst. Prof. Dr., Balıkesir University, Artificial Intelligence Operator Program, Balıkesir, Türkiye, ORCID 0000-0002-1896-6853.

accuracy. Two findings from the forecasting literature call for caution. First, large forecasting studies and competitions show that simple statistical methods and their combinations are hard to beat. In these studies, pure machine-learning methods often performed poorly, and the more successful neural approaches learned across many series at once (Makridakis, Spiliotis, & Assimakopoulos, 2018, 2020; Smyl, 2020). Second, high reported accuracy is often an artefact of the evaluation design. Random or overlapping data splits can make an ordinary forecast look excellent. Metrics such as the coefficient of determination ( $R^2$ ) have the same effect, because they are dominated by differences between units. So does turning a regression problem into an artificial classification task (Kapoor & Narayanan, 2023; Tashman, 2000).

Prevalence series for mental disorders put both findings to a demanding test. The series are short, with thirty annual values per country in GBD 2019. They change slowly, and they are themselves the output of a statistical model. Between 1990 and 2019, the age-standardised world prevalence of depressive disorders changed from 3.49% to 3.44% (-1.3%). Within a country, the median absolute change from one year to the next was only 0.008 percentage points. A model that simply repeats the last observed value therefore already has a very small absolute error. The relevant question is not whether a deep network is accurate. It is whether the network is more accurate than such simple benchmarks on years it has never seen.

This chapter addresses that question with age-standardised GBD 2019 estimates for five disorder groups in 204 countries. Five deep-learning architectures are compared with five statistical benchmarks and with an equal-weight combination of methods. All methods are evaluated strictly out of sample, with scale-free error measures and country-level significance tests. An eleven-year backtest then provides

prediction intervals for the 2030 projections. The chapter thus offers transparent projections and a worked example of how such forecasts should be evaluated before they inform policy.

## **2. BACKGROUND**

### **2.1. Measuring and Projecting the Prevalence of Mental Disorders**

The GBD study produces consistent estimates of prevalence, incidence and burden for hundreds of conditions, for every country and year since 1990 (GBD 2019 Diseases and Injuries Collaborators, 2020). For mental disorders, prevalence is estimated with a Bayesian meta-regression. This model pools population surveys and adjusts them for differences in case definitions and study designs. Where surveys are missing, values are borrowed from neighbouring years, countries and covariates (Ferrari et al., 2013; GBD 2019 Mental Disorders Collaborators, 2022). The resulting series are smooth by construction. They can nevertheless shift when new survey data enter the model, and they may underestimate the true burden of mental illness (Vigo, Thornicroft, & Atun, 2016).

Rising treatment has not reduced prevalence so far. In four English-speaking industrialised countries, the prevalence of common mental disorders did not fall despite substantial increases in treatment (Jorm, Patten, Brugha, & Mojtabai, 2017). The GBD collaboration itself forecasts mortality to 2040 and disease burden to 2050 from drivers such as income, education and fertility (Foreman et al., 2018; GBD 2021 Forecasting Collaborators, 2024). The present chapter asks how well purely data-driven models extrapolate the prevalence series themselves.

## **2.2. Deep Learning for Time-Series Forecasting**

The LSTM network introduced gated memory cells. These cells let recurrent networks learn long-range dependencies without vanishing gradients (Hochreiter & Schmidhuber, 1997). The GRU simplifies this gating with fewer parameters (Cho et al., 2014). Bidirectional recurrent networks read the input sequence in both directions (Schuster & Paliwal, 1997). In the CNN-LSTM model of Kim and Cho (2019), convolutional layers placed before a recurrent layer extract local patterns. Attention mechanisms, in turn, let a model weight the time steps that are most informative for the prediction (Bahdanau, Cho, & Bengio, 2015; Vaswani et al., 2017).

Reviews of the field stress that these architectures work best as global models. Such models are trained across many related series, each normalised appropriately, instead of being fitted to one short series (Hewamalage, Bergmeir, & Bandara, 2021; Lim & Zohren, 2021; Salinas, Flunkert, Gasthaus, & Januschowski, 2020). The winning method of the M4 competition, for example, combined exponential smoothing with a recurrent network trained across series (Smyl, 2020). On small data sets, however, the picture is different. On 1,045 monthly M3 series, the machine-learning methods examined were less accurate than simple statistical methods at all forecast horizons (Makridakis et al., 2018). A single linear layer can also outperform Transformer models on standard long-horizon benchmarks (Zeng, Chen, Zhang, & Xu, 2023). Forecast combinations, by contrast, have been among the most reliable strategies since the work of Bates and Granger (1969) (Makridakis et al., 2020; Wang, Hyndman, Li, & Kang, 2023).

## **2.3. Evaluating Forecasts Correctly**

A forecast can only be judged on observations that were not available when it was made. Out-of-sample evaluation from

a fixed or rolling forecast origin is therefore the standard design (Tashman, 2000). When cross-validation is used for time series, blocked schemes that respect the time order are recommended (Bergmeir & Benítez, 2012). Test data that precede the training data let information from the future leak into the model (Kapoor & Narayanan, 2023). Errors should also be comparable across series of different scales. For this purpose, Hyndman and Koehler (2006) recommended the mean absolute scaled error (MASE). Finally, differences between methods should be tested across the units of analysis, here countries, and not inferred from a single summary number. The design used in this chapter follows these principles.

### **3. DATA AND METHODS**

#### **3.1. Data**

The data are GBD 2019 estimates of age-standardised prevalence for schizophrenia, depressive, anxiety, bipolar and eating disorders, as a percentage of the population for both sexes (Institute for Health Metrics and Evaluation [IHME], 2020). They were obtained in the version processed by Our World in Data (Dattani, Rodés-Guirao, Ritchie, & Roser, 2023), as distributed in the Kaggle Mental Health dataset (Momeni, 2023). The file covers 1990-2019 for 204 countries and territories, the world and several regional and income groupings. The 204 countries form the modelling panel of  $204 \times 30 \times 5 = 30,600$  values, with no missing data. The world and the four World Bank income groups were not used for training. They were forecast with the trained models in the same way as a country. Table 1 summarises the data.

Figure 1 shows the main patterns. Age-standardised world prevalence was almost flat for schizophrenia, anxiety and bipolar disorders. It rose steadily for eating disorders. For

depressive disorders it increased in the 1990s and fell between 2005 and 2010. This decline is concentrated in the lower-middle-income group. It is driven mainly by India, whose estimate fell from 4.50% in 2005 to 3.74% in 2010, with smaller drops in Brazil, Uganda and Nigeria. Such breaks may reflect changes in the underlying data as much as in population health, and no extrapolative method can anticipate them.

**Table 1. Age-Standardised Prevalence (%) of the Five Disorder Groups**

| Disorder      | World 1990 | World 2019 | Change (%) | Countries 2019 (min / median / max) | Median  annual change |
|---------------|------------|------------|------------|-------------------------------------|-----------------------|
| Schizophrenia | 0.290      | 0.287      | -0.9       | 0.19 / 0.28 / 0.43                  | 0.0002                |
| Depressive    | 3.486      | 3.440      | -1.3       | 1.60 / 3.64 / 6.58                  | 0.0079                |
| Anxiety       | 3.792      | 3.780      | -0.3       | 2.03 / 4.02 / 8.03                  | 0.0041                |
| Bipolar       | 0.490      | 0.490      | -0.1       | 0.18 / 0.58 / 1.51                  | 0.0000                |
| Eating        | 0.151      | 0.174      | 15.6       | 0.05 / 0.16 / 1.03                  | 0.0011                |

*Note.* Median |annual change| is the median absolute year-to-year change within the 204 countries in 1990-2019, in percentage points. Data from GBD 2019 via Our World in Data, own calculations.

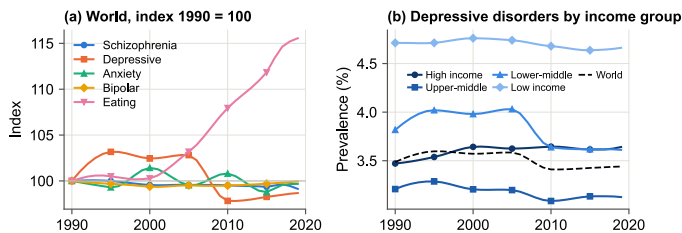


**Figure 1. Age-standardised prevalence, 1990-2019. Panel (a) shows the world, indexed to 1990 = 100. Panel (b) shows depressive disorders by income group (%). Data from GBD 2019 via Our World in Data.**

### 3.2. Forecasting Task and Evaluation Design

All learning-based methods are global models. A single model is trained on the series of all 204 countries. For every

country, it predicts the next-year change of all five disorders at once. The input window covers 5 years, and each year contributes ten features. Five features describe the shape of the recent trajectory. They are the logarithm of each disorder's prevalence relative to its value in the last year of the window. The other five describe the level, as the standardised logarithm of each prevalence. The target is the vector of next-year log-changes, standardised with the training mean and standard deviation. This normalisation lets one network learn from countries whose prevalence differs by a factor of three or more (Hewamalage et al., 2021; Salinas et al., 2020). Forecasts for several years ahead are produced recursively. Each prediction is appended to the window and fed back as input.

Figure 2 shows three settings that apply the same procedure from different forecast origins. In the main test, the models are trained on target years up to 2009. The years 2010-2013 are used for early stopping and for tuning the ridge penalty. Forecasts are then made from 2013 for 2014-2019, that is, for horizons 1-6. The long-horizon backtest repeats this procedure with origin 2008. Training targets end in 2004, validation covers 2005-2008, and the forecasts cover 2009-2019 (horizons 1-11). The errors of this backtest give the prediction intervals for the projections. In the projection setting, the models are trained on targets up to 2016, validated on 2017-2019 and used to forecast 2020-2030. In no setting is any observation after the forecast origin used for training, early stopping or model selection.

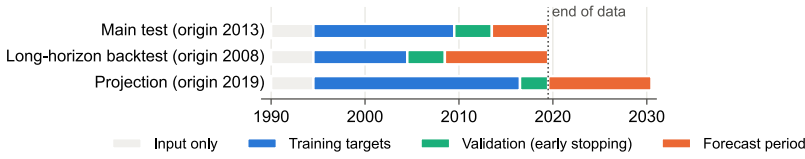


**Figure 2. Evaluation design. Grey bars mark years used only as model inputs (the first five-year window). Blue bars mark target years used for training, and green bars target years used for early stopping and tuning. Orange bars mark the forecast period.**

### 3.3. Deep-Learning Models

The five architectures follow common designs in the applied literature. LSTM has two LSTM layers with 64 and 32 units, each followed by dropout (0.2), and a linear output layer with five units. BiLSTM uses the same design with bidirectional layers. CNN-LSTM places two one-dimensional convolutions (64 and 32 filters, kernel size 3) with max-pooling before two LSTM layers. GRU has two GRU layers with 64 and 32 units. StackedLSTM-ATT adds a two-head self-attention block with a residual connection to two LSTM layers of 64 units. The networks have between 24,165 (GRU) and 79,941 (BiLSTM) trainable parameters.

All networks were trained with the Adam optimiser at a learning rate of 0.001 (Kingma & Ba, 2015). The loss was the mean squared error of the standardised target, with a batch size of 64 and at most 300 epochs. Training stopped after 25 epochs without improvement of the validation loss, and the best weights were restored. The learning rate was halved after 10 epochs without improvement. Dropout served as regularisation (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). In the main test, the best validation loss was reached after 9-12 epochs on average. One training run took about half a minute on a laptop CPU. Neural networks depend on their random initialisation, so every architecture was trained with five

random seeds. Single-seed results are reported as mean  $\pm$  standard deviation. The average of the five seeds, called a seed ensemble, is used in all further comparisons.

### **3.4. Statistical Benchmarks and Combination**

Five statistical benchmarks were fitted. The naïve method repeats the last observed value. The random walk with drift extends the average annual change between 1990 and the forecast origin. Damped Holt exponential smoothing (Gardner & McKenzie, 1985) and ARIMA(p,1,q) models with drift (Box, Jenkins, Reinsel, & Ljung, 2015) were fitted to the logarithm of each country-disorder series separately, which gives 1,020 series. For ARIMA, the orders  $p \leq 2$  and  $q \leq 1$  were chosen with the corrected Akaike information criterion. The fifth benchmark is a global ridge regression (Hoerl & Kennard, 1970). It uses exactly the same input features and recursive scheme as the networks. Its penalty was chosen on the validation years from  $10^{-4}$  to  $10^2$ . Very weak penalties of  $10^{-4}$  or  $10^{-3}$  were selected in every setting, so the model is practically an ordinary least-squares autoregression. It differs from the networks only in being linear.

The literature favours forecast combinations (Bates & Granger, 1969; Wang et al., 2023). An equal-weight combination was therefore specified before any test result was examined. It is the arithmetic mean of the drift, damped Holt, ARIMA and ridge forecasts and of the five deep-learning seed ensembles. The combination is also used for the 2030 projections. In this way, the projections do not depend on choosing the method that happened to score best in the test years.

### **3.5. Accuracy Measures, Tests and Prediction Intervals**

Accuracy was measured with four measures. The mean absolute error (MAE) and the root mean squared error (RMSE) are expressed in percentage points of prevalence. The mean absolute percentage error (MAPE) is expressed in percent. The fourth measure is the MASE (Hyndman & Koehler, 2006). The MASE divides each absolute error by the mean absolute year-to-year change of the same country and disorder between 1990 and the forecast origin. It is therefore comparable across countries and disorders. A value of 1 equals the typical one-step error of the naïve method in the training period. All measures were averaged over the 204 countries and the forecast horizons. The pooled  $R^2$  is reported only for comparison with applied studies.

The Wilcoxon signed-rank test (Wilcoxon, 1945) was used to check whether differences between methods were systematic. For each country, the MASE for depressive disorders was averaged over horizons 1-6. These country values were compared between each method and two references, the naïve method and ARIMA. The p values were adjusted for the ten comparisons per reference with Holm's (1979) procedure.

Prediction intervals were obtained empirically from the long-horizon backtest. For each disorder and each horizon  $h = 1, \dots, 11$ , the log ratio between observed and forecast values of the combination was computed for the 204 countries. The 5th and 95th percentiles of this ratio were then applied to the projection for the same horizon, which gives a 90% interval. The intervals were validated on the independent main-test period 2014-2019. Because they come from country-level errors, they are conservative for the smoother world and income-group series. The analysis was carried out in Python 3.10 with TensorFlow

2.19, statsmodels 0.14 and scikit-learn 1.6. All code and result files are available from the author.

## **4. RESULTS**

### **3.5. Accuracy for Depressive Disorders in the Main Test**

Table 2 reports the accuracy of all methods for depressive disorders in 2014-2019. In absolute terms, every method was accurate. The MAE ranged from 0.025 to 0.048 percentage points, and the MAPE from 0.65% to 1.31%. The pooled  $R^2$  exceeded 0.990 for every method, including the naïve forecast ( $R^2 = 0.996$ ). Such measures therefore cannot discriminate between methods on this kind of data.

The scaled measures are more informative. The naïve method achieved the lowest mean MASE (2.08). It was followed by ridge regression (2.21), ARIMA (2.27), damped Holt (2.27) and the combination (2.30). Ridge regression had the lowest MAE, RMSE, MAPE and median MASE (1.46). All five deep-learning seed ensembles were less accurate than these methods, with a mean MASE between 2.79 and 3.66. Single seeds were worse still (3.23-4.12). The spread between seeds, with a standard deviation of up to 0.74, was larger than most differences between architectures. GRU was the most accurate network and StackedLSTM-ATT the least accurate. One year ahead, damped Holt, ARIMA and ridge regression were far better than the naïve forecast, with a MASE of about 0.23-0.30 against 0.90. By the sixth year, however, all of them had fallen behind it.

**Table 2. Forecast Accuracy for Depressive Disorders, Main Test (2014-2019, 204 Countries)**

| Method                   | MAE    | RMSE   | MAPE | MASE | MdMASE | h =<br>1 | h =<br>6 | Single<br>seeds |
|--------------------------|--------|--------|------|------|--------|----------|----------|-----------------|
| Naïve (last value)       | 0.0309 | 0.0591 | 0.79 | 2.08 | 1.60   | 0.90     | 4.04     | -               |
| Random walk with drift   | 0.0479 | 0.0779 | 1.31 | 3.25 | 3.16   | 1.12     | 6.23     | -               |
| Damped Holt              | 0.0380 | 0.0883 | 0.98 | 2.27 | 1.67   | 0.23     | 4.54     | -               |
| ARIMA (AICc)             | 0.0335 | 0.0642 | 0.90 | 2.27 | 1.87   | 0.24     | 5.31     | -               |
| Ridge regression         | 0.0252 | 0.0471 | 0.65 | 2.21 | 1.46   | 0.30     | 5.03     | -               |
| LSTM                     | 0.0351 | 0.0572 | 0.96 | 3.03 | 2.05   | 0.78     | 6.00     | 3.62 ±<br>0.74  |
| BiLSTM                   | 0.0372 | 0.0595 | 1.02 | 3.59 | 2.37   | 1.05     | 7.02     | 3.91 ±<br>0.22  |
| CNN-LSTM                 | 0.0351 | 0.0616 | 0.95 | 2.87 | 2.16   | 0.77     | 5.66     | 3.84 ±<br>0.18  |
| GRU                      | 0.0326 | 0.0548 | 0.88 | 2.79 | 1.90   | 0.74     | 6.02     | 3.23 ±<br>0.55  |
| StackedLSTM-ATT          | 0.0383 | 0.0592 | 1.02 | 3.66 | 2.90   | 1.21     | 7.13     | 4.12 ±<br>0.20  |
| Equal-weight combination | 0.0286 | 0.0512 | 0.77 | 2.30 | 1.78   | 0.59     | 5.14     | -               |

**Note:** MAE and RMSE are in percentage points and MAPE in percent. MASE is the mean absolute scaled error averaged over countries and horizons 1-6. MdMASE is the median of the country MASE values. The columns h = 1 and h = 6 give the MASE at horizons 1 and 6. Deep-learning rows show the five-seed ensemble, and the last column gives the mean ± SD of the MASE of the five single seeds. Lower values are better for all measures.

Table 3 tests these differences at the country level. After Holm correction, no method was significantly more accurate than the naïve forecast. Ridge regression was better in 112 of 204 countries, which is not a significant majority (adjusted p = .693). All five deep-learning ensembles were significantly less accurate than the naïve method, and so was the random walk with drift. Ridge regression was also significantly better than ARIMA, with a lower MASE in 125 countries (adjusted p = .031). LSTM, BiLSTM and StackedLSTM-ATT were significantly worse than ARIMA, while CNN-LSTM and GRU did not differ significantly from it.

**Table 3. Country-Level Comparison of MASE for Depressive Disorders (Wilcoxon Signed-Rank Tests)**

| Method                   | Better<br>(n) | Median<br>$\Delta$ | p<br>(Holm) | Better<br>(n) | Median<br>$\Delta$ | p<br>(Holm) |
|--------------------------|---------------|--------------------|-------------|---------------|--------------------|-------------|
| Random walk with drift   | 50            | 1.32               | < .001      | 46            | 1.01               | < .001      |
| Damped Holt              | 92            | 0.12               | .441        | 116           | -0.21              | .344        |
| ARIMA (AICc)             | 94            | 0.07               | .441        | -             | -                  | -           |
| Ridge regression         | 112           | -0.12              | .693        | 125           | -0.33              | .031        |
| LSTM                     | 83            | 0.52               | < .001      | 84            | 0.33               | .012        |
| BiLSTM                   | 83            | 0.48               | < .001      | 83            | 0.36               | < .001      |
| CNN-LSTM                 | 90            | 0.25               | .005        | 90            | 0.18               | .136        |
| GRU                      | 86            | 0.28               | .013        | 101           | 0.02               | .344        |
| StackedLSTM-ATT          | 49            | 0.83               | < .001      | 70            | 0.93               | < .001      |
| Equal-weight combination | 95            | 0.06               | .693        | 106           | -0.08              | .544        |

*Note.* Columns 2-4 compare each method with the naïve method, and columns 5-7 compare it with ARIMA. Better (n) is the number of the 204 countries in which the method had the lower MASE. Median  $\Delta$  is the median country-level difference in MASE (method minus reference). Negative values favour the method. The p values were adjusted with Holm's procedure within each reference.

#### **4.2. Forecast Horizon and the Long-Horizon Backtest**

Figure 3 shows how the error grows with the forecast horizon. In both evaluations, damped Holt, ARIMA and ridge regression were much more accurate than the naïve method one and two years ahead, because they continued short-run movements. The networks did so only in the long-horizon backtest. Beyond about five years, the order reversed. The long-horizon backtest starts in 2008, in the middle of the decline in depressive prevalence shown in Figure 1. The methods extrapolated a fall that ended around 2010, and their errors grew rapidly. At eleven years, the MASE was 6.68 for the naïve method and 8.16 for ridge regression. It was 11.30 for the

combination, 12.62 for ARIMA and between 9.82 and 14.75 for the networks. Among the networks, StackedLSTM-ATT degraded least at long horizons. It was, however, the weakest network in the main test, so no architecture was consistently superior.

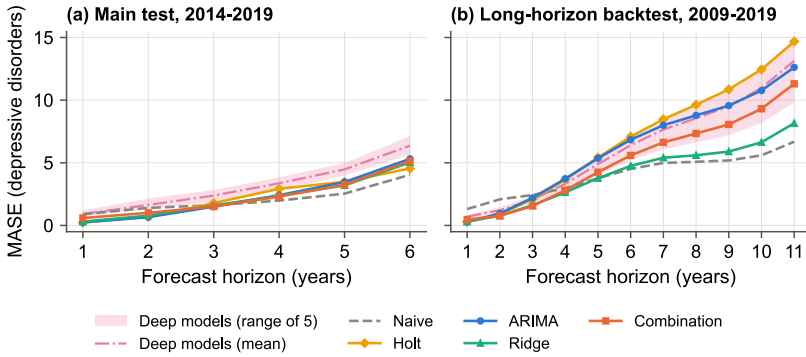


**Figure 3. Mean absolute scaled error (MASE) for depressive disorders by forecast horizon, averaged over 204 countries. Panel (a) shows the main test (origin 2013) and panel (b) the long-horizon backtest (origin 2008). The shaded band spans the five deep-learning seed ensembles. Values above 1 mean that the error exceeds the typical one-year change of the series.**

#### 4.3. Other Disorders and Aggregated Series

The pattern was the same for the other disorder groups. By the median country MASE, ridge regression was the most accurate method for schizophrenia (1.54), anxiety (2.11) and eating disorders (1.07). Damped Holt was the most accurate for bipolar disorder (0.90). The best network reached 1.77, 3.26, 1.29 and 5.04 for the same four disorders. No network was therefore the best method for any disorder. The naïve method did well for depressive disorders but poorly for eating disorders (3.87), whose prevalence rose steadily. Bipolar disorder series are almost constant. All methods forecast them with a MAPE below 0.1%. Even a small drift, however, produces a large

scaled error, which explains the high MASE of the global models for this disorder.

The trained models were also applied to the world series, which they had not seen in training. Here, too, the statistical methods were more accurate. For world depressive prevalence in 2014-2019, the MAPE was 0.13% for ARIMA, 0.22% for damped Holt and 0.45% for the naïve method. For the networks it ranged from 1.35% to 1.92%. The dynamics the networks learned from single countries did not transfer well to a smooth average.

#### **4.4. Projections to 2030**

The empirical 90% prediction intervals from the long-horizon backtest covered 91.1% of the observed values in the independent main-test period. Coverage ranged from 87.2% to 94.4% across disorders and was 90.6% for depressive disorders. The intervals are therefore well calibrated at the country level. Table 4 and Figure 4 present the projections of the combination.

The projections point to stability rather than change in the age-standardised rates. World prevalence of depressive disorders is projected at 3.44% in 2030, with a 90% interval of 3.25-3.77%. This is practically identical to the 3.44% observed in 2019. The individual methods range from 3.38% to 3.59%. Anxiety, schizophrenia and bipolar disorders are likewise projected to change by less than 1%. The only clear movement is a continued rise in eating disorders, by 5.6% to 0.184%. At the country level, depressive prevalence is projected to rise in 130 countries and to fall in 74. The median change is 0.65%, with an interquartile range from -0.67% to 2.34% (Figure 4f).

One projection calls for caution. For anxiety disorders in high-income countries, the combination projects an increase of 13.2%. This increase extends a rise between 2017 and 2019. The individual methods range from 4.89%, which means no change,

to 6.09%. This spread is wider than the prediction interval. It signals uncertainty about the trend and is not evidence of an increase.

**Table 4. Projected Age-Standardised Prevalence (%) in 2030**

| Population          | Disorder      | 2019  | 2030  | 90% PI      | Change (%) | Range of methods |
|---------------------|---------------|-------|-------|-------------|------------|------------------|
| World               | Schizophrenia | 0.287 | 0.288 | 0.279-0.293 | +0.1       | 0.286-0.289      |
| World               | Depressive    | 3.44  | 3.44  | 3.25-3.77   | +0.0       | 3.38-3.59        |
| World               | Anxiety       | 3.78  | 3.76  | 3.56-3.93   | -0.4       | 3.54-3.82        |
| World               | Bipolar       | 0.490 | 0.491 | 0.486-0.492 | +0.2       | 0.490-0.494      |
| World               | Eating        | 0.174 | 0.184 | 0.160-0.200 | +5.6       | 0.174-0.190      |
| High-income         | Depressive    | 3.64  | 3.73  | 3.52-4.08   | +2.3       | 3.62-3.81        |
| High-income         | Anxiety       | 4.89  | 5.53  | 5.23-5.78   | +13.2      | 4.89-6.09        |
| Upper-middle-income | Depressive    | 3.13  | 3.11  | 2.93-3.40   | -0.6       | 2.99-3.28        |
| Upper-middle-income | Anxiety       | 3.93  | 3.97  | 3.75-4.14   | +0.8       | 3.83-4.07        |
| Lower-middle-income | Depressive    | 3.61  | 3.57  | 3.37-3.91   | -1.2       | 3.52-3.63        |
| Lower-middle-income | Anxiety       | 3.33  | 3.27  | 3.09-3.41   | -2.0       | 3.11-3.36        |
| Low-income          | Depressive    | 4.66  | 4.70  | 4.44-5.15   | +0.8       | 4.62-4.77        |
| Low-income          | Anxiety       | 3.68  | 3.70  | 3.50-3.86   | +0.4       | 3.64-3.74        |

**Note:** 2019 is the observed GBD estimate and 2030 the equal-weight combination. 90% PI is the empirical prediction interval from the long-horizon backtest. Range of methods gives the lowest and highest 2030 value of the ten individual methods.

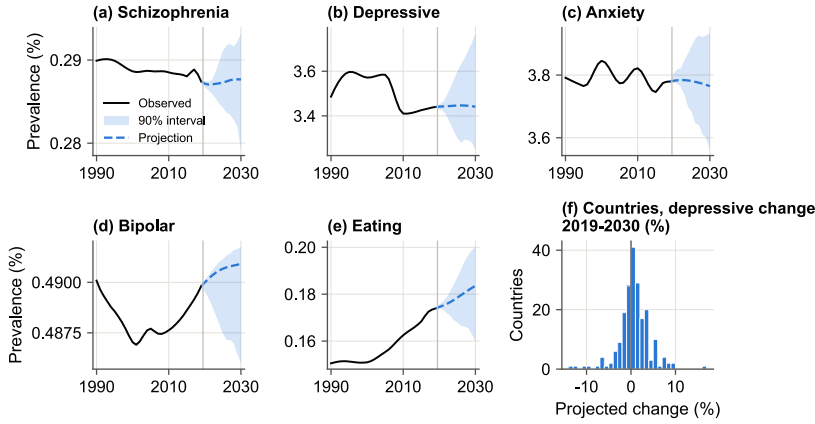


**Figure 4. Projections of world age-standardised prevalence to 2030. Dashed lines show the equal-weight combination and shaded areas the 90% prediction intervals. Panel (f) shows the distribution of projected changes in depressive prevalence across 204 countries. The vertical axes of panels (a)-(e) have different scales and do not start at zero.**

## 5. DISCUSSION

### 5.1. Why Deep Learning Did Not Win

The central result of this chapter is negative but robust. On thirty-year GBD prevalence series, five widely used deep-learning architectures were not more accurate than a naïve forecast. They were less accurate than simple linear and exponential-smoothing methods, in line with Makridakis et al. (2018) and Zeng et al. (2023). Three features of the data explain the result. First, the series are short and smooth. Most of their predictable information lies in the current level and the latest slope, which a linear model captures with a few parameters. Second, the series change mainly through occasional breaks, such as the 2005-2010 fall in depressive prevalence in several large countries, which past values cannot predict. Third, the

networks were unstable across random seeds, so a single training run may misrepresent an architecture.

Ridge regression used the same global training set, features and recursive scheme as the networks, and it was the most accurate method on most measures. Training across countries is therefore useful, but the benefit does not require nonlinearity. With richer data the balance may change (Lim & Zohren, 2021). For annual prevalence estimates, however, a network should only be preferred when it beats simple benchmarks in an out-of-sample test.

## **5.2. What "High Accuracy" Means**

These results also show why claims of "98% accuracy" or of an  $R^2$  close to one carry little information. Here, simply repeating the last observed value gives a pooled  $R^2$  of 0.996 and an "accuracy" ( $100 - \text{MAPE}$ ) above 99%. The reason is that prevalence differs much more between countries than within a country over a few years. An evaluation becomes informative only with out-of-sample forecasts, scale-free errors, simple benchmarks, several seeds and country-level tests (Hyndman & Koehler, 2006; Kapoor & Narayanan, 2023).

## **5.3. Implications of the Projections**

Without new shocks, age-standardised prevalence is projected to stay close to its 2019 level until 2030, except for a rise in eating disorders. Two qualifications apply. First, stable rates do not mean stable needs. The world population is growing and ageing. The number of people living with these disorders will therefore keep increasing even if the age-standardised rates do not (GBD 2019 Mental Disorders Collaborators, 2022; Patel et al., 2018). Second, the projections use data up to 2019 only. In 2020, the pandemic raised the number of cases by about 28% for major depressive disorder and by about 26% for anxiety disorders (COVID-19 Mental Disorders Collaborators, 2021).

The projections are therefore best read as a pre-pandemic baseline. In addition, prevalence has stayed stable despite expanding treatment (Jorm et al., 2017). The WHO targets for 2030 therefore depend on better coverage and quality of care (WHO, 2021).

#### **5.4. Limitations**

The GBD values are model-based estimates that are revised in every round of the study, so part of their movement reflects changes in data and methods. Each setting used a single forecast origin. More extensive hyperparameter tuning might improve the networks, but it would risk overfitting the validation years. No covariates such as income, education or population structure were used, unlike in the GBD forecasting models (GBD 2021 Forecasting Collaborators, 2024). The intervals exclude the uncertainty of the GBD estimates, and only age-standardised prevalence for both sexes was analysed.

### **6. CONCLUSION**

In this strictly out-of-sample comparison across 204 countries, no deep-learning model was more accurate than a naïve forecast. All five networks were significantly less accurate than it for depressive disorders. A linear ridge autoregression trained on the same data was the most accurate method on most measures. Trend-following methods helped at horizons of one or two years, but over a decade simple persistence was hard to beat. The equal-weight combination projects age-standardised world prevalence of depressive disorders at about 3.44% in 2030 (3.25-3.77%). This is essentially unchanged from 2019, while eating disorders continue to rise.

Two recommendations follow. Researchers should always report deep-learning forecasts of health indicators

against simple benchmarks and on years after the training data. Headline accuracies near 100% are a warning sign rather than a result. For policymakers, flat age-standardised prevalence, population growth and the pandemic mean that the need for mental-health services will not resolve itself by 2030.

## REFERENCES

- Bahdanau, D., Cho, K., & Bengio, Y. (2015, May). *Neural machine translation by jointly learning to align and translate*. Paper presented at the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA. Retrieved from <https://arxiv.org/abs/1409.0473>
- Bates, J. M., & Granger, C. W. J. (1969). The combination of forecasts. *OR*, 20(4), 451-468. <https://doi.org/10.2307/3008764>
- Bergmeir, C., & Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191, 192-213. <https://doi.org/10.1016/j.ins.2011.12.028>
- Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: Forecasting and control* (5th ed.). Hoboken, NJ: Wiley. Retrieved from <https://www.wiley.com/en-us/Time+Series+Analysis%3A+Forecasting+and+Control%2C+5th+Edition-p-9781118675021>
- Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1724-1734). Stroudsburg, PA: Association for Computational Linguistics. <https://doi.org/10.3115/v1/D14-1179>
- COVID-19 Mental Disorders Collaborators. (2021). Global prevalence and burden of depressive and anxiety

disorders in 204 countries and territories in 2020 due to the COVID-19 pandemic. *The Lancet*, 398(10312), 1700-1712. [https://doi.org/10.1016/S0140-6736\(21\)02143-7](https://doi.org/10.1016/S0140-6736(21)02143-7)

Dattani, S., Rodés-Guirao, L., Ritchie, H., & Roser, M. (2023). Mental health. *Our World in Data*. Retrieved from <https://ourworldindata.org/mental-health>

Ferrari, A. J., Charlson, F. J., Norman, R. E., Patten, S. B., Freedman, G., Murray, C. J. L., . . . Whiteford, H. A. (2013). Burden of depressive disorders by country, sex, age, and year: Findings from the Global Burden of Disease Study 2010. *PLoS Medicine*, 10(11), e1001547. <https://doi.org/10.1371/journal.pmed.1001547>

Foreman, K. J., Marquez, N., Dolgert, A., Fukutaki, K., Fullman, N., McGaughey, M., . . . Murray, C. J. L. (2018). Forecasting life expectancy, years of life lost, and all-cause and cause-specific mortality for 250 causes of death: Reference and alternative scenarios for 2016-40 for 195 countries and territories. *The Lancet*, 392(10159), 2052-2090. [https://doi.org/10.1016/S0140-6736\(18\)31694-5](https://doi.org/10.1016/S0140-6736(18)31694-5)

Gardner, E. S., Jr., & McKenzie, E. (1985). Forecasting trends in time series. *Management Science*, 31(10), 1237-1246. <https://doi.org/10.1287/mnsc.31.10.1237>

GBD 2019 Diseases and Injuries Collaborators. (2020). Global burden of 369 diseases and injuries in 204 countries and territories, 1990-2019: A systematic analysis for the Global Burden of Disease Study 2019. *The Lancet*, 396(10258), 1204-1222. [https://doi.org/10.1016/S0140-6736\(20\)30925-9](https://doi.org/10.1016/S0140-6736(20)30925-9)

- GBD 2019 Mental Disorders Collaborators. (2022). Global, regional, and national burden of 12 mental disorders in 204 countries and territories, 1990-2019: A systematic analysis for the Global Burden of Disease Study 2019. *The Lancet Psychiatry*, 9(2), 137-150. [https://doi.org/10.1016/S2215-0366\(21\)00395-3](https://doi.org/10.1016/S2215-0366(21)00395-3)
- GBD 2021 Forecasting Collaborators. (2024). Burden of disease scenarios for 204 countries and territories, 2022-2050: A forecasting analysis for the Global Burden of Disease Study 2021. *The Lancet*, 403(10440), 2204-2256. [https://doi.org/10.1016/S0140-6736\(24\)00685-8](https://doi.org/10.1016/S0140-6736(24)00685-8)
- Hewamalage, H., Bergmeir, C., & Bandara, K. (2021). Recurrent neural networks for time series forecasting: Current status and future directions. *International Journal of Forecasting*, 37(1), 388-427. <https://doi.org/10.1016/j.ijforecast.2020.06.008>
- 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>
- Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1), 55-67. <https://doi.org/10.1080/00401706.1970.10488634>
- Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2), 65-70. Retrieved from <https://www.jstor.org/stable/4615733>
- Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4), 679-688. <https://doi.org/10.1016/j.ijforecast.2006.03.001>

- Institute for Health Metrics and Evaluation. (2020). *Global Burden of Disease Study 2019 (GBD 2019) results*. Seattle, WA: IHME. Retrieved from <https://vizhub.healthdata.org/gbd-results/>
- Jorm, A. F., Patten, S. B., Brugha, T. S., & Mojtabai, R. (2017). Has increased provision of treatment reduced the prevalence of common mental disorders? Review of the evidence from four countries. *World Psychiatry*, 16(1), 90-99. <https://doi.org/10.1002/wps.20388>
- Kapoor, S., & Narayanan, A. (2023). Leakage and the reproducibility crisis in machine-learning-based science. *Patterns*, 4(9), 100804. <https://doi.org/10.1016/j.patter.2023.100804>
- Kim, T.-Y., & Cho, S.-B. (2019). Predicting residential energy consumption using CNN-LSTM neural networks. *Energy*, 182, 72-81. <https://doi.org/10.1016/j.energy.2019.05.230>
- Kingma, D. P., & Ba, J. (2015, May). *Adam: A method for stochastic optimization*. Paper presented at the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA. Retrieved from <https://arxiv.org/abs/1412.6980>
- Lim, B., & Zohren, S. (2021). Time-series forecasting with deep learning: A survey. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 379(2194), 20200209. <https://doi.org/10.1098/rsta.2020.0209>
- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2018). Statistical and machine learning forecasting methods: Concerns and ways forward. *PLoS ONE*, 13(3), e0194889. <https://doi.org/10.1371/journal.pone.0194889>

- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2020). The M4 Competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1), 54-74. <https://doi.org/10.1016/j.ijforecast.2019.04.014>
- Momeni, M. (2023). *Mental health* [Data set]. Kaggle. Retrieved from <https://www.kaggle.com/datasets/imtkaggleteam/mental-health>
- Patel, V., Saxena, S., Lund, C., Thornicroft, G., Baingana, F., Bolton, P., . . . Unützer, J. (2018). The Lancet Commission on global mental health and sustainable development. *The Lancet*, 392(10157), 1553-1598. [https://doi.org/10.1016/S0140-6736\(18\)31612-X](https://doi.org/10.1016/S0140-6736(18)31612-X)
- Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T. (2020). DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3), 1181-1191. <https://doi.org/10.1016/j.ijforecast.2019.07.001>
- 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>
- Smyl, S. (2020). A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *International Journal of Forecasting*, 36(1), 75-85. <https://doi.org/10.1016/j.ijforecast.2019.03.017>
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56), 1929-1958.

- Retrieved from  
<https://jmlr.org/papers/v15/srivastava14a.html>
- Tashman, L. J. (2000). Out-of-sample tests of forecasting accuracy: An analysis and review. *International Journal of Forecasting*, 16(4), 437-450.  
[https://doi.org/10.1016/S0169-2070\(00\)00065-0](https://doi.org/10.1016/S0169-2070(00)00065-0)
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., . . . Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems* 30. Red Hook, NY: Curran Associates.  
Retrieved from  
[https://proceedings.neurips.cc/paper\\_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf)
- Vigo, D., Thornicroft, G., & Atun, R. (2016). Estimating the true global burden of mental illness. *The Lancet Psychiatry*, 3(2), 171-178.  
[https://doi.org/10.1016/S2215-0366\(15\)00505-2](https://doi.org/10.1016/S2215-0366(15)00505-2)
- Wang, X., Hyndman, R. J., Li, F., & Kang, Y. (2023). Forecast combinations: An over 50-year review. *International Journal of Forecasting*, 39(4), 1518-1547.  
<https://doi.org/10.1016/j.ijforecast.2022.11.005>
- Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6), 80-83.  
<https://doi.org/10.2307/3001968>
- World Health Organization. (2021). *Comprehensive mental health action plan 2013-2030*. Geneva, Switzerland: Author.  
Retrieved from  
<https://www.who.int/publications/i/item/9789240031029>

- World Health Organization. (2022). *World mental health report: Transforming mental health for all*. Geneva, Switzerland: Author. Retrieved from <https://www.who.int/publications/i/item/9789240049338>
- Zeng, A., Chen, M., Zhang, L., & Xu, Q. (2023). Are transformers effective for time series forecasting? *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(9), 11121-11128. <https://doi.org/10.1609/aaai.v37i9.26317>

# **AKADEMİK DOKÜMANLARIN OTOMATİK ANALİZİ: METİN SINIFLANDIRMA, BİLGİ ÇIKARIMI VE OTOMATİK ÖZETLEME ÜZERİNE BİR İNCELEME**

**Erhan AKAGÜNDÜZ<sup>1</sup>**

**Cengiz HARK<sup>2</sup>**

**Taner UÇKAN<sup>3</sup>**

## **1. GİRİŞ**

Bilimsel yayınların sayısındaki artış, bilim insanlarının ve araştırmacıların ilgili çalışmaları belirlemesini ve geniş doküman koleksiyonlarını sistematik biçimde incelemesini güçleştirmektedir. Akademik makaleler; araştırma problemi, yöntem, veri kümesi, deneysel süreç ve bulgular gibi farklı bilgi türlerini içermektedir. Bunun yanında alan terminolojisi, atıflar, tablolar, şekiller ve makalelerin bölüm temelli yapısı, akademik metinlerin otomatik analizini genel amaçlı metin işlemeye göre daha karmaşık hale getirebilmektedir.

Doğal dil işleme yöntemleri bu içeriklerin otomatik olarak sınıflandırılması, yapılandırılmış bilginin çıkarılması ve uzun dokümanların özetlenmesi için kullanılmaktadır. Bu bölümde akademik doküman analizi üç temel görev üzerinden ele alınacaktır. Bunlar; metin sınıflandırma, bilgi çıkarımı ve

---

<sup>1</sup> Öğr. Gör., Van Yüzüncü Yıl Üniversitesi, Özalp Meslek Yüksekokulu, Bilgisayar Programcılığı Bölümü, ORCID: 0000-0002-6477-3796.

<sup>2</sup> Doç. Dr., İnönü Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, ORCID: 0000-0002-5190-3504.

<sup>3</sup> Doç. Dr., Van Yüzüncü Yıl Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, ORCID: 0000-0001-5385-6775.

otomatik metin özetleme konularıdır. Amaç, geleneksel yöntemlerden Transformer ve graf tabanlı yaklaşımlara uzanan temel gelişimi, özellikle akademik metinler bağlamında özetlemektir.

## **2. AKADEMİK DOKÜMANLARDA METİN SINIFLANDIRMA**

### **2.1. Geleneksel ve Derin Öğrenme Tabanlı Yaklaşımlar**

Metin sınıflandırma, bir cümle, paragraf, bölüm veya dokümanın önceden tanımlanmış kategorilere atanmasıdır. Akademik dokümanlarda konu, bilim alanı veya bölüm türünün belirlenmesi gibi amaçlarla kullanılabilir. Geleneksel sınıflandırma yaklaşımlarında TF-IDF gibi metin özellikleri ile Naive Bayes, destek vektör makineleri ve lojistik regresyon gibi yöntemler; derin öğrenme yaklaşımlarında ise CNN ve RNN/LSTM mimarileri ele alınmaktadır (Kowsari vd., 2019).

### **2.2. Transformer ve Bilimsel Metinlere Özgü Yaklaşımlar**

Transformer tabanlı ön eğitim, metin sınıflandırmada bağlamsal gösterimlerin kullanılmasını yaygınlaştırmıştır. BERT, büyük miktarda etiketlenmemiş metin üzerinde ön eğitimden sonra farklı görevlere uyarlanabilmektedir (Devlin, Chang, Lee ve Toutanova, 2019). Bilimsel metinlere yönelik SciBERT ise bilimsel yayınlar üzerinde eğitilmiş ve cümle sınıflandırma dahil çeşitli bilimsel doğal dil işleme görevlerinde değerlendirilmiştir (Beltagy, Lo ve Cohan, 2019). Bu modeller, akademik terminoloji ve bağlamın temsil edilmesi açısından çalışılan alana özel ön eğitimin önemini göstermektedir.

### **3. AKADEMİK DOKÜMANLARDA BİLGİ ÇIKARIMI**

#### **3.1. Adlandırılmış Varlık Tanıma**

Bilgi çıkarımı, yapılandırılmamış akademik metinden yöntem, model, veri kümesi, görev ve değerlendirme ölçütü gibi öğelerin yapılandırılmış biçimde elde edilmesini amaçlar. Bunun temel görevlerinden biri Adlandırılmış Varlık Tanımadır (NER). Genel amaçlı NER sistemleri kişi, kurum ve yer gibi sınıfları hedeflerken bilimsel metinlerde varlık türleri alana göre değişebilir. Lample, Ballesteros, Subramanian, Kawakami ve Dyer (2016), kelime ve karakter düzeyindeki temsilleri birlikte kullanan sinir ağı tabanlı NER mimarileri önermiştir. Akademik terminolojinin alanlara göre farklılaşması, bilimsel metinlerde alana özel veri ve modellerin önemini artırmaktadır.

#### **3.2. İlişki ve Eşgönderim Çıkarımı**

Bilimsel bilgiyi yapılandırmak için varlıkların belirlenmesinin yanında aralarındaki ilişkilerin de çıkarılması gerekir. Örneğin bir yöntem belirli bir görevle veya bir model belirli bir veri kümesiyle ilişkilendirilebilir. Luan, He, Ostendorf ve Hajishirzi (2018), bilimsel makalelerde varlık, ilişki ve eşgönderim bilgisini ortak biçimde ele alan SciIE yaklaşımını ve SciERC veri kümesini sunmuştur. Bu tür çıktılar, varlıkların düğüm ve ilişkilerin kenar olarak temsil edildiği bilimsel bilgi graflarının oluşturulmasına temel sağlayabilir. Bununla birlikte çıkarım hatalarının sonraki analizlere aktarılabilmesi, doğruluk ve kaynak metinle ilişkilendirilebilirliği kritik hale getirmektedir.

## **4. AKADEMİK DOKÜMANLARDA OTOMATİK METİN ÖZETLEME**

### **4.1. Çıkarımsal ve Soyutlayıcı Özetleme**

Otomatik metin özetleme, kaynak dokümandaki temel bilginin daha kısa bir metinle sunulmasını amaçlar (Hark, Uçkan, Seyyarer ve Karcı 2019). Çıkarımsal yaklaşımlar kaynak metindeki önemli cümleleri seçerken (Hark, 2025) soyutlayıcı yaklaşımlar kaynak içerikten yeni ifadeler üretir (Lewis vd., 2020). Akademik dokümanların uzunluğu ve bölüm temelli yapısı, özetleme sürecinde hem içerik seçimini hem de uzun bağlamın modellenmesini önemli hale getirmektedir.

### **4.2. Geleneksel ve Makine Öğrenmesi Tabanlı Yaklaşımlar**

Geleneksel çıkarımsal yöntemlerde cümle konumu, kelime sıklığı, başlıkla benzerlik ve cümle uzunluğu gibi özellikler kullanılabilir. Makine öğrenmesiyle birlikte cümle seçimi sınıflandırma veya sıralama problemi olarak da ele alınmıştır. Uçkan ve Karabalut (2024), çıkarımsal özetlemede K-Means, Rastgele Orman, Gradient Boosting Machine, Lojistik Regresyon ve Destek Vektör Makinelerini karşılaştırmalı olarak incelemiştir.

### **4.3. Çizge Tabanlı ve Hibrit Yaklaşımlar**

Çizge tabanlı özetlemede cümleler düğüm, cümleler arasındaki benzerlikler ise kenar olarak temsil edilerek önemli cümleler merkezilik, bölümlenme veya optimizasyon yöntemleriyle belirlenebilmektedir. Uçkan, Hark ve Karcı (2021) tarafından geliştirilen CatSumm, spektral çizge bölmeleme ile düğüm merkeziliklerini çoklu doküman özetleme bağlamında bir araya getirmiştir.

Çizge kuramındaki kombinatoriyal yaklaşımlar da özetlemede kullanılmaktadır. Hark (2025), maksimum bağımsız

küme tabanlı çizge yapısını büyük dil modelleriyle birleştiren çıkarımsal bir özetleme yaklaşımı önermiştir.

Özetlerde bilgi kapsamının korunması ve tekrarlı içeriğin azaltılması da temel bir problemdir. Hark, Uçkan ve Karcı (2024), çoklu doküman özetlemede maksimum kapsama ve minimum tekrarı birlikte ele alan Saplings Growing-up algoritması temelli bir optimizasyon yaklaşımı geliştirmiştir. Uçkan (2025) ise çizge entropisini BERT, RoBERTa ve XLNet gibi dil modelleriyle bütünleştiren hibrit bir çıkarımsal özetleme modeli önermiştir.

Malatya Merkezilik Algoritması, düğüm önemine dayalı farklı çizge problemlerinde değerlendirilmiştir. Karcı, Yakut ve Öztemiz (2022), algoritmayı minimum vertex cover problemine; Yakut, Öztemiz ve Karcı (2023a) ise maksimum bağımsız küme problemine uygulayan yaklaşımlar sunmuştur. Minimum vertex cover için Malatya merkeziliğine dayalı farklı bir algoritma da önerilmiştir (Yakut, Öztemiz ve Karcı, 2023b). Öztemiz ve Yakut (2024b) ise düğüm baskınlık değerlerinin belirlenmesinde Malatya Merkezilik Algoritmasını ele almıştır.

Malatya merkeziliğine dayalı yöntemler ayrıca klik ve ağ analizi problemlerinde de incelenmiştir. Yakut ve Öztemiz (2024), kemoinformatik çizgelerde maksimum klik belirlemeye yönelik Malatya Merkezilik Algoritması tabanlı bir yaklaşım önermiştir. Öztemiz ve Yakut (2024a), Malatya merkeziliği tabanlı klik yöntemini DIMACS benchmark çizgeleri ve farklı rastgele çizge modelleri üzerinde değerlendirmiştir. Yakut, Öztemiz ve Karcı (2025), Malatya Merkezilik Algoritması ile bilinen merkezilik ölçütleri arasındaki sıralama ilişkisini Kendall rank correlation kullanarak analiz etmiştir. Çağlar, Öztemiz ve Yakut (2024) ise gerçek ulaşım ağı verilerinde bağlantı tahmini için Jaccard benzerliğini ve maksimum akış analizi için Ford-Fulkerson yöntemini kullanmıştır.

#### **4.4. Transformer Tabanlı Yaklaşımlar**

Önceden eğitilmiş Transformer modelleri hem çıkarımsal hem de soyutlayıcı özetlemede yaygınlaşmıştır. Liu ve Lapata (2019), BERT temelli doküman kodlayıcısını özetleme görevine uyarlamıştır. BART, bozulmuş metni yeniden oluşturma hedefiyle eğitilen kodlayıcı-kod çözücü yapısıyla metin üretimi ve özetlemede kullanılmaktadır (Lewis vd., 2020). PEGASUS ise ön eğitim hedefini özetleme görevine yaklaştırarak dokümandan çıkarılan önemli cümlelerin yeniden üretilmesini esas almıştır (Zhang, Zhao, Saleh ve Liu, 2020). Uzun akademik dokümanlarda standart öz-dikkatin maliyeti önemli bir sınırlılık oluşturduğundan, Longformer gibi seyrek dikkat kullanan mimariler uzun dizilerin daha düşük maliyetle işlenmesine yönelik çözümler sunmaktadır (Beltagy, Peters ve Cohan, 2020).

### **5. TEMEL SORUNLAR VE GELECEK ARAŞTIRMA YÖNLERİ**

Akademik doküman analizinde uzun metinlerin işlenmesi, dokümanın bölüm yapısının dikkate alınması ve alanlar arasındaki terminolojik farklılıklar temel sorunlar arasındadır. Bilimsel metinlere yönelik çalışılan alana özel ön eğitimin yararı SciBERT çalışmasında da gösterilmiştir (Beltagy vd., 2019).

Diğer bir araştırma yönü, çizge tabanlı yapısal bilginin bağlamsal dil modelleriyle birlikte kullanılmasıdır. Bu tür hibrit örnekler önceki bölümde ele alınmıştır. Bilgi çıkarımı görevlerinde ardışık hataların azaltılması ise çok görevli yaklaşımların ele aldığı sorunlardan biridir (Luan vd., 2018).

### **6. SONUÇ**

Akademik dokümanların otomatik analizi, bilimsel bilgiye erişimi destekleyen sınıflandırma, bilgi çıkarımı ve

özetleme görevlerini bir araya getiren geniş bir doğal dil işleme alanıdır. Geleneksel özellik tabanlı yöntemlerden derin öğrenme ve Transformer modellerine uzanan gelişim, metnin hem bağlamsal hem de yapısal özelliklerinin daha güçlü biçimde temsil edilmesini sağlamıştır.

Bu bölümde metin sınıflandırma, bilimsel bilgi çıkarımı ve otomatik özetleme yaklaşımları temel yöntemleriyle ele alınmıştır. Özellikle çizge tabanlı özetlemede merkezilik, spektral bölümlendirme, baskın ve bağımsız kümeler, optimizasyon ve dil modeli tabanlı hibrit yaklaşımların birlikte kullanıldığı görülmektedir. Gelecekte uzun dokümanların etkin biçimde işlenmesi, Türkçe akademik veri kaynaklarının geliştirilmesi ve kaynak metne dayalı güvenilir sistemlerin oluşturulması başlıca araştırma yönleri arasında değerlendirilebilir.

## KAYNAKÇA

- Beltagy, I., Lo, K., ve Cohan, A. (2019). SciBERT: A pretrained language model for scientific text. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 3615–3620). Association for Computational Linguistics. doi:10.18653/v1/D19-1371
- Beltagy, I., Peters, M. E., & Cohan, A. (2020). Longformer: The long-document Transformer. arXiv preprint arXiv:2004.05150.
- Çağlar, A., Öztemiz, F., ve Yakut, S. (2024). Link prediction and maximum flow in transportation network. Computer Science, 9(2), pp. 169–177. doi:10.53070/bbd.1593501
- Devlin, J., Chang, M.-W., Lee, K., ve Toutanova, K. (2019). BERT: Pre-training of deep bidirectional Transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (pp. 4171–4186). Association for Computational Linguistics. doi:10.18653/v1/N19-1423
- Hark, C., Uçkan, T., Seyyarer, E., ve Karcı, A. (2019). Metin özetlemesi için düğüm merkezliklerine dayalı denetimsiz bir yaklaşım. Bitlis Eren Üniversitesi Fen Bilimleri Dergisi, 8(3), 1109–1118. doi:10.17798/bitlisfen.568883
- Hark, C. (2025). Using graph-based maximum independent sets with large language models for extractive text summarization. Applied Sciences, 15(12), 6395. doi:10.3390/app15126395

- Hark, C., Uçkan, T., ve Karcı, A. (2024). A new multi-document summarisation approach using saplings growing-up optimisation algorithms: Simultaneously optimised coverage and diversity. *Journal of Information Science*, 50(3), 635–650. doi:10.1177/01655515221101841
- Karcı, A., Yakut, S., ve Öztemiz, F. (2022). A new approach based on centrality value in solving the minimum vertex cover problem: Malatya Centrality Algorithm,” *JCS*, vol. 7, no. 2, pp. 81–88, Dec. 2022, doi: 10.53070/bbd.1195501
- Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., ve Brown, D. (2019). Text Classification Algorithms: A Survey. *Information*, 10(4), 150. <https://doi.org/10.3390/info10040150>
- Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., ve Dyer, C. (2016). Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 260–270). doi:10.18653/v1/N16-1030
- Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., ve Zettlemoyer, L. (2020). BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 7871–7880). doi: 10.18653/v1/2020.acl-main.703
- Liu, Y., ve Lapata, M. (2019). Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp.

- 3730–3740). Association for Computational Linguistics. doi:10.18653/v1/D19-1387
- Luan, Y., He, L., Ostendorf, M., ve Hajishirzi, H. (2018). Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (pp. 3219–3232). Association for Computational Linguistics. doi:10.18653/v1/D18-1360
- Öztemiz, F., ve Yakut, S. (2024a). Analysis of the Malatya Centrality-Based Clique Method on DIMACS Benchmarks and Random Graphs”, NATURENGS, vol. 5, no. 2, pp. 63–69, Dec. 2024, doi: 10.46572/naturengs.1591530.
- Öztemiz, F., ve Yakut, S. (2024b). An effective method for determining node dominance values: Malatya Centrality Algorithm. In 2024 32nd Signal Processing and Communications Applications Conference (SIU) (pp. 1–4). IEEE. doi:10.1109/SIU61531.2024.10601155
- Uçkan, T. (2025). A hybrid model for extractive summarization: Leveraging graph entropy to improve large language model performance. Ain Shams Engineering Journal, 16(5), 103348. doi: 10.1016/j.asej.2025.103348
- Uçkan, T., Hark, C., ve Karcı, A. (2021). CatSumm: Spektral çizge bölmeleme ve düğüm merkeziliklerine dayalı çıkarıcı metin özetleme. Bitlis Eren Üniversitesi Fen Bilimleri Dergisi, 10(4), 1349–1365. doi:10.17798/bitlisfen.949052
- Uçkan, T., ve Karabulut, K. (2024). The effectiveness of machine learning algorithms in extractive text summarization: A comparative analysis of K-Means, Random Forest, GBM,

Logistic Regression, and SVM. Doğu Fen Bilimleri Dergisi, 7(2), 77–91. doi:10.57244/dfbd.1538959

Yakut, S., ve Öztemiz, F. (2024). An effective and robust approach based on Malatya Centrality Algorithm for interpreting cheminformatics graphs using maximum clique. Journal of Physical Chemistry and Functional Materials, 7(2), 192–199. doi:10.54565/jphcfum.1590385

Yakut, S., Öztemiz, F., ve Karcı, A. (2023a). A new approach based on centrality value in solving the maximum independent set problem: Malatya Centrality Algorithm. Computer Science, 8(1), 16–23. doi:10.53070/bbd.1224520

Yakut, S., Öztemiz, F., ve Karcı, A. (2023b). A new robust approach to solve minimum vertex cover problem: Malatya vertex-cover algorithm. The Journal of Supercomputing, 79(17), 19746–19769. doi:10.1007/s11227-023-05397-8

Yakut, S., Öztemiz, F., ve Karcı, A. (2025). Kendall rank correlation analysis of Malatya Centrality Algorithm with well-known centrality measures. Sigma Journal of Engineering and Natural Sciences, 43(4), 1339–1354. doi:10.14744/sigma.2025.00124

Zhang, J., Zhao, Y., Saleh, M., ve Liu, P. J. (2020). PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. In Proceedings of the 37th International Conference on Machine Learning (ICML'20), Vol. 119. JMLR.org, Article No: 1051, 11328–11339. doi/10.5555/3524938.3525989

# **JOINT EVALUATION OF SOLUTION QUALITY AND COMPUTATIONAL RESOURCES IN INDEPENDENT SET ALGORITHMS**

**Furkan ÖZTEMİZ<sup>1</sup>**

## **1. INTRODUCTION**

The independent set problem can be interpreted as determining the largest subset that can be selected together from among mutually conflicting options (Hark, 2025): vertices represent candidates, and edges represent relationships that prevent joint selection. In practice, the value of a solution is not limited to cardinality alone; validity, proof of optimality, running time, and memory consumption also determine the choice of method.

The maximum independent set problem is NP-hard on general graphs; however, structural reductions, graph decompositions, and bounding rules make exact solutions of some large instances possible. Furthermore, centrality approaches, MIS, and graph theory (Hark, 2024) provide solutions to numerous real-world problems ((Uckan, Aslan, & Hark, 2025), (Yunus, Hark, & Okumuş, 2025)). Akiba and Iwata (2014) demonstrate the practical value of branching/reduction through vertex cover, while Lamm et al. (2017) demonstrate that of reduction combined with evolutionary search. Therefore, a time limit observed for a single method is not a universal limit for all exact or heuristic methods.

---

<sup>1</sup> İnönü University, Faculty of Engineering, Department of Software Engineering, ORCID: 0000-0001-5425-3474.

In an experimental comparison, it must be clear which metric is calculated on which set of observations and under which implementation conditions. High average solution quality may conceal excessive runtime on some instances, and low median memory may conceal high cost on dense graphs; comparing only completed runs may also make methods with missing records appear more advantageous than they are (Beiranvand et al., 2017).

This chapter examines MISA together with six comparison methods within this framework. Selection based on Malatya centrality was presented by Yakut et al. (Yakut, Öztemiz, & Karcı, 2023a), and its previous theoretical/experimental results were compared with limited algorithmic approaches (Yakut, Öztemiz and Karcı (2025)). The numerical basis of this study consists of the 170 Erdős–Rényi instances used in the general random graphs section (Öztemiz, 2026). The aim is not to present previous graphs as new experiments, but to evaluate, through secondary analysis, the comparability of solution quality, time, memory, and record coverage; Pareto dominance, realized density, failure instances, and quality attainment under a time budget.

## **2. PROBLEM DEFINITION AND LEVELS OF PROOF**

### **2.1. Independence, Maximality, and Maximum Cardinality**

Let  $G = (V, E)$  be a finite, simple, undirected graph, with  $|V| = n$  and  $|E| = m$ .  $I \subseteq V$  is independent if no edge exists between any two selected vertices; maximal if no additional vertex can be added from outside; and maximum if it has the greatest cardinality among all independent sets. Every maximum set is maximal, but the converse does not hold in general.

$$\alpha(G) = \max \{ |I| : I \subseteq V, \forall u, v \in I, u \neq v \Rightarrow \{u, v\} \notin E \} \quad (1)$$

Maximality guarantees that every unselected vertex is adjacent to at least one selected vertex and that the solution cannot be extended by adding a single vertex; it does not rule out finding a larger solution by removing one selected vertex and adding multiple vertices. Thus, the distinction between valid/maximal output and optimal output is maintained.

$$\forall v \in V \setminus I, N_G(v) \cap I \neq \emptyset \quad (2)$$

When  $C = V \setminus I$  is taken,  $C$  is a vertex cover; if  $I$  is maximal,  $C$  is inclusion-minimal, and if  $I$  is maximum,  $C$  has minimum cardinality.  $|I| + |C| = n$  holds for every complementary pair and is not, by itself, proof of optimality; the optimal values satisfy  $\alpha(G) + \tau(G) = n$ .

## 2.2. Objective Functions and Reference Solutions

Let the binary variable  $x_v$  indicate the selection of vertex  $v$ . The maximum independent set is expressed by a binary model that maximizes the number of selected vertices and permits at most one endpoint of each edge to be selected. A valid heuristic solution provides a lower bound; maximum cardinality requires an upper bound attaining the same value or an equivalent certificate.

$$\max \sum_{v \in V} x_v; \quad x_u + x_v \leq 1 \quad (\forall \{u, v\} \in E), \quad x_v \in \{0, 1\} \quad (3)$$

The problem can also be formulated in penalized binary quadratic form. With unit vertex weights and each edge counted once in the sum,  $\lambda > 1$  makes retaining two conflicting vertices together disadvantageous at the global optimum; however, this mathematical property does not imply that a particular solver will prove optimality within its time budget. The QUBO/Ising framework was examined by Lucas (2014).

$$\max F(x) = \sum_{v \in V} x_v - \lambda \sum_{\{u, v\} \in E} x_u x_v, \quad \lambda > 1 \quad (4)$$

### **2.3. Transfer of Solution Error to Vertex Cover**

If the absolute shortfall of an independent set relative to the optimum is  $e = \alpha(G) - |I|$ , the excess of the complementary cover over the minimum is also  $e$ ; however, the denominators of the relative errors differ. Therefore, the percentage error reported for the independent set should not be transferred directly to the vertex cover.

For example, for  $\alpha(G)=20$ ,  $\tau(G)=80$ , and  $|I|=18$ , the independent set shortfall is 10%, whereas the relative excess of the complementary cover is 2.5%. Since  $\tau(G)=0$  may hold in an edgeless graph, relative error for the cover must be defined separately; the primary quality metric in this chapter is the cardinality of the unweighted independent set.

## **3. DECISION PRINCIPLES OF THE ALGORITHMS**

### **3.1. Families of Methods Compared**

Among greedy methods, MDG selects a vertex of minimum degree in the remaining graph; the tendency to delete few neighbors does not directly model the value of future selections, and the implementation here does not represent all MDG variants in the literature (Halldórsson and Radhakrishnan, 1997). VSA constructs the cover side using a support measure based on the sum of neighboring degrees (Balaji et al., 2010); the exact correspondence of the VSA label in the experimental file with the original implementation could not be verified because the source code was unavailable. The Luby approach produces a maximal independent set using random local priorities (Luby, 1986); because details of parallel execution are absent from the records, the measured times are not evidence of parallel speedup.

Exact methods serve a different verification role. CPLEX was used in the source as a binary quadratic model solver; Color B&B exploits the equivalence between an independent set in  $G$  and a clique in the complement graph, along with coloring upper bounds (Tomita and Kameda, 2007); exhaustive search provides a reference for small instances through inclusion/exclusion branches. The names in the table refer to specific implementations of these families.

**Table 1. Decision principles of the methods examined**

| Method            | Basic operation                                        | Role in the evaluation                      |
|-------------------|--------------------------------------------------------|---------------------------------------------|
| MISA              | Selection using minimum Malatya centrality             | Quality-oriented greedy comparison          |
| MDG               | Selection using current minimum degree                 | Low-cost baseline approach                  |
| VSA               | Constructing a cover using the sum of neighbor degrees | Alternative local selection criterion       |
| Luby              | Random local priorities                                | Speed-oriented maximal-set approach         |
| CPLEX             | Binary quadratic optimization                          | Reference classified as exact in the source |
| Color B&B         | Complementary clique and coloring upper bound          | Bounded exact search                        |
| Exhaustive search | Inclusion and exclusion branches                       | Reference for small instances               |

### 3.2. Effect of Malatya Centrality on Selection

Let  $H$  be the current residual graph. Malatya centrality is the sum of the ratios of a vertex's degree to the degrees of its neighbors; for an isolated vertex, the empty sum is zero, and no division by zero occurs because neighbor degrees are at least one. The definition and selection based on minimum centrality are based on (Yakut, Öztemiz, & Karci, 2023b).

$$\Psi_H(v) = \sum_{u \in N_H(v)} \frac{d_H(v)}{d_H(u)}, \quad \Psi_H(v) = 0 \text{ eğer } d_H(v) = 0 \quad (5)$$

Because centrality uses not only vertex degree but also the relative degree structure within the neighborhood, among vertices of equal degree, the one connected to high-degree neighbors may have a lower  $\Psi$  value; therefore, minimum-centrality and minimum-degree selection do not always

coincide. Although the metric indirectly accounts for the structure of the neighborhood to be deleted, it does not resolve all future selections. The Malatya centrality algorithm also provides solutions to many different graph problems. (Yakut (2025a), Yakut (2025b), Karaca and Yakut (2025))

In each iteration, MISA recalculates the degree and  $\Psi$  values of the residual graph, selects the vertex with the smallest  $\Psi$  value, and deletes its closed neighborhood; ties are resolved by fixed vertex identity. Updating the values after each deletion is part of the algorithm's definition; keeping the initial values fixed constitutes a different procedure.

---

Algorithm 1. MISA procedure on the current residual graph

---

Input  $G = (V, E)$ , fixed and unique vertex identifiers

Output Maximal independent set  $I$  and minimal cover  $C$

```
1  $H \leftarrow G$ ;  $I \leftarrow \emptyset$ 
2 While vertices remain in  $H$ 
3   Compute the degrees and  $\Psi$  values in  $H$ 
4    $v \leftarrow \operatorname{argmin}(\Psi(v), \operatorname{identifier}(v))$ 
5    $I \leftarrow I \cup \{v\}$ 
6    $H \leftarrow H[V(H) \setminus (\{v\} \cup N(v))]$  ( $N$  denotes the current neighborhood)
7  $C \leftarrow V \setminus I$ 
8 Return the pair  $(I, C)$ 
```

---

The closed-neighborhood deletion rule ensures that the selected vertices are not adjacent to one another and that each unselected vertex is deleted as a neighbor of some selected vertex; the output is independent and maximal. When the output contains  $k$  vertices, an upper bound of  $O(n^2)$  per iteration and  $O(kn^2)$  in total can be used for a straightforward matrix-based implementation, and this is polynomial because  $k \leq n$ . For an adjacency-list implementation, the residual-graph scanning cost

expressed as  $O(\sum_i(n_i+m_i))$  is more informative; these are theoretical bounds depending on the data structure, not profiling measurements of the experimental code.

**3.3. Illustrated Seven-Vertex Example**

In the illustrative example,  $V=\{1,2,3,4,5,6,7\}$  and  $E=\{(1,4),(1,5),(2,3),(2,6),(2,7),(3,4),(3,5),(3,6),(3,7),(6,7)\}$  are taken, and this graph is not included in the experimental set of 170 instances. Initially, although vertices 1, 4, and 5 have degree two,  $\Psi(1)=2$  and  $\Psi(4)=\Psi(5)=7/5$ . MISA selects 4 and deletes  $\{1,3,4\}$ , then selects isolated vertex 5, followed by vertex 2, which has the smallest identifier in the triangle  $\{2,6,7\}$ , producing the size-3 solution  $\{4,5,2\}$ ; when MDG selects 1 first under the same identifier ordering, it selects only one more vertex from the remaining complete graph  $\{2,3,6,7\}$  and stops at size 2.

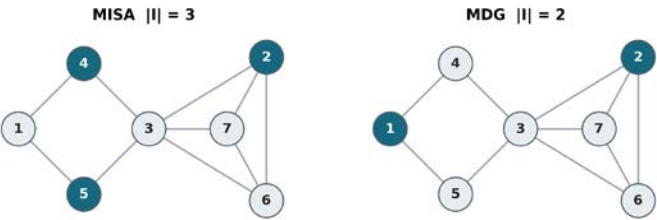


**Figure 1. Different outcomes of centrality and minimum-degree selections on the same graph. Colored vertices indicate the independent set. The example was prepared for this chapter.**

**Table 2. MISA selection sequence in the illustrative example**

| Step | Remaining vertices  | Selected | Deleted closed neighborhood | $\Psi$ value of selected vertex |
|------|---------------------|----------|-----------------------------|---------------------------------|
| 1    | 1, 2, 3, 4, 5, 6, 7 | 4        | 1, 3, 4                     | 7/5                             |
| 2    | 2, 5, 6, 7          | 5        | 5                           | 0                               |
| 3    | 2, 6, 7             | 2        | 2, 6, 7                     | 2                               |

MISA's size-3 output is optimal: at most two vertices can be selected from the four-cycle  $\{1,3,4,5\}$  and at most one from the triangle  $\{2,6,7\}$ ; the upper bound is three, and direct examination of 128 subsets also confirms this. The purpose of the example is not to prove general superiority but to demonstrate the effect of selection criteria. A fixed identifier provides reproducibility for identically labeled input, but relabeling may change tie order and the maximal set found; seed and labeling effects should be examined separately.

## **4. DATA PROVENANCE AND COMPARISON PROTOCOL**

### **4.1. Sources and Experimental Scope**

The solutions, times in seconds, and memory values labeled KB in the 170 records examined were taken from the tables while jointly preserving row order,  $n$ ,  $m$ , and the displayed  $p$ . Vertex counts are 50, 100, 150, 200, 500, 1,000, 5,000, and 10,000, with 21, 21, 21, 21, 17, 17, 26, and 26 instances, respectively; edge counts range from 15 to 12,499,569. The source reports Erdős–Rényi generation, a 600-second limit, and an Intel Core i7-11370H/16 GB/Windows 11/Python-igraph environment. The experiments were not rerun; because code versions, solver logs, seeds, measurement code, and raw graphs were not supplied, the audit is limited to the internal consistency of the records and derived statistics.

At least one result classified as exact in the source exists for 78 instances; 77 have  $n \leq 200$ , and the other has  $n=500$  and source  $p=0.01$ . In instances with records from more than one exact method, the cardinalities agree; this is an important indication of internal consistency but does not replace an independent optimality certificate. Because the IBM (2026) documentation also distinguishes a proven optimum from an

optimum within tolerance, the optimality gaps below are conditional on the source classification.

#### **4.2. Parameter Precision and Record Coverage**

In the Erdős–Rényi model,  $p$  is the generation probability, whereas realized density is  $\rho=2m/[n(n-1)]$ , and these need not be equal in a finite instance. Since the two-decimal precision of the source tables may display different  $p$  values as 0.00/0.01 in large graphs, density analyses were performed using  $\rho$  calculated from  $n$  and  $m$ .

$$\rho = \frac{2m}{n(n-1)}, \quad 0 \leq \rho \leq 1 \quad (6)$$

For example, for  $n=5,000$  and  $m=12,158$ ,  $\rho \approx 0.000973$ ; 0.00 in the  $p$  column does not mean an edgeless graph, and the original  $p$  cannot be recovered exactly from the number of edges. Because the same rounded  $(n,p)$  pair may occur in multiple rows, row order together with  $n,m$  was retained as the instance identifier.

**Table 2. Result generation values based on methods**

| Method            | Solution | Time | Memory |
|-------------------|----------|------|--------|
| MISA              | 170      | 170  | 170    |
| MDG               | 170      | 170  | 170    |
| VSA               | 129      | 129  | 118    |
| Luby              | 170      | 170  | 170    |
| CPLEX             | 76       | 76   | 76     |
| Color B&B         | 70       | 70   | 55     |
| Exhaustive search | 28       | 28   | 24     |

The total number of instances is 170, and record counts were counted separately from the solution, time, and memory appendices of the source report.

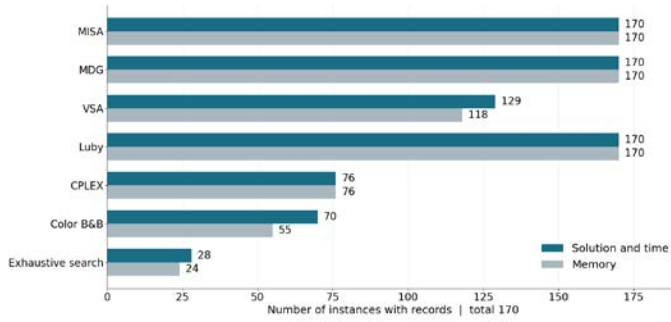


**Figure 2. Separate coverage of solution and time records and memory records. Missing memory data are not equivalent to missing solution data.**

All three metrics are available for all 170 instances for MISA, MDG, and Luby; solution/time records cover 129 instances and memory 118 for VSA, 70 and 55 for Color B&B, and 28 and 24 for exhaustive search. Therefore, all metrics cannot be assumed to refer to the same observation set. The 600-second limit in the source is part of the experimental design; the actual termination reason for every empty cell cannot be determined from the existing tables.

#### 4.3. Quality and Cost Metrics

For an instance with an exact reference, the relative gap is the difference between the reference and the algorithm output divided by the reference. For large graphs without a reference, the largest value among MISA, MDG, and Luby on the same instance,  $b_i$ , is used; this second metric is only the loss relative to the three-method portfolio, and if the portfolio changes,  $b_i$  and the gaps may also change.

$$g_{a,i} = 100 \frac{\alpha(G_i) - |I_{a,i}|}{\alpha(G_i)} \quad (7)$$

$$\delta_{a,i} = 100 \frac{b_i - |I_{a,i}|}{b_i}, \quad b_i = \max \{ |I_{MISA,i}|, |I_{MDG,i}|, |I_{Luby,i}| \} \quad (9)$$

An average that gives equal weight to each instance is a descriptive summary per graph; dividing the sum of all cardinalities by one total reference gives greater weight to larger independent sets. This chapter uses per-instance percentages for quality, the geometric mean for time when appropriate, and the median for memory, and specifies the denominator/type of average in each table.

The paired time ratio is  $t(\text{MISA})/t(\text{MDG})$ , and the quality difference is  $|I(\text{MISA})|-|I(\text{MDG})|$ . If one method provides at least the same quality in at most the same time and is strictly better on one of the metrics, it Pareto-dominates the other; thus, the quality/time weighting is not fixed in advance.

Time-budget-dependent quality attainment  $H_a(B, \varepsilon)$  is the proportion of large instances whose recorded completion time does not exceed  $B$  and whose relative loss does not exceed threshold  $\varepsilon$ ; here  $\varepsilon=1\%$ . The metric uses only final solutions, is not a time series of intermediate solutions, and, although related to the approach of Dolan–Moré (2002), is not identical to the classical time-ratio performance profile.

$$H_a(B, \varepsilon) = \frac{1}{86} \sum_{i \in D_{\text{büyük}}} \mathbb{1}_{\{t_{a,i} \leq B \text{ ve } \delta_{a,i} \leq \varepsilon\}} \quad (9)$$

## **5. COMPARATIVE FINDINGS**

### **5.1. Quality and Common Observation Sets in Small Graphs**

Among the 77 small instances with an exact reference in the source, MISA has a mean gap of 3.871%, a median of 1.064%, and a reference match in 38/77; for MDG, the mean gap is 5.654% and the match count is 25/77, while the gaps for VSA and Luby are higher. These values are findings of previous experiments recalculated from supplementary tables, are not

presented as new experiments, and reference exactness depends on the source classification.

**Table 4. Solution quality in 77 small instances with a reference**

| Method | Mean gap (%) | Reference matches | Match rate (%) |
|--------|--------------|-------------------|----------------|
| MISA   | 3,871        | 38/77             | 49,35          |
| MDG    | 5,654        | 25/77             | 32,47          |
| VSA    | 21,536       | 2/77              | 2,60           |
| Luby   | 20,198       | 3/77              | 3,90           |

In the same 77 instances, MISA yielded a larger result than MDG 32 times, an equal result 42 times, and a smaller result three times; the reduction in mean gap is approximately 1.78 percentage points, and the percentage-point difference should not be confused with a relative percentage improvement.

The 28 instances with solution and time records common to all seven methods consist of 19 graphs with  $n=50$ , five with  $n=100$ , three with  $n=150$ , and one with  $n=200$ ; only 24 of these also have complete memory records for all methods (19 with  $n=50$ , four with  $n=100$ , and one with  $n=150$ ). Accordingly, in Table 5, quality/time is calculated over  $N=28$  and memory over  $N=24$ .

**Table 5. Quality, time, and memory on common instances**

| Method            | Mean gap (%)<br>N = 28 | Geometric mean time<br>(ms)<br>N = 28 | Median memory (KB)<br>N = 24 |
|-------------------|------------------------|---------------------------------------|------------------------------|
| MISA              | 4,04                   | 1,277                                 | 7,83                         |
| MDG               | 5,35                   | 0,238                                 | 2,30                         |
| VSA               | 21,49                  | 3,685                                 | 2,95                         |
| Luby              | 17,41                  | 0,233                                 | 8,03                         |
| CPLEX             | 0,00                   | 409,634                               | 91,12                        |
| Color B&B         | 0,00                   | 4,197                                 | 15,08                        |
| Exhaustive search | 0,00                   | 8.006,893                             | 2,87                         |

For the 28 common instances, the geometric mean runtime is  $\approx 1.28$  ms for MISA,  $\approx 4.20$  ms for Color B&B,

$\approx 409.63$  ms for CPLEX, and  $\approx 8,006.89$  ms for exhaustive search. These ratios are meaningful only for the same instances; the concentration of the common set in small graphs cannot be generalized to large graphs, and selecting common instances does not completely remove the selection effect of conditioning on solvable instances.

## 5.2. Per-Instance Quality and Time in Large Graphs

The large-graph set contains 86 instances. Relative to the best value in the three-method core portfolio, the average loss is 0.207% for MISA, 2.491% for MDG, and 23.790% for Luby; MISA equals the best core value in 77 instances. These are not estimates of the distance from the optimum; the loss may be zero when all methods in the portfolio produce the same low-quality solution.

Of the 66 instances in which MISA produced a larger set than MDG, it was also faster in six, while in 60 the quality gain was obtained at the cost of additional time. MDG was faster in all 11 instances with equal quality and all nine instances in which MISA produced a smaller set; in the pairwise Pareto comparison, MISA dominates in 6 instances and MDG in 20, while the remaining 60 instances require a quality–time trade-off.

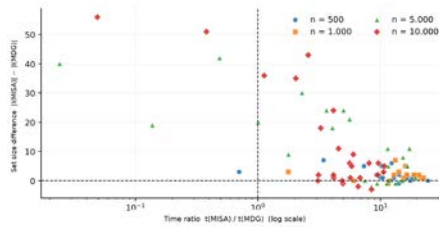


**Figure 3. Paired quality difference and time ratio for MISA and MDG on large graphs. Above the horizontal zero line indicates a quality advantage for MISA, while left of the vertical line at one indicates a time advantage for MISA. Each point compares two records for the same graph.**

The non-dominated solution–time points among the three core methods are 66/86 for MISA, 80/86 for MDG, and 72/86 for Luby; the sum exceeds 86 because multiple methods may remain in the Pareto set on the same graph with different trade-offs. Non-dominance does not mean highest quality; a very fast but lower-quality method may also be non-dominated.

**5.3. Behavior According to Realized Density**

Large graphs were divided into three descriptive groups:  $\rho \leq 0.01$ ,  $0.01 < \rho \leq 0.10$ , and  $\rho > 0.10$ . The thresholds are not universal sparse/dense classes, and the distributions of  $n$  across groups are not equal; because 20 of the 22 graphs in the first group have 5,000 or 10,000 vertices, the differences cannot be attributed solely to a causal effect of density.

**Table 6. Paired comparison by density in large graphs**

| Density                 | N  | MISA superior | Equal | MDG superior | Mean vertex difference | Geom. time ratio |
|-------------------------|----|---------------|-------|--------------|------------------------|------------------|
| $\rho \leq 0,01$        | 22 | 22            | 0     | 0            | 24,64                  | 1,36             |
| $0,01 < \rho \leq 0,10$ | 33 | 24            | 5     | 4            | 3,09                   | 12,00            |
| $\rho > 0,10$           | 31 | 20            | 6     | 5            | 1,06                   | 9,97             |

The vertex difference was calculated as MISA – MDG, and the time ratio as MISA / MDG. Density  $\rho$  differs from the generation parameter  $p$ .

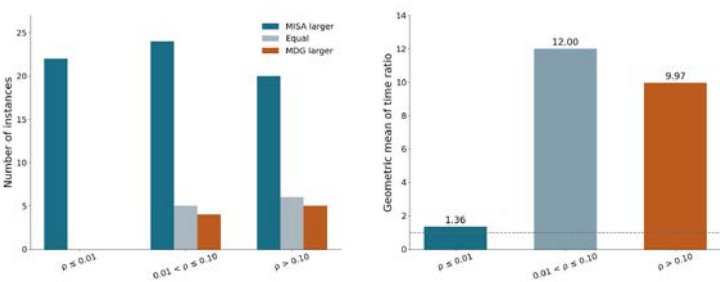


**Figure 4. Quality comparison and geometric mean of the MISA/MDG time ratio in realized-density groups. The number of observations differs between groups. The groups represent a secondary analysis of existing data.**

MISA is superior in all 22 instances in the  $p \leq 0.01$  group; the mean absolute gain is 24.64 vertices and the geometric time ratio is 1.36. At medium/high density, the mean gain falls to 3.09 and 1.06 vertices, while the time ratios increase to 12.00 and 9.97. At the same  $n=5,000$  level, the group differences are 24.70/5.13/0.38; at  $n=10,000$ , they are 28.90/2.25/1.25. However, because the relative cardinality gains are approximately 2.47%/2.37%/2.37%, a substantial part of the variation in absolute gains is associated with scale.

Density also matters in interpreting complexity: under fixed positive  $p$ , the expected number of edges in  $G(n,p)$  increases quadratically as  $p \cdot n(n-1)/2$ ; if mean degree is to be kept constant,  $p$  must decrease approximately on the order of  $1/n$ . These two designs address different scaling questions; the source  $p=0.01$  log–log time slope is not a general complexity theorem, and it was not re-estimated here because rounded  $p$  does not distinguish the original choices.

#### 5.4. Cases Where MISA Produces a Smaller Solution

In the nine large instances where MISA produced a smaller set than MDG, the loss ranges from 1 to 3 vertices and the MISA time ratio from 4.95 to 14.18; MDG is preferable in terms of quality–time alone. Table 7 presents all nine instances.

**Table 7. Nine large-graph instances in which MDG produced a larger set**

| n      | m         | $p$      | MISA | MDG | Time ratio |
|--------|-----------|----------|------|-----|------------|
| 500    | 8.715     | 0,069860 | 66   | 67  | 14,18      |
| 1.000  | 29.770    | 0,059600 | 88   | 89  | 11,60      |
| 5.000  | 1.251.837 | 0,100167 | 77   | 78  | 13,37      |
| 5.000  | 1.376.891 | 0,110173 | 69   | 70  | 11,30      |
| 5.000  | 1.875.895 | 0,150102 | 51   | 53  | 8,44       |
| 5.000  | 2.500.875 | 0,200110 | 38   | 39  | 9,44       |
| 10.000 | 3.499.882 | 0,070005 | 115  | 118 | 8,51       |
| 10.000 | 4.500.833 | 0,090026 | 90   | 92  | 6,60       |
| 10.000 | 6.001.826 | 0,120049 | 71   | 72  | 4,95       |

The largest absolute loss occurs at  $n=10,000$ ,  $m=3,499,882$ , with 115 versus 118; the highest relative core loss is  $2/53 \approx 3.77\%$  at  $n=5,000$ ,  $m=1,875,895$ . Absolute and relative losses highlight different instances; without raw edge lists, attributing failures to a particular motif, degree distribution, or local selection sequence would exceed the evidential limits of the data.

**5.5. Attainment of the Quality Target Within the Time Budget**

The time-budget analysis counts, among 86 large instances, runs that finish within time B and have at most 1% loss relative to the best final solution of the core portfolio; because records of earlier intermediate solutions are unavailable, no assumptions are made about their existence or quality.

**Table 8. Attainment under a time budget and a one-percent quality threshold**

| Budget B (s) | MISA          | MDG           | Luby        |
|--------------|---------------|---------------|-------------|
| 0,1          | 12/86 (14,0%) | 8/86 (9,3%)   | 0/86 (0,0%) |
| 1,0          | 34/86 (39,5%) | 17/86 (19,8%) | 0/86 (0,0%) |
| 10,0         | 56/86 (65,1%) | 23/86 (26,7%) | 0/86 (0,0%) |
| 60,0         | 77/86 (89,5%) | 24/86 (27,9%) | 0/86 (0,0%) |

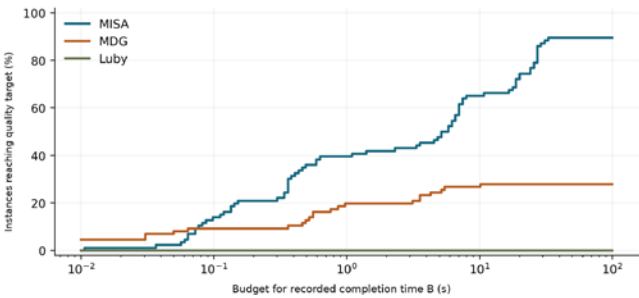


**Figure 5. Time-budget-dependent quality attainment based on recorded final solutions. The target is at most 1% loss from the best core solution on the same instance. The curves do not represent records of intermediate-solution progress.**

Within one second, MISA meets the quality condition in 34/86 instances and MDG in 17/86; within 60 seconds, they do so in 77/86 and 24/86 instances, respectively. Although Luby fails to meet the strict 1% threshold in every large instance, it has a recorded solution for all of them; zero attainment indicates a threshold violation, not failure to find a solution. A looser quality requirement or a requirement for only a fast maximal set may lead to a different choice.

This evaluation is not a comparison with candidates found by exact solvers before proof of optimality or with early-terminated partial solutions from MISA/MDG/Luby; because time-stamped intermediate records are unavailable, the budget results are summaries conditional on final termination times.

### **5.6. Density-Dependent Variation in Memory Behavior**

For  $n=10,000$ , median memory is 11,715.79 KB for MISA, 13,341.52 KB for MDG, and 14,170.21 KB for Luby; nevertheless, MISA uses less memory than MDG in only 14 of the 26 paired graphs and more in 12; for  $n=5,000$ , the distribution is 13 to 13. Therefore, the median advantage cannot be generalized to all densities.

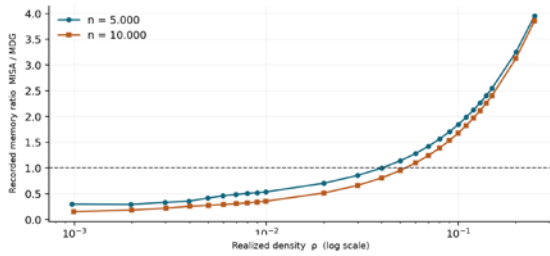


**Figure 6. Paired memory ratio by realized density at large sizes. Values below one indicate lower recorded memory for MISA, and those above one indicate lower recorded memory for MDG. Whether the source measurement represents total process memory or additional allocation is unknown.**

In the instance with  $n=10,000$  and  $m=12,499,569$ , MISA uses 51,506.47 KB and MDG 13,367.11 KB, with a ratio of  $\approx 3.85$ . Using a ratio removes differences arising from decimal/binary conversion of KB but does not correct the measurement definition; the results should not be generalized to total system memory without knowing whether the main graph, temporary copies, Python objects, and solver memory were included.

### **5.7. Interpretation of Statistical Evidence**

In the Wilcoxon signed-rank recalculation, zeros were excluded, average ranks were used for equal absolute differences, and the asymptotic p-value without continuity correction was used. For MISA–MDG,  $W=53.5$ ,  $p \approx 8.22 \times 10^{-6}$ ,  $r=0.830$  in the small reference set, and  $W=149.5$ ,  $p \approx 1.42 \times 10^{-11}$ ,  $r=0.895$  in the large set, consistent with the report. Because 42 of the 77 differences in the small set are zero, the effective number of ranks is 35; in the large set, 75 of the 86 pairs have nonzero differences. SciPy notes that zeros/ties affect the distribution. Because there are no repeated graph generations or random runs, the p-values are not proof of universal superiority but supplementary evidence for the existing paired differences; density analyses are exploratory, and the main interpretation rests on the observed distributions of superiority/ties/losses/costs.

## **6. METHOD SELECTION ACCORDING TO APPLICATION REQUIREMENTS**

### **6.1. Verification Requirement and Resource Budget**

Method selection should first establish whether the application needs an optimality certificate or a valid and sufficiently large solution. If the optimum is required, a heuristic

lower bound alone is insufficient; for the exact solver, the best valid solution, upper bound, termination status, and remaining gap should be reported together. Under quality–time flexibility, MISA and MDG should be selected according to the additional quality/additional time trade-off on the same instance rather than given a fixed ranking; with strict latency, a fast initial solution may be useful, but a multistage system was not measured in these data.

**Table 9. Comparison approach according to requirements**

| Requirement                    | Option to examine                   | Decision criterion                               |
|--------------------------------|-------------------------------------|--------------------------------------------------|
| Proof of optimality            | Exact solver and certificate        | Valid solution, upper bound, and remaining gap   |
| Flexible time for a larger set | MISA and strong improvement methods | Additional quality and time on the same instance |
| Valid output with low latency  | MDG or random-priority approach     | Completion within budget and accepted quality    |
| Limited memory on dense graphs | Measured low-memory implementation  | Peak memory and input cost by density            |
| Reproducible decision          | Fixed labeling and selection rule   | Code version, tie-breaking rule, and seed        |

## **6.2. Combined Use and Directions for Improvement**

Running multiple methods and retaining the largest independent set is a simple portfolio approach; sequential execution adds runtimes and management overhead, whereas parallel execution may reduce wall-clock time but increase processor/memory requirements. The best core reference in this chapter represents the output size of such a portfolio in the records, not a measured portfolio runtime.

Directions for improving MISA include tracking changing neighborhoods through safe local updates verified to preserve the same selection order, rather than recalculating all centrality values from scratch in every iteration, and enlarging the maximal set found through local exchanges; such extensions should be compared with strong references (Lamm et al., 2017).

If a new method selector is developed, training and evaluation must be separated; the information value of realized density, degree variability, number of components, and local structural features must be independently validated on raw graphs, otherwise success on the same data is not evidence of generalizability.

## **7. REPRODUCIBILITY AND LIMITATIONS OF VALIDITY**

For reproducibility, solution, time, memory, and the optimality certificate should be treated as separate data fields; each graph should be assigned a unique identifier, and  $n$ ,  $m$ , the full-precision generation parameter, the generation seed, and the file checksum should be retained. Each run should include the algorithm version, tie-breaking rule, random seed, start/end time, termination reason, and solution vertices; validation should confirm that no edges exist between selected vertices, and maximality checking should confirm that every unselected vertex has a selected neighbor. For timing measurements, it should be specified which of graph generation, file reading, complement construction, model building, solution search, and verification are included; repetitions, warm-up, background load, and measurement resolution for millisecond-scale runs, as well as different seeds for random methods and independent instances in graph generation, should be considered. A sufficient number of repetitions should be determined by the observed variance and target uncertainty. For memory, peak process memory and additional allocation should be distinguished, the shared initial graph should be counted in the same manner, coverage of external processes/libraries should be explained, and KB/kB/KiB/MB labels should be used consistently; because no source unit definition is available, this chapter uses the KB

values without conversion and employs ratios for the same instances. External validity is limited: Erdős–Rényi instances do not represent all networks with community structure, heavy-tailed degree distributions, geometric constraints, or domain-specific constraints, and the seven implementations examined do not encompass the entire current independent set literature. Claims of superiority across the literature cannot be made without new comparisons against reduction, local search, and advanced exact solvers.

## **8. CONCLUSION**

The 170 experiments show that MISA exhibits a strong quality profile compared with simple heuristics, but that the advantage is not independent of time and density. Of its 66 quality advantages over MDG on large graphs, six were achieved in less time and 60 with additional time; MDG dominated MISA in quality–time terms in 20 instances. While the absolute quality gain becomes more pronounced at low realized density, the memory advantage was not maintained in every instance, and memory records were complete for only 24 of the 28 common solution/time records across all seven methods.

From an application perspective, the quality target, time/memory budget, and verification requirement should be determined before method selection. MISA is meaningful in many quality-prioritized instances; other methods may be suitable when exact optimality, very low latency, or limited memory on a dense graph takes priority. Independent experimental replications, raw graphs, strong additional references, and transparent measurement records are the next step toward extending these conditional findings to broader graph classes.

## REFERENCES

- Akiba, T., & Iwata, Y. (2014). Branch-and-reduce exponential/FPT algorithms in practice: A case study of vertex cover [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.1411.2680>
- Balaji, S., Swaminathan, V., & Kannan, K. (2010). A simple algorithm to optimize maximum independent set. *Advanced Modeling and Optimization*, 12(1), 107–118. <https://camo.ici.ro/journal/vol12/v12a9.pdf>
- Beiranvand, V., Hare, W., & Lucet, Y. (2017). Best practices for comparing optimization algorithms. *Optimization and Engineering*, 18, 815–848. <https://doi.org/10.1007/s11081-017-9366-1>
- Dolan, E. D., & Moré, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91, 201–213. <https://doi.org/10.1007/s101070100263>
- Halldórsson, M. M., & Radhakrishnan, J. (1997). Greed is good: Approximating independent sets in sparse and bounded-degree graphs. *Algorithmica*, 18, 145–163. <https://doi.org/10.1007/BF02523693>
- IBM. (2026). CPXMIP\_OPTIMAL. IBM ILOG CPLEX Optimization Studio 22.1.2 documentation. Accessed October 4, 2026. <https://www.ibm.com/docs/en/icos/22.1.2?topic=api-cpxmip-optimal>
- Karaca, C., & Yakut, S. (2025). Graph-Based Course Scheduling Using the Malatya Vertex Coloring Algorithm for Constraint Optimization. *Turkish Journal of Nature and Science*, 14(3), 46-56. <https://doi.org/10.46810/tdfd.1629184>

- Lamm, S., Sanders, P., Schulz, C., Strash, D., & Werneck, R. F. (2017). Finding near-optimal independent sets at scale. *Journal of Heuristics*, 23, 207–229. <https://doi.org/10.1007/s10732-017-9337-x>
- Luby, M. (1986). A simple parallel algorithm for the maximal independent set problem. *SIAM Journal on Computing*, 15(4), 1036–1053. <https://doi.org/10.1137/0215074>
- Lucas, A. (2014). Ising formulations of many NP problems. *Frontiers in Physics*, 2, Article 5. <https://doi.org/10.3389/fphy.2014.00005>
- Öztemiz, F. (2026). MIS-and-MVC- [Computer software]. GitHub. Retrieved October 9, 2026, from <https://github.com/furkanoztemiz/MIS-and-MVC-/tree/main>
- Tomita, E., & Kameda, T. (2007). An efficient branch-and-bound algorithm for finding a maximum clique with computational experiments. *Journal of Global Optimization*, 37, 95–111. <https://doi.org/10.1007/s10898-006-9039-7>
- Hark, C. (2025). Using graph-based maximum independent sets with large language models for extractive text summarization. *Applied Sciences*, 15(12), 6395. doi:10.3390/app15126395
- Uckan, T., Aslan, C., & Hark, C. (2025). A comprehensive hybrid approach for indoor scene recognition combining CNNs and text-based features. *Sensors*, 25(17), 5350. doi:10.3390/s25175350
- Hark, C. (2024). The power of graphs in medicine: Introducing BioGraphSum for effective text summarization. *Heliyon*, 10(11), e31813. doi:10.1016/j.heliyon.2024.e31813.

- Yakut, S., Öztemiz, F., & Karci, A. (2025). Kendall rank correlation analysis of Malatya Centrality Algorithm with well-known centrality measures. *Sigma Journal of Engineering and Natural Sciences*, 43(4), 1339–1354. <https://doi.org/10.14744/sigma.2025.00124>
- Yakut, S. (2025a). "Efficient and Robust Graph Coloring Algorithm for Graph Coloring Problem: Malatya Vertex Coloring Algorithm," in *IEEE Access*, vol. 13, pp. 168512-168526, 2025, doi: 10.1109/ACCESS.2025.3614296
- Yakut, S. (2025b), A robust and efficient algorithm for graph coloring problem based on Malatya centrality and sequent independent sets, *Egyptian Informatics Journal*, Volume 30, 2025, <https://doi.org/10.1016/j.eij.2025.100676>.
- Yakut, S., Öztemiz, F., & Karci, A. (2023b). A new robust approach to solve minimum vertex cover problem: Malatya vertex-cover algorithm. *The Journal of Supercomputing*, 79, 19746–19769. doi:10.1007/s11227-023-05397-8
- Yakut, S., Öztemiz, F., & Karcı, A. (2023a). A new approach based on centrality value in solving the maximum independent set problem: Malatya Centrality Algorithm. *Computer Science*, 8(1), 16–23. <https://doi.org/10.53070/bbd.1224520>
- Yunus, S., Hark, C., & Okumuş, F. (2025). Enhancing extractive multi-documents summarization with a novel dominating set model for semantic relationship detection. *Engineering Science and Technology, an International Journal*, 69, 102127. doi:10.1016/j.jestch.2025.102127

# **COMPARATIVE EVALUATION OF MALATYA CENTRALITY AND MACHINE LEARNING ALGORITHMS FOR SPOTIFY SONG POPULARITY ANALYSIS**

**Selman YAKUT<sup>1</sup>**

## **1. INTRODUCTION**

The digital transformation of the music industry has made song popularity analysis an important research area in music data mining and machine learning. Spotify datasets provide multidimensional acoustic features, including tempo, energy, danceability, valence, acousticness, speechiness, and loudness, which can be used to investigate musical characteristics and popularity patterns (Duman, Neto, Mavrolampados, Toiviainen, & Luck, 2022). However, relationships between these features and popularity are often nonlinear and influenced by dataset size, listener preferences, and temporal trends. Consequently, different analytical approaches may produce varying results depending on their methodological characteristics.

Previous studies have investigated song popularity using various machine learning algorithms. Middlebrook and Sheik (2019) and Saragih (2023) examined machine learning approaches to song popularity prediction, including tree-based methods such as Random Forest, while K. Li (2024) and Yao (2024) investigated popularity prediction and model generalization across different data distributions. Additionally,

---

<sup>1</sup> Doçent Dr., İnönü Üniversitesi, Mühendislik Fakültesi, Yazılım Mühendisliği bölümü, ORCID: 0000-0002-0649-1993.

Random Forest (Breiman, 2001) provide complementary approaches to prediction. Unlike these methods, Malatya Centrality (MC) was originally developed to evaluate relational structures in graphs (Karci, Yakut, & Öztemiz, 2022) and has subsequently been investigated in graph optimization and other application domains (Yakut, Öztemiz, & Karci, 2023a, 2023b).

This study comparatively evaluates MC, K-Means, KNN, Random Forest, and Naive Bayes using six Spotify dataset sizes comprising 10, 50, 100, 500, 1,000, and 5,000 observations. The analysis focuses on song rankings, overlap among algorithms, and score variations across dataset sizes. MC-inspired relational proximity scores are calculated using distances between standardized acoustic features. The main contribution is to investigate the applicability of an MC-inspired approach to Spotify song analysis and evaluate its similarities and differences with conventional machine learning approaches across six dataset sizes.

## **2. RELATED WORK**

Research on music popularity prediction has employed various machine learning approaches using Spotify's acoustic features. Middlebrook and Sheik (2019) investigated hit-song prediction, while Saragih (2023) examined popularity prediction using machine learning methods. K. Li (2024) investigated relationships between musical attributes and popularity, whereas Y. Li (2024) and Yao (2024) examined machine learning approaches, data distribution, and model generalization in song popularity prediction. Duman, Neto, Mavrolampados, Toiviainen, and Luck (2022) investigated relationships between Spotify audio features and reasons for listening, demonstrating the broader analytical relevance of acoustic characteristics.

Different algorithms provide complementary perspectives on music data analysis. Random Forest combines decision trees for predictive modeling (Breiman, 2001). In contrast, centrality-based approaches examine relational structures. Furthermore, Özavci and Yakut (2023) explored its applicability to relational data in social platforms and commercial applications, suggesting its potential for other analytical contexts.

Although previous studies have extensively investigated Spotify popularity prediction, comparative analyses incorporating centrality-based methods remain limited. Accordingly, this study evaluates MC, K-Means, KNN, Random Forest, and Naive Bayes across six dataset sizes (10, 50, 100, 500, 1,000, and 5,000 observations). The analysis focuses on song-ranking similarities, overlap among highly ranked songs, and score variations, aiming to assess the potential of an MC-inspired relational proximity approach for analyzing Spotify music data.

### **3. METHODOLOGY**

This section presents the methodological framework, including data preprocessing, the application of MC and four machine learning algorithms, and comparative evaluation across different dataset sizes.

#### **3.1. Dataset and Preprocessing**

A Spotify dataset containing song popularity, release year, and various acoustic and structural features was used. During preprocessing, numerical variables were identified, missing or invalid values were addressed, and features were standardized to minimize scale-related effects on distance-based calculations:

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j} \quad (1)$$

Here;  $z_{ij}$ , the standardized value represents the  $j$ -th feature of the  $i$ -th song;  $x_{ij}$ , the original value denotes the corresponding unstandardized feature value;  $\mu_j$ , the arithmetic mean represents the mean of the  $j$ -th feature; and  $\sigma_j$ , the standard deviation denotes the standard deviation of the  $j$ -th feature. Additionally, song popularity was classified into five categories: 0–20 (Very Low), 20–40 (Low), 40–60 (Medium), 60–80 (High), and 80–100 (Very High). The popularity variable was used as the target variable in the prediction models.

### **3.2. Algorithms Employed**

#### **3.2.1. Malatya Centrality**

MC, introduced by Karci et al. (2022), evaluates vertex importance using neighborhood relationships and degree information. Initially developed for the minimum vertex cover problem, MC was subsequently extended to maximum independent set problems and other graph-based optimization tasks (Öztemiz, 2025; Yakut, Öztemiz, & Karci, 2023a, 2023b). Further studies investigated its effectiveness in determining node dominance and its relationships with conventional centrality measures (Öztemiz & Yakut, 2024; Yakut, Öztemiz, & Karci, 2025), demonstrating its applicability across different graph-theoretic contexts. Graph-based optimization methods have been employed in automatic text summarization through dominating set models (Yunus, Hark, & Okumuş, 2025) and maximum independent sets integrated with large language models (Hark, 2025). Additionally, convolutional neural networks combined with text-based features have been applied to indoor scene recognition (Uckan, Aslan, & Hark, 2025).

In this study, a distance-based adaptation inspired by MC was employed to evaluate relational proximity among Spotify songs. Each song was represented as a node, and similarities were assessed using Euclidean distances calculated from standardized acoustic features:

$$d(x_i, x_j) = \sqrt{\sum_{p=1}^m (z_{ip} - z_{jp})^2} \quad (2)$$

Here, (p) represents the number of features, while  $z_{ip}$  ve  $z_{jp}$  denote the standardized feature values of the respective songs. Subsequently, the relational proximity score for each song relative to the base song set was calculated as follows:

$$MC(x_i) = \sum_{x_j \in B} \frac{1}{1 + d(x_i, x_j)} \quad (3)$$

Here, B represents the reference set of songs. To Eq. 3, shorter distances yield higher proximity scores, indicating stronger song similarity.

### **3.2.2. K-Means**

K-Means is an unsupervised clustering algorithm that groups similar observations. In this study, three clusters ( $K = 3$ ) were formed, and each song's proximity score was calculated based on its distance to the nearest cluster centroid:

$$s = \frac{b - a}{\max(a, b)} \quad (4)$$

Here,  $\mu_k$  denotes the centroid of cluster k.

### **3.2.3. KNN**

KNN is a distance-based algorithm that evaluates relationships among neighboring observations. In this study, the

mean distance to the five nearest neighbors ( $k = 5$ ) was used to calculate a proximity score:

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (5)$$

Here,  $n_j$  denotes the  $j$ -th nearest neighbor of the song under consideration. The score was used to analyze local neighborhood relationships among tracks.

#### **3.2.4. Random Forest**

Random Forest is an ensemble learning algorithm that combines multiple decision trees. In this study, a RandomForestRegressor with 50 trees was employed to predict numerical song popularity scores:

$$y(x_i) = \frac{1}{50} \sum_{t=1}^{50} h_t(x_i) \quad (6)$$

Here,  $y(x_i)$ , represents the predicted popularity score of song  $i$ , while  $h_t(x_i)$ , denotes the prediction generated by decision tree  $t$  for the same song.

#### **3.2.5. Naive Bayes**

Naive Bayes is a probabilistic classifier based on Bayes' theorem. In this study, GaussianNB was used to classify songs according to whether their popularity scores exceeded 50:

$$P(X/Y) = \frac{P\left(\frac{X}{Y}\right)P(Y)}{P(X)} \quad (7)$$

Here,  $c$  denotes the popularity class. The highest class probability generated by the model was used as the score.

### 3.3. Experimental Design

Six dataset sizes (10, 50, 100, 500, 1000, and 5000 observations) were analyzed to assess the effects of sample size on algorithm performance. Random sampling, an 80% training/20% test split, and ten experimental repetitions were employed. Computational times and algorithm-specific scores were evaluated, considering the distinct objectives of proximity analysis, regression, and classification.

### 3.4. Pseudocode of the Study

Algorithm 1. Comparative experimental procedure for Spotify song popularity analysis

---

```
1. Input:
D: Spotify dataset
S: {10, 20, 50, 100, 200, 500, 1000, 2000, 5000}
A: {MC, K-Means, KNN, Random Forest, Naive Bayes}
Output:
R: Algorithm results
1. Load dataset D.
2. Identify numerical features.
3. Handle missing and invalid values.
4. Separate the popularity target from input features.
5. FOR EACH  $s \in S$ :
6.   Select  $s$  observations from D to form  $D_s$ .
7.   Split  $D_s$  into training and test sets.
8.   Estimate standardization parameters using training data.
9.   Transform training and test data using the same parameters.
10.  Identify the reference song set for MC.
11.  Calculate Euclidean distances between songs.
12.  Compute MC proximity scores.
13.  Fit the K-Means model with  $K = 3$ .
14.  Calculate scores based on the nearest cluster centroid.
15.  Identify  $k = 5$  nearest neighbors for KNN.
16.  Calculate scores based on mean neighbor distance.
17.  Train Random Forest using 50 trees.
18.  Predict popularity values for test songs.
19.  Construct a binary target using popularity  $> 50$ .
20.  Train the GaussianNB model.
21.  Calculate class probabilities for test songs.
22.  Record results and computational times.
23.   $R[s] \leftarrow$  Results for the corresponding dataset size.
24. END FOR
25. Compare the results using tables and figures.
26. Return R.
```

---

## 4. EXPERIMENTAL RESULTS

This section presents the experimental findings in book-chapter format, retaining the comparative tables for six dataset sizes (10, 50, 100, 500, 1,000, and 5,000 observations) and preserving the original experimental results.

### 4.1. Malatya Centrality Analysis

The results indicate that mean MC values increase with dataset size and stabilize at approximately 3.76–3.81 for 500–5,000 nodes. The maximum MC reached 4.3868 at 5,000 nodes, while “Together As One” frequently ranked highest in larger samples. These findings suggest that MC can identify songs with prominent relational positions in the analyzed feature space.

**Table 1. MC Analysis Results for Different Dataset Sizes**

| Dataset Size | Number of Nodes | Number of Edges | Maximum MC | Song with Highest MC | Mean MC |
|--------------|-----------------|-----------------|------------|----------------------|---------|
| 10           | 10              | 7               | 3.2000     | Walking in a winter  | 2.4000  |
| 50           | 50              | 645             | 4.0345     | Everything is Sound  | 3.5964  |
| 100          | 100             | 3,111           | 3.9718     | Everything is Sound  | 3.6917  |
| 500          | 500             | 84,526          | 4.0914     | Father's day         | 3.7871  |
| 1000         | 1000            | 343,63          | 4.1003     | Together As One      | 3.7834  |
| 5000         | 5000            | 7,766,082       | 4.3868     | Together As One      | 3.7651  |

### 4.4. Song Ranking Outputs Across Dataset Sizes

This section compares song rankings generated by the algorithms across six dataset sizes (10, 50, 100, 500, 1,000, and 5,000 observations), highlighting similarities and differences in the highest-scoring songs as sample size increases.

### 4.5. Small-Scale Samples

For the 10- and 50-node datasets, the algorithms produced different song rankings, reflecting their distinct scoring approaches. As shown in Table 2, for the 10-node dataset, MC shared all ten songs with Random Forest, Naive

Bayes, and K-Means, and nine with KNN, although their ranking positions varied. Both MC and Random Forest ranked “Walking in a Winter” first, indicating substantial agreement in song selection despite differences in ordering.

**Table 2. Ten Highest-Scoring Songs by Method for Base = 10**

| Rank | MC                                 | K-Means                           | KNN                               | Random Forest                      | Naive Bayes                       |
|------|------------------------------------|-----------------------------------|-----------------------------------|------------------------------------|-----------------------------------|
| 1    | Walking in a Winter (119.6993)     | What They Say (0.6973)            | Say Anything (0.9543)             | Walking in a Winter (38.4283)      | Dancing Shoes (0.9958)            |
| 2    | Dancing Shoes (119.1196)           | Living in the Moment (0.5222)     | Dancing Shoes (0.9504)            | Dancing Shoes (38.4276)            | Say Anything (0.9874)             |
| 3    | I Shall Be Released (117.2281)     | Money (That's What I...) (0.4761) | Money (That's What I...) (0.9503) | Say Anything (38.0234)             | Teenage Dream (0.9755)            |
| 4    | Say Anything (115.1615)            | I Shall Be Released (0.4296)      | Living in the Moment (0.9477)     | I Shall Be Released (38.0093)      | Walking in a Winter (0.9741)      |
| 5    | Teenage Dream (110.0669)           | Holding On and Letti (0.3948)     | Teenage Dream (0.9444)            | Teenage Dream (37.2194)            | Fix You (0.9676)                  |
| 6    | Holding On and Letti (91.0488)     | Walking in a Winter (0.3673)      | Walking in a Winter (0.9424)      | Holding On and Letti (34.4139)     | Holding On and Letti (0.9589)     |
| 7    | Fix You (79.8068)                  | Fix You (0.3554)                  | Fix You (0.9352)                  | Fix You (32.4258)                  | I Shall Be Released (0.9478)      |
| 8    | Money (That's What I...) (69.6451) | Teenage Dream (0.3452)            | I Shall Be Released (0.9260)      | Money (That's What I...) (29.2225) | Money (That's What I...) (0.9239) |
| 9    | Living in the Moment (55.6514)     | Say Anything (0.3183)             | Holding On and Letti (0.9153)     | Living in the Moment (26.0217)     | Living in the Moment (0.9004)     |
| 10   | What They Say (36.3521)            | Dancing Shoes (0.2865)            | What They Say (0.8929)            | What They Say (20.8141)            | What They Say (0.8088)            |

For the 50-node dataset, Table 3 shows that MC and Random Forest shared nine of the top ten songs in identical order. MC shared six songs with Naive Bayes, one with KNN, and none with K-Means. Although MC and Random Forest exhibited substantial ranking agreement, this does not necessarily indicate superior predictive accuracy due to differences in their scoring methods.

**Table 3. Ten Highest-Scoring Songs by Method for Base = 50**

| Rank | MC                                  | K-Means                       | KNN                           | Random Forest                       | Naive Bayes                       |
|------|-------------------------------------|-------------------------------|-------------------------------|-------------------------------------|-----------------------------------|
| 1    | Everything Is Sound (611.8869)      | Fuck You (0.7600)             | kanashi yume o mita (0.9863)  | Everything Is Sound (191.2012)      | Firework (0.9986)                 |
| 2    | For the First Time (610.4276)       | The Freedom Song (0.5873)     | St Louis Elegy (0.9838)       | For the First Time (191.1958)       | Best of You (0.9976)              |
| 3    | Firework (589.4630)                 | Ima Killer (0.5788)           | Airplanes (0.9817)            | Firework (188.8118)                 | Everything Is Sound (0.9908)      |
| 4    | In My Bones (588.4965)              | Stuck On You (0.5437)         | Black Dog (0.9799)            | In My Bones (188.3980)              | Breakeven (Falling t...) (0.9903) |
| 5    | Just a Kiss (588.2728)              | The Gravedigger's So (0.5352) | Bleeding Muddy Water (0.9793) | Just a Kiss (187.9973)              | Someday (Electric) (0.9864)       |
| 6    | Breaking the Rules (584.0558)       | Ode to Sad Disco (0.5220)     | Simple Things (0.9786)        | Breaking the Rules (186.0039)       | Nothin' on You / My (0.9859)      |
| 7    | Someday (Electric) (553.7161)       | hikari no furusato (0.5035)   | hikari no furusato (0.9781)   | Someday (Electric) (184.0009)       | For the First Time (0.9845)       |
| 8    | Best of You (547.9177)              | kanashi yume o mita (0.4820)  | With Arms Wide Open (0.9779)  | Best of You (183.6120)              | Walking in a Winter (0.9839)      |
| 9    | Breakeven (Falling t...) (539.8375) | 希望のうた (0.4662)                | Best of You (0.9779)          | Breakeven (Falling t...) (182.4409) | Will You Be There (0.9817)        |
| 10   | Still Here (525.6264)               | Everybody Knows My N (0.4458) | Faithfully (0.9768)           | Winter Wonderland (177.1960)        | Sweet As Whole (0.9811)           |

#### 4.6. Medium-Scale Samples

For the 100-node dataset, Table 4 shows that MC and Random Forest shared all ten top-ranked songs, despite differences in ordering. MC shared seven songs with Naive Bayes but none with K-Means or KNN. This substantial agreement between MC and Random Forest does not necessarily indicate equivalent predictive accuracy.

**Table 4. Ten Highest-Scoring Songs by Method for Base = 100**

| Rank | MC                               | K-Means                           | KNN                           | Random Forest                   | Naive Bayes                   |
|------|----------------------------------|-----------------------------------|-------------------------------|---------------------------------|-------------------------------|
| 1    | Everything Is Sound (1242.1023)  | Dilo Como Yo (Te Est...) (0.7540) | The Old Guitar (0.9861)       | Everything Is Sound (382.8001)  | Paradise - Acoustic (0.9986)  |
| 2    | Hate Song for Brains (1233.3545) | Fuck You (0.7088)                 | Walking in a Winter (0.9858)  | Hate Song for Brains (381.5857) | Hate Song for Brains (0.9965) |
| 3    | Beer (Acoustic) (1212.0468)      | Not There Yet (0.7050)            | I Heard The Bells On (0.9858) | Beer (Acoustic) (379.5913)      | Everything Is Sound (0.9958)  |
| 4    | Just a Kiss (1207.5206)          | Tiers monde (0.6713)              | Hate & Love (0.9843)          | I Was Gonna Marry Yo (378.7873) | Beer (Acoustic) (0.9953)      |
| 5    | I Was Gonna Marry Yo (1204.4192) | Knock Knock (0.6513)              | Bleeding Muddy Water (0.9819) | For the First Time (377.9954)   | Mean (0.9933)                 |
| 6    | Shimmer (1203.9266)              | Acid Rock (0.6183)                | Have Yourself a Merr (0.9819) | Just a Kiss (377.5563)          | I Was Gonna Marry Yo (0.9915) |
| 7    | For the First Time (1201.0501)   | My Weakness (0.6146)              | Black Dog (0.9814)            | Shimmer (377.1937)              | Ghost (0.9912)                |
| 8    | Don't Speak (1186.5364)          | Akula Ovu Onyarea (0.5817)        | I'll Never Get Over (0.9814)  | Don't Speak (375.9823)          | Hate & Love (0.9908)          |
| 9    | Paradise - Acoustic (1158.9211)  | Hardbover Hospital (0.5671)       | Coal War - from Sons (0.9813) | Paradise - Acoustic (373.6017)  | Shimmer (0.9897)              |
| 10   | In My Bones (1158.5089)          | Mother's Song (0.5649)            | Angels We Have Heard (0.9813) | In My Bones (371.9972)          | For the First Time (0.9890)   |

For the 500-node dataset, Table 5 shows that MC shared eight of its top ten songs with Random Forest, three with Naive Bayes, one with KNN, and none with K-Means. This substantial overlap with Random Forest indicates ranking similarity but does not establish predictive accuracy or algorithmic superiority.

**Table 5. Highest-Scoring Songs by Method for Base = 500**

| Rank | MC                               | K-Means                        | KNN                         | Random Forest                   | Naive Bayes                   |
|------|----------------------------------|--------------------------------|-----------------------------|---------------------------------|-------------------------------|
| 1    | Father's Day (1231.1520)         | Sunday Best - Live (2011.4572) | Cargo Cult (0.9886)         | Streetlight in the E (381.1601) | Streetlight in the E (0.9988) |
| 2    | Burn the House Down (1230.1687)  | Juan Locho - Area 52 (0.7592)  | Insecurity (0.9885)         | Father's Day (380.3549)         | Flying For The First (0.9985) |
| 3    | Counting Stars (1227.3837)       | Jacob's Ladder (0.7268)        | Kingdom of Heaven (0.9875)  | Pedals Press the Flo (379.9410) | Hobo Lullaby (0.9981)         |
| 4    | Streetlight in the E (1227.3507) | Stories To Tell (0.7254)       | Psalm 86 (0.9873)           | Counting Stars (379.9367)       | Pedals Press the Flo (0.9980) |
| 5    | Crocodile Rock (1226.6286)       | Master Maqui - Area (0.7234)   | I Can Only Imagine (0.9869) | Flying For The First (379.5551) | 1 vs 7 Billion (0.9978)       |
| 6    | They Can't If You Do (1219.0009) | Ob-La-Di (0.7211)              | Coffee Break (0.9861)       | Crocodile Rock (379.5521)       | Amen (0.9973)                 |
| 7    | 40 Miles (1216.1699)             | [Japonca başlık] (0.7189)      | So Long (0.9860)            | Burn the House Down (379.5473)  | Every Mother's Son (0.9972)   |
| 8    | Times Are Not Changi (1215.3512) | On Freedom (A Carol) (0.7185)  | Wayfaring Stranger (0.9857) | They Can't If You Do (379.1537) | Pinocchio's Demise (0.9968)   |
| 9    | Pedals Press the Flo (1214.9742) | And Still (0.7037)             | Goodbye (0.9853)            | God Gave Me You (379.1460)      | Make It Through (0.9964)      |
| 10   | Flying For The First (1212.0655) | Doctor (0.7027)                | Father's Day (0.9851)       | Healing Touch (378.7532)        | God Gave Me You (0.9963)      |

#### 4.7. Large-Scale Samples

For the 1,000-node dataset, Table 6 shows that MC and Random Forest shared 18 of their top 20 songs, while no overlap was observed with K-Means, KNN, or Naive Bayes. This substantial ranking similarity is noteworthy despite differences in the methods' scoring principles.

**Table 6. Highest-Scoring Songs by Method for Base = 1,000**

| Rank | MC                                   | K-Means                         | KNN                             | Random Forest                      | Naive Bayes                     |
|------|--------------------------------------|---------------------------------|---------------------------------|------------------------------------|---------------------------------|
| 1    | Together As One (feat...) (20120.00) | Sunday Best - Live a (20120.00) | Kingpin (9.78)                  | Together As One (fea...) (5030.00) | Piloto de Fuga - Leo (8.77)     |
| 2    | Baby (793.81)                        | Luta (20120.00)                 | 2012 - Feat. The Par (9.76)     | Baby (395.70)                      | Agita (feat. El Mich...) (8.28) |
| 3    | I'm Down (feat. Dee...) (91.81)      | Lights (20120.00)               | Dilo Como Yo (Te Est...) (9.73) | I'm Down (feat. Dee...) (44.45)    | 2012 - Feat. The Par (7.99)     |
| 4    | Everybody Talks (86.24)              | E Go Listen (feat. J...) (9.41) | Jericho (9.73)                  | Everybody Talks (40.81)            | E Go Listen (feat. J...) (7.95) |
| 5    | Immigrant Visa (82.29)               | Piloto de Fuga (9.34)           | March of the Idiots (9.73)      | Immigrant Visa (39.32)             | Yo Momma (7.85)                 |
| 6    | Bless the Lord (78.42)               | Thank You Lord (feat...) (9.28) | Lazy Bones Harold Bu... (9.73)  | Bless the Lord (37.60)             | Human Disco Ball (7.84)         |
| 7    | In The End (77.39)                   | Piloto de Fuga - Leo (9.24)     | La Bostelle (9.72)              | Courtesy Call (37.09)              | Tempos Insanos (7.74)           |
| 8    | Courtesy Call (77.38)                | Dancing Groove Featu... (9.19)  | 生きてる生きてく (9.71)                 | Rosemary (37.03)                   | Maga Traz a Lenha - (7.74)      |
| 9    | Rosemary (77.13)                     | Darker Shade of Blac... (9.09)  | Stella and Sam Theme (9.71)     | No Te Apartes de Mi (36.37)        | Maga Traz a Lenha (7.74)        |
| 10   | War of Change (75.87)                | Darker Shade of Pale (9.03)     | Yan-Nah Mama-Loo (9.70)         | In The End (36.35)                 | El Pacino (7.60)                |
| 11   | No Te Apartes de Mi (75.49)          | Agita (feat. El Mich...) (9.01) | J.B R.I.P (9.70)                | War of Change (35.72)              | Tempos Insanos - Fri (7.55)     |
| 12   | Lo malo de ser bueno (74.51)         | The Penetrator (9.00)           | Not There Yet (9.68)            | Lo malo de ser bueno (35.13)       | Fatty Boom Boom (7.54)          |
| 13   | Absolute Zero (73.88)                | 2012 - Feat. The Par (8.98)     | 健世の色 (9.68)                     | I Won't Give Up (34.76)            | Tempos Insanos - Cop (7.48)     |
| 14   | No Me Importa el Din... (72.62)      | Tempos Insanos - Cop (8.96)     | Soldier Boy (9.68)              | Absolute Zero (34.47)              | Mugabe (7.46)                   |
| 15   | Take a Walk (72.55)                  | Felicia (8.95)                  | Juliana (9.68)                  | No Me Importa el Din... (34.41)    | Thank You Lord (feat...) (7.42) |
| 16   | Entombed (72.53)                     | Legalise It (8.91)              | Best Days of Our Liv... (9.67)  | Take a Walk (34.39)                | Rapinoblada (7.39)              |
| 17   | Another Way Out (72.46)              | No Balanço de Canoa (8.87)      | Point of No Return (9.67)       | Entombed (34.38)                   | Movin (7.36)                    |
| 18   | Enamorado tuyo (72.26)               | Fatty Boom Boom (8.86)          | Jumbo (9.67)                    | Enamorado tuyo (34.06)             | Tá Amarrado (7.35)              |
| 19   | A Little Death (71.63)               | Tá Amarrado (8.81)              | Assomdwee (9.67)                | A Little Death (33.91)             | J.B R.I.P (7.33)                |
| 20   | Me Estoy Enamorando (71.27)          | Cyber Space Jam (8.73)          | Time for Me (9.67)              | Another Way Out (33.87)            | Rapinoblada - Dr. Da (7.23)     |

For the 5,000-node dataset, Table 7 shows that MC and Random Forest produced identical song-title rankings in the first 19 positions. Overlap with other methods was more limited. However, duplicate song titles and tied scores complicate ranking interpretation, and ranking similarity does not necessarily indicate predictive performance.

**Table 7. Highest-Scoring Songs by Method for Base = 5,000**

| Rank | MC                              | K-Means                         | KNN                             | Random Forest                  | Naive Bayes                      |
|------|---------------------------------|---------------------------------|---------------------------------|--------------------------------|----------------------------------|
| 1    | Together As One (20120.00)      | Sunday Best - Live a (20120.00) | 5 (10.00)                       | Together As One (5030.00)      | Don Giovanni (724320.00)         |
| 2    | Don Giovanni (20120.00)         | Luta (20120.00)                 | Thundercats (10.00)             | Don Giovanni (5030.00)         | All-night Vigil (645840.00)      |
| 3    | All-night Vigil (20120.00)      | Lights (20120.00)               | Taaveti laul 104 (10.00)        | All-night Vigil (5030.00)      | All-night Vigil (523120.00)      |
| 4    | 12 Trio Sonatas (20120.00)      | Fall (20120.00)                 | All-night Vigil (9.95)          | 12 Trio Sonatas (5030.00)      | 12 Trio Sonatas (482880.00)      |
| 5    | All-night Vigil (20120.00)      | Quizás (20120.00)               | All-night Vigil (9.94)          | All-night Vigil (5030.00)      | Taaveti laul 141 (482880.00)     |
| 6    | Octet in E Flat Majo (20120.00) | Ja (20120.00)                   | Komm (9.94)                     | Octet in E Flat Majo (5030.00) | Komm (462760.00)                 |
| 7    | Taaveti laul 141 (20120.00)     | It's Dark (20120.00)            | Taaveti laul 141 (9.90)         | Taaveti laul 141 (5030.00)     | Bach - 3 chorales (402400.00)    |
| 8    | Don Giovanni (20120.00)         | Sonata For Solo Cell (20120.00) | All-night Vigil (9.90)          | Don Giovanni (5030.00)         | Harold en Italie (382280.00)     |
| 9    | Komm (20120.00)                 | I Dreamed of Choirs (20120.00)  | Star Runner (9.84)              | Komm (5030.00)                 | Octet in E Flat Majo (342040.00) |
| 10   | All-night Vigil (20120.00)      | Night Woven of Snow (20120.00)  | P.I. High Theme (9.83)          | All-night Vigil (5030.00)      | All-night Vigil (342040.00)      |
| 11   | Don Giovanni (20120.00)         | Goodnight (20120.00)            | 情がついた夜 (9.82)                   | Don Giovanni (5030.00)         | Don Giovanni (321920.00)         |
| 12   | Harold en Italie (20120.00)     | I Don't Wanna Lose (20120.00)   | Bach - 3 chorales (9.80)        | Harold en Italie (5030.00)     | Concerto for Harpsic (301800.00) |
| 13   | Bach - 3 chorales (20120.00)    | Strobe Lights (20120.00)        | Lovergirls Precinct (9.79)      | Bach - 3 chorales (5030.00)    | Don Giovanni (241440.00)         |
| 14   | Dornröschen (20120.00)          | Strobe Lights (20120.00)        | Twinkle (Little Chri...) (9.79) | Dornröschen (5030.00)          | Dornröschen (221320.00)          |

|    |                                 |                                 |                             |                                |                                 |
|----|---------------------------------|---------------------------------|-----------------------------|--------------------------------|---------------------------------|
| 15 | Concerto for Harpsic (20120.00) | Etudes symphoniques (20120.00)  | Kingpin (9.78)              | Concerto for Harpsic (5030.00) | Nocturne in C-Sharp (105000.00) |
| 16 | Baby (793.81)                   | Bach - 3 chorales (20120.00)    | Always Something (9.78)     | Baby (395.70)                  | Lohengrin: Act III S (580.00)   |
| 17 | Symphony No. 88 in G (526.78)   | Das Wohltemperierte (20120.00)  | Zones Without People (9.77) | Symphony No. 88 in G (261.94)  | Bach (570.00)                   |
| 18 | Stills & Nash (463.68)          | Bach (20120.00)                 | 2012 - Feat. The Par (9.76) | Stills & Nash (230.92)         | Don Giovanni (500.00)           |
| 19 | River Deep (391.17)             | Nocturne in E-Flat M (20120.00) | 午夜電影 (9.76)                 | River Deep (194.79)            | Piloto de Fuga - Leo (8.77)     |
| 20 | Lohengrin: Act III S (290.00)   | 2 Pieces (20120.00)             | Dance Face 2000 (9.76)      | Concerto for Recorde (134.39)  | 2 Turntables and a C (8.60)     |

Across the six dataset sizes (10, 50, 100, 500, 1,000, and 5,000 nodes), MC and Random Forest exhibited substantial ranking overlap. Their top-ten song sets were identical for 10 and 100 nodes, with eight overlapping songs at 500 nodes and 18 of the top 20 at 1,000 nodes. At 5,000 nodes, the first 19 ranking positions contained identical song titles. Overlap with K-Means and KNN was generally limited, while Naive Bayes showed varying agreement. However, ranking similarity does not establish predictive accuracy or algorithmic superiority.

## 5. DISCUSSION OF FINDINGS

The experimental findings demonstrate that the algorithms exhibit different behaviors depending on dataset size and methodological characteristics. Random Forest effectively models complex relationships among acoustic features, consistent with previous studies (Saragih, 2023). Meanwhile, Naive Bayes provides probabilistic classification, KNN evaluates neighborhood relationships, and K-Means identifies acoustically similar groups. Since these methods have distinct objectives, their scores are not directly comparable.

The MC results indicate its potential for relational analysis of Spotify songs. Mean MC values increased from 2.4000 for 10 nodes to approximately 3.76–3.81 for 500–5,000 nodes, while the maximum reached 4.3868 at 5,000 nodes. Across the six dataset sizes (10, 50, 100, 500, 1,000, and 5,000 nodes), MC and Random Forest exhibited substantial ranking overlap. Their top-ten song sets were identical for 10 and 100 nodes, with nine overlapping songs at 50 nodes and eight at 500 nodes. Additionally, 18 of the top 20 songs overlapped at 1,000 nodes, while the first 19 positions contained identical song titles at 5,000 nodes. Agreement with the other algorithms was generally lower or varied across datasets.

Although these findings suggest that MC and Random Forest can identify similar tracks, ranking overlap does not establish predictive accuracy or algorithmic superiority. Differences in scoring mechanisms, potential duplicate records, and the influence of dataset size require careful interpretation. Future research should investigate score normalization, statistical relationships between MC and observed popularity, and the integration of MC-derived features into predictive models.

## **6. CONCLUSIONS AND FUTURE WORK**

This study comparatively evaluated MC, K-Means, KNN, Random Forest, and Naive Bayes for Spotify song popularity analysis across six dataset sizes (10, 50, 100, 500, 1,000, and 5,000 observations). The findings revealed substantial ranking overlap between MC and Random Forest, with identical top-ten song sets at 10 and 100 nodes, nine overlapping songs at 50 nodes, eight at 500 nodes, and 18 of the top 20 at 1,000 nodes. At 5,000 nodes, identical song titles appeared in the first 19 ranking positions. These results

highlight the potential of MC-inspired proximity measures for relational analysis of Spotify tracks. However, ranking similarity does not establish predictive accuracy or algorithmic superiority, as the methods serve different analytical objectives.

The main contribution of this study is the comparative assessment of MC-inspired relational proximity and conventional machine learning approaches across varying dataset sizes. Future research should explore MC score normalization, its relationship with song popularity, and its integration into predictive models. Larger datasets and independent validation procedures could further improve the reliability of the findings.

## REFERENCES

- Breiman, L. (2001). Random forests. *Machine Learning*, 45, 5–32. <https://doi.org/10.1023/A:1010933404324>
- Cover, T. M., & Hart, P. E. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), 21–27. <https://doi.org/10.1109/TIT.1967.1053964>
- Duman, D., Neto, P., Mavrolampados, A., Toivainen, P., & Luck, G. (2022). Music we move to: Spotify audio features and reasons for listening. *PLOS ONE*, 17(9), e0275228. <https://doi.org/10.1371/journal.pone.0275228>
- Hark, C. (2025). Using graph-based maximum independent sets with large language models for extractive text summarization. *Applied Sciences*, 15(12), 6395. <https://doi.org/10.3390/app15126395>
- Karci, A., Yakut, S., & Öztemiz, F. (2022). A new approach based on centrality value in solving the minimum vertex cover problem: Malatya Centrality Algorithm. *JCS*, 7(2), 81–88. <https://doi.org/10.53070/bbd.1195501>
- Li, K. (2024). Predicting song popularity in the digital age through Spotify's data. *Theoretical and Natural Science*, 39, 68–75. <https://doi.org/10.54254/2753-8818/39/20240600>
- Middlebrook, K., & Sheik, K. (2019). *Song hit prediction: Predicting Billboard hits using Spotify data* (arXiv:1908.08609). <https://doi.org/10.48550/arXiv.1908.08609>
- Özavci, A., & Yakut, S. (2023). Based on Malatya Centrality algorithm development of suggestion system in social platforms and commercial applications. In *Proceedings of the 7th International Artificial Intelligence and Data Processing Symposium (IDAP'23)*.

- Öztemiz, F. (2025). A greedy approach to solve maximum independent set problem: Differential Malatya independent set algorithm. *Engineering Science and Technology, an International Journal*, 63, 101995. <https://doi.org/10.1016/j.jestch.2025.101995>
- Öztemiz, F., & Yakut, S. (2024). An effective method for determining node dominance values: Malatya Centrality Algorithm. In *Proceedings of the 2024 32nd Signal Processing and Communications Applications Conference (SIU)* (pp. 1–4). <https://doi.org/10.1109/SIU61531.2024.10601155>
- Saragih, H. S. (2023). Predicting song popularity based on Spotify's audio features: Insights from the Indonesian streaming users. *Journal of Management Analytics*, 10(4), 693–709. <https://doi.org/10.1080/23270012.2023.2239824>
- Uckan, T., Aslan, C., & Hark, C. (2025). A comprehensive hybrid approach for indoor scene recognition combining CNNs and text-based features. *Sensors*, 25(17), 5350. <https://doi.org/10.3390/s25175350>
- Yakut, S., Öztemiz, F., & Karci, A. (2023a). A new approach based on centrality value in solving the maximum independent set problem: Malatya Centrality Algorithm. *JCS*, 8(1), 16–23. <https://doi.org/10.53070/bbd.1224520>
- Yakut, S., Öztemiz, F., & Karci, A. (2023b). A new robust approach to solve minimum vertex cover problem: Malatya vertex-cover algorithm. *The Journal of Supercomputing*, 79, 19746–19769. <https://doi.org/10.1007/s11227-023-05397-8>
- Yakut, S., Öztemiz, F., & Karci, A. (2025). Kendall rank correlation analysis of Malatya Centrality Algorithm

with well-known centrality measures. *Sigma Journal of Engineering and Natural Sciences*, 43(4), 1339.

Yao, H. (2024). Improving model generalization in songs' popularity prediction based on datasets with diverse distributions. In *Proceedings of the 1st International Conference on Modern Logistics and Supply Chain Management*.

Yunus, S., Hark, C., & Okumuş, F. (2025). Enhancing extractive multi-documents summarization with a novel dominating set model for semantic relationship detection. *Engineering Science and Technology, an International Journal*, 69, 102127. <https://doi.org/10.1016/j.jestch.2025.102127>

# **GRAPH-BASED MOVIE RECOMMENDATION USING MALATYA CENTRALITY: A COMPARATIVE STUDY**

**Selman YAKUT<sup>1</sup>**

## **INTRODUCTION**

The expansion of digital movie catalogs has increased the need for recommender systems that rank unwatched films according to users' viewing preferences and support personalized content discovery (Bozkurt & Acı, 2021; Raza et al., 2024). Content-based filtering uses item attributes, whereas collaborative filtering exploits user-item interactions (Darban & Valipour, 2022). Neighbor-based and clustering methods have also been applied to movie recommendation, while graph-based approaches incorporate relationships among movies into recommendation scoring (Elmas et al., 2025).

A methodological question is how relationships within a user's viewing history can inform candidate assessment beyond pairwise similarity. Malatya Centrality (MC) provides a means of evaluating relative relationships among connected nodes (Karcı et al., 2022). Building on Özavci Şahin and Yakut (2023), this study applies MC to a larger movie collection and compares its rankings with established similarity measures.

The proposed method constructs a weighted graph from previously watched movies. Edge weights reflect shared release

---

<sup>1</sup> Doçent Dr., İnönü Üniversitesi, Mühendislik Fakültesi, Yazılım Mühendisliği bölümü, ORCID: 0000-0002-0649-1993.

year, genre, country, runtime, and language attributes, while node strengths equal the sums of incident edge weights. Each candidate is temporarily added, scored using its strength relative to its neighbors, and removed before the next evaluation. This procedure avoids constructing a graph containing the entire movie collection.

Using the movies\_metadata dataset of 45,571 records, the study compares MC rankings with Jaccard similarity, cosine similarity, and Pearson correlation across candidate collections of different sizes. The analysis examines overlap and differences among the highest-ranked recommendations.

## **RELATED WORK**

Recommender systems employ content-based, collaborative, hybrid, and graph-based approaches across diverse domains. Applications include journal selection through text mining (Kartal et al., 2018), news recommendation (Özgöbek & Erdur, 2015). 2021). Pinar, Okumuş, Turgut, Kalıpsız, and Aktaş (2017) emphasized the importance of hyperparameter optimization when processing large datasets.

In movie recommendation, Bozkurt and Acı (2021) compared three systems using accuracy and computational speed. Nilla and Setiawan (2024) used movie reviews and sentiment analysis for content-based recommendation. Graph-based approaches include Neo4j modeling (Dalgın & Daş, 2024) and hybrid filtering designed to address sparsity (Darban & Valipour, 2022). Elmas, Yalcin, and Kaynar (2025) combined genre analysis with graph-based interactions, reporting lower prediction errors but weaker performance for rare genres.

Cold-start challenges have been addressed through deep learning and graph analysis (Kherad & Bidgoly, 2022). Chen et

al. (2024) introduced Graph Neural Patching, combining GWarmer and Patching Networks to support existing and cold-start users and items. Raza et al. (2024) reviewed practical recommendation challenges and emerging graph- and LLM-based approaches.

Centrality-based methods have been investigated for citation recommendation (Samad et al., 2019) and movie recommendation using MC (Özavci Şahin & Yakut, 2023). Related research examined publication similarity (Öztemiz & Karıcı, 2020), MC rankings relative to established centrality measures (Yakut et al., 2025), and node dominance (Öztemiz & Yakut, 2024). Broader applications include dominating-set and maximum-independent-set approaches to text summarization (Hark, 2025; Yunus et al., 2025), alongside hybrid CNN and textual-feature methods for scene recognition (Uckan, Aslan, & Hark, 2025).

Overall, the literature highlights opportunities for incorporating structural relationships into recommendation, alongside persistent sparsity, cold-start, and computational challenges. Previous MC applications motivate the present comparison with established similarity measures, while recommendation quality and scalability require further systematic evaluation.

## **MATERIALS AND METHODS**

The MC-based recommendation approach was compared with Jaccard similarity, cosine similarity, and Pearson correlation using the `movies_metadata` dataset of 45,571 records. Identical preprocessing procedures were applied across all methods.

A user-specific weighted graph was constructed from ten previously watched movies, whose attributes are presented in

Table 1. Nodes represented movies, and edge weights reflected shared release year, genre, country, runtime, and language attributes. Runtime differences of no more than 10 minutes counted as a match, and edges were established for positive weights.

Each node's weighted strength was calculated by summing its incident edge weights and used in MC-based scoring. These strengths are presented in Table 2.

**Table 1. Previously watched movies and their attributes**

| <b>Movie</b>                       | <b>Year</b> | <b>Genre</b>                           | <b>Duration</b> | <b>Country</b> | <b>Language</b> |
|------------------------------------|-------------|----------------------------------------|-----------------|----------------|-----------------|
| <b>Toy Story</b>                   | 1995        | Animation,<br>Comedy, Family           | 81 min          | USA            | English         |
| <b>Jumanji</b>                     | 1995        | Adventure,<br>Fantasy, Family          | 104 min         | USA            | English         |
| <b>Grumpier Old Men</b>            | 1995        | Romance,<br>Comedy                     | 101 min         | USA            | English         |
| <b>Waiting to Exhale</b>           | 1995        | Comedy, Drama,<br>Romance              | 127 min         | USA            | English         |
| <b>Father of the Bride Part II</b> | 1995        | Comedy                                 | 106 min         | USA            | English         |
| <b>Heat</b>                        | 1995        | Action, Crime,<br>Drama, Thriller      | 170 min         | USA            | English         |
| <b>Sabrina</b>                     | 1995        | Comedy,<br>Romance                     | 127 min         | Germany        | English         |
| <b>Tom and Huck</b>                | 1995        | Action,<br>Adventure,<br>Drama, Family | 97 min          | USA            | English         |
| <b>Sudden Death</b>                | 1995        | Action,<br>Adventure,<br>Thriller      | 106 min         | USA            | English         |
| <b>GoldenEye</b>                   | 1995        | Adventure,<br>Action, Thriller         | 130 min         | UK             | English         |

Each movie's weighted node strength was calculated by summing its incident edge weights and used in MC-based scoring. Table 2 presents these strengths for the ten previously watched movies.

**Table 2. Weighted node strengths of previously watched movies**

| Movie                       | Node Value |
|-----------------------------|------------|
| Toy Story                   | 31         |
| Jumanji                     | 34         |
| Grumpier Old Men            | 35         |
| Waiting to Exhale           | 35         |
| Father of the Bride Part II | 33         |
| Heat                        | 32         |
| Sabrina                     | 26         |
| Tom and Huck                | 39         |
| Sudden Death                | 37         |
| GoldenEye                   | 28         |

After constructing the viewing-history graph, the movies\_metadata dataset of 45,571 records was loaded, and unwatched movies were evaluated as candidates:

1. Each candidate was temporarily added and connected to previously watched movies through positive-weight edges based on shared attributes.
2. The candidate's and its neighbors' weighted strengths were calculated, and its MC score was stored.
3. The candidate and its edges were removed, restoring the original graph.

Candidates were ranked by descending MC scores, and the top ten were recommended. Figure 1 illustrates graph construction and candidate evaluation.

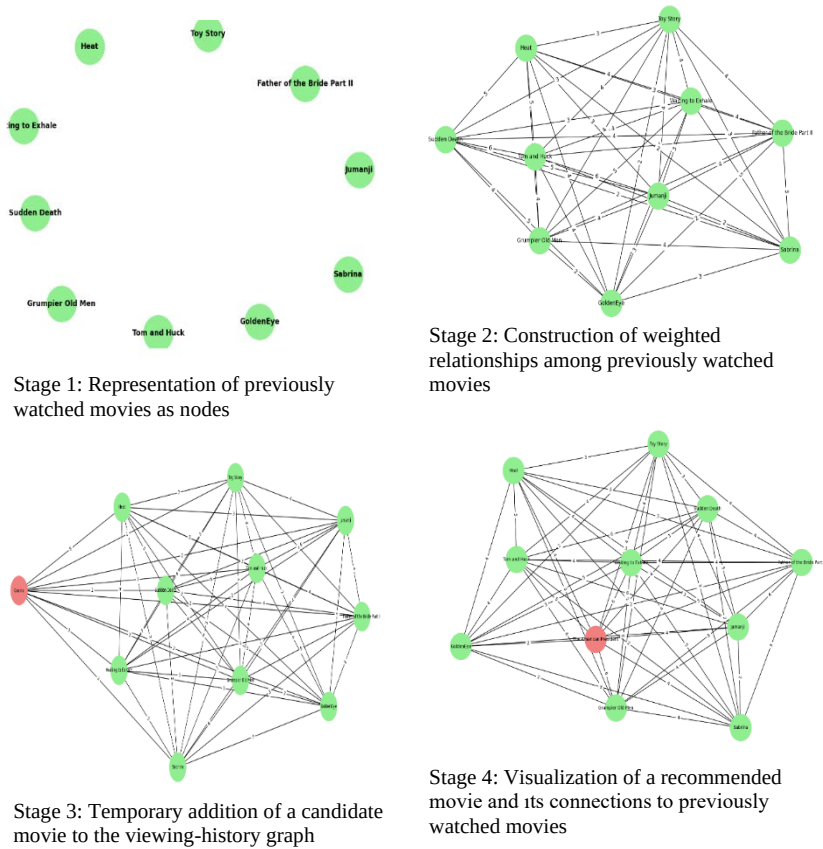


**Figure 1. Weighted graph construction and candidate evaluation for MC-based movie recommendations**

### **3.1. Application of the MC in Movie Recommendation Systems**

MC evaluates nodes using their degrees relative to those of their neighbors and has been applied to graph optimization problems (Yakut et al., 2023a, 2023b). Building on Özavci Şahin and Yakut (2023), this study replaces degree with weighted node strength. Previously watched movies form a graph weighted by shared attributes. Each unwatched candidate is temporarily

added, connected through positive-weight edges, and scored using its strength relative to those of its neighbors.

Let  $u$  denote the target user,  $c$  a candidate movie, and  $MC_u$  the augmented graph obtained by adding  $c$  to the user's viewing-history graph. The weighted strength of a movie node  $v$  in  $G_c$  is defined as the sum of the weights of its incident edges:

$$s_{G_c}(v) = \sum_{z \in N_{G_c}(v)} w(v, z) \quad (1)$$

Here,  $v$  and  $z$  represent movie nodes,  $N_{G_c}(v)$  denotes the set of nodes directly connected to  $v$  in  $G_c$  and  $w(v, z)$  denotes the edge weight calculated from the shared attributes of movies  $v$  and  $z$ . Consequently,  $s_{G_c}(v)$ , represents the aggregate weight of the connections associated with  $v$ , rather than its number of neighbors. When a candidate movie is added, the strengths of its connected neighbors are also updated to include the newly established edges.

The MC-based recommendation score of candidate movie  $c$  for user  $u$  is calculated as follows:

$$MC_u(x_i) = \sum_{v \in N_{G_c}(v)} \frac{s_{G_c}(v)}{s_{G_c}(v)} \quad (2)$$

In Equation (2),  $N_{G_c}(v)$  denotes the set of previously watched movies connected to the candidate, while  $MC_u(c)$  represents its user-specific recommendation score. The MC score sums the ratios of the candidate's weighted strength to those of its neighbors; unconnected candidates receive zero. After scoring, the candidate and its edges are removed to restore the original graph. All candidates are ranked by descending MC scores, and the top ten are recommended.

Algorithm 1 constructs a weighted graph from previously watched movies using shared attributes. Each candidate is

temporarily added, scored by summing its strength-to-neighbor-strength ratios, and removed, ensuring independent evaluation against the same viewing-history graph.

### **Algorithm 1. Pseudocode for MC-Based Movie Recommendation**

---

```
1. Input:
   U = set of user movies
   E = set of Excel movies (candidate movies)
2. Output:
   MC score for each candidate movie
3. BuildGraph(G, U){ // create base graph from user movies
4.   For each movie m in U
5.     AddNode(G, m)
6.   For each pair (m1, m2) in U
7.     w <- CalculateEdgeWeight(m1, m2)
8.     If w > 0 then AddEdge(G, m1, m2, weight=w)
9.   return G
10.}
11. CalculateEdgeWeight(m1, m2){ // similarity-based weight
12.   weight <- 0
13.   If m1.year = m2.year then weight++
14.   weight <- weight + |genres (m1) ∩ genres (m2)|
15.   If |m1.duration - m2.duration| ≤ 10 then weight++
16.   If m1.country = m2.country then weight++
17.   If m1.language = m2.language then weight++
18.   return weight
19.}
20. NodeWeight(G, node){
21.   return sum(edge.weight for edges of node)
22.}
23. MalatyaCentrality(G, node){ // proposed centrality measure
24.   d_vi <- NodeWeight(G, node)
25.   For each neighbor vj of node
26.     d_vj <- NodeWeight(G, vj)
27.     If d_vj > 0 then score <- score + (d_vi / d_vj)
28.   return score
29.}
30. MainAlgorithm(U, E){
31.   G <- BuildGraph(G, U)
32.   For each movie m in E
33.     AddNode(G, m)
34.     For each node n in U
35.       w <- CalculateEdgeWeight(m, n)
36.       If w > 0 then AddEdge(G, m, n, weight=w)
37.     malatya <- MalatyaCentrality(G, m)
38.     Print(m, malatya) // output result
39.     VisualizeGraph(G, highlight=m)
40.     RemoveNode(G, m)
41.   EndFor
42.   return Results
43.}
44. End
```

---

### **3.2. Application of Jaccard Similarity in Movie Recommendation Systems**

Jaccard similarity measures the overlap between two sets of movie attributes. Let (A) and (B) represent the attribute sets of two movies. Their similarity is calculated as the number of shared elements divided by the number of distinct elements in their union:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (3)$$

Here,  $A \cap B$  denotes the number of attributes shared by both movies, whereas  $A \cup B$  denotes the total number of distinct attributes across the two sets. Attribute values are represented as feature-specific elements, such as genre—comedy or language—English. The similarity score ranges from 0 to 1, with higher values indicating greater overlap.

Each candidate's Jaccard similarities with neighboring movies in the viewing-history graph are summed to obtain its recommendation score. Candidates are ranked by descending scores, and the top ten are recommended.

### **3.3. Application of Cosine Similarity in Movie Recommendation Systems**

Cosine similarity compares movie feature vectors, with categorical attributes numerically encoded in a shared feature space. Let  $x_A$  and  $x_B$  denote the feature vectors of movies A and B, respectively. Their cosine similarity is defined as:

$$\text{CosSim}(A, B) = \frac{x_A \cdot x_B}{\|x_A\|_2 \|x_B\|_2} \quad (4)$$

Here,  $x_A \cdot x_B$  denotes the dot product of the two vectors, while  $\|x_A\|_2$  and  $\|x_B\|_2$  denote their Euclidean norms. For nonzero vectors, cosine similarity ranges from  $-1$  to  $1$ ; when all feature values are nonnegative, it ranges from  $0$  to  $1$ . Higher values indicate greater alignment between the movie feature vectors.

Each candidate's cosine similarities with neighboring movies in the viewing-history graph are summed to obtain its recommendation score. Candidates are ranked by descending scores, and the top ten are recommended..

### **3.4. Application of Pearson Correlation in Movie Recommendation Systems**

Pearson correlation assesses the linear similarity between candidate and previously watched movies, ranging from  $-1$  (perfect negative correlation) to  $1$  (perfect positive correlation), with  $0$  indicating no linear correlation. Movie attributes, including categorical features, are numerically encoded in a shared feature space.

Let  $x_A = (x_{A,1}, \dots, x_{A,p})$  and  $x_B = (x_{B,1}, \dots, x_{B,p})$  denote the feature vectors of movies A and B, respectively, where  $p$  is the number of vector components. Pearson correlation is calculated as follows:

$$r(A, B) = \frac{\sum_{i=1}^p (x_{A,i} - \bar{x}_A)(x_{B,i} - \bar{x}_B)}{\sqrt{\sum_{i=1}^p (x_{A,i} - \bar{x}_A)^2} \sqrt{\sum_{i=1}^p (x_{B,i} - \bar{x}_B)^2}} \quad (5)$$

Here,  $x_{A,i}$  and  $x_{B,i}$  denote the values of the (i)th feature components, while  $\bar{x}_A$  and  $\bar{x}_B$  represent the mean component values of the respective vectors. The coefficient therefore compares the feature vectors after centering each vector around its mean.

Each candidate's Pearson correlations with neighboring movies in the viewing-history graph are summed to obtain its recommendation score. Candidates are ranked by descending scores, and the top ten are recommended.

## **4. EXPERIMENTAL RESULTS**

The MC-based method was compared with Jaccard similarity, cosine similarity, and Pearson correlation using the

movies\_metadata dataset of 45,571 records. Previously watched movies formed a weighted graph based on shared attributes, with node strengths calculated from incident edge weights. The methods' top-ten rankings were examined for candidate collections of 10 movies, 100 movies, and the full dataset.

For the ten candidates (Table 3), MC and Jaccard ranked The American President, Money Train, and Balto identically, while differing in the positions of several other movies. Cosine and Pearson produced identical rankings, placing Cutthroat Island first, compared with seventh under MC and sixth under Jaccard. Since every list contained the same ten movies, these results demonstrate differences in ordering rather than selection.

**Table 3. Rankings Produced by the Four Recommendation Methods for 10 Candidate Movies**

|    | MC                             | Jaccard                        | Kosinüs                        | Pearson                        |
|----|--------------------------------|--------------------------------|--------------------------------|--------------------------------|
| 1  | The American President         | The American President         | Cutthroat Island               | Cutthroat Island               |
| 2  | Money Train                    | Money Train                    | The American President         | The American President         |
| 3  | Ace Ventura: When Nature Calls | Four Rooms                     | Money Train                    | Money Train                    |
| 4  | Four Rooms                     | Ace Ventura: When Nature Calls | Four Rooms                     | Four Rooms                     |
| 5  | Balto                          | Balto                          | Sense and Sensibility          | Sense and Sensibility          |
| 6  | Nixon                          | Cutthroat Island               | Ace Ventura: When Nature Calls | Ace Ventura: When Nature Calls |
| 7  | Cutthroat Island               | Sense and Sensibility          | Dracula: Dead and Loving It    | Dracula: Dead and Loving It    |
| 8  | Sense and Sensibility          | Nixon                          | Balto                          | Balto                          |
| 9  | Dracula: Dead and Loving It    | Dracula: Dead and Loving It    | Casino                         | Casino                         |
| 10 | Casino                         | Casino                         | Nixon                          | Nixon                          |

For the 100 candidates (Table 4), MC and Jaccard shared eight top-ten recommendations, with The Big Green ranked first by both methods. Other shared movies differed in position, while MC uniquely selected To Die For and Get Shorty, and Jaccard selected Mighty Aphrodite and Friday.

Cosine and Pearson produced identical top-ten rankings, with no overlap with either MC or Jaccard. These differences illustrate the influence of the scoring method on recommendation selection, but rankings alone cannot establish which method better reflects user preferences.

**Table 4. Top-10 Recommendations Produced by the Four Methods for 100 Candidate Movies**

|    | Proposed Method (MC)   | Jaccard                | Kosinüs                                  | Pearson                                  |
|----|------------------------|------------------------|------------------------------------------|------------------------------------------|
| 1  | The Big Green          | The Big Green          | Once Upon a Time... When We Were Colored | Once Upon a Time... When We Were Colored |
| 2  | To Die For             | The American President | Taxi Driver                              | Taxi Driver                              |
| 3  | Now and Then           | Clueless               | Georgia                                  | Georgia                                  |
| 4  | The American President | Home for the Holidays  | How To Make An American Quilt            | How To Make An American Quilt            |
| 5  | Clueless               | Kicking and Screaming  | Things to Do in Denver When You're Dead  | Things to Do in Denver When You're Dead  |
| 6  | Home for the Holidays  | It Takes Two           | Restoration                              | Restoration                              |
| 7  | Kicking and Screaming  | Mighty Aphrodite       | Leaving Las Vegas                        | Leaving Las Vegas                        |
| 8  | It Takes Two           | Friday                 | Beautiful Girls                          | Beautiful Girls                          |
| 9  | Money Train            | Now and Then           | The Juror                                | The Juror                                |
| 10 | Get Shorty             | Money Train            | Angels and Insects                       | Angels and Insects                       |

For the full dataset (Table 5), MC and Jaccard shared only The Big Green and Operation Dumbo Drop, ranked first and second under MC but sixth and seventh under Jaccard, respectively.

Cosine and Pearson selected the same ten movies, differing only in fourth and fifth positions. Neither list overlapped with MC or Jaccard, highlighting distinct recommendation outputs across the scoring methods.

**Table 5. Top 10 Recommended Movies Based on the Application of Four Methods**

|    | MC                                          | Jaccard                                          | Kosinüs                                  | Pearson                                  |
|----|---------------------------------------------|--------------------------------------------------|------------------------------------------|------------------------------------------|
| 1  | The Big Green                               | Bye Bye Love                                     | Crooklyn                                 | Crooklyn                                 |
| 2  | Operation Dumbo Drop                        | Nine Months                                      | I'll Do Anything                         | I'll Do Anything                         |
| 3  | Bushwhacked                                 | Something to Talk About                          | The Philadelphia Story                   | The Philadelphia Story                   |
| 4  | Congo                                       | To Wong Foo, Thanks for Everything! Julie Newmar | Daylight                                 | Once Upon a Time... When We Were Colored |
| 5  | Fall Time                                   | Tommy Boy                                        | Once Upon a Time... When We Were Colored | Daylight                                 |
| 6  | To Die For                                  | The Big Green                                    | Little Women                             | Little Women                             |
| 7  | A Goofy Movie                               | Operation Dumbo Drop                             | Only You                                 | Only You                                 |
| 8  | First Knight                                | The American President                           | The Man without a Face                   | The Man without a Face                   |
| 9  | Far from Home: The Adventures of Yellow Dog | Clueless                                         | Rudy                                     | Rudy                                     |
| 10 | Hackers                                     | Home for the Holidays                            | Kim                                      | Kim                                      |

The scoring method affects recommendation rankings and selection. MC–Jaccard overlap decreases with larger candidate collections, while cosine and Pearson remain closely aligned. Further evaluation using held-out user preferences and computational measurements is needed to assess accuracy and efficiency.

## 5. DISCUSSION AND FINDINGS

The results show that the scoring method affects both recommendation ordering and selection. MC and Jaccard shared eight top-ten movies for 100 candidates, but only two for the full dataset. This reduced overlap indicates different recommendation behavior as the candidate collection expands; it does not establish which method provides better recommendations.

These differences reflect the scoring mechanisms. Jaccard measures attribute overlap, whereas MC evaluates a candidate's

weighted strength relative to those of its neighbors, incorporating relationships within the viewing-history graph. Lower-strength neighbors contribute larger ratios for a fixed candidate strength. Explaining specific ranking differences, however, requires examining the corresponding edge weights and node strengths.

Cosine and Pearson produced identical rankings for the smaller collections and nearly identical rankings for the full dataset. This agreement may reflect the feature representation, but it does not imply identical scores or values close to one. Feature encoding, scaling, and score distributions require further examination.

Evaluating candidates through temporary insertion and removal avoids constructing a graph for the entire catalog, although all eligible candidates are still assessed. Since the study reports neither held-out user evaluations nor runtime and memory benchmarks, its findings characterize ranking behavior.

## **6. CONCLUSION AND FUTURE STUDY**

This study examined MC-based movie recommendation using a weighted viewing-history graph, where shared attributes defined edge weights and node strengths determined candidate scores. Candidates were temporarily inserted, evaluated, and removed individually.

Using 45,571 movie records, MC rankings were compared with Jaccard, cosine, and Pearson. MC–Jaccard top-ten overlap decreased to two movies for the full dataset, while cosine and Pearson remained closely aligned. These findings demonstrate differences in ranking behavior without establishing superior accuracy or computational efficiency.

Future work should include multi-user evaluation with held-out preferences, runtime and memory measurements, and

comparisons with weighted-strength scoring. Feature ablation and weighting experiments could further clarify MC's contribution to recommendation.

## REFERENCES

- Bozkurt, M., & Acı, Ç. İ. (2021). Öneri algoritmalarının film önerme problemi üzerinde karşılaştırılması: MovieLens örneği. *Bilgisayar Bilimleri ve Teknolojileri Dergisi*, 2(2), 36–42.
- Chen, H., Yang, Y., Bei, Y., Wang, Z., Xu, Y., & Huang, F. (2024). Graph neural patching for cold-start recommendations. In *Australasian Database Conference* (pp. 334–346). Singapore: Springer Nature Singapore.
- Dalgın, G. T., & Daş, R. (2024). Sinema verilerinin Neo4j çizge veritabanı ile modellenmesi ve analizi. *Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi*, 15(1), 1–13.
- Darban, Z. Z., & Valipour, M. H. (2022). GHRS: Graph-based hybrid recommendation system with application to movie recommendation. *Expert Systems with Applications*, 200, 116850.
- Elmas, Y., Yalcin, E., & Kaynar, O. (2025, February). *Kişiselleştirilmiş film önerileri için hibrit öneri modelleri: Tür ve grafik tabanlı bir yaklaşım*. Paper presented at the 4. BİLSEL International Aspendos Scientific Researches Congress, Antalya, Turkey.
- Hark, C. (2025). Using graph-based maximum independent sets with large language models for extractive text summarization. *Applied Sciences*, 15(12), 6395. doi:10.3390/app15126395
- Karci, A., Yakut, S., & Öztemiz, F. (2022). A new approach based on centrality value in solving the minimum vertex cover problem: Malatya Centrality Algorithm. *JCS*, 7(2), 81–88. doi:10.53070/bbd.1195501

- Kherad, M., & Bidgoly, A. J. (2022). Recommendation system using a deep learning and graph analysis approach. *Computational Intelligence*, 38(5), 1859–1883.
- Nilla, A., & Setiawan, E. B. (2024). Film recommendation system using content-based filtering and the convolutional neural network (CNN) classification methods. *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, 10(1), 17.
- Özavci Şahin, A., & Yakut, S. (2023). Based on Malatya Centrality Algorithm development of suggestion system in social platforms and commercial applications. In *Proceedings of the 7th International Artificial Intelligence and Data Processing Symposium (IDAP'23)*.
- Özgöbek, Ö., & Erdur, R. C. (2015). Öneri sistemleri ve bir uygulama alanı olarak haber öneri sistemleri. Paper presented at the Akademik Bilişim Konferansı, Eskişehir, Turkey.
- Öztemiz, F., & Karıcı, A. (2020). Akademik yazarların yayınları arasındaki ilişkinin sosyal ağ benzerlik yöntemleri ile tespiti. *Uludağ Üniversitesi Mühendislik Fakültesi Dergisi*, 25(1), 591–608. doi:10.17482/uumfd.533476
- Öztemiz, F., & Yakut, S. (2024). An effective method for determining node dominance values: Malatya Centrality Algorithm. In *2024 32nd Signal Processing and Communications Applications Conference (SIU)* (pp. 1–4). IEEE. doi:10.1109/SIU61531.2024.10601155
- Pınar, M., Okumuş, O., Turgut, U. O., Kalıpsız, O., & Aktaş, M. S. (2017). Büyük veri içeren öneri sistemleri için hiperparametre optimizasyonu. In *Ulusal Yazılım Mühendisliği Sempozyumu*.
- Raza, S., Rahman, M., Kamawal, S., Toroghi, A., Raval, A., Navah, F., & Kazemeini, A. (2024). A comprehensive

*review of recommender systems: Transitioning from theory to practice* (Technical report).

- Samad, M. A., Islam, M. A., Iqbal, M. A., & Aleem, M. (2019). Centrality based paper citation recommender system. *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, 6(19), e2.
- Uckan, T., Aslan, C., & Hark, C. (2025). A comprehensive hybrid approach for indoor scene recognition combining CNNs and text-based features. *Sensors*, 25(17), 5350. doi:10.3390/s25175350
- Yakut, S., Öztemiz, F., & Karci, A. (2023a). A new approach based on centrality value in solving the maximum independent set problem: Malatya Centrality Algorithm. *JCS*, 8(1), 16–23. doi:10.53070/bbd.1224520
- Yakut, S., Öztemiz, F., & Karci, A. (2023b). A new robust approach to solve minimum vertex cover problem: Malatya vertex-cover algorithm. *The Journal of Supercomputing*, 79, 19746–19769. doi:10.1007/s11227-023-05397-8
- Yakut, S., Öztemiz, F., & Karci, A. (2025). Kendall rank correlation analysis of Malatya Centrality Algorithm with well-known centrality measures. *Sigma Journal of Engineering and Natural Sciences*, 43(4), 1339–1354. doi:10.14744/sigma.2025.00124
- Yunus, S., Hark, C., & Okumuş, F. (2025). Enhancing extractive multi-documents summarization with a novel dominating set model for semantic relationship detection. *Engineering Science and Technology, an International Journal*, 69, 102127. doi:10.1016/j.jestch.2025.102127

**BİLGİSAYAR BİLİMLERİ VE MÜHENDİSLİĞİ ALANINDA  
GÜNCEL KONULAR VE AKADEMİK DEĞERLENDİRMELER**

**yaz**  
yayınları

YAZ Yayınları  
M.İhtisas OSB Mah. 4A Cad. No:3/3  
İscehisar / AFYONKARAHİSAR  
Tel : (0 531) 880 92 99  
yazyayinlari@gmail.com • www.yazyayinlari.com