



# SAUCIS

Sakarya University Journal of Computer and Information Sciences

e-ISSN 2636-8129 | VOLUME 4 | NUMBER 1 | APRIL 20201

*Performance Evaluation of MANET Routing Protocols AODV and DSDV Using NS2 Simulator*

*A Study on the Efficacy of Deep Reinforcement Learning for Intrusion Detection*

*Detection of Pneumonia with a Novel CNN-based Approach*

*Sentiment Analysis on Social Media Reviews Datasets with Deep Learning Approach*

*Determination of The Critical Success Factors in Disaster Management Through  
The Text Mining Assisted Ahp Approach*

*Performance Assessment of a Turn Around Ranging in Communication Satellite Orbit  
Determination*

*A Conjoint Analysis of Propellant Budget and Maneuver Life for a Communication Satellite*

*Electrical-Thermal-Mechanical Analysis of  
Cable Connection with Screw-Connected  
Terminal Strips Using Finite Element  
Method*

*A V-Model Software Development  
Application for Sustainable and Smart  
Campus Analytics Domain*

*Effect of the Chaotic Crossover Operator on  
Breeding Swarms Algorithm*

*Deep Neural Networks Based on Transfer  
Learning Approaches to Classification of  
Gun and Knife Images*

*Deep Learning Methods for Autism  
Spectrum Disorder Diagnosis Based on  
fMRI Images*

*Designing a Data Warehouse for  
Earthquake Risk Assessment of Buildings:  
A Case Study for Healthcare Facilities*





# SAUCIS

**Sakarya University Journal of Computer and Information Sciences**  
**Volume: 4 – Issue No: 1 (April 2021)**  
<http://saucis.sakarya.edu.tr/issue/59732>

## Editor in Chief

Nejat Yumusak, Sakarya University, Turkey, [nyumusak@sakarya.edu.tr](mailto:nyumusak@sakarya.edu.tr)

## Associate Editors

Muhammed Fatih Adak, Sakarya University, Turkey, [fatihadak@sakarya.edu.tr](mailto:fatihadak@sakarya.edu.tr)

Mustafa Akpınar, Sakarya University, Turkey, [akpinar@sakarya.edu.tr](mailto:akpinar@sakarya.edu.tr)

Unal Cavusoglu, Sakarya University, Turkey, [unalc@sakarya.edu.tr](mailto:unalc@sakarya.edu.tr)

Veysel Harun Sahin, Sakarya University, Turkey, [vsahin@sakarya.edu.tr](mailto:vsahin@sakarya.edu.tr)

## Editorial Assistants - Secretary

Deniz Balta, Sakarya University, Turkey, [ddural@sakarya.edu.tr](mailto:ddural@sakarya.edu.tr)

Fatma Akalin, Sakarya University, Turkey, [fatmaakalin@sakarya.edu.tr](mailto:fatmaakalin@sakarya.edu.tr)

Gozde Yolcu Oztel, Sakarya University, Turkey, [gyolcu@sakarya.edu.tr](mailto:gyolcu@sakarya.edu.tr)

Ibrahim Delibasoglu, Sakarya University, Turkey, [ibrahimdelibasoglu@sakarya.edu.tr](mailto:ibrahimdelibasoglu@sakarya.edu.tr)

Muhammed Kotan, Sakarya University, Turkey, [mkotan@sakarya.edu.tr](mailto:mkotan@sakarya.edu.tr)

Sumeyye Kaynak, Sakarya University, Turkey, [sumeyye@sakarya.edu.tr](mailto:sumeyye@sakarya.edu.tr)

## Editorial Board

Ahmet Ozmen, Sakarya University, Turkey, [ozmen@sakarya.edu.tr](mailto:ozmen@sakarya.edu.tr)

Aref Yelghi, Istanbul Ayvansaray University, [ar.yelqi@gmail.com](mailto:ar.yelqi@gmail.com)

Ayhan Istanbulu, Balikesir University, Turkey, [iayhan@balikesir.edu.tr](mailto:iayhan@balikesir.edu.tr)

Aysegul Alaybeyoglu, Izmir Katip Celebi University, Turkey, [alaybeyoglu@gmail.com](mailto:alaybeyoglu@gmail.com)

Bahadir Karasulu, Canakkale Onsekiz Mart University, [bahadirkarasulu@comu.edu.tr](mailto:bahadirkarasulu@comu.edu.tr)

Celal Ceken, Sakarya University, Turkey, [celalceken@sakarya.edu.tr](mailto:celalceken@sakarya.edu.tr)

Cihan Karakuzu, Bilecik Seyh Edebali University, [cihan.karakuzu@bilecik.edu.tr](mailto:cihan.karakuzu@bilecik.edu.tr)

Fahri Vatansever, Bursa Uludag University, [fahriv@uludag.edu.tr](mailto:fahriv@uludag.edu.tr)

Ibrahim Turkoglu, Firat University, Turkey, [iturkoglu@firat.edu.tr](mailto:iturkoglu@firat.edu.tr)

Levent Alhan, Sakarya University, Turkey, [leventalhan@sakarya.edu.tr](mailto:leventalhan@sakarya.edu.tr)

Kamal Z Zamli, Malaysia Pahang University, Malaysia, [kamalz@ump.edu.my](mailto:kamalz@ump.edu.my)

Muhammed Fatih Adak, Sakarya University, Turkey, [fatihadak@sakarya.edu.tr](mailto:fatihadak@sakarya.edu.tr)

Mustafa Akpınar, Sakarya University, Turkey, [akpinar@sakarya.edu.tr](mailto:akpinar@sakarya.edu.tr)



# SAUJCIS

## **Editorial Board (Cont.)**

Nuri Yilmazer, Texas A&M University, US, [nuri.yilmazer@tamuk.edu](mailto:nuri.yilmazer@tamuk.edu)

Orhan Er, Bozok University, Turkey, [orhan.er@bozok.edu.tr](mailto:orhan.er@bozok.edu.tr)

Priyadip Ray, Lawrence Livermore National Laboratory, [priyadipr@gmail.com](mailto:priyadipr@gmail.com)

Resul Das, Firat University, Turkey, [rdas@firat.edu.tr](mailto:rdas@firat.edu.tr)

Veysel Harun Sahin, Sakarya University, Turkey, [vsahin@sakarya.edu.tr](mailto:vsahin@sakarya.edu.tr)



# SAUCIS

Sakarya University Journal of Computer and Information Sciences  
Volume: 4 – Issue No: 1 (April 2021)  
<http://saucis.sakarya.edu.tr/issue/59732>

## Contents


Author(s), Paper Title	Pages
<i>Amenah Mhmood, Ahmet Zengin,</i> Performance Evaluation of MANET Routing Protocols AODV and DSDV Using NS2 Simulator	1-10
<i>Halim Gorkem Gulmez, Pelin Angin,</i> A Study on the Efficacy of Deep Reinforcement Learning for Intrusion Detection	11-25
<i>Ebru Erdem, Tolga Aydin,</i> Detection of Pneumonia with a Novel CNN-based Approach	26-34
<i>Muhammet Sinan Basarlan, Fatih Kayaalp,</i> Sentiment Analysis on Social Media Reviews Datasets with Deep Learning Approach	35-49
<i>Halil İbrahim Cebeci, Yasemin Korkut,</i> Determination of The Critical Success Factors in Disaster Management Through The Text Mining Assisted Ahp Approach	50-72
<i>İbrahim Oz, Umit Cezmi Yilmaz, Umit Guler,</i> Performance Assessment of a Turn Around Ranging in Communication Satellite Orbit Determination	73-83
<i>Ibrahim Oz,</i> A Conjoint Analysis of Propellant Budget and Maneuver Life for a Communication Satellite	84-95
<i>Mustafa Tosun, Huseyin Aksoy,</i> Electrical-Thermal-Mechanical Analysis of Cable Connection with Screw-Connected Terminal Strips Using Finite Element Method	96-110
<i>Onur Dogan, Semih Bitim, Kadir Hiziroglu,</i> A V-Model Software Development Application for Sustainable and Smart Campus Analytics Domain	111-119



# SAUJCIS

Author(s), Paper Title	Pages
<i>Huseyin Demirci, Nilufer Yurtay,</i> Effect of the Chaotic Crossover Operator on Breeding Swarms Algorithm	120-130
<i>Mehmet Teyfik Agdas, Muammer Turkoglu, Sevinc Gulsecen,</i> Deep Neural Networks Based on Transfer Learning Approaches to Classification of Gun and Knife Images	131-141
<i>Muhammed Ali Bayram, Ilyas Ozer, Feyzullah Temurtas,</i> Deep Learning Methods for Autism Spectrum Disorder Diagnosis Based on fMRI Images	142-155
<i>Mert Ozcan, Serhat Peker,</i> Designing a Data Warehouse for Earthquake Risk Assessment of Buildings: A Case Study for Healthcare Facilities	156-165

# Performance Evaluation of MANET Routing Protocols AODV and DSDV Using NS2 Simulator

 Amenah Sufyan Mhmood Thabet<sup>1</sup>,  Ahmet Zengin<sup>2</sup>

<sup>1</sup>Corresponding Author; Computer Engineering Department, Faculty of Computer and Information Sciences, Sakarya University; amenah.mhmood@ogr.sakarya.edu.tr; +90 (552) 719 02 42

<sup>2</sup>Computer Engineering Department, Faculty of Computer and Information Sciences, Sakarya University; azengin@ogr.sakarya.edu.tr

Received 14 August 2020; Revised 16 November 2020; Accepted 22 December 2020; Published online 03 February 2021

## Abstract

Mobile Ad hoc Network (MANET) is a collection of wireless mobile nodes that dynamically form a network temporarily without any support of central management. Moreover, every node in MANET moves arbitrarily making the multi-hop network topology to change randomly at uncertain times. There are several familiar routing protocols like AODV, DSR, and DSDV etc. which have been proposed for providing communication among all the nodes in the wireless network. This paper presents a performance comparison and study of reactive (AODV) and proactive (DSDV) protocols based on metrics such as throughput, packet delivery ratio, average end-to-end delay, packet loss rate and consumed energy by using the NS-2 simulator. The simulation results showed that AODV performance is better than DSDV regarding packet delivery ratio and end-to-end delay, while DSDV performance is better than AODV regarding packet loss rate and consumed energy then the performance of AODV and DSDV protocols in throughput parameter is equal close. For small networks, DSDV works well and AODV is best suited for larger networks.

**Keywords:** MANET, AODV, DSDV, NS-2 simulator, Routing algorithms, Protocol comparison

## 1. Introduction

Due to the fast growth of mobile communication in recent years, especially observed in the field of mobile system, wireless local area network, and ubiquitous computing. The set of mobile terminals that are placed in a close location communicating with each other, sharing services, resources or computing time during a limited period of time and in a limited space forms spontaneous ad hoc network. Network management should be transparent to the user. These types of networks have independent centralized administration; user can enter the networks and leave the networks easily. One of the important research areas in MANET is establishing and maintaining the ad hoc network through the use of routing protocols [1],[2].

Routing is the method of selecting a traffic path in a network or over multiple networks, which to send and receive data. It directs the passing of logically addressed packets from their source toward their ultimate destination through intermediary nodes. Routing protocol is the routing of packets based on the defined rules and regulations. Every routing protocol has its own algorithm on the basis of which it discovers and maintains the route. Each routing protocol has a data structure which stores the information of route and modifies the table as route maintenance is requires. A routing metric is a value used by a routing algorithm to determine whether one route should perform better than another. Metrics can cover such information as bandwidth, delay, hop count, path cost, load, reliability and communication cost. The routing table stores only the best possible routes while link-state or topological databases may store all other information as well. MANETs are currently the greatest innovation in the field of telecommunications.[3],[4].

Routing is a core problem in networks for sending data from one node to another. Several routing protocols have been proposed for mobile Ad-Hoc networks. In this paper we present the classification of these routing protocols and the review of an AODV and DSDV routing protocols [5].



The rest of the paper is organized as follows: Section II presented the related works; Section III provides a classification details of routing protocols in Mobile Ad-hoc network; Section IV provides the simulation methodology. The simulation results and discussion will be explained in section V. The last section VI concludes the paper.

## 2. Related Works

There are numerous investigate endeavors that have been done during the previous year comparing the execution of different routing protocols in MANET. These routing protocol employments diverse strategy or metric to choose the best path between the source node and the destination node, some of them they utilize the accessible transmission capacity and a few of them they utilize the bounce check between the sets. All of these protocols have their pros and cons [6].

Daas et al (2015) presented a comparison study for evaluating the performance of AODV and DSDV routing protocol based on node speed using NS2 simulator. The simulation results indicated that AODV has better performance than DSDV in terms of throughput, delay, and PDR factor [2]. DSDV performance is however better than AODV in terms of energy consumption [7].

A similar study is presented by [8]. The authors climbed that AODV is superior to AOMDV, DSR, and DSDV in terms of CBR connection. DSR is however perform excellently than AODV, AMDV, and DSDV in terms of TCP connections.

Sharma et al [9] are evaluated the performance of proactive and reactive protocols with different mobility models. Simulation results uncover that proactive protocols perform superior for smaller networks and reactive protocols perform way better for larger networks in terms of the performance metrics such as PDR, delay and bundle misfortune.

This study investigates and compare between two different routing protocol categories which are Proactive protocols (DSDV) and Reactive protocols (AODV). There have been several efforts to implement and test the efficiency of the network protocols in various contexts, such as AODV, DSR, and DSDV routing protocols.

## 3. Mobile Ad hoc Network Routing Protocols

Mobile Ad-Hoc network (MANET) is a kind of wireless network and self-configuring network of moving routers associated with wireless network. In MANET, the routers are free to move randomly and organize themselves arbitrarily, thus, the network's wireless topology may change rapidly and unpredictably [3], [10]. Figure 1 represent MANET overview.

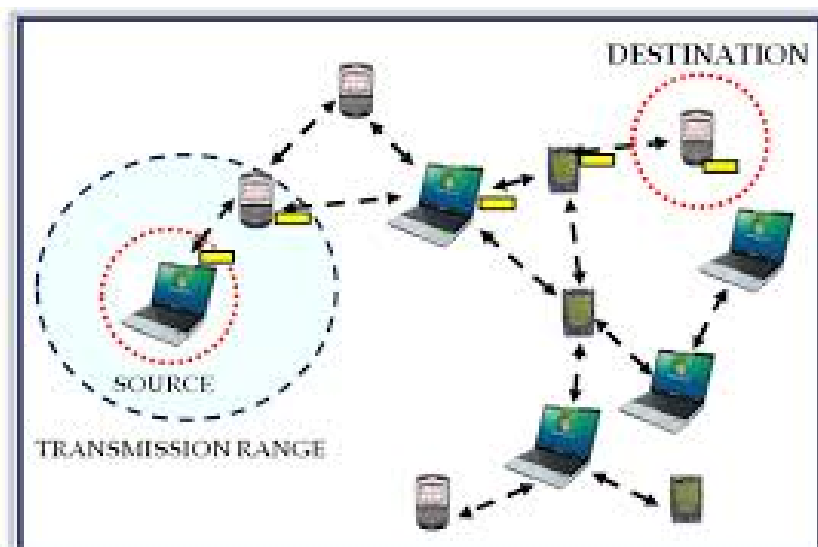


Figure 1 Mobile Ad Hoc Networks

MANET is a collection of wireless mobile hosts forming a temporary network without the aid of any established infrastructure or centralized administration. They are characterized by a dynamic, multi-hop, rapid changing topology [5]. The main objective of ad-hoc routing protocols is to deliver data packets among mobile nodes efficiently without predetermined topology or centralized control. The various mobile ad-hoc routing protocols have been proposed and have their unique characteristics. Hence, in order to find out the most efficient routing protocol for the highly dynamic topology in ad-hoc networks, the behavior of routing protocols has to be analyzed under different traffic patterns respect to their metrics [11]. Figure 2 shows the classification of MANET routing protocols depending on how the protocols are handle the packet to deliver from source to destination. Due to their functionality of Routing protocols are broadly classified into three types: Reactive, Proactive and Hybrid protocols [12].

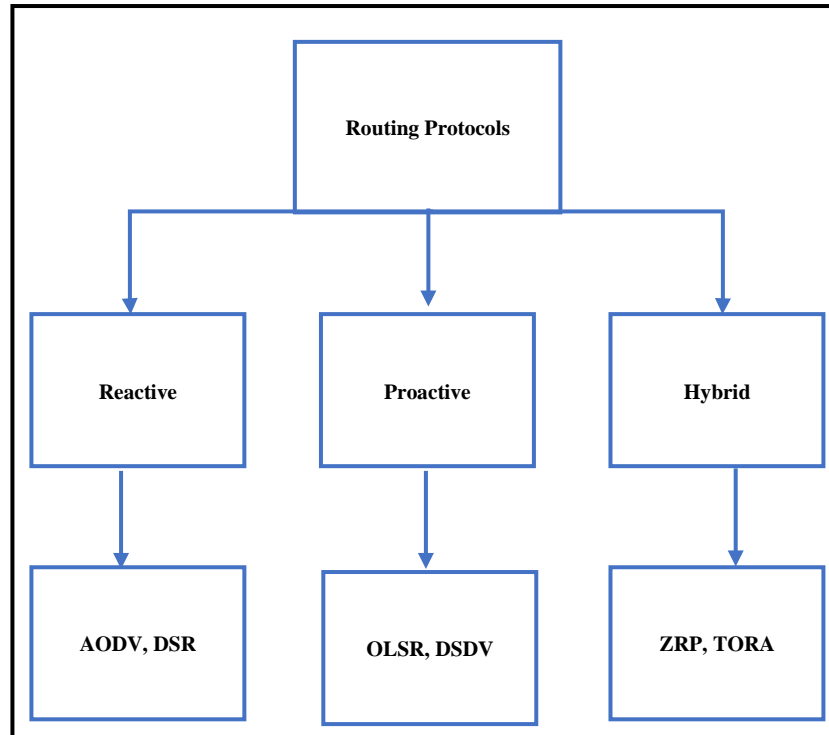


Figure 2 Routing protocols in MANET

### 3.1 Proactive (table-driven) Routing Protocol

The proactive routing is also known as table-driven routing protocol. Each node maintains routing information for every possible destination. This causes more overhead in the routing table leading to consumption of more bandwidth. DSDV and OLSR are the main representative protocols [13].

#### Destination Sequenced Distance Vector (DSDV)

(DSDV) is a table-driven routing protocol for ad-hoc mobile networks works based on the Bellman-ford algorithm. Each node acts as a router where a routing table is maintained and periodic routing updates are transfer, even if the routes are not necessary. A sequence number is associated with each route or path to the destination to prevent routing loops. The Routing updates are exchanged even if the network is idle which uses up battery and network bandwidth. So, it is not preferable for highly dynamic networks. The DSDV eliminates two problems of routing loops and counting to infinity. Dissemination of an update, however, remains quite slow. Mobility for high losses are mainly due to the use of outdated table entries.[14],[15].



### 3.2. Reactive (on-demand) routing protocol

This type creates a route when a source node require from distination node. It is based on flooding algorithm which employs on the technique that a node just broadcasts the packet to all of its neighbors and intermediate nodes just forward that packet to nearby nodes and this technique will repetitive until it reaches the destination.The major representative protocols are AODV, DYMO and DSR. [16].

#### Ad Hoc On-demand Distance Vector Routing (AODV)

In AODV, route establishment takes place only when there is a demand for new route. The network remains stable till the connection is desirable. At the point where the network node wants the connection then it broadcast the demand for the connection. The intermediate nodes progress these messages, and record the node from which they heard it, and creates the outbursts of temporary routes backward to the source node. When the node receives such a message and already route is present to the preferred node, afterward it sends a message backwards throughout the provisional route to the requesting node. As a result, AODV does not load any extra protocol over data packets because it doesn't use resource routing [17],[4].

## 4. Simulation Methodology

Nowadays simulation helps in analyzing the performance and behavior of complex networks before implementing it. Several network simulators are available such OMNET, NS2, and OPNET, whose output depicts as close as possible to real time implementation. In this work, we have used NS-2.34 network simulator to compare and evaluate the performance of AODV and DSDV routing protocols in MANET. The simulation has been used a different number of nodes to deeply verify the performance of these protocols in terms of the performance measures. The number of nodes were 10, 20, 30, 40 and 50. Where as the nodes have deployed in the network and they move randomly as shown in Figure 3.

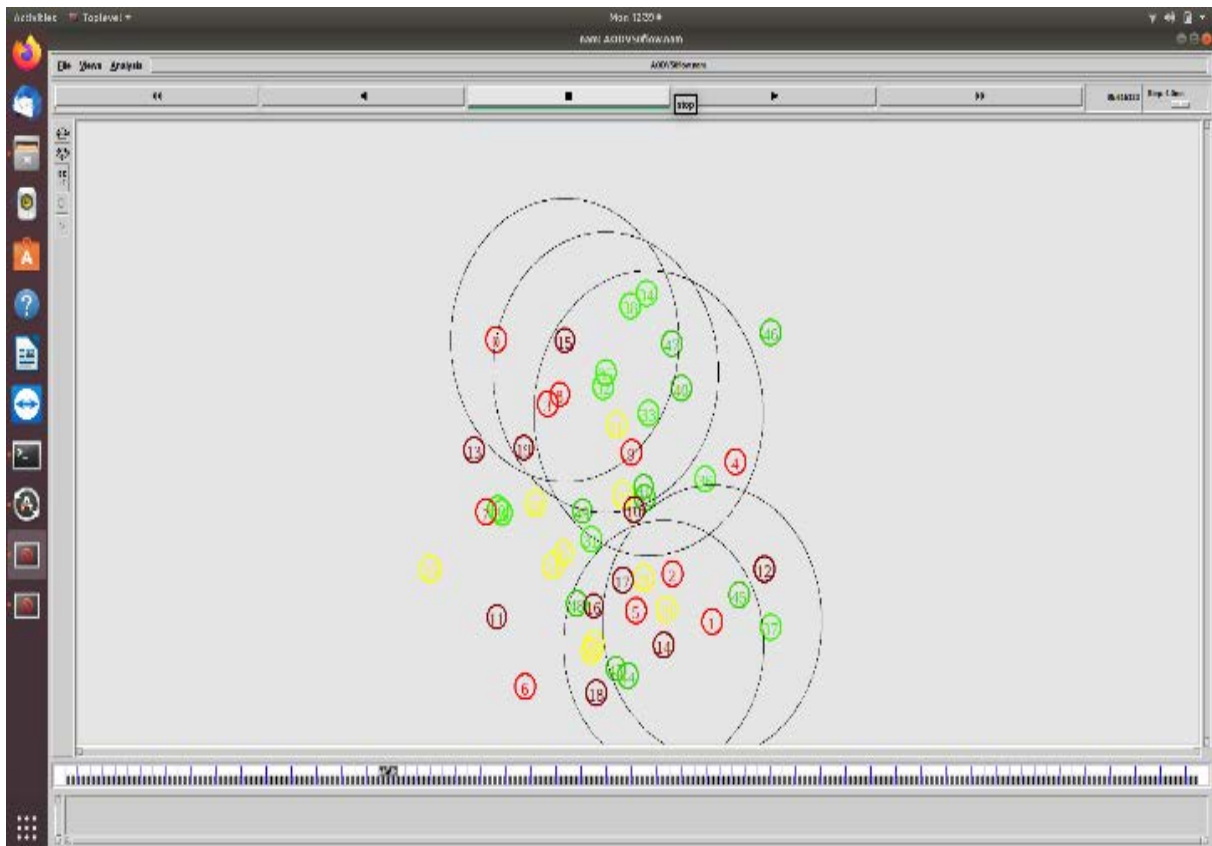


Figure 3 The Simulation Environment.

The transmission range in all nodes is set to be 250m in the network. The simulation area is 900m x 900m and the simulation time is 300 sec. The packet size in this simulation is 512 bytes. Table 1 shows the simulation parameters.

Table 1 Simulation parameters

Parameter	Value
Simulator	NS-2 (Version 2.34 )
Channel type	Channel/Wireless channel
Radio-propagation model	Propagation/Two ray
Network interface type	Phy/WirelessPhy
MAC Type	Mac /802.11
Interface queue Type	Queue/Drop
Link Layer Type	LL
Antenna	Antenna/Omni Antenna
Packet Size	512
Area ( M*M)	900 * 900
Number of mobile node	50
Source Type	TCP
Simulation Time	300
Routing Protocols	DSDV, AODV
Transmission Range of	250

## 5. Simulation Results and Discussion

Simulations were done by varying the number of nodes and keeping speed of the node constant (50). The deviation was done respectively varying the routing protocol from AODV and DSDV. The number of flows for each comparison was also varied from 10 to 20 to 30 to 40 to 50 to identify the result. In all scenarios the comparison were based on performance metric: Packet Delivery Ratio, End to End Delay and Throughput by also using NS-2 simulator and the results have been analyzed using Excel as shown in table 2.

Table 2 comparison of AODV and DSDV in terms of throughput, end to end delay, packet delivery ratio, packet loss rate and consumed energy

No. of Flow	Throughput		End to End delay		Packet delivery ratio		Packet Loss Rate		Energy	
	AODV	DSDV	AODV	DSDV	AODV	DSDV	AODV	DSDV	AODV	DSDV
10	691	553	354	373	626	312	480	608	390	380
20	668	715	478	505	888	532	177	420	353	334
30	673	691	654	853	1347	735	243	548	580	382
40	640	655	661	992	1284	817	410	312	430	270
50	525	601	942	989	1513	846	518	89	443	190

The performance metrics helps to characterize the network that is substantially affected by the routing algorithm to achieve the required Quality of Service (QoS). In this work, the following metrics are considered.

### 5.1 Average Throughput (TP)

It is the measure of the number of packets or data successfully transmitted to their final destination via a communication link per unit time [5] as shown in figure (4).

From Fig 4 it's clearly seen that DSDV has the high throughput for almost scenarios. The throughput values of DSDV and AODV Protocols for 10, 20, 30, 40 and 50 flow Nodes at 300s are noted in Table

1, the throughput value of DSDV is less than AODV in the case of 10 flow, and it increases gradually until reach (715) in case of 20 flow.

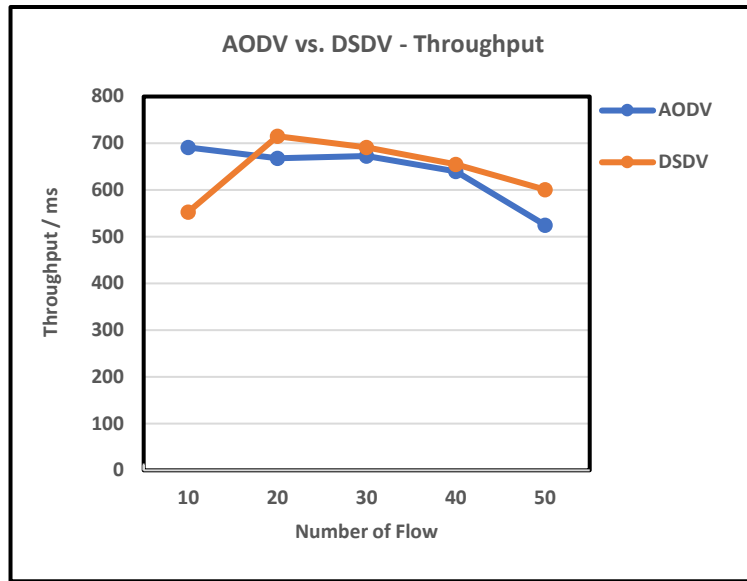


Figure 4 AODV vs DSDV Throughput

After that the DSDV values have range between 601 and 691. The throughput value for AODV start largest than DSDV value and decreased value when the number of flow increases, the throughput of AODV is between 525 and 691. Hence, DSDV performs close equal with AODV.

### 5.2 Packet delivery ratio (PDR)

It is the ratio of the total data bits received to total data bits sent from source to destination.[9] Figure (5) show the average packet delivery ratio for AODV and DSDV for all scenarios.

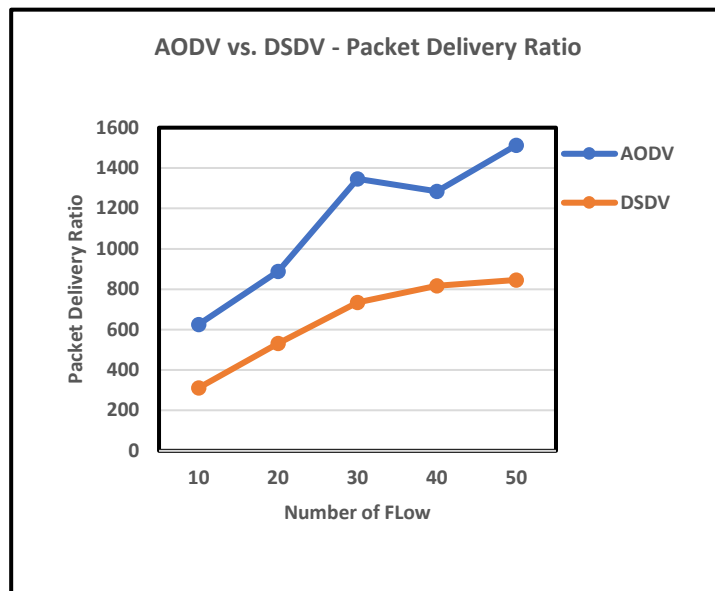


Figure 5 AODV vs DSDV Packet Delivery Ratio

Based on Figure 5, AODV has shown a better performance than DSDV when the number of flow nodes increased. The packet delivery ratio of AODV is between 1513 and 626. The packet delivery ratio of DSDV is between (846-312).

### 5.3 End-to-End Delay (EED)

It is the time delay for send data packet from the source node to the destination node. Total time difference over the total number of packets received is dividing with single packet send and received time [8]. Figure (6) shows the delay values for AODV and DSDV for all scenarios. **End to End Delay= (time packet received - time packet sent)**

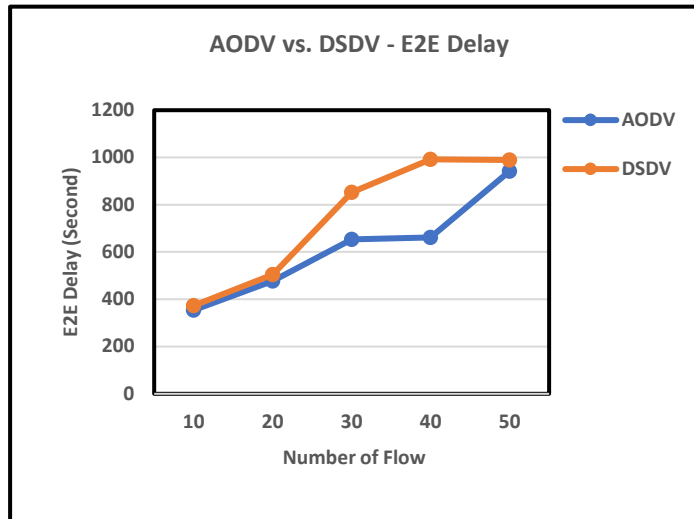


Figure 6 AODV vs DSDV End – End Delay

Figure 6 representing the delay graph for AODV and DSDV routing protocols. AODV performance has little delay in comparison with DSDV even the number of nodes increased. Resulting in, AODV is better than DSDV. End to end delay of DSDV have greater value than AODV. The average end to end delay for AODV and DSDV for all scenarios is between 354 and 942 and (373-992) respectively. DSDV keeps routing tables to deliver packets, and hence it sets up the new routes when there is a change in the network topology and AODV is the on-demand protocols, and it has to initiate the routing discovery mechanism whenever a new route is to be established. AODV delivers required packets on demand of communication between the nodes.

### 5.4 Packet Loss Rate (PLR)

Packet Loss rate is characterized as those packets that are sent by the source and fizzled to be gotten by the goal. It is calculated by separating the whole lost packets for directing by add up to packets sent by equation as below [18]. Figure (7) shows the packet loss rate values for AODV and DSDV for all scenarios

$$\text{Packet Loss Rate} = \frac{\text{Packet Lost}}{\text{Packet Sent}}$$

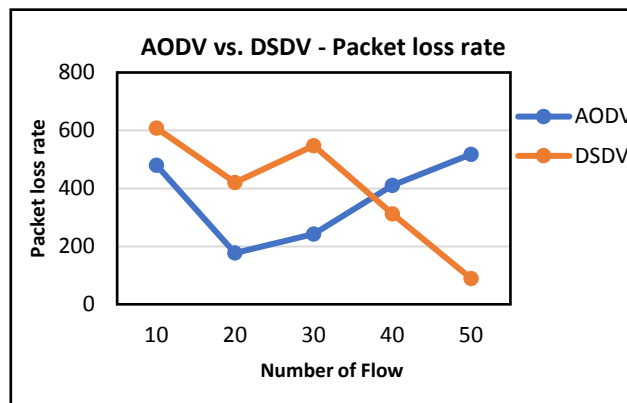


Figure 7 AODV vs DSDV Packet loss rate

We will see from figure (7) over that the number of nodes decreased the impact on the packet loss rate with DSDV routing protocol but rises obviously in AODV, the alter of the coming about packet loss values tends to improve in denser nodes.

A decrease in packet loss on DSDV in node (40, 50) demonstrates that the number of packets lost to the goal is exceptionally small compared with AODV within the same nodes.

### 5.5 Energy Consumption (joules)

The average of energy consumed by the mobile nodes while routing and communication [19].

Figure (8) shows the energy consumption values for AODV and DSDV for all scenarios.

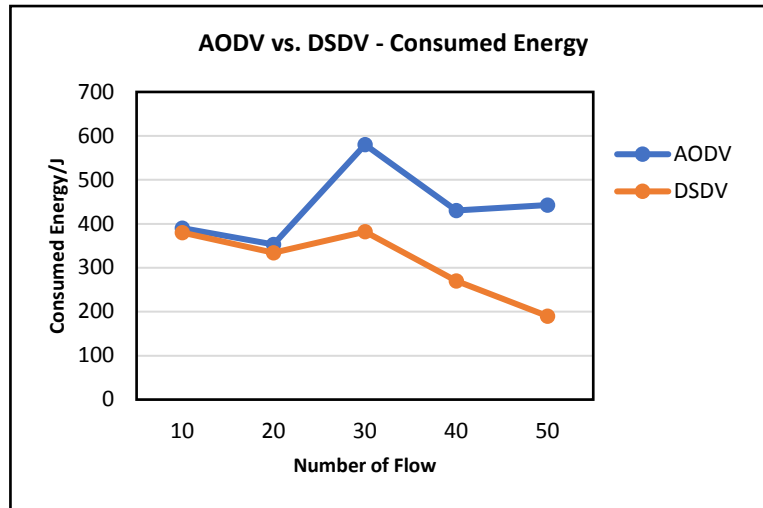


Figure 8 AODV vs DSDV consumed energy

In figure 8, the proactive protocol DSDV has yielded consistent energy consumption by the mobile nodes as the number of nodes increases from 10 to 20. At that point the energy consumption is increased in node (30), while the following nodes (40, 50) decreasing. On the other hand, the reactive protocol AODV energy consumption was similar to the DSDV within the run from 10 to 20 nodes. It then increased from 20 to 30 nodes to reach the highest consumption at 30 nodes, after it decreased from 30 to 40 nodes and finally, it tends to increase once more from 40 to 50 nodes over DSDV. For medium estimate MANETs, AODV consumed more energy recently decreasing for larger MANETs, whereas DSDV consumed a moderately lower energy for small, medium and larger MANETs.

### 6. Conclusion

MANET is a collection of mobile nodes, dynamically establishing short-lived networks in the absence of fixed infrastructure. This project compares of AODV and DSDV routing protocols which are proposed for ad-hoc mobile networks. In DSDV routing protocol, mobile nodes periodically broadcast their routing information to the neighbors. Each node requires to maintain their routing table. AODV protocol finds routes by using the route request packet and route is discovered when needed. The comparison of these protocols is done in terms of the parameters packet delivery ratio, throughput, end to end delay, packet loss rate and consumed energy. The simulation results showed that AODV performance is better than DSDV regarding packet delivery ratio and end to end delay while DSDV performance superior to AODV on packet loss and consumed energy. In throughput parameter the AODV and DSDV performance was closed. For small networks, DSDV works well and AODV is best suited for larger networks.

## Acknowledgments

The authors would like to thank the Iraqi parliament council for financial support.

## References

- [1] A. Arya and J. Singh, "Comparative Study of AODV , DSDV and DSR Routing Protocols in Wireless Sensor Network Using NS-2 Simulator," vol. 5, no. 4, pp. 5053–5056, 2014.
- [2] M. Manjunath and D. H. Manjaiah, "Performance Comparative of AODV, AOMDV and DSDV Routing Protocols in MANET Using NS2 Alamsyah1,2," *Int. J. Commun. Netw. Syst.*, vol. 004, no. 001, pp. 18–22, 2018, doi: 10.20894/ijcnes.103.004.001.005.
- [3] A. Tuteja, R. Gujral, and S. Thalia, "Comparative performance analysis of DSDV, AODV and DSR routing protocols in MANET using NS2," *ACE 2010 - 2010 Int. Conf. Adv. Comput. Eng.*, vol. 1, pp. 330–333, 2010, doi: 10.1109/ACE.2010.16.
- [4] M. G. K. Alabdullah, B. M. Atiyah, K. S. Khalaf, and S. H. Yadgar, "Analysis and simulation of three MANET routing protocols: A research on AODV, DSR & DSDV characteristics and their performance evaluation," *Period. Eng. Nat. Sci.*, vol. 7, no. 3, pp. 1228–1238, 2019, doi: 10.21533/pen.v7i3.717.
- [5] N. Gupta and R. Gupta, "Comparative Analysis of CBRP, AODV, DSDV Routing Protocols in Mobile Ad-hoc Networks," *Int. Conf. "Emerging Trends Robot. Commun. Technol. INTERACT-2010*, pp. 173–177, 2010, doi: 10.1109/INTERACT.2010.5706220.
- [6] T. H. Sureshbhai, M. Mahajan, and M. K. Rai, "An investigational analysis of DSDV, AODV and DSR routing protocols in mobile Ad Hoc networks," *Proc. - 2nd Int. Conf. Intell. Circuits Syst. ICICS 2018*, pp. 286–289, 2018, doi: 10.1109/ICICS.2018.00064.
- [7] S. El Khediri, N. Nasri, A. Benfradj, A. Kachouri, and A. Wei, "Routing protocols in MANET: Performance comparison of AODV, DSR and DSDV protocols using NS2," *2014 Int. Symp. Networks, Comput. Commun. ISNCC 2014*, 2014, doi: 10.1109/SNCC.2014.6866519.
- [8] B. Paul, K. A. Bhuiyan, K. Fatema, and P. P. Das, "Analysis of AOMDV, AODV, DSR, and DSDV routing protocols for wireless sensor network," *Proc. - 2014 6th Int. Conf. Comput. Intell. Commun. Networks, CICN 2014*, pp. 364–369, 2014, doi: 10.1109/CICN.2014.88.
- [9] R. Sharma, T. Sharma, and A. Kalia, "A Comparative Review on Routing Protocols in MANET," *Int. J. Comput. Appl.*, vol. 133, no. 1, pp. 33–38, 2016, doi: 10.5120/ijca2016907748.
- [10] A. V. Biradar and S. R. Tandle, "Detailed Performance Analysis of Energy based AODV Protocol in Comparison with Conventional AODV, and DSDV Protocols in MANET," *Int. J. Comput. Appl.*, vol. 49, no. 10, pp. 49–58, 2012, doi: 10.5120/7667-0785.
- [11] M. A. Mohammed, "Performance Study of AODV and DSDV Routing Protocols for Mobile Ad Hoc Networks Based on Network Simulator NS2," *Sulaimani J. Eng. Sci.*, vol. 2, no. 2, pp. 7–12, 2015, doi: 10.17656/sjes.10020.
- [12] S. K. Gupta and R. K. Saket, "Performance Metric Comparison of AODV and DSDV Routing Protocol In MANETs Using NS-2," *Ijrras 7*, vol. 7, no. June, pp. 339–350, 2011.
- [13] J. Rahman, M. A. M. Hasan, and M. K. Ben Islam, "Comparative analysis the performance of AODV, DSDV and DSR routing protocols in wireless sensor network," *2012 7th Int. Conf. Electr. Comput. Eng. ICECE 2012*, pp. 283–286, 2012, doi: 10.1109/ICECE.2012.6471541.
- [14] N. F. Rozy, R. Ramadhiansya, P. A. Sunarya, and U. Rahardja, "Performance Comparison Routing Protocol AODV, DSDV, and AOMDV with Video Streaming in Manet," *2019 7th Int. Conf. Cyber IT Serv. Manag. CITSM 2019*, 2019, doi: 10.1109/CITSM47753.2019.8965386.
- [15] M. Y. and E. A., "Comparative Analysis of Routing Protocols AODV DSDV and DSR in MANET," *Ijarcece*, vol. 5, no. 12, pp. 470–475, 2016, doi: 10.17148/ijarcece.2016.512107.
- [16] B. Moussaoui, S. Djahel, M. Smati, and J. Murphy, "A cross layer approach for efficient multimedia data dissemination in VANETs," *Veh. Commun.*, vol. 9, no. May, pp. 127–134, 2017, doi: 10.1016/j.vehcom.2017.05.002.
- [17] M. Gupta and S. Kumar, "Performance evaluation of DSR, AODV and DSDV routing protocol for wireless Adhoc network," *Proc. - 2015 IEEE Int. Conf. Comput. Intell. Commun. Technol.*



- CICT 2015*, pp. 416–421, 2015, doi: 10.1109/CICT.2015.95.
- [18] A. A. Al-khatib and R. Hassan, “Performance evaluation of AODV, DSDV, and DSR routing protocols in MANET using NS-2 simulator,” *Lect. Notes Data Eng. Commun. Technol.*, vol. 5, pp. 276–284, 2018, doi: 10.1007/978-3-319-59427-9\_30.
- [19] I. M. A. Fahmy, L. Nassef, and H. A. Hefny, “Energy consumption efficiency and performance evaluation of DSDV and AODV routing protocols,” *Int. J. Comput. Netw. Wirel. Commun.*, vol. 4, no. 2, pp. 113–118, 2014.

## Biography





Ms. Amenah Thabet received her B.Sc in software engineering from Computer Science and Mathematics college, Mosul University, in 2009. She is working as software engineer at the Iraqi parliament council. She is currently studying Master degree at the Computer Engineering Department/ Faculty of Computer and Information Sciences/ Sakarya University. Her research interests include Mobile Ad hoc Network (MANET), Vehicle Ad hoc Network (VANET), and smart city.



Ahmet Zengin is Professor at Sakarya University, Turkey. His experience with modeling and simulation includes a one-year-stay in ACIMS Lab at the Arizona State University. His research topics include DEVS theory, multi-formalism modeling, parallel and distributed simulation, modeling and simulation of large-scale networks, distributed systems management, biologically-inspired optimization schemes. His main research interest lies in parallel and distributed simulation and the High Level Architecture.

# A Study on the Efficacy of Deep Reinforcement Learning for Intrusion Detection

 Halim Görkem Gülmez<sup>1</sup>,  Pelin Angin<sup>2</sup>

<sup>1</sup>Middle East Technical University; halim.gorkem.gulmez@gmail.com

<sup>2</sup>Corresponding Author; Middle East Technical University; pangin@ceng.metu.edu.tr

Received 30 November 2020; Revised 14 December 2020; Accepted: 26 December 2020; Published online 03 February 2021

## Abstract

The world has witnessed a fast-paced digital transformation in the past decade, giving rise to all-connected environments. While the increasingly widespread availability of networks has benefited many aspects of our lives, providing the necessary infrastructure for smart autonomous systems, it has also created a large cyber attack surface. This has made real-time network intrusion detection a significant component of any computerized system. With the advances in computer hardware architectures with fast, high-volume data processing capabilities and the developments in the field of artificial intelligence, deep learning has emerged as a significant aid for achieving accurate intrusion detection, especially for zero-day attacks. In this paper, we propose a deep reinforcement learning-based approach for network intrusion detection and demonstrate its efficacy using two publicly available intrusion detection datasets, namely NSL-KDD and UNSW-NB15. The experiment results suggest that deep reinforcement learning has significant potential to provide effective intrusion detection in the increasingly complex networks of the future.

**Keywords:** security, deep reinforcement learning, intrusion detection

## 1. Introduction

The fast-paced developments in computing and network infrastructures in the past two decades have led to the rise of the Internet of Things (IoT) paradigm with ubiquitous connectivity along with increasingly widespread usage of cloud computing. While these developments have greatly facilitated daily operations in many industries and enterprises in addition to touching the daily lives of people in positive ways, the resulting cyber security issues have created deterrents for the more widespread adoption of IoT due to an enlarged attack surface with many security vulnerabilities. The number of zero-day attacks, which are security incidents whose signatures were not previously observed, is rising every day with the increasing number of vulnerabilities in these networked systems. Some of these attacks can have devastating consequences, as they are now capable of destroying not only software, but also hardware components through IoT connections.

Modern network intrusion detection and prevention systems (IDPS) have the purpose of detecting and mitigating various attacks on networked systems with sub-second response times. While IDPS in legacy systems mostly relied on attack signature-based solutions, which would create rules for each observed attack pattern and compare incoming traffic with the rules in the IDPS's database, this solution is not sufficient to cover the variety of attacks in today's complex systems both because of the high volume of traffic that needs to be analyzed in real time and due to the inability to generalize and detect attacks with unknown signatures. Security researchers thus have turned to machine learning (ML) and deep learning (DL) techniques that are capable of learning patterns of attacks and normal behavior of systems so that anomalous network traffic can be detected and classified in real time, and the IDPS can adjust itself to deal with new types of attacks over time.

Reinforcement learning (RL) algorithms, which are based on agents interacting with a runtime environment under a variety of states to learn to maximize their rewards, has been a popular technique for many learning-based tasks since their introduction. More recently, deep reinforcement learning (DRL) algorithms, which utilize deep neural networks within RL to facilitate representation of many possible state-action pairs and provide generalizability, have been applied successfully to a variety of

problem domains. Among successful applications of DRL are Atari games [1], chess [2], solving arithmetic problems [3], medication treatment plans [4], optimization of chemical reactions [5], and extraction of biological sequence data [6] among many others. Despite its success in various fields, the application of DRL to network security has been rather limited so far.

In this paper we propose a DRL-based approach for network intrusion detection and evaluate its effectiveness on two real-world benchmark datasets that have been commonly used in the evaluation of ML-based approaches for detecting cyber attacks in legacy networks, namely NSL-KDD and UNSW-NB15. The evaluation results demonstrate that DRL is a promising method for network intrusion detection, achieving F-1 scores of over 96% on both datasets. We also show that the effectiveness of the algorithm is significantly affected by the structure of the embedded deep neural network, i.e., the number of hidden neurons, as well as the number of training iterations. Performance comparison of the model with various state-of-the-art ML/DL models demonstrates its promise, especially in terms of F-1 score, on the two benchmarks.

The remainder of this paper is organized as follows: Section 2 provides an overview of related work in ML-based intrusion detection systems. Section 3 provides details of the proposed DRL model for network intrusion detection. Section 4 provides an experimental evaluation of the model on two public network intrusion detection datasets. Section 5 concludes the paper with future work directions.

## 2. Related Work

The advances in the field of machine learning have paved the way for their use in the field of cyber security for the past two decades. Most existing anomaly-based intrusion detection systems rely on ML techniques. Beehive, a successful solution for detecting intrusion from network logs, was proposed in [7]. Beehive uses four types of features and utilizes k-means clustering to detect anomalies. One downside is that it does not work in real time. Another successful approach utilizing k-means clustering includes the work of [8]. While k-means clustering can be effective for detecting anomalies, predefining the value of k can be a problem in many settings.

Balogun and Jimoh [9] proposed a method utilizing the k-nearest neighbor (KNN) classifier and decision trees. Their approach was shown to be capable of detecting new attacks with high accuracy. [10] utilized a variety of ML algorithms including k-means clustering, isolation forest, histogram based outlier score and cluster-based local outlier factor in their approach called CAMLPAD, and achieved an accuracy of 95% in an intrusion detection task. Pervez and Farid [11] proposed using Support Vector Machines (SVM) for intrusion detection on the NSL-KDD dataset. Although SVM was successful on the training set, it failed to detect many attacks in the test set. In [12], a multi-layer perceptron based model with 3 layers was proposed, which achieved 81% accuracy for binary classification on NSL-KDD. Kamel et al. [13] proposed an AdaBoost-based intrusion detection model and reported 99.9% accuracy on NSL-KDD, however their training and test sets consisted of subsets of the whole dataset, which were not clearly described. Hu et al. [14] also applied AdaBoost for intrusion detection on the KDD Cup'99 dataset and achieved 91% detection rate. Engly et al. [15] evaluated the performance of Gradient Boosting Machines on NSL-KDD and achieved successful results with an ensemble model. Moustafa and Slay [16] applied Expectation-Maximization Clustering, Logistic Regression (LR) and Naive Bayes classification on the UNSW-NB15 datasets, and achieved the best results with an accuracy of 83% for LR.

Following the success of deep learning in many fields in recent years, security researchers have started employing it in many intrusion detection systems. [17] and [18] proposed using recurrent neural networks (RNN) in intrusion detection on data with time dependencies and achieved successful results. A variant of the RNN-based intrusion detection model was proposed by Yin et al. [19], achieving over 83% accuracy on the KDD Cup'99 dataset. An LSTM-based model, which is a special RNN-structure, was proposed by Li et al. [20], which achieved 83% accuracy and F-1 score on NSL-KDD. Behera et al. [21] also proposed the use of convolutional neural networks (CNN) for intrusion detection and achieved high accuracy on the NSL-KDD dataset. They also stated their approach can be adapted to detect zero-day attacks. Another CNN-based intrusion detection model was proposed by Li et al. [22],

which also achieved successful results on NSL-KDD. Lopez-Martin et al. [23] proposed a conditional variational auto-encoder based model for unsupervised intrusion detection, which achieved 80.10% accuracy on NSL-KDD. A two-stage stacked auto-encoders based model was proposed by Khan et al. [24], which achieved 89% accuracy on the UNSW-NB15 dataset. A hybrid model consisting of deep neural networks and spectral clustering was proposed by Ma et al. [25], which achieved around 72% accuracy on NSL-KDD. A scalable, hybrid intrusion detection approach, which utilizes deep neural networks (DNNs), was proposed by Vinayakumar et al. [26]. The distributed DNN-based model was shown to achieve better performance than traditional ML-based classifiers on a set of benchmarks. Gao et al. [27] developed a deep belief networks-based model for intrusion detection, and demonstrated its superior performance in comparison to SVM and MLP.

Researchers have also utilized RL for detecting attacks in networks. Various types of log files were used in the solution of [28] where a rule-based approach was taken to create association rules signaling attacks. Their approach utilized RL as a helper rather than basing the solution on it. [29] also proposed an RL-based approach with multiple agents watching over the network states in a hierarchical manner, which was shown to provide accurate results, although it was not evaluated with different datasets. A cyber security simulation was set up in [30] to apply RL for finding the best strategies of both attackers and defenders in a Markov game. Their experiments demonstrated the tool can be used both for intrusion detection systems and for launching successful cyber attacks on systems. The approach we propose in this work differs from existing RL-based approaches in that it utilizes fully connected deep neural networks for allowing the RL agents to make decisions based on unstructured input data, obviating the need to manually create large state spaces.

### 3. Proposed Intrusion Detection Approach

In this section, we describe our proposed DRL-based approach for network intrusion detection. We first provide a brief overview of deep neural networks, and continue with an explanation of how they are integrated into RL to achieve a highly accurate intrusion detection model.

#### 3.1 Deep Neural Networks (DNN)

Neural networks are a special category of ML models the design of which resembles the functioning of the human brain in the sense that it simulates the processing and transmission of information through the complex networks of neurons, which get excited or inhibited by the signals in the network [31]. One of the first examples of neural network structures is the *perceptron*, which contains a single input layer connected to an output. The perceptron represents the simplest processes in the brain's neurons using an activation function and a set of weights, as depicted in Figure 1(a). Machine learning with a perceptron involves random assignment of weights to each of the input nodes, and the passage of the weighted sum of the input values through an activation function to produce the output value. The weights are adjusted throughout the training process in multiple iterations and the goal of the training process is to minimize the aggregate error in the output. The error is calculated as the difference between the ground truth output, and the output that is calculated by the model.

Multi-layer perceptrons (MLP) are feedforward neural networks containing a number of hidden layers in between the input and output layers, as demonstrated in Figure 1(b). The figure shows a fully-connected deep neural network with one hidden layer, with every input node connected to every hidden node and likewise, every hidden node connected to every output node. When the fully-connected neural network consists of more hidden layers, each node in a hidden layer will be connected to each node in the following hidden layer. As seen in the figure, each edge connecting the nodes has a weight that is updated throughout the training process to achieve minimum output error. The number of hidden neurons in each layer can be different from the number of input and output layer neurons. Training of the network involves running a back-propagation algorithm [31] updating the weights of the edges in each iteration. While the number of input nodes is decided by the dimensionality of the input feature vector, the number of output nodes is decided by the specific learning task, e.g. multi-class classification,

regression, binary classification etc. Among commonly used activation functions in MLP are sigmoids including  $y(v_i) = \tanh(v_i)$  and  $y(v_i) = \frac{1}{1 + e^{-v_i}}$ .

DNNs are yet more complex artificial neural networks with many more hidden layers than MLPs. Their complexity allows them to express more complex hypotheses by better modeling the nonlinear relationships in the network. DNNs provide the inherent ability to learn higher level representations from possibly unstructured data, which makes them very valuable for a variety of machine learning tasks. In this work, we utilize fully connected DNNs integrated into the RL process as described below to achieve highly accurate intrusion detection.

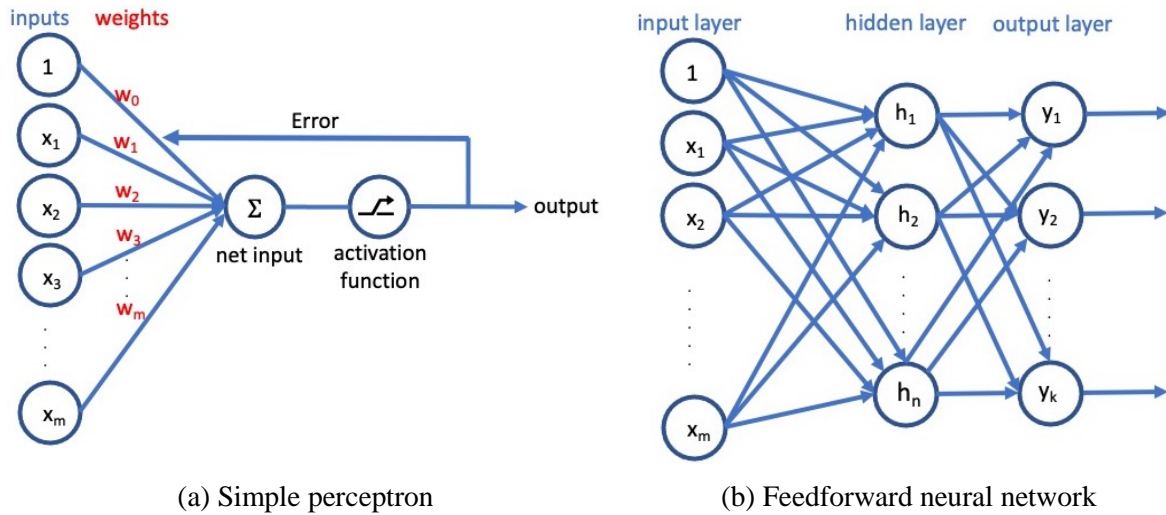


Figure 1 Structure of neural networks

### 3.2 DRL-Based Intrusion Detection

Reinforcement learning is an ML technique based on an agent learning through rewards and punishments it receives through its interactions with the environment. Each state of the agent is associated with a set of actions that could have different rewards, and the agent learns over time what action to perform based on its history of actions-rewards at that same state. An agent in RL takes actions from among a set of possible actions for its current state, and receives a positive or negative reward for taking that particular action, which it saves in its memory. These rewards are then used by the agent to decide which action to take in later states, where the ultimate goal of the agent is to maximize its total reward value. Agents are connected to their environments with action and recognition as described by Kaelbling et al. [32]. Picking a certain action at a certain state results in an output, which modifies the state of the agent, and the agent receives the value of this change with a reinforcement signal. The agent learns to choose the most rewarding action over time by trial and error using different algorithms. The environment is not always deterministic, i.e. choosing the same action can have different consequences at different points in time in the same state.

As apparent from the description above, RL is quite different from supervised learning. While supervised learning utilizes training datasets consisting of labeled input/output pairs, an agent in RL receives immediate rewards based on its actions after performing the action. Learning does not actually stop in RL; it is a continuous process in which the agent keeps receiving new rewards or punishments as it interacts with its environment, however it is expected that the rewards will keep increasing over time, as the agent learns which actions provide the greatest rewards at each state.

The state-action space could get very large in RL in complex environments, causing the algorithm not to generalize well. DRL is an improvement of RL algorithms that provides improved generalization power by augmenting RL with deep neural networks in the state-action input formation. i.e., DRL utilizes deep neural networks for function approximation in policy and value functions in RL. This

capability is important in a network intrusion detection setting, as generalizability matters especially for cases like zero-day attacks.

In an RL algorithm based on Q-learning, the value function is as follows:

$$Q(s,a)=r(s)+\gamma \max_{a'}\Sigma P(s'|s,a)Q(s',a') \quad (1)$$

Equation 1 is the Bellman Equation. Here  $s$  represents the state,  $a$  represents the action,  $r$  represents the reward, and  $P$  represents state change possibility. Based on the equation, the Q value of a state-action pair is equal to the sum of the current reward and potential future Q-values. While this equation is discrete, many real-life applications involve continuous actions and states. Thus, we need an effective function approximation technique for the value function. This requirement is met by integrating DNNs into RL. In the value function using DNNs, every state and Q-value are calculated by utilizing hidden layers of neural networks, which are trained using backpropagation.

Algorithm 1 Deep Q-learning

1	Initialize replay memory $D$ to capacity $N$
2	Initialize Q-function with random weights
3	<b>for</b> episode = 1, $M$ <b>do</b>
4	Initialize neural network from a random state $s$
5	<b>for</b> $t = 1, T$ <b>do</b>
6	Find Q-values for all actions using DNN algorithm: $a_t = \max_a Q^*(s_t, a; \theta)$
7	Choose an action $a_t$ for current state $s_t$ by using e-greedy exploration
8	Move to the next state $s_{t+1}$ with action $a_t$ , pick reward $r_t$
9	Store transition $(s_t, a_t, r_t, s_{t+1})$ in $D$
10	Sample random minibatch of transitions $(s_t, a_t, r_t, s_{t+1})$ in $D$
	Set $y_j =$
11	$\begin{cases} r_j, & \text{for terminal } s_{j+1} \\ r_j + \gamma \max_{a'} Q(s_{j+1}, a'; \theta), & \text{for nonterminal } s_{j+1} \end{cases}$
12	Perform a gradient descent step on $(y_j - Q(s_j, a_j; \theta))^2$

We describe deep Q-learning [1] in Algorithm 1 above. As the algorithm describes, DNNs are used as part of the RL, forming the DRL algorithm. In RL, immediate rewards are valued more than distant rewards in the future. DNNs provide the capability for the Q-functions to more accurately take future rewards into account when deciding about the actions to take. Another advantage of using DNNs in RL is that the number of interactions needed is reduced by sampling, resulting in better performance and efficiency.

In this paper, we propose a simple DRL model for network intrusion detection, where the learning agent has two different states, i.e. under attack or normal traffic, and four possible actions. Table 1 provides a high-level overview of states, actions and corresponding reward values. The main difference of this approach from state-of-the-art ML/DL models for intrusion detection is the overall learning process, which involves exploration of the different classification options by the learning agent, which is penalized when it incorrectly classifies an instance and rewarded for correct classification. Through this process, the agent learns to take the optimal actions over time to maximize its reward. Here the states refer to the network traces. Unlike traditional deep learning, DNNs are only used as part of the process in DRL to enable representation of the policies of the agent, i.e. actions to be taken to achieved the maximum reward at a specific state, without having to enumerate all possible states manually. The learning process continues throughout the lifetime of the agent. This is an important feature for especially online learning systems, which will be instrumental in successful intrusion detection in the era of ever increasing zero-day attacks. Figure 2 shows an activity diagram of the deep Q-learning algorithm.



Table 1 RL States, Actions and Rewards

State	Action	Reward
Normal	No Alarm	+1
Normal	Alarm	-1
Attack	Alarm	+1
Attack	No Alarm	-1

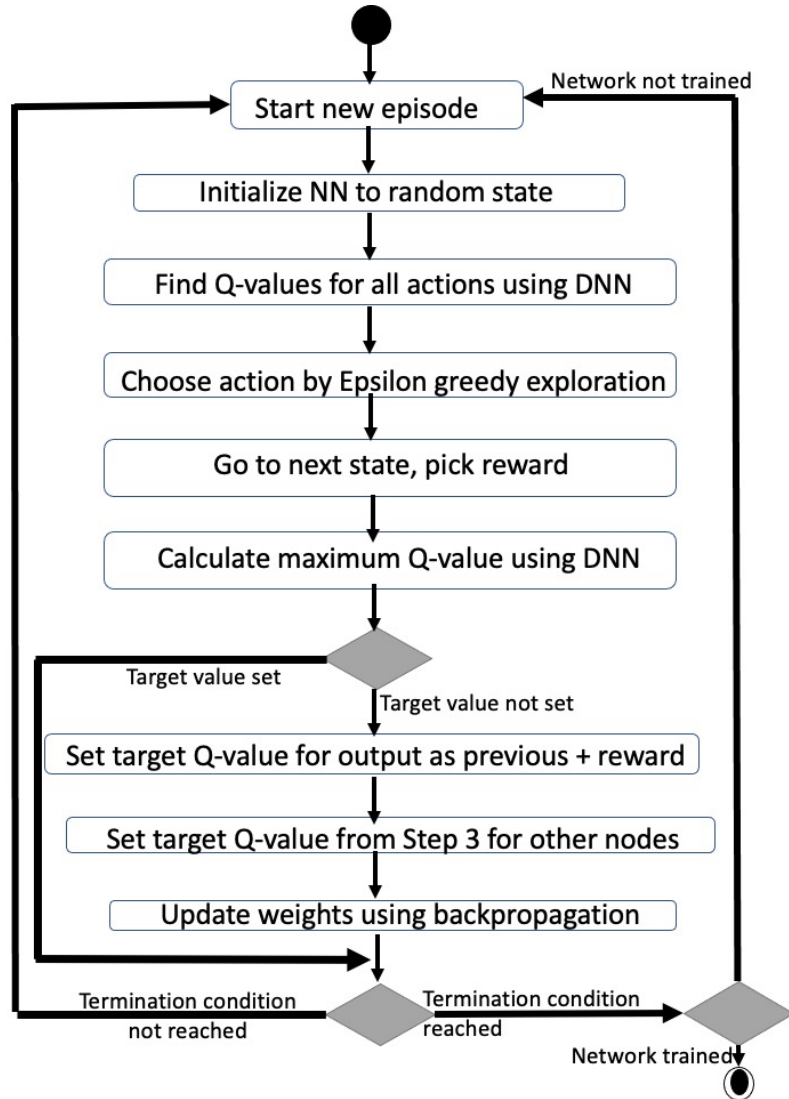


Figure 2 Deep Q-learning activity diagram

#### 4. Experimental Evaluation

We have evaluated the effectiveness of the proposed DRL model for intrusion detection using two benchmark datasets, UNSW-NB15 and NSL-KDD, where the task was to perform binary classification of records into attack and normal classes. Below we describe the datasets and provide results of the performed experiments.

##### 4.1 Datasets

*UNSW-NB15:*

The UNSW-NB15 [33] dataset was created by the University of New South Wales in 2015, using the IXIA tool for generating network traffic including attacks. It has 49 features, two of which are labels for binary classification (i.e. attack or normal traffic) and multi-class classification (i.e. type of attack). The dataset contains 9 types of attack traffic in addition to normal traffic, where attacks include DoS, DDoS, fuzzing, backdoor, analysis, worm, exploit, shellcode and generic. The remaining fields include network packet and connection details like IP addresses, ports, communication protocols. A subset of the features of this dataset are listed in Table 2 below. The dataset consists of about two million network packet traces, which is quite extensive.

Table 2 UNSW-NB15 Features

Feature	Type	Description	Feature	Type	Description
srcip	nominal	Source IP	sloss	integer	Source packets retransmitted or dropped
sport	integer	Source port	dloss	integer	Destination packets retransmitted or dropped
dsip	nominal	Destination IP	service	nominal	http, ftp, ...
dsport	integer	Destination port	Sload	float	Source bits/sec
proto	nominal	Protocol	Dload	float	Destination bits/sec
dur	float	Total duration	Spkts	integer	Source-to-destination packet count
sbytes	integer	Source-to-destination transaction bytes	Dpkts	integer	Destination-to-source packet count
dbytes	integer	Destination-to-source transaction bytes	stime	timestamp	Record start time
sttl	integer	Source-to-destination time to live value	ltime	timestamp	Record last time
dttl	integer	Destination-to-source time to live value	label	binary	0 for normal, 1 for attack
...					

#### NSL-KDD:

KDD CUP'99 [34] has been one of the most frequently used datasets in the evaluation of ML-based intrusion detection techniques since it was released in 1999. This dataset was generated by extracting features from DARPA98 [35], which is a dataset consisting of traffic obtained from the U.S. Air Force LAN. The dataset consists of 41 features and 4 attack categories: probing, denial of service (DoS), R2L, U2R. Despite its age, this dataset is still used by many researchers due to its large size (about 5 million records) and its modeling of a variety of conditions obtained from real network traffic. It also has some drawbacks including the presence of many duplicate records, unbalanced numbers of records from different classes in the training set, which could create biased classification models and the unbalanced distribution of records in the training and test sets [36].

The NSL-KDD dataset [36] was created to solve the abovementioned issues with KDD CUP'99. It involved removal of duplicate records and balancing of the number of records for different classes to prevent bias in the classification. The researchers also provided more balanced training and test sets. All original features from KDD CUP'99 were retained. Different sets were provided in the dataset, including sets with binary classification labels as in UNSW-NB15, sets with attack type labels and difficulty levels, as well as sets not including the hardest-to-detect cases.

## 4.2 Experimental Results

To evaluate the effectiveness of the solution, we have utilized metrics commonly used to in ML to judge the goodness of algorithms, which are precision, recall, accuracy, and F-1 score. The description of each metric is provided in Table 3 below. The abbreviations used in the table are as follows:

TP (true positive): The number of instances correctly classified as attacks

TN (true negative): The number of instances correctly classified as normal traffic

FP (false positive): The number of instances incorrectly classified as attacks

FN (false negative): The number of instances incorrectly classified as normal traffic

Table 3 Evaluation Metrics

Metric	Formula
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$
Precision	$\frac{TP}{TP + FP}$
Recall	$\frac{TP}{TP + FN}$
F-1 Score	$\frac{2 * Precision * Recall}{Precision + Recall}$

Here, the recall is quite important, as it demonstrates the ability of the algorithm to detect attack traffic. However, equally important is precision, which will ensure that the system will not block legitimate traffic by creating many false positives. Therefore, the value of the F-1 score, which combines both metrics, is a good measure for the efficacy of the algorithm. Accordingly, for the optimization of the models, the F-1 score has been taken as the main performance measure. In the below subsections, we provide performance results of the DRL model on the two benchmark datasets discussed above and compare them with the results of previous work that have utilized the same datasets for evaluation.

### 4.2.1 Experiments with NSL-KDD

The first set of experiments were performed using the NSL-KDD dataset. The learning system was set up in a Gym environment as explained by Koduvally [37]. Gym provides an environment for testing and comparison of RL algorithms. We used the full training and test datasets for the experiments. We experimented with different numbers of training iterations to evaluate the effect of the number of training iterations on the accuracy of the algorithm. Table 4 lists the precision, recall, accuracy and F-1 score values for the experiments with a low number of training iterations (1) and a high number of training iterations (20). As seen in the table, the algorithm achieves very high precision and recall when the number of training iterations is high.

Table 4 Precision, Recall and Accuracy for Varying Number of Training Iterations in NSL-KDD

	Precision	Recall	Accuracy	F-1 Score
Low Iterations	0.715	0.719	0.725	0.72
High Iterations	0.951	0.925	0.940	0.93

We also experimented with different DNN architectures to see the effects of the number of hidden neurons on the performance of the algorithm. As opposed to the number of iterations, we observe that increasing the number of hidden neurons in the DNN does not always lead to better performance. We have tried five different settings and the results are reported in Table 5 below.

In the first experiment, we set the number of neurons at the hidden layers to be 2/3 of the input layer's size. We achieved satisfying results with an accuracy close to %97. In the second experiment, the number of hidden neurons was set equal to the size of the input layer. The performance was much lower than that of the first setting.

In the third experiment, the number of hidden neurons was one and a half times the input layer's size. This made the performance degrade even further. In the fourth experiment the number of hidden neurons was half the size of the input layer, and while the precision and recall values were quite balanced, this setting did not achieve the performance of the first setting either.

In Experiment 5, we used the square root of the input layer's size as the number of hidden neurons. This provided an increase in performance over the previous settings except for the first experiment.

Table 5 Precision, Recall, Accuracy, and F-1 Score for Varying Number of Hidden Neurons in NSL-KDD

	#of hidden neurons	Precision	Recall	Accuracy	F-1 Score
Experiment 1	$\frac{2 * Input\ size}{3}$	0.98	0.96	0.97	0.97
Experiment 2	Input size	0.65	0.91	0.70	0.76
Experiment 3	$\frac{3 * Input\ size}{2}$	0.72	0.54	0.68	0.62
Experiment 4	$\frac{Input\ size}{2}$	0.77	0.79	0.79	0.78
Experiment 5	$\sqrt{Input\ size}$	0.89	0.92	0.89	0.90

After the initial set of experiments with different numbers of hidden neurons, we optimized the training process by automating the setting of hyperparameter values for the DNN component of the model. The optimization process performs a grid search [38] over all given possible values of the different hyperparameters, calculates F-1 scores achieved with the specific hyperparameter settings on the validation dataset and reports the hyperparameter values resulting in the best F-1 score. Grid search is currently one of the most commonly used hyperparameter optimization techniques in DL, as it has been proven to find the most optimal parameter settings when compared to random search and function approximation techniques for hyperparameter optimization. It involves determining a range of possible values for each hyperparameter and training the model with all combinations of those values to find the combination with the optimal performance. In this work, we included the following hyperparameters for DNN in the automated grid search: (a) learning rate (in the range [0, 0.1]) (b) dropout rate (in the range [0, 0.4]) (c) number of hidden neurons (in the range [6, 60]). Adam optimizer and L2 regularization were used for DNN. The best performance was achieved with a learning rate of 0.01, dropout rate of 0.3 and 27 hidden neurons. Before performing grid search for the selected hyperparameters, we performed trials for the other hyperparameters including the number of epochs, batch size and reward decay rate, and the best performance was achieved with 30 epochs, a batch size of 1000 and a reward decay rate of 0.9. Note that although it is possible to include many hyperparameter types and hyperparameter values in the grid search, the more parameter values included, the longer it takes to train the model. For a large hyperparameter space, the optimization process could take days of training, which has been avoided in the DNN literature, as the resulting model could also overfit the training data, decreasing the usefulness of the model for real-world application. The increase in the training time would also hurt the performance of online learning, which is important in intrusion detection systems that need to continuously update their models with new data.

Table 6 provides performances of state-of-the-art ML algorithms in the literature in terms of precision, recall, accuracy, and F-1 score on the NSL-KDD dataset and example related works in the literature utilizing these algorithms. The models compared against include logistic regression, SVM with the Radial Basis Function (RBF) kernel, random forest, Gradient Boosting Machine (GBM), Adaboost, multi-layer perceptron (MLP), convolutional neural networks (CNN) (results of these are provided by Lopez-Martin et al. [39]), variational autoencoder, deep belief network and fully connected deep neural network (results of these are provided by Yang et al. [40]). All of the included models are state-of-the-art ML/DL models that have been utilized in a variety of intrusion detection systems.

As seen in Table 6, the proposed DRL model achieved good results in all of the performance measures for the NSL-KDD experiments. While models such as random forest, GBM, Adaboost, MLP and CNN achieved quite high precision values, their low recall values caused a lower F-1 score. As recall values demonstrate the ability of the models to detect attacks, it is quite an important metric for the goodness of the models in practice.

Table 6 Performance Comparison of Proposed Approach and Existing ML Approaches on NSL-KDD

ML/DL algorithm	Precision	Recall	Accuracy	F-1 Score	Example Work	Method
DRL (proposed)	<b>0.98</b>	<b>0.96</b>	<b>0.97</b>	<b>0.97</b>	--	
Logistic regression	0.90	0.55	0.71	0.68	Moustafa and Slay [16]	A network intrusion detection system using logistic regression in its decision engine along with association rule mining is proposed.
SVM	0.91	0.88	0.88	0.89	Lopez-Martin et al. [39]	Application of an optimized SVM model on intrusion detection datasets is evaluated.
Random forest	0.97	0.57	0.75	0.72	Lopez-Martin et al. [39]	Application of an optimized random forest model on intrusion detection datasets is evaluated.
GBM	0.97	0.63	0.78	0.76	Engly et al. [15]	The performance of GBM on intrusion detection datasets is evaluated by itself vs. in an ensemble with random forests and neural networks.
Adaboost	0.97	0.60	0.76	0.74	Hu et al. [14]	A computationally lightweight intrusion detection model based on direct application of the AdaBoost algorithm is proposed.
MLP	0.97	0.67	0.80	0.79	Ingre and Yadav [12]	An artificial neural network with Backpropagation (BFG) and tansig activation function is proposed for intrusion detection.
CNN	0.97	0.68	0.81	0.80	Li et al. [22]	An image conversion method for network data is proposed and the resulting data is fed into a convolutional neural network for intrusion detection.
Variational Autoencoder	0.95	0.80	0.80	0.87	Yang et al. [40]	A supervised variational auto-encoder with regularization is proposed, which utilizes Wasserstein GAN for learning latent data distribution.
Deep belief network	0.89	0.55	0.57	0.68	Gao et al. [27]	A DNN classifier comprising multilayer unsupervised learning networks, and a supervised backpropagation learning network is proposed for intrusion detection.
Fully connected DNN	0.89	0.61	0.62	0.73	Vinayakumar et al. [26]	A distributed, fully connected DNN architecture is proposed for intrusion detection in large networks.

#### 4.2.2 Experiments with UNSW-NB15

The second set of experiments was performed with the UNSW-NB15 dataset. As in the previous experiments, the optimal hyperparameters were found using grid search with the same set of possible values as in Section 4.2.1. The best performance was achieved with a learning rate of 0.01, dropout rate of 0.3 and 32 hidden neurons. We report the best performance results in Figure 3 below. We performed two different experiments, where we utilized the default training set consisting of 175341 records and test set consisting of 82332 records in the first experiment. In the second experiment we randomly selected training and test data over the dataset. 100000 records were selected for both sets. The results did not change much in this experiment as compared to the first experiment.

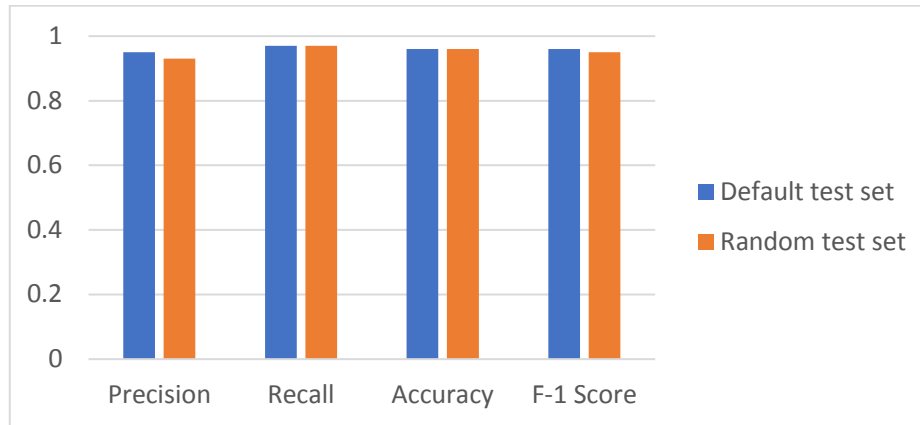


Figure 3 Precision, Recall, Accuracy, and F-1 Score on UNSW-NB15

Table 7 provides a performance comparison of the proposed approach with existing state-of-the-art ML-based approaches in the literature in terms of precision, recall, accuracy, and F-1 score on the UNSW-NB15 dataset. The models in the table are the same as those in Section 4.2.1 and their results are provided by Lopez-Martin et al. [39] and Yang et al. [40] as before.

Table 7 Performance Comparison of Proposed Approach and Existing ML Approaches on UNSW-NB15

Algorithm	Precision	Recall	Accuracy	F-1 Score
DRL (proposed)	<b>0.95</b>	0.97	<b>0.96</b>	<b>0.96</b>
Logistic regression	0.81	0.94	0.84	0.87
SVM	0.75	<b>0.99</b>	0.82	0.86
Random forest	0.83	<b>0.99</b>	0.88	0.90
GBM	0.80	<b>0.99</b>	0.86	0.88
Adaboost	0.80	0.98	0.85	0.88
MLP	0.81	0.98	0.87	0.89
CNN	0.86	0.98	0.90	0.91
Variational Autoencoder	<b>0.95</b>	0.92	0.93	0.94
Deep belief networks	0.85	0.97	0.89	0.91
Fully connected DNN	0.82	0.98	0.87	0.90

As seen in Table 7, high precision, accuracy and F-1 scores are achieved by the proposed DRL-based approach. While for this dataset SVM, random forest and GBM achieve higher recall values, their precision values are much lower than that of the DRL approach, which means they would create many false positives at runtime. The DRL model achieves a better balance between false positives and false negatives, with high precision, recall and F-1 values. This makes it promising for both accurately detecting attacks and achieving high network reliability by avoiding unnecessary interruption of traffic.



The good performance of the DRL model is attributable to the exploration of a wide set of network states and penalizing all incorrect classifications with the same penalty function, which limits the number of false positives and false negatives as the RL agent continues to learn.

## 5. Conclusion

In this work, we proposed a deep reinforcement learning based approach for network intrusion detection. The proposed approach overcomes the generalization shortcomings of reinforcement learning and achieves high performance on binary intrusion detection tasks trying to differentiate between normal and attack traffic. The efficacy of the model was evaluated with two widely used intrusion detection benchmark datasets and F-1 scores of over 96% were achieved for both datasets. We also demonstrated the effects of the number of hidden neurons and number of iterations on the performance of the proposed algorithm. This study has shown that deep reinforcement learning is a promising method for network intrusion detection. We aim to expand upon this study in future work by evaluating the performance of the model on additional datasets as well as creating extensions of the model with different reward functions to achieve optimal performance in a variety of settings.

## References

- [1] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra and M. Riedmiller, "Playing Atari with Deep Reinforcement Learning," pp. 1–9, 2013. Retrieved from <http://arxiv.org/abs/1312.5602>
- [2] M. Lai, "Giraffe: Using Deep Reinforcement Learning to Play Chess," September, 2015. Retrieved from <http://arxiv.org/abs/1509.01549>
- [3] L. Wang, D. Zhang, L. Gao, J. Song, L. Guo and H. T. Shen, "MathDQN: Solving arithmetic word problems via deep reinforcement learning," *32nd AAAI Conference on Artificial Intelligence*, pp. 5545–5552, 2018.
- [4] S. Nemati, M. M. Ghassemi and G. D. Clifford, "Optimal medication dosing from suboptimal clinical examples: A deep reinforcement learning approach," *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2016.
- [5] Z. Zhou, X. Li and R. N. Zare, "Optimizing Chemical Reactions with Deep Reinforcement Learning," *ACS Central Science*, vol. 3, no. 12, pp. 1337–1344, 2017.
- [6] M. Mahmud, M. S. Kaiser, A. Hussain and S. Vassanelli, "Applications of Deep Learning and Reinforcement Learning to Biological Data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2063–2079, 2018.
- [7] T. Yen, A. Oprea and K. Onarlioglu, "Beehive: large-scale log analysis for detecting suspicious activity in enterprise networks," *Proc. 29th Annual Computer Security Applications Conference*, pp. 199–208, 2013.
- [8] A. Razaq, H. Tianfield and P. Barrie, "A big data analytics based approach to anomaly detection," *Proc. - 2016 IEEE/ACM 3rd International Conference on Big Data Computing Applications and Technologies (BDCAT)*, pp. 187–193, 2016.
- [9] A. O. Balogun and R. G. Jimoh, "Anomaly intrusion detection using a hybrid of decision tree

- and K-nearest neighbor," *Journal of Advances in Scientific Research & Applications (JASRA)*, vol. 2, no. 1, pp. 67-74, 2015.
- [10] A. Hariharan, A. Gupta and T. Pal, "CAMLPAD: Cybersecurity Autonomous Machine Learning Platform for Anomaly Detection," *Proc. Future of Information and Communication Conference (FICC), San Francisco, CA, USA*, pp. 705-720, 2020.
- [11] M. S. Pervez and D. M. Farid, "Feature selection and intrusion classification in NSL-KDD cup 99 dataset employing SVMs," *SKIMA 2014 - 8th International Conference on Software, Knowledge, Information Management and Applications*, pp. 1-6, 2014.
- [12] B. Ingre and A. Yadav, "Performance analysis of NSL-KDD dataset using ANN," *2015 International Conference on Signal Processing and Communication Engineering Systems*, pp. 92-96, 2015.
- [13] S.O.M. Kamel, N. Hegazi, H. Harb, A. ElDein and H. ElKader, "AdaBoost Ensemble Learning Technique for Optimal Feature Subset Selection," *International Journal of Computer Networks and Communications Security* vol. 4, no. 1, pp. 1-11, 2016.
- [14] W. Hu, W. Hu, and S. Maybank, "AdaBoost-Based Algorithm for Network Intrusion Detection," *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol. 38, no. 2, pp. 577-583, 2008.
- [15] A. H. Engly, A. R. Larsen, and W. Meng, "Evaluation of Anomaly-Based Intrusion Detection with Combined Imbalance Correction and Feature Selection," *Proc. 14<sup>th</sup> International Conference on Network and System Security*, Melbourne, Australia, pp. 277-291, 2020.
- [16] N. Moustafa and J. Slay, "A hybrid feature selection for network intrusion detection systems: central points and association rules," arXiv:1707.05505, (2017) [cs.CR].
- [17] J. Kim and H. Kim, "Applying Recurrent Neural Network to Intrusion Detection with Hessian Free Optimization." In: Kim H., Choi D. (eds) Information Security Applications. WISA 2015. *Lecture Notes in Computer Science*, vol. 9503, 2016, Springer, Cham.
- [18] Y. Chuan-long, Z. Yue-fei, F. Jin-long and H. Xin-zheng, "A Deep Learning Approach for Intrusion Detection using Recurrent Neural Networks," *IEEE Access*, vol. 5, pp. 21954 - 2196, 2017.
- [19] C. Yin, Y. Zhu, J. Fei, and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," *IEEE Access*, vol. 5, pp. 21954-21961, 2017.
- [20] Z. Li, A. L. G. Rios, G. Xu, and L. Trajkovic, "Machine learning techniques for classifying network anomalies and intrusions," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, pp. 1-5, 2019.
- [21] S. Behera, A. Pradhan, and R. Dash, "Deep Neural Network Architecture for Anomaly Based Intrusion Detection System," *5th International Conference on Signal Processing and Integrated Networks (SPIN 2018)*, pp. 270- 274, 2018.

- [22] Z. Li, Z. Qin, K. Huang, X. Yang, and S. Ye, "Intrusion detection using convolutional neural networks for representation learning," in *Proc. Int. Conf. Neural Inf. Process.* pp. 858–866, 2017.
- [23] M. Lopez-Martin, B. Carro, A. Sanchez-Esguevillas, and J. Lloret, "Conditional variational autoencoder for prediction and feature recovery applied to intrusion detection in IoT," *Sensors*, vol. 17, no. 9, p. 1967, Aug. 2017.
- [24] F. A. Khan, A. Gumaiei, A. Derhab, and A. Hussain, "TSDL: A twostage deep learning model for efficient network intrusion detection," *IEEE Access*, vol. 7, pp. 30373–30385, 2019.
- [25] T. Ma, F. Wang, J. Cheng, Y. Yu, and X. Chen, "A hybrid spectral clustering and deep neural network ensemble algorithm for intrusion detection in sensor networks," *Sensors*, vol. 16, no. 10, p. 1701, Oct. 2016.
- [26] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.
- [27] N. Gao, L. Gao, Q. Gao, and H. Wang, "An Intrusion Detection Model Based on Deep Belief Networks," *Proc. 2<sup>nd</sup> International Conference on Advanced Cloud and Big Data*, Huangshan, China, pp. 247-252, 2014.
- [28] B. Deokar and A. Hazarnis, "Intrusion Detection System using Log Files and Reinforcement Learning," *International Journal of Computer Applications*, vol. 45, no. 1919, pp. 28–35, 2012.
- [29] A. Servin and D. Kudenko, "Multi-agent reinforcement learning for intrusion detection: A case study and evaluation," *Frontiers in Artificial Intelligence and Applications*, vol. 178, pp. 873–874, 2008.
- [30] R. Elderman, L. J. J. Pater, A. S. Thie, M. M. Drugan and M. A. Wiering, "Adversarial reinforcement learning in a cyber security simulation," *ICAART 2017- Proceedings of the 9th International Conference on Agents and Artificial Intelligence*, pp. 559–566, 2017.
- [31] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [32] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement Learning: A Survey," *Journal of Artificial Intelligence Research*, vol. 4, 1996.
- [33] N. Moustafa, J. Slay, "UNSW-NB15: A Comprehensive Data Set for Network i Intrusion Detection Systems (UNSW-NB15 Network Data Set)," *Proceedings of the 2015 IEEE Military Communications and Information Systems Conference (MilCIS)*, pp. 1–6, 2015.
- [34] KDD Cup 1999. Available online: <https://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html> (Accessed on 20 November 2020).
- [35] 1998 DARPA Intrusion Detection Evaluation Dataset. Available online: <https://www.ll.mit.edu/r-d/datasets/1998-darpa-intrusion-detection-evaluation-dataset> (Accessed on 20 November 2020).

- [36] M. Tavallaee, E. Bagheri, W. Lu, and A.A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," *IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pp. 1–6, 2009.
- [37] H. Koduvely, "Github repository, gym-network\_intrusion," Retrieved from [https://github.com/harik68/gym-network\\_intrusion](https://github.com/harik68/gym-network_intrusion), 2018.
- [38] Y. Sun, B. Xue, M. Zhang, and G. G. Yen, "An Experimental Study on Hyper-parameter Optimization for Stacked Auto-Encoders," *Proc. IEEE Congress on Evolutionary Computation*, Rio de Janeiro, Brazil, pp. 1-8, 2018.
- [39] M. Lopez-Martin, B. Carro, A. Sanchez-Esguevillas, and J. Lloret, "Shallow neural network with kernel approximation for prediction problems in highly demanding data networks," *Expert Systems with Applications*, vol. 124, pp. 196-208, 2019.
- [40] Y. Yang, K. Zheng, B. Wu, Y. Yang, and X. Wang, "Network intrusion detection based on supervised adversarial variational auto-encoder with regularization," *IEEE Access*, vol. 8., pp. 42169-42184, 2020.

## Detection of Pneumonia with a Novel CNN-based Approach

 Ebru Erdem<sup>1</sup>,  Tolga Aydın<sup>2</sup>

<sup>1</sup>Corresponding Author; Ataturk University, Department of Computer Engineering, 25240, Erzurum, Turkey; ebruerdem@atauni.edu.tr;

<sup>2</sup>Ataturk University, Department of Computer Engineering, 25240, Erzurum, Turkey; atolga@atauni.edu.tr;

Received 28 August 2020; Revised 25 December 2020; Accepted 28 December 2020; Published online 05 February 2021

### Abstract

Pneumonia is a seasonal infectious lung tissue inflammatory disease. According to the World Health Organization (WHO), early diagnosis of the disease reduces the risk of its transmission and death. Various deep learning and machine learning algorithms were used for pneumonia detection. This study aims to analyze the lung images and diagnose pneumonia disease by employing deep learning approaches. We have suggested a novel deep learning framework for the detection of pneumonia in lung. A comparison was made between the proposed new deep learning model and pre-trained deep learning models. 88.62% accuracy rate has been obtained from the proposed deep learning structure. It was observed that by utilizing the new deep neural network developed, the accuracy results of VGG16 (88.78%) and VGG19 (88.30%), which are among the popular deep learning architectures, can be approximated. The test results show that our proposed model has a better recall value (97.43%) (VGG16 (93.33%) and VGG19 (96.92%)), and a better F1-Score (91.45%) (VGG16 (91.22%) and VGG19 (91.19%)).

**Keywords:** Pneumonia, CNN, VGG16, VGG19

### 1. Introduction

Lung is a vital organ, and lung abnormalities are highly risky among people. An example of a risk-bearing condition is lung pneumonia. Lung pneumonia (pneumonia) is an inflammation of the lung tissue by various microorganisms. Pneumonia can be detected from chest x-ray images. However, this practice requires highly qualified radiologists. Since there is the risk of confusing pneumonia with other lung diseases, pneumonia detection has turned out to a time consuming process. Computer aided systems (CAD) are being developed to overcome these problems. Thanks to CAD, early detection enables effective treatment and reduces the risk.

With the recently developed deep learning techniques, early diagnosis can be made and the progression of the disease can be prevented. For this reason, the subject to be studied and the technique to be applied is a prominent phenomenon in current medical fields. Automated detection studies of pneumonia with machine learning or deep learning solutions are found in the literature. The detection of the pneumonia was made with AI techniques and results have been obtained as follows;

Among these studies, in the study conducted by Ilyas Sirazitdinov et al, lung pneumonia was detected using images in the "Chest X-ray" database. For this purpose, two convolutional neural networks, Mask R-CNN [1] and RetinaNet [2], were used. The proposed solution was tested on 26,684 image sets from Kaggle Pneumonia Detection Struggle. Good results were obtained for the diagnosis of automatic pneumonia with 79.3% recall [3]. In the study conducted by Enes Ayan et al., CNN [4] [5] networks' performance in pneumonia disease detection was investigated. For this purpose, VGG16 [6] and Xception [7] were compared. VGG16 has been observed to exceed the Xception's accuracy rate of 87%. The Xception network was observed to be more successful than the detection of the VGG16 network. VGG16's success in detecting normal situations was higher than Xception [8]. Abhir Bhandary et al. a deep learning approach that identifies lung abnormalities on chest x-ray images is presented. Performance comparisons were performed by using pre-trained deep learning techniques such as AlexNet [9], VGG16, VGG19, ResNet50 [10]. 96.80% success was achieved with the proposed MAN-SVM method [11]. Gaobo Liang et al. proposed a method of transfer learning for the diagnosis of pediatric pneumonia. The proposed network includes 49 convolutional layers and the ReLU activation function, 1 global mean pooling layer and 2 dense layers. 96.7% recall and 92.7% F1-score rates were

achieved in the classification of pneumonia of children. Also, Liang et al. applied CNN and VGG16. They achieved an accuracy rate of, recall rate, precision rate, and F1-score rate of 90.5%, 96.7%, 89.1%, and 92.7%, respectively for CNN model. They achieved an accuracy rate of, recall rate, precision rate, and F1-score rate of 74.2%, 95.1%, 72.3%, and 82.2%, respectively for VGG16 [12]. 96.4% success was achieved with the transfer learning method proposed by Vikash Chouhan et al. on the dataset received from the Guangzhou Women's and Children's Medical Center [13]. Pneumonia was diagnosed using a sequential convolutional neural network customized by Raheel Siddiqi. 93.75% success rate was achieved using an 18-layer neural network [14]. Yadav and Jadhav applied CapsNet. They obtained an accuracy rate of 82.50% [15]. Asnaoui et al. proposed a CNN model and achieved the accuracy, recall, precision, and F1-score rates of 84.18%, 78.33%, 94.05%, and 85.66%, respectively [16]. Mittal et al. applied E3CC and VGG16+CapsNet. They achieved an accuracy rate of 81.54% for E3CC and an accuracy rate of 88.30% for VGG16+CapsNet [17]. Jain et al. proposed a CNN model. They achieved the accuracy, recall, and F1-score rates of 85.26%, 94%, and 89%, respectively [18]. Chakraborty et al. proposed a CNN model. They achieved the accuracy, recall, and precision rates of 95.62%, 95%, and 96%, respectively [19].

CNNs are shown as the most recent technique applied, as can be seen from the studies examined. With the classification made using convolutional networks, the detection of the disease can be done with a high level of success.

The contributions of the paper are as follows:

- We proposed a new CNN model in addition to existing models.
- The classifications were conducted on chest x-ray images with the CNN-based pre-trained models. Comparisons were made between the proposed new model and the pre-trained deep learning models.

## 2. Dataset and Pre-processing

In this study, evaluations were performed on chest x-ray images. The data set used in the relevant study has open access permission. These images were taken from Guangzhou Women and Children's Medical Center [20]. Data were selected from past cohort subjects (Figure 1). The dataset contains a total of 5856 chest x-ray images 4273 of which having pneumonia and 1583 normal. The dataset is organized in 3 folders (train, validation, test), as given in Table 1 [21]. Each folder contains x-ray images of all patients.

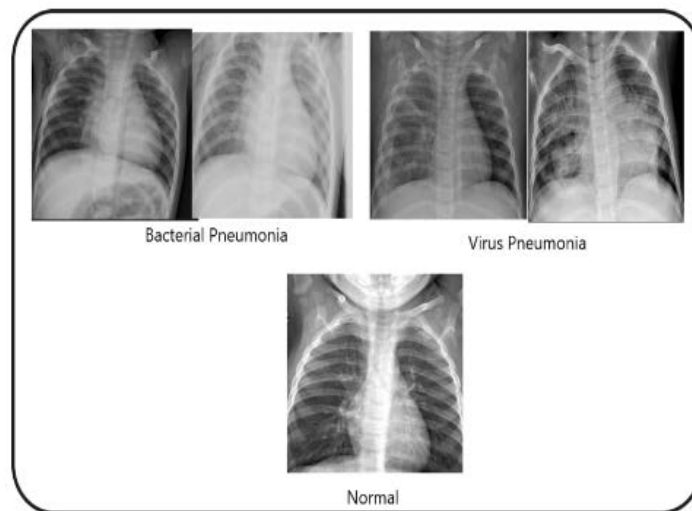


Figure 1 Chest x-ray images

Before the training phase, images with varying width and height values have been reshaped to 224 \* 224. Because, pre-trained models accept images at 224x224x3 dimensions. Deep learning algorithms

need a huge amount of data to improve performance. Augmentation of data is one of the solutions to get rid of sparse data. Using the train data generator, images were rescaled using 1./255 ratio, zoom which the range for a random zoom that was set to 0.3.

Table 1 Distribution of dataset

	<b>Pneumonia</b>	<b>Normal</b>
<b>Training Set</b>	3875	1341
<b>Test Set</b>	390	234
<b>Validation Set</b>	8	8

### 3. Method

In this study, in addition to the pre-trained deep learning models VGG-16 and VGG-19 applied in the literature, a new deep learning model has been proposed.

#### 3.1. Proposed Model

CNNs contain filters that allow us to collect important information embedded within the image. Since convolutional neural networks do not require pre-treatment or feature extraction on the image, they are run directly on the pixels. The separable convolutional neural networks are a variation of the convolutional neural network. In this network, the filter to be applied on the image is implemented in 2 stages: depth convolution and point convolution. Each color channel is moved on the image during depth convolution phase. Then the resulting images are stacked. Point convolution is applied on the stacked image. At this stage, by increasing the number of channels, the 1x1 filter is passed through every point on the picture. With these two stages, it is aimed to reduce the total number of impacts applied in classical convolutional layers and to save memory and time. Separable convolutional layers, when used instead of convolutional layers in the Inception model, created a new architecture, Xception. This architecture has been shown to perform better in 350 million image classifications than Inception V3 with evolutionary layers [22].

The proposed model consists of 3 convolutional blocks and 3 separable convolutional blocks. 3x3 filters are used in the blocks. Each block is separated by the maximum pooling layer. 2x2 filters are used in the maximum pooling layers. There are 6 convolutional layers in the first 3 blocks and 6 separable convolutional layers in the last 3 blocks. 16, 32, and 64 filters are used in the structure of the first, second, and third convolutional blocks, respectively. 32, 64, and 128 filters are used in the structure of the first, second, and third separable convolutional blocks, respectively (Fig. 2). ReLU function is used as the activation process for hidden layers. To increase the training performance, overfitting has been prevented by using the dropout layer in the model. The first, second, third, fourth and fifth dropout layers have drop rates of 25%, 20%, 90%, 70%, and 50%, respectively. In the last stage, Flatten and 4 fully connected (fc) layers are used. There are 2048 neurons (nodes) in the first fc layer, 1024 neurons in the second fc layer, 512 neurons in the third fc layer, and 1 neuron in the last fc layer. At the last layer of the model, the sigmoid function is implemented. Binary cross-entropy is used as the loss function because of the binary classification. Details of the layers are presented in Table 2.

For the proposed model, images have been reshaped to 150\*150 with three channels before the training phase. This is due to obtaining higher accuracy vales when compared to images of 224 \* 224 with three channels.

Table 2 Details of layers used in proposed model

<b>Layer</b>	<b>Stride</b>	<b>Filter size</b>	<b>Pool size</b>	<b>Padding</b>	<b>Activation</b>
Input Layer	-	-	-	-	-
Conv1	1	3x3	-	same	relu
Conv2	1	3x3	-	same	relu
MaxPool1	2	-	2x2	-	-
Conv3	1	3x3	-	same	relu
Conv4	1	3x3	-	same	relu
MaxPool2	2	-	2x2	-	-
Conv5	1	3x3	-	same	relu
Conv6	1	3x3	-	same	relu

Table 2 Details of layers used in proposed model (cont.)

Layer	Stride	Filter size	Pool size	Padding	Activation
MaxPool3	2	-	2x2	-	-
Dropout1	-	-	-	-	-
SepConv1	1	3x3	-	same	relu
SepConv2	1	3x3	-	same	relu
BatchNorm	-	-	-	-	-
MaxPool4	2	-	2x2	-	-
SepConv3	1	3x3	-	same	relu
SepConv4	1	3x3	-	same	relu
BatchNorm	-	-	-	-	-
MaxPool5	2	-	2x2	-	-
SepConv5	1	3x3	-	same	relu
SepConv6	1	3x3	-	same	relu
BatchNorm	-	-	-	-	-
MaxPool6	2	-	2x2	-	-
Dropout2	-	-	-	-	-
Flatten	-	-	-	-	-
Dense	-	-	-	-	relu
Dropout3	-	-	-	-	-
Dense	-	-	-	-	relu
Dropout4	-	-	-	-	-
Dense	-	-	-	-	relu
Dropout5	-	-	-	-	-
Dense	-	-	-	-	sigmoid

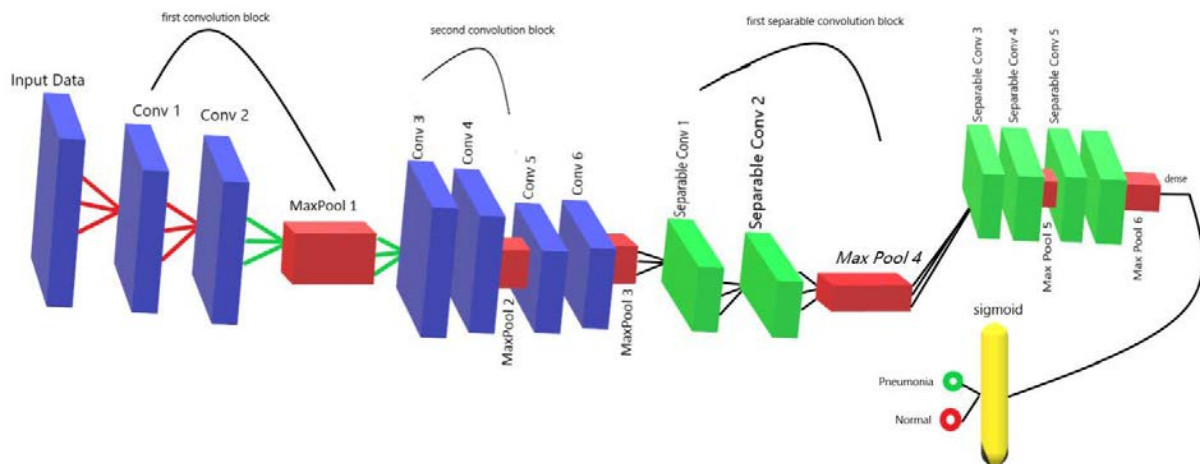


Figure 2 Architecture of the proposed model

### 3.2. Transfer Learning

With a data set containing a lot of data and classes, the pre-trained models can benefit from the transfer learning method. Using pre-trained models with the transfer learning method can yield successful results, especially in smaller data sets. In this study, deep learning architectures previously trained on larger data sets were used on chest X-ray images with transfer learning.

In 2014, a new model was introduced that improved the success of the AlexNet model. In this model approach, the error rate was decreased by increasing the depth and by reducing the filter size. This model, called VGGNet, has different network structures such as 16-layer VGG-16 and 19-layer VGG-19. The VGG-16 and VGG-19 neural networks have a convolutional layer, a pooling layer, a flatten layer, a dropout layer and a dense layer (Fig. 2.). The images that will enter the models must be 224x224. The weights obtained to recognize the 1000 class problem in the Imagenet competition with these proposed models were used for the problem in this study. The model was customized to solve the 2-class problem by performing fine tuning in the last layer in the study (Fig. 3).



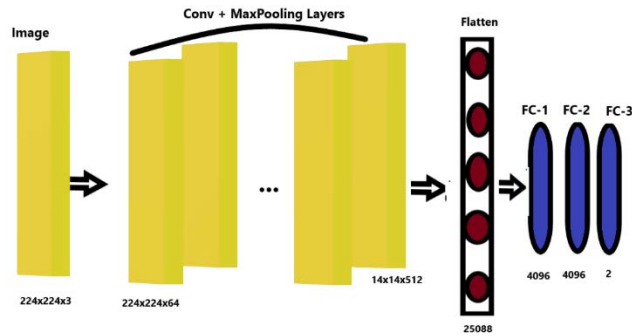


Figure 3 VGG-16 and VGG-19 model architectures

Architectural structures of the proposed and the compared models are given in Table 2. In the new model, additionally, a separable convolution layer is used. The number of convolution layers in VGG-16 and VGG-19 almost corresponds to the number of convolutions and separable convolutions in the new model. Relu, dense and dropout layers are used in all three models. It is remarkable that the number of parameters of the new model is quite low when compared to the number of parameters of the other two models.

Table 2 Details of models

Models	Convolution Layer Number	Separable Convolution Layer Number	Relu Layer Number	Dense Layer Number	Parameter Value
Proposed Model	6	6	15	4	23,936,513
VGG-16	13	---	15	3	134,264,641
VGG-19	16	---	18	3	139,574,337

#### 4. Results

The deep learning algorithms used in the study were run on the computer with the configurations given in Table 3.

Table 3 Configurations

<b>Memory</b>	245676MB, 24GB
<b>Processor</b>	Intel(R) Core(TM) i7-7700 CPU @3.60GHz (8 CPUs), ~3.6GHz
<b>Graphics Card</b>	Intel(r) Hd Graphics 630, NVIDIA GeForce GT 730
<b>Operating System</b>	Windows 10 Pro 64 Bit (10.0, build 18362)

In the application section, keras, tensorflow and matplotlib libraries were used for all models in Python environment. When the hyper parameters in the model are examined, the number of epoch indicates how many times the data should pass through the model. The model was asked to work in 20 iterations. During the training phase, it was decided to process 32 data (Table 4).

In this study; sensitivity, accuracy, precision and F1-score measurement metrics have been used for performance evaluation. The success rates of the algorithms are given in Table 4. Accuracy is considered as the main performance evaluation metric of the study. Accordingly, while the batch size is 32, the proposed model outperformed VGG-19 and did not exceed VGG-16. VGG-16 gave the best results in terms of precision. On the other hand, for the recall criterion, the proposed model gave the best result. In order to evaluate the success of recall and precision criteria together, the F1-measurement was examined and the proposed model gave the highest result. When the results obtained in Table 4 are analyzed thoroughly, it is observed that the proposed model behaves between the VGG-16 and VGG-19 architectures.

Table 4 Comparative results

Model	Batch Size	Epoch	Optimizer	Recall (%)	Precision (%)	F1-Score (%)	Test Accuracy (%)
VGG-19	32	20	ADAM	96.92	86.10	91.19	88.30
<b>Proposed Model</b>	32	20	ADAM	97.43	86.16	91.45	88.62
VGG-16	32	20	ADAM	93.33	89.21	91.22	88.78

Table 5 shows the comparison of the proposed model with the previous studies using the same dataset [20]. Liang and Zheng proposed a CNN-based model to detect the disease. Our model had a higher recall rate (97.43%) when compared to their model (96.7%). Accuracy, precision and F1-Scores were close to the values of their study. Liang and Zheng also studied on VGG16 to detect the disease. In this context, our model has proven to be better in terms of accuracy, precision, recall, and F1-Score. Yadav and Jadhav studied CapsNet to detect the disease and obtained an accuracy of 82.50%. Mittal et al. studied E3CC and VGG16+CapsNet to detect the disease and obtained accuracy rates of 81.54% and 88.30%, respectively. In this context, our model outperformed in terms of accuracy performance criterion when compared to these two methods. Asnaoui et al. proposed a new CNN-based model. Our model was better in terms of accuracy, recall, and F1-Score performance criteria. Ayan and Ünver studied VGG16 and Xception to detect the disease. Our model, again, outperformed in terms of accuracy and recall performance criteria. When compared to another study conducted by Chakraborty et al., we obtained a higher recall value of 97.43% with respect to 95%. Finally, when compared to the study conducted by Jain and et al., our accuracy, recall, and F1-Score performance criteria values were better than their results.

Table 5 Comparison with the previous studies

Article	Year	Method	Accuracy(%)	Recall(%)	Precision(%)	F1-score(%)
Ayan and Ünver [8]	2019	VGG16	87	82	-	-
Ayan and Ünver [8]	2019	Xception	82	85	-	-
Chakraborty et al. [19]	2019	CNN	95.62	95	96	-
Yadav and Jadhav [15]	2019	CapsNet	82.50	-	-	-
Liang and Zheng [12]	2020	CNN	90.5	96.7	89.1	92.7
Liang and Zheng [12]	2020	VGG16	74.2	95.1	72.3	82.2
Asnaoui et al. [16]	2020	CNN	84.18	78.33	94.05	85.66
Mittal et al. [17]	2020	E3CC	81.54	-	-	-
Mittal et al. [17]	2020	VGG16+CapsNet	88.30	-	-	-
Jain et al. [18]	2020	CNN (Model 1)	85.26	94	-	89
<b>Proposed Model</b>	2020	CNN	88.62	97.43	86.16	91.45

“—” denotes that the information is not mentioned in the associated paper.

Considering the processing time as a benchmark criterion, it was observed that the proposed model run in a shorter time. The reason is possibly due to usage of less number of parameters (Table 6).

Table 6 Proposed model, VGG16, and VGG19 running times

	Time to reach result
<b>Proposed Model</b>	2 hours 14 minutes
VGG16	11 hours 7 minutes
VGG19	14 hours 4 minutes

The accuracy and loss graphs of VGG16, VGG19, and the proposed model are shown in Figs. 4, 5, and 6, respectively. When the accuracy and loss graphs are examined, it is observed that the training accuracy values are higher than the test (validation) accuracy values and the test (validation) loss values are higher than the training loss values for the same epoch values.

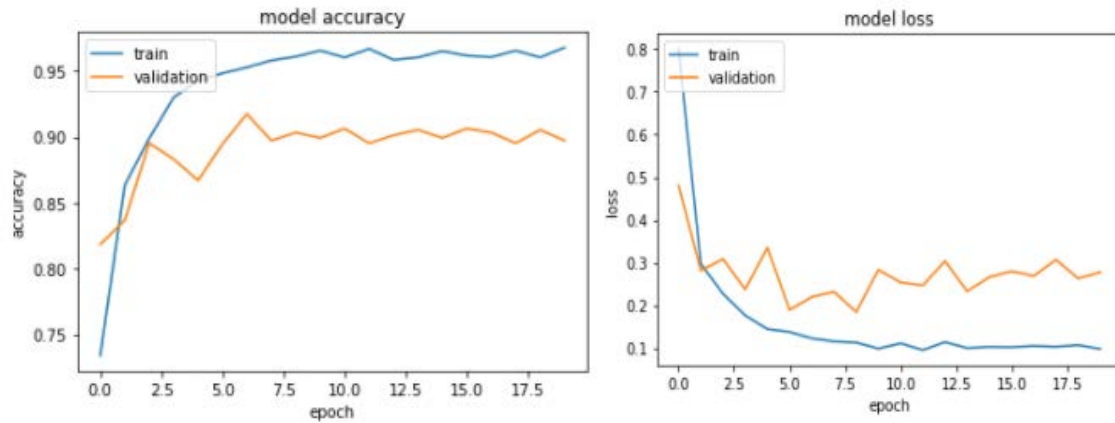


Figure 4 The accuracy and loss graph of VGG16

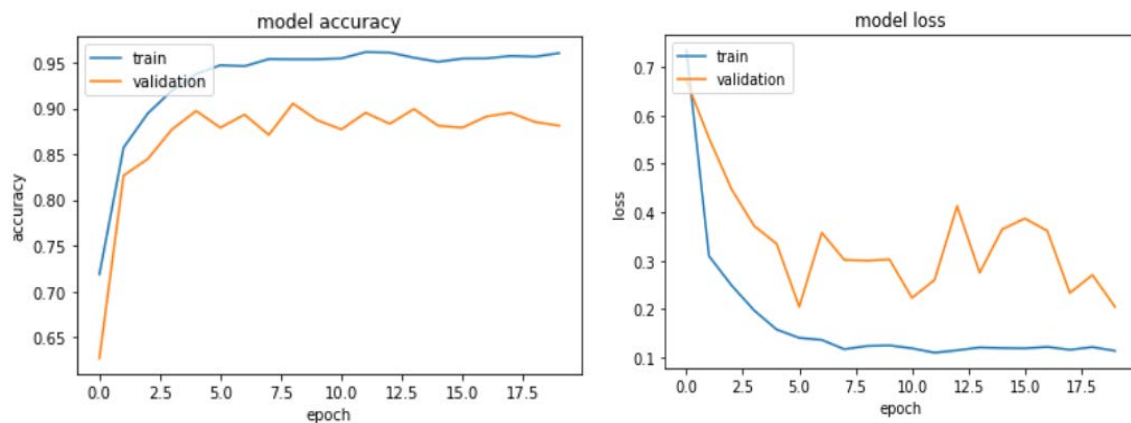


Figure 5 The accuracy and loss graph of VGG19

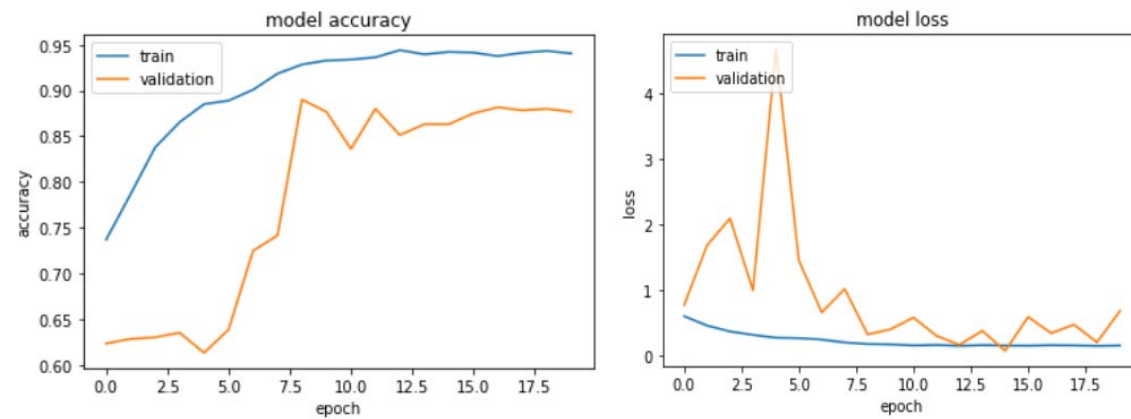


Figure 6 The accuracy and loss graph of the proposed model

## 5. Conclusions

The study focused on biomedical image interpretation and medical decision in lung pneumonia, which is life-threatening for children and the elderly. As computer aided systems are instrumental in early diagnosis, new models can be proposed in order to reduce risks.

Thanks to their higher performance in solving many related problems, deep learning algorithms have proven to be stronger than artificial intelligence algorithms. Therefore, this study utilized pre-trained deep learning models and a new deep learning model has been developed. Transfer learning has been carried out through pre-trained models. A comparison of these models with the proposed model has been made. It has been observed that the developed model gives better results than VGG-19 and worse

results than VGG-16. We think that the proposed model is open to development. Due to this situation, it is concluded that more experiments should be conducted with different hyper parameters of the model. Performance may possibly be improved by making changes in learning ratio, optimizer method and batch size.



In the future studies, it is aimed to detect COVID-19 from lung film images by using the advanced version of this model.

## References

- [1] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn," in Proceedings of the IEEE international conference on computer vision., 2017.
- [2] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in Proceedings of the IEEE international conference on computer vision., 2017.
- [3] I. Sirazitdinov, M. Kholiavchenko, T. Mustafaev, Y. Yixuan, R. Kuleev, and B. Ibragimov, "Deep neural network ensemble for pneumonia localization from a large-scale chest x-ray database," *Computers & Electrical Engineering.*, vol. 78, pp. 388-399, 2019.
- [4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE.*, vol. 86, no. 11, pp. 2278-2324, 1998.
- [5] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks.*, vol. 3361, no. 10, pp. 1995, 1995.
- [6] K. Simonyan, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.*, vol. 1556, 2014.
- [7] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in Proceedings of the IEEE conference on computer vision and pattern recognition., 2017.
- [8] E. Ayan, and H.M. Ünver, "Diagnosis of Pneumonia from Chest X-Ray Images Using Deep Learning," in 2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT)., 2019.
- [9] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems.*, 2012.
- [10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition., 2016.
- [11] A. Bhandary, G.A. Prabhu, V. Rajinikanth, K.P. Thanaraj, S.C. Satapathy, D.E. Robbins, C. Shasky, Y.-D. Zhang, J.M.R.S. Tavares, and N.S.M. Raja, "Deep-learning framework to detect lung abnormality–A study with chest X-Ray and lung CT scan images," *Pattern Recognition Letters.*, vol. 129, pp. 271-278, 2020.
- [12] G. Liang and L. Zheng, " A transfer learning method with deep residual network for pediatric pneumonia diagnosis," *Computer methods and programs in biomedicine.*, vol. 187, pp. 104964, 2020.
- [13] V. Chouhan, S.K. Singh, A. Khamparia, D. Gupta, P. Tiwari, C. Moreira, R. Damasevicius, and V.H.C. de Albuquerque, "A Novel Transfer Learning Based Approach for Pneumonia Detection in Chest X-ray Images," *Applied Sciences.*, vol. 10, no. 2, pp. 559, 2020.

- [14] R. Siddiqi, "Automated Pneumonia Diagnosis using a Customized Sequential Convolutional Neural Network," in *Proceedings of the 2019 3rd International Conference on Deep Learning Technologies.*, 2019.
- [15] S.S. Yadav, S.M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," *J Big Data.*, vol. 6, pp. 113, 2019.
- [16] K.E. Asnaoui, Y. Chawki, and A. Idri, "Automated methods for detection and classification pneumonia based on x-ray images using deep learning," *arXiv preprint arXiv:2003.*, pp. 14363, 2020.
- [17] A. Mittal, D. Kumar, M. Mittal, T. Saba, I. Abunadi, A. Rehman, and S. Roy, "Detecting Pneumonia Using Convolutions and Dynamic Capsule Routing for Chest X-ray Images," *Sensors.*, vol. 20, no. 4, pp. 1068, 2020.
- [18] R. Jain, P. Nagrath, G. Kataria, V.S. Kaushik, and D.J. Hemanth, "Pneumonia detection in chest X-ray images using convolutional neural networks and transfer learning," *Measurement.*, vol. 165, pp. 108046, 2020.
- [19] S. Chakraborty, S. Aich, J.S. Sim, and H.C. Kim, "Detection of pneumonia from chest x-rays using a convolutional neural network architecture," in *International Conference on Future Information & Communication Engineering.*, vol. 11, no. 1, pp. 98-102, 2019.
- [20] D. Kermany, K. Zhang, and M. Goldbaum, "Labeled optical coherence tomography (oct) and chest X-ray images for classification," *Mendeley data.*, vol. 2, 2018.
- [21] P. Mooney, "Chest X-Ray Images (Pneumonia)," 2017. [Online]. Available: <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>. [Accessed: 27-June-2020].
- [22] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition.*, 2016.

# Sentiment Analysis on Social Media Reviews Datasets with Deep Learning Approach

 Muhammet Sinan Başarslan<sup>1</sup>,  Fatih Kayaalp<sup>2</sup>

<sup>1</sup>Corresponding Author; Dogus University, Advanced Vocational School, Computer Programming;  
mbasarslan@dogus.edu.tr;

<sup>2</sup>Duzce University; Engineering Faculty, Department of Computer Engineering; fatihkayaalp@duzce.edu.tr

Received 28 November 2020; Revision 20 December 2020; Accepted 4 February 2021; Published online 15 February 2021

## Abstract

Thanks to social media, people are now able to leave guiding comments quickly about their favorite restaurants, movies, etc. This has paved the way for the field of sentiment analysis, which brings together various disciplines. In this study, Yelp restaurant reviews and IMDB movie reviews dataset were used together with the data collected from Twitter. Word2Vec (W2V), Global Vector (GloVe) and Bidirectional Encoder Representation (BERT) word embedding methods, Term Frequency-Reverse Document Frequency (TF-IDF), and the Bag-of-Words (BOW) were used on these datasets. Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), Recurrent Neural Network (RNN), Support Vector Machine (SVM), and Naive Bayes (NB) were used in the sentiment analysis models. Accuracy, F-measure (F), Sensitivity (Sens), Precision (Pre), and Receiver Operating Characteristics (ROC) were used in the evaluation of the model performance. The Accuracy rates of the models created by the Machine Learning (ML) and Deep Learning (DL) methods using the IMDB dataset were in the range of 81%-90% and 84%-94%, respectively. These rates were in the range of 80%-86% and 81%-89% for the Yelp dataset, and in the range of 75%-79% and 85%-98% for the Twitter dataset. The models that incorporated the BERT word embedding method have the best performance, compared to the other models with ML and DL. Therefore, BERT method is recommended for this type of analysis in future studies.

**Keywords:** sentiment analysis, deep learning, machine learning, text representation, word embedding.

## 1. Introduction

In parallel with the advances in technology, visual and print communication channels have shifted towards social media. Social events such as movies, restaurants, concerts are now publicized through articles published on social media or websites, instead of recommendations on newspapers and magazines, thanks to the Internet technologies.

The fact that social media is an indispensable tool for people and that they constantly express their opinions about social issues, economy, health, products, and brands paves the way for sentiment analysis. Sentiment analysis is carried out using natural language processing, an important part of artificial intelligence. In the sentiment analysis studies, underlying sentiments in textual expressions are identified. This analysis is used to see whether the sentiment of the texts shared by people is positive, negative, or neutral. Sentiment analysis are used by companies to see whether they receive a positive feedback [1].

The purpose of the text classification is to assign single or multiple tags to a text string. Conventional approaches for text classification, and the classification in the feature extraction step of BOW, usually utilizes the TF-IDF probabilities. With the advances in natural language processing, BERT, Word2Vec, and GloVe have started to be widely used in feature extraction. However, these methods often ignore the contextual information or word order in texts and they have data flexibility issues, which affect classification accuracy. NB, support vector machines, decision trees, networks such as CNN and LSTM based RNN are used in recent ML algorithms.

In this study, architectures that increase the classification performance in ML and DL models was investigated by applying the traditional text representation method and word embedding methods, which

are widely used in sentiment analysis studies. The model with the best result was proposed as the recommended framework.

In the study, five different datasets were obtained using traditional text representation methods of TF-IDF, BOW, and the word embedding methods BERT, Word2Vec, and GloVe were used on three different datasets. After obtaining these datasets, sentiment analysis, which is one of the natural language processing tasks, was carried out by using ML algorithms of support vector machine and Naive Bayes classifier algorithms, and by using the DL methods of CNN, RNN, and LSTM. Accuracy, F, Sens, Pre, and ROC performance criteria were used in the evaluation of the models created by ML and DL.

As a contribution to the literature, hybrid classifier models of DL and ML were created by using word representation methods for meaning, context, and syntax on public data sources and datasets collected by the researchers.

As shown in the related studies section, classifier models created by ML such as SVM, ANN, and NB, CNN, RNN, LSTM DL are popular and have good performances in sentiment analysis studies. As another contribution, this study evaluates the performance of these algorithms by comparing them with traditional frequency-based text representation (TF-IDF, BOW) and prediction based text representation (W2V, GloVe, BERT) methods.

In the second section, sentiment analysis studies with ML and DL are discussed. In the third section, under the methodology subtitle, datasets used in the study, word representation and embedding methods, ML, and DL algorithms are discussed. The fourth section explains the proposed framework in the study. In the fifth section, the experiments made with the created models and their results are presented. Finally, the sixth section draws the conclusions. The flowchart of the study is shown in Figure 1.

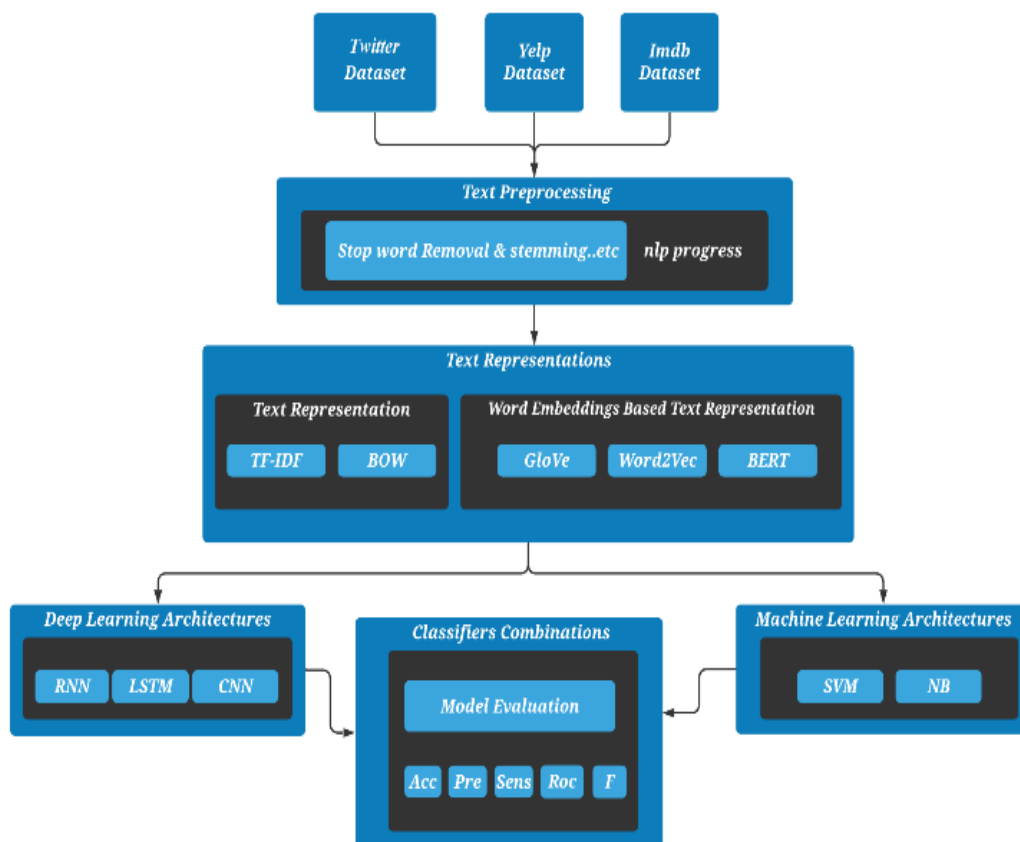


Figure 1 The flowchart of study

## 2. Related Works

Sentiment analysis studies with various datasets in different languages are introduced in this section. In their sentiment analysis study, Pang et al. have created a pre-classification vector space model on the movie comments present in the Internet Movie Database archive, and conducted a sentiment analysis via classifying algorithms, such as NB, Maximum Entropy (ME), and SVM. Of the classification algorithms, they achieved the best performance with SVM, by 82.9% accuracy, using unigrams on the dataset [2].

In their study on movie reviews, Kaynar et al. used NB, Multilayered Artificial Neural Network (ANN), and SVM. They also used TF-IDF for feature extraction. SVM has yielded better results in terms of accuracy, compared to other methods [3].

Hamoud et al. have used the BOW, TF, and TF-IDF for the classification of political tweets on the Twitter data. They used SVM and NB classification algorithms. According to the results, BOW-enabled SVM provides the highest accuracy and F-measure [4].

Symeonidis et al. used Linear SVC, Bernoulli NB, Logistic Regression (LR), and CNN, which are four popular ML algorithms. They achieved the best results by CNN in terms of accuracy [5].

A deep-learning-based approach using convolutional neural network (CNN) and word2vec on Twitter dataset to detect opportunities for improving the quality of their products or services through sentiment analysis has also been proposed in [6]. The study has obtained encouraging results with 88.7% precision, 88.7% recall, and 88.7% F-measure.

Zheng et al. have proposed a model based on the hybrid bidirectional RNN in their study conducted with various datasets such as Sogou, Yelp and Douban Movies. The accuracy rates of the method they proposed varies between 73.46% and 96.81% [7].

Huq et al. have used feature extraction with n-grams on Twitter data and then applied SVM and K-Nearest Neighbor algorithms on the dataset. According to their experiments, accuracy values were between 58.39% and 79.99% [9].

Amolik et al. have classified tweets correctly by using Feature-Vector, NB, and SVM classifier algorithms. Despite its lower recall and accuracy, NB had better sensitivity compared to SVM [10].

Liao et al. have created a simple CNN model with W2V on the data collected from Twitter, and have used this model for comparison against SVM and NB. As a result, CNN has shown to have higher classification performance in terms of accuracy compared to other models [11].

Li et al. have achieved a classification accuracy in the range of 52.23-55.93% in their experiments with DL architectures, such as CNN, LSTM, MemNET, AttNet, applied to three different datasets of Online debates, Restaurants, and laptop reviews [12].

Li et al. have proposed an improved version of the Sliced RNN and have compared this model against various DL models in a sentiment study. According to the results, their proposed model had the highest accuracy by 73.36% [13].

Zhao et al. obtained the highest accuracy rate of 87.9% in the models they created with CNN and LSTM DL algorithms on the Amazon product reviews dataset [14].

Al-Smadi et al. have shown better results in the models they created with the DL RNN and ML SVM algorithms, on the data of Arabic hotel reviews. They obtained an accuracy rate of 87% with RNN and 95.4% with SVM [15].

In their study, Tang, Qin et al. have achieved an accuracy of 80.95% on the restaurant views dataset with the DL algorithm, while they achieved an accuracy of 72.37% in laptop views [16].

In the study of Chen et al. on Chinese Twitter data with RNN-based models, an accuracy of 73.89% was obtained [17].



Altrabsheh et al. used NB, SVM, ME, and Random Forest (RF) algorithms in sentiment analysis with unigram, bigram, trigram-based text representations on the tweets about courses such as mathematics, database, engineering, molecular biology, chemistry, and physics. Models created with SVM and text representations had better performance compared to the other models [18].

H. Ghulam et al. have created models with LSTM, RF, NB in a sentiment analysis study on Roman Urdu tweets. Models created with LSTM had better performance compared to the other models [19].

J. Singh et al. have combined sentiment analysis and morphological assessment in Punjabi language, using DL. The accuracy rate of the model, created using DL and morphological text classification with 275 suicide cases in Punjab, was 95.45% [20].

As seen above, mostly traditional word representation methods were used in previous studies. In this study, the performances of traditional machine learning and deep learning classification algorithms were investigated also by using different text representation and word embedding techniques.

As seen above, DL algorithms such as RNN, LSTM, CNN, and ML algorithms such as NB and SVM are so popular in sentiment analysis studies. In addition, different word embedding methods such as BERT, W2V, GloVe, TF-IDF and BOW have also been used in various studies.

### 3. Methodology

In this section, the datasets, word embedding techniques, ML and DL algorithms, and details of the proposed system are discussed.

#### 3.1 Datasets

Three different datasets were used in the study. These datasets include the IMDB movie review dataset, which is often used in sentiment analysis studies, Yelp hotel and restaurant comments, and Twitter API.

Yelp (restaurant reviews) dataset consists of 598,000 reviews of various restaurants. 560,000 of the reviews were reserved for training and 38,000 for testing [21]. Dataset attributes and descriptions of these features are presented in Table 1.

Table 1 Yelp Dataset

Attribute	Description
Text	Review from yelp
Sentiment class	Positive, negative

IMDB (movie reviews) dataset consists of 50,000 positive and negative movie reviews [22]. In this dataset, 50,000 reviews were split into 25,000 testing and 25,000 training data. Dataset attributes and descriptions of these features are presented in Table 2.

Table 2 IMDB Dataset

Attribute	Description
Text	Review from IMDB
Sentiment class	Positive, negative

4500 health-related Twitter data were collected using the Twitter API. The pre-processing and sentiment analysis of these data were carried out using the Python programming language. The collected tweets were labeled as 1680 neutral, 1220 positive, 1600 negative tweets. The neutral-tagged tweets were the drug ads, and their attribute information is presented in Table 3. Tweets marked as negative seem to belong to those with various diseases. On the other hand, the positive ones are the tweets indicating that diseases such as cancer have successfully treated.

Table 3 Twitter Dataset

Attribute	Description
id	Order of tweet data frame
text	tweet
created_at	Date and time the Tweet was posted
retweeted	Tweet rerun status (bool)
retweet_count	Number of retweets
user_screen_name	Username
user_followers_count	Number of followers
user_location	Followers location
hashtags	Tweet tag
sentiment_score	Sentiment score
sentiment_class	positive, negative, neutral

Since the datasets were scraped from the web, some HTML (Hyper Text Markup Language) codes were also present in the datasets. Therefore, it was necessary to clear these texts by removing HTML tags. The numbers, punctuation, and stop words were removed. Although BERT gives successful results in splitting compound names made with word representation dashes, other methods have problems. A set of NLTK (Natural Language Tool Kit) stop words was used to remove stop words. Since BERT embedding was trained on Wikipedia data, we allowed numbers and some of the punctuations like [, / () : ; '] and compound nouns with a hyphen, which may cause a more reliable embedding to remain in the text. Moreover, we saved [! ? .] to detect the end of the sentence for a later purpose (generate BERT for each sentence). Stemming and lemmatization according to POS (Part of Speech) tags of words were used for BOW and TF-IDF embedding. Finally, we replaced white spaces with only one space.

### 3.2 Text Representation

The representation of documents in text processing is important for successful results. In the text classification applications, texts are represented as vectors in the dataset. Such vector corresponds to the words in the document. Vector representation of documents. A document-word matrix is created. Thus, the words in the document are of importance. Vectors are calculated using various word weighting methods. TF-IDF is a weighting method widely used in text processing. In this method, the frequency of each word is represented by multiplying the inverse document frequency (IDF). This decreases the importance of highly repetitive words and increases the importance of words with fewer words.

There are also word embedding techniques used without document representation. In this study, however, the following document representation methods, BOW and TF-IDF, were used.

#### 3.2.1 TF-IDF

TF is the method used to calculate term weights in a document. Eq. (1) is seen. The IDF tries to find out the number of words in more than one document and to determine whether the word is a term or not (Stop Words). For this, the absolute value of the logarithm of the number of documents passed by the term must be divided by the number of documents. Eq. (2) is seen [23]. In Eq. (2),  $t$  is the term and  $j$  is the document. TF-IDF score  $i$  in document  $j$  is calculated as in Eq. (3).

$$TF(i, j) = \frac{\text{Term } i \text{ frequency in document } j}{\text{Total words in document } j} \quad (1)$$

$$IDF(i) = \log \left( \frac{\text{Total documents}}{\text{documents with term } i} \right) \quad (2)$$

$$j = TF(i, j) * IDF(i) \quad (3)$$

### 3.2.2 BOW

BOW is the document representation model widely used in text processing. In the BOW model, the word order of text documents is not preserved, but only the word counts are taken into account [24]. The BOW model, which shows the frequency of words in documents, was used by the classifier to create a learning model with a set of features.

### 3.3 Word Embedding Based Text Representation

Word2Vec, GloVe, BERT word embedding methods are explained in this section.

#### 3.3.1 W2V

W2V method is a word embedding method that learns the vector representations of words using a training set with ANN [25] - [27]. It has two models, the Continuous Bag of Words (CBOW) and Skip-gram, which matches close vectors with similar meaningful words in the vector space. While the CBOW model predicts a word in a certain context, the Skip-gram model predicts the context of a particular word.

W2V extracts vector representations of words from datasets. The skip-gram and the CBOW model are shown in Figure 2.

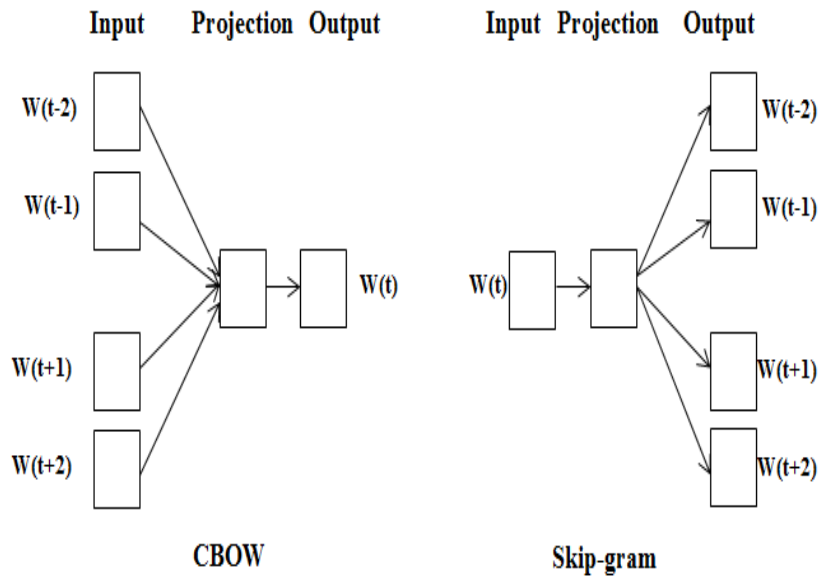


Figure 2 W2V models

#### 3.3.2 GloVe

The gloVe is an advanced method from W2V that makes embedding words in documents more efficient. The gloVe is regression-based and the objective function is given in Equation. (4):

$$J = \sum_{i,j=1}^v f(X_{ij})(w_i^T V_j + b_i + b_j - \log X_{ij})^2 \tag{4}$$

where  $v$  denotes the vocabulary size,  $w \in R^d$  represents the word vectors,  $V$  represents context word vectors,  $X_{ij}$  is the number of times the word pair  $(i, j)$  occurs together in the corpus.  $f(X_{ij})$  denotes a weighting function and  $b_i, b_j$  are bias parameters [27].

### 3.3.3 BERT

BERT is a word embedding model that stands for bi-directional encoder representations. The BERT model is designed to condition the word in right and left contexts by pre-training the dataset in each layer and in both directions. Figure 3 shows the architecture of the BERT model.

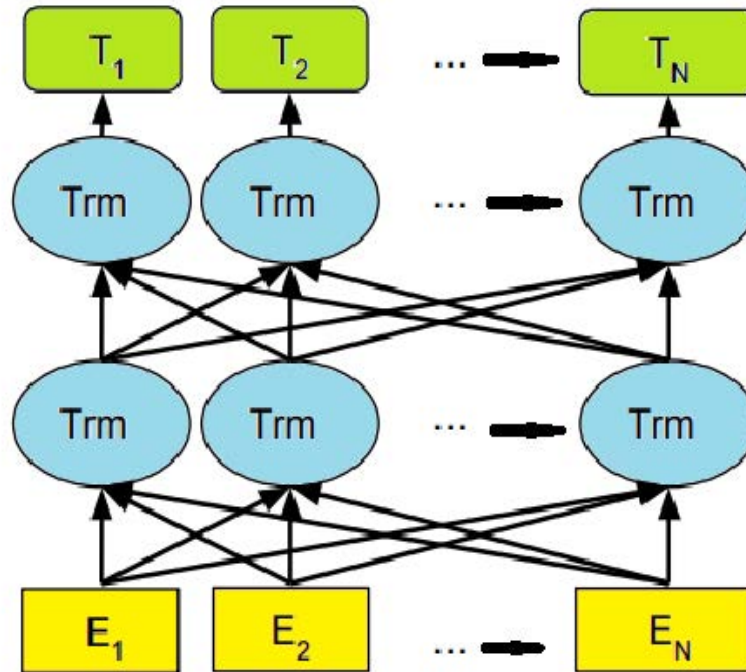


Figure 3 BERT model

## 3.4 Machine Learning

It has been introduced in the 1980s and has become popular in data mining. These are self-training systems that make better decisions by making simulations with the data and parameters given for learning purposes.

### 3.4.1 Naïve Bayes Classifier

The NB algorithm, named after Thomas Bayes, is based on Bayes' theorem.

Let  $X = \{ x_1, x_2, x_3, \dots, x_n \}$  is the sample set, and  $C_1, C_2, C_3, \dots, C_m$  is the class set. The sample to be classified:

$$P(X|C_i) = \frac{P(X|C_i)P(C_i)}{P(X)} \quad (5)$$

As seen in Eq. (5), the probability value is calculated according to the data of the class with the highest probability [28].

### 3.4.2 Support Vector Machine

SVM is a ML method that sets a boundary between any point in the training data and another furthest point [28]. One feature of SVM is the inherent risk minimization in statistical learning theory [29].

### 3.5 Deep Learning

Intelligent systems have been developed in various fields with ML algorithms in recent years. Various classifier algorithms are successfully used for tagging in data classification, as one of the ML methods. With the increase in the amount of data, however, the performance of the models decreases. Hence, different algorithms and methods have been developed to overcome hardware problems. One of these methods is the DL algorithms that emerged in line with the neural networks introduced in the 1940s [30].

Although there were some limited achievements before the early 2000s due to the limitations in the computing power, it was not practical to train neural networks as today [31].

DL is a structure consisting of an increasing number of ANN layers that function like neurons in the human brain. Recent years witnessed its widespread use in sentiment analysis. Of the DL algorithms, LSTM, CNN, and RNN algorithms were used in this study.

#### 3.5.1 Recurrent Neural Network

Thanks to recent advances in technology, RNN can be used easily. RNN is a neural network model developed to learn existing patterns by taking advantage of sequential information [28]-[29]. In RNN, each output is determined by the continuous processing of the same task on each instance of the array. The output is determined according to previous calculations [32].

In RNN, the resulting output is based not only on the current input, but also on the other inputs. In addition to the input data at time  $t$ , the results of the hidden layer at the time  $t-1$  are used as the input of the hidden layer at the time  $t$ . The decision regarding the input at the time  $t-1$  also affects the decision to be made at the time  $t$ . In other words, the inputs of these networks generate output by combining current and previous information. Eq. (6) shows the result of the hidden layer  $s_t$  at the time  $t$ . Eq. (6), shows the input  $x_t$  at the time  $t$ , the hidden state  $S_t$ , the activation function of the  $f$  value, and the weight at  $U$  and  $W$  [33]:

$$s_t = f(Ux_t + Ws_{t-1}) \quad (6)$$

#### 3.5.2 Long Short-Term Memory

LSTM is an RNN architecture. Unlike standard feed-forward neural networks, LSTM has feedback links. It consists of a cell, and three types of gates: an input gate, an output gate, and a forget gate. Based on the open-closed state of the gates, the cells determine the information to be preserved and the time to access the units [34].

Through these gates, the cell decides what to store, when to read, write or delete. These gates have a network structure and activation function. Just like neurons, they pass or stop the incoming information according to their weights. These weights are calculated during the learning phase of the recurrent network.

#### 3.5.3 Convolutional Neural Networks

Although CNN is one of the deep learning algorithms used in artificial intelligence fields such as Natural Language processing, it is also often used in the field of Image processing. It consists of three main layers [35]:

The first layer is the Convolutional Layer where a filter is used to transform the input matrix. In this layer, each filter maps the input matrix to a gap, and the output size depends on the size of the filter.

The second layer is the pooling layer. It is usually placed after the convolutional layer and used to reduce the size of the mapped elements.

The third layer is the fully connected layer. It is placed after the last pooling layer. The activation functionality in each layer is determined by the network for classification.

### 3.6 Evaluation Metrics

The confusion matrix used in the model evaluation gives the number of correctly and incorrectly classified samples according to binary classification. ( $T_P$ ) represents false positive ( $F_N$ ), true positive ( $F_P$ ), false negative, and ( $T_N$ ) true negative numbers (Table 4) [36].

Table 4 Confusion Matrix

		Predicted	
		Positive	Negative
Actual	Positive	$T_P$	$F_N$
	Negative	$F_P$	$T_N$

Accuracy, Sens, Pre, F used in the study are given between Eq. (7) and Eq. (10).

$$Accuracy = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \quad (7)$$

$$Sensitivity = \frac{T_P}{T_P + F_N} \quad (8)$$

$$Precision = \frac{T_P}{T_P + F_P} \quad (9)$$

$$F - measure = \frac{2 * Precision * Sensitivity}{Precision + Sensitivity} \quad (10)$$

In order to partition the dataset as training and testing, 10-fold cross-validation method is used in the experiments. The original dataset is randomly partitioned into 10 equal sized partitions. Each time, one of the partitions is used for testing and the others are used for training. The process is repeated ten times and the average results across all steps are calculated.

### 4. Proposed Framework

The image of the proposed model for sentiment analysis on the publicly available and privately collected datasets is shown in Figure 4. Text processing such as the stop-word elimination was performed in all datasets. On the collected Twitter data, hashtags and URLs were removed.

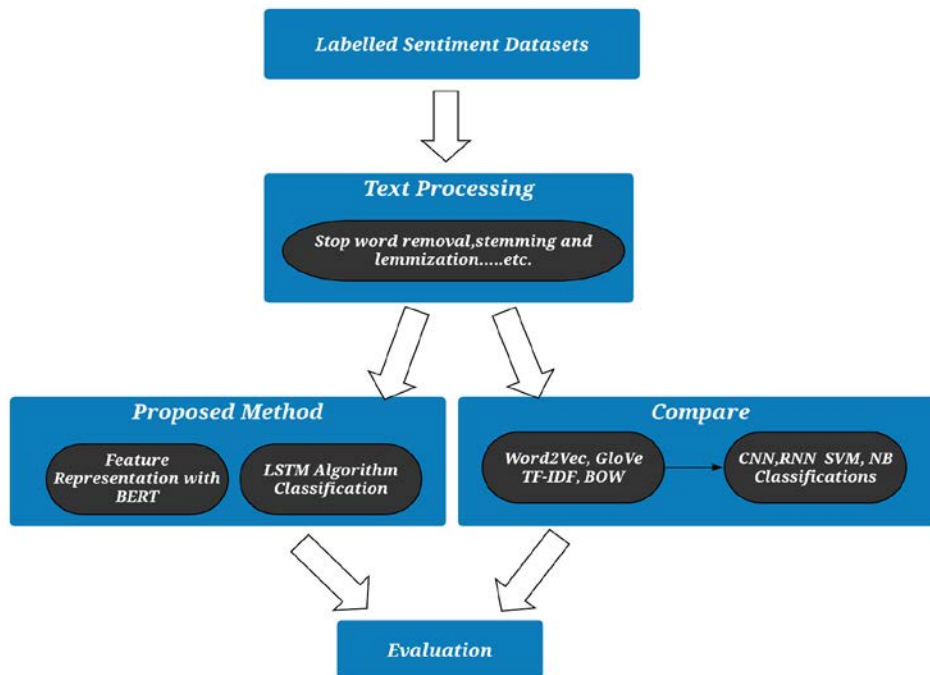


Figure 4 The proposed framework

As shown in Figure 4, the model created by the combination of the BERT word embedding representation method and the LSTM DL algorithm was compared to the models created by other word representation and learning algorithms.

The results of the proposed method are shown as bold and red in all tables. Besides, the text representation and word embedding method that gives the best results in each classification algorithm categories are shown as bold.

As shown in Table 5-7, SVM, one of the ML algorithms, gave a better performance in all performance criteria compared to NB, followed by the word representation and embedding methods. In DL algorithms, word embedding, and representation methods, the LSTM classifier model used after the

Table 5 Performance of Classification Algorithms on IMDB Review Dataset with Word Embedding and Text Representations

Classifier Algorithms	Text Representations	Accuracy	Pre	Sens	F	ROC
<b>SVM</b>	BOW	81%	81%	83%	82%	88%
	TF-IDF	83%	84%	84%	84%	90%
	W2V	84%	84%	86%	85%	92%
	GloVe	89%	88%	90%	88%	91%
	<b>Bert</b>	<b>90%</b>	<b>90%</b>	<b>91%</b>	<b>90%</b>	<b>91%</b>
<b>NB</b>	BOW	81%	82%	81%	81%	89%
	TF-IDF	82%	82%	83%	82%	90%
	W2V	83%	84%	85%	84%	92%
	GloVe	86%	86%	86%	86%	84%
	<b>Bert</b>	<b>87%</b>	<b>86%</b>	<b>87%</b>	<b>88%</b>	<b>90%</b>
<b>CNN</b>	BOW	84%	82%	81%	81%	89%
	TF-IDF	85%	82%	83%	82%	90%
	W2V	87%	84%	85%	84%	92%
	GloVe	88%	86%	86%	86%	84%
	<b>Bert</b>	<b>93%</b>	<b>86%</b>	<b>87%</b>	<b>88%</b>	<b>90%</b>
<b>RNN</b>	BOW	85%	82%	81%	81%	89%
	TF-IDF	85%	82%	83%	82%	90%
	W2V	88%	84%	85%	84%	92%
	GloVe	90%	86%	86%	86%	84%
	<b>Bert</b>	<b>92%</b>	<b>90%</b>	<b>88%</b>	<b>88%</b>	<b>90%</b>
<b>LSTM</b>	BOW	86%	82%	81%	81%	89%
	TF-IDF	86%	82%	83%	82%	90%
	W2V	89%	84%	85%	84%	92%
	GloVe	91%	86%	86%	86%	84%
	<b>Bert</b>	<b>94%</b>	<b>94%</b>	<b>93%</b>	<b>89%</b>	<b>94%</b>

BERT word embedding method was found to perform better than other DL methods. Similarly, performances of the word representation methods with ML and DL algorithms were obtained for the classifier models created with BERT, GloVe, Word2Vec, TF-IDF, BOW, respectively. The results also confirmed that the GloVe is the improved version of W2V.

In addition, the results showed that the models with BERT word embedding method, used both with ML and DL, have better performance than the others. This reveals that the BERT is more successful than other text representation methods.

Table 6 Performance of Classification Algorithms on Yelp Review Dataset with Word Embedding and Text Representations

Classifier Algorithms	Text Representations	Accuracy	Pre	Sens	F	ROC
<b>SVM</b>	<b>BOW</b>	81%	80%	81%	81%	81%
	<b>TF-IDF</b>	81%	82%	81%	81%	82%
	<b>W2V</b>	83%	84%	85%	84%	83%
	<b>GloVe</b>	84%	84%	86%	85%	86%
	<b>Bert</b>	<b>86%</b>	<b>87%</b>	<b>83%</b>	<b>86%</b>	<b>90%</b>
<b>NB</b>	<b>BOW</b>	74%	73%	73%	74%	78%
	<b>TF-IDF</b>	76%	77%	77%	77%	85%
	<b>W2V</b>	78%	78%	78%	81%	86%
	<b>GloVe</b>	79%	79%	78%	79%	88%
	<b>Bert</b>	<b>81%</b>	<b>83%</b>	<b>81%</b>	<b>80%</b>	<b>91%</b>
<b>CNN</b>	<b>BOW</b>	81%	82%	81%	81%	89%
	<b>TF-IDF</b>	82%	82%	83%	82%	90%
	<b>W2V</b>	84%	84%	85%	84%	92%
	<b>GloVe</b>	86%	86%	86%	86%	94%
	<b>Bert</b>	<b>87%</b>	<b>86%</b>	<b>87%</b>	<b>88%</b>	<b>95%</b>
<b>RNN</b>	<b>BOW</b>	82%	82%	81%	82%	86%
	<b>TF-IDF</b>	83%	83%	84%	83%	88%
	<b>W2V</b>	85%	84%	85%	85%	91%
	<b>GloVe</b>	87%	86%	86%	86%	92%
	<b>Bert</b>	<b>88%</b>	<b>86%</b>	<b>87%</b>	<b>88%</b>	<b>94%</b>
<b>LSTM</b>	<b>BOW</b>	83%	77%	75%	76%	82%
	<b>TF-IDF</b>	84%	78%	76%	75%	83%
	<b>W2V</b>	84%	82%	81%	81%	85%
	<b>GloVe</b>	85%	82%	83%	82%	87%
	<b>Bert</b>	<b>89%</b>	<b>84%</b>	<b>85%</b>	<b>84%</b>	<b>91%</b>



Table 7 Performance of Classification Algorithms on Twitter Dataset with Word Embedding and Text Representations

Classifier Algorithms	Text Representations	Accuracy	Pre	Sens	F	ROC
<b>SVM</b>	<b>BOW</b>	80%	78%	77%	80%	89%
	<b>TF-IDF</b>	83%	83%	82%	81%	86%
	<b>W2V</b>	89%	88%	86%	87%	90%
	<b>GloVe</b>	89%	88%	86%	88%	90%
	<b>Bert</b>	<b>89%</b>	<b>87%</b>	<b>89%</b>	<b>87%</b>	<b>93%</b>
<b>NB</b>	<b>BOW</b>	70%	72%	73%	74%	75%
	<b>TF-IDF</b>	72%	73%	73%	76%	78%
	<b>W2V</b>	72%	76%	75%	76%	79%
	<b>GloVe</b>	75%	77%	75%	76%	80%
	<b>Bert</b>	<b>79%</b>	<b>78%</b>	<b>76%</b>	<b>77%</b>	<b>82%</b>
<b>CNN</b>	<b>BOW</b>	84%	82%	81%	81%	89%
	<b>TF-IDF</b>	85%	82%	83%	82%	90%
	<b>W2V</b>	87%	84%	85%	84%	92%
	<b>GloVe</b>	88%	86%	86%	86%	84%
	<b>Bert</b>	<b>93%</b>	<b>86%</b>	<b>87%</b>	<b>86%</b>	<b>90%</b>
<b>RNN</b>	<b>BOW</b>	85%	82%	81%	81%	89%
	<b>TF-IDF</b>	85%	82%	83%	82%	90%
	<b>W2V</b>	88%	84%	85%	84%	92%
	<b>GloVe</b>	90%	86%	86%	86%	84%
	<b>Bert</b>	<b>94%</b>	<b>86%</b>	<b>87%</b>	<b>86%</b>	<b>90%</b>
<b>LSTM</b>	<b>BOW</b>	87%	86%	87%	84%	85%
	<b>TF-IDF</b>	89%	89%	87%	86%	88%
	<b>W2V</b>	91%	94%	91%	94%	95%
	<b>GloVe</b>	96%	96%	96%	96%	96%
	<b>Bert</b>	<b>98%</b>	<b>98%</b>	<b>99%</b>	<b>99%</b>	<b>98%</b>

## 5. Conclusion And Discussion

This study was conducted on the public and privately collected data to compare the word representation and embedding methods for sentiment analysis tasks with ML and DL algorithms. The Accuracy, Pre, Sens, F, and ROC were used as performance metrics.

In the study, learning algorithms CNN, LSTM, RNN from DL; SVM, NB from ML were used for classifying the sentiments. Word embedding methods BERT, GloVe, Word2Vec, and traditional word representation methods TF-IDF, BOW were also used.

According to the results of the experiments, the model created with Bert and LSTM has shown the best performance among the model combinations created on all datasets. Besides, the models that incorporated the BERT word embedding method have the best performance, among the other text representations and word embedding method.

In future studies, methods such as EIMo that yield successful results in sentiment analysis studies and the performance of the transformers such as RoBERTa and DistilBERT in neural networks such as LSTM and RNN are planned.



## References

- [1] E. Park, J. Kang, D. Choi, and J. Han, "Understanding Customers' Hotel Revisiting Behaviour: a sentiment analysis of Online Feedback Reviews," *Current Issues in Tourism*, vol. 23, pp. 605-611, 2020, doi: 10.1080/13683500.2018.1549025.
- [2] B. Pang and L. Lee, "Opinion mining and sentiment analysis", *Foundations Trends Information Retrieval*, vol. 2, no. 2, 2008, pp. 1-135.
- [3] O. Kaynar, H. Arslan, Y. Görmez and F. Demirkoparan, "Makine Öğrenmesi Yöntemleri ile Duygu Analizi," *International Artificial Intelligence and Data Processing Symposium (IDAP)*, pp. 1-5, Malatya, 2017.
- [4] A. Al Hamoud, A. Alwehaibi, K. Roy, and M. Bikdash, "Classifying Political Tweets Using Naïve Bayes and Support Vector Machines," *In International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pp. 736-744, 2018, doi: 10.1007/978-3-319-92058-0\_71.
- [5] S. Symeonidis, D. Effrosynidis, and A. Arampatzis, "A Comparative Evaluation of Pre - Processing Techniques and Their Interactions for Twitter Sentiment Analysis," *Expert System Applications*, vol. 110, pp. 298-310, 2018, doi: 10.1016/j.eswa.2018.06.022.
- [6] M. A. Paredes-Valverde, R. Colomo-Palacios, M. P. Salas-Zárate, and R. Valencia-García, "Sentiment Analysis in Spanish for Improvement of Products and Services: A Deep Learning Approach," *Scientific Programming*, vol. 2017, 2017, doi: 10.1155/2017/1329281.
- [7] J. Zheng and L. Zheng, "A Hybrid Bidirectional Recurrent Convolutional Neural Network Attention-Based Model for Text Classification," *IEEE Access*, vol. 7, 2019, pp. 106673-106685, doi: 10.1109/ACCESS.2019.2932619.
- [8] S. Liu, "Sentiment Analysis of Yelp Reviews: A Comparison of Techniques and Models", *arXiv preprint*, arXiv:2004.13851, 2020.
- [9] M. R. Huq, A. Ali, and A. Rahman, "Sentiment Analysis on Twitter Data Using KNN and SVM," *International Journal of Advanced Computer Science and Applications*, vol. 8, pp. 19-25, 2017, doi: 10.14569/IJACSA.2017.080603.
- [10] A. Amolik, N. Jivane, M. Bhandari, and M. Venkatesan "Twitter Sentiment Analysis of Movie Reviews Using Machine Learning Techniques," *International Journal of Engineering and Technology*, vol. 7, no. 6, pp. 1-7, 2016.
- [11] S. Liao J. Wang R. Yu, K. Sato, and Z., Cheng, "CNN for Situations Understanding Based on Sentiment Analysis of Twitter Data," *Procedia Computer Science*, vol. 111, 2017, pp. 376–381, 2017, doi: 10.1016/j.procs.2017.06.037
- [12] Li C, Guo X, Mei Q (2017b) Deep Memory Networks for Attitude Identification. *In: Proceedings of the tenth ACM International Conference on Web Search and Data Mining, WSDM*, Cambridge, United Kingdom, pp 671–680, 2017.
- [13] B. Li, Z. Cheng, Z. Xu, W. Ye, T. Lukasiwicz and S. Zhang, "Long Text Analysis Using Sliced Recurrent Neural Networks with Breaking Point Information Enrichment," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, pp. 7550-7554, 2019,doi: 10.1109/ICASSP.2019.8683812.
- [14] W. Zhao et al., "Weakly-Supervised Deep Embedding for Product Review Sentiment Analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 1, 1 Jan. pp. 185-197, 2018, doi: 10.1109/TKDE.2017.2756658.
- [15] M. Al-Smadi, O. Qawasmeh, M. Al-Ayyoub, Y. Jararweh, and B. Gupta, "Deep Recurrent

- Neural Network vs. Support Vector Machine for Aspect-Based Sentiment Analysis of Arabic Hotels' Reviews," *Journal of Computational Science*, 2017, doi: 10.1016/j.jocs.2017.11.006.
- [16] D. Tang, F. Wei, B. Qin, N. Yang, T. Liu, and M. Zhou, "Sentiment Embeddings with Applications to Sentiment Analysis," *In IEEE Transactions on Knowledge and Data Engineering*: vol. 28, pp. 496–509, 2016, doi: 10.1109/TKDE.2015.2489653.
- [17] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent Attention Network on Memory for Aspect Sentiment Analysis," *Empirical Methods in Natural Language Processing*, pp. 452–461, 2017.
- [18] F. Tian et al., "Recognizing and Regulating Elearners' Emotions Based on interactive Chinese Texts in E-Learning Systems," *Knowledge Based System*, vol. 55, 148–164, 2014, doi: 10.1016/j.knosys.2013.10.019
- [19] H. Ghulam, F. Zeng, W. Li, and Y. Xiao, "Deep learning-based Sentiment Analysis for Roman Urdu Text," *Procedia Computer Science*, vol. 147, pp.131-135, 2019, doi: 10.1016/j.procs.2019.01.202
- [20] J. Singh, R. Singh, and P. Singh, "Morphological evaluation and sentiment analysis of Punjabi text using deep learning classification," *Journal King Saud University-Computer and Information Science*, 2018, doi: 10.1016/j.jksuci.2018.04.003.
- [21] Yelp Polarity Dataset, "TensorFlow Datasets Catalog homepage," 2015. [online]. Available: [https://www.tensorflow.org/datasets/catalog/yelp\\_polarity\\_reviews](https://www.tensorflow.org/datasets/catalog/yelp_polarity_reviews)
- [22] A. L. Maas, R.E. Daly, P.T. Pham, D. Huang, A.Y. Ng and C. Potts, "Learning Word Vectors for Sentiment Analysis", *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, pp. 142-150, 2011.
- [23] R. Sjögren, K. Stridh, T. Skotare, J. and J. Trygg, "Multivariate Patent Analysis-Using Chemometrics to Analyze Collections of Chemical and Pharmaceutical Patents," *Journal of Chemometrics*, vol. 34, pp. e3041, 2020, doi: 10.1002/cem.3041
- [24] A. Onan "Mining opinions from instructor evaluation reviews: A Deep Learning Approach, " *Computer Application in Engineering Education*, vol. 28, pp. 117–138, 2020, doi: 10.1002/cae.22179.
- [25] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space", *arXiv preprint*, arXiv:1301.3781, 2013.
- [26] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and Their Compositionality," *Neural Information Processing Systems Conference*, Lake Tahoe, pp. 3111–3119, 2013.
- [27] R. Ni and H. Cao, "Sentiment Analysis based on GloVe and LSTM-GRU," *39th Chinese Control Conference (CCC)*, Shenyang, China, pp. 7492-7497, 2020, doi: 10.23919/CCC50068.2020.9188578.
- [28] M. M. Saritas, A. Yasar, "Performance Analysis of ANN and Naive Bayes Classification Algorithm for Data Classification," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 7, pp. 88-91, 2019, doi: 10.18201/ijisae.2019252786.
- [29] S. Qing, H. Wenjie and X. Wenfang, "Robust Support Vector Machine with Bullet Hole Image Classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 32, no. 4, pp. 440-448, 2002, doi: 10.1109/TSMCC.2002.807277.
- [30] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [31] S. Karita et al., "A Comparative Study on Transformer vs RNN in Speech Applications," *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, SG, Singapore, , pp. 449-456, 2019, doi: 10.1109/ASRU46091.2019.9003750.
- [32] L. M. Rojas-Barahona, "Deep Learning for Sentiment Analysis," *Language Linguistic Compass*, vol. 10, no. 12, 2016, doi: 10.1111/lnc3.12228
- [33] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015, doi: 10.1038/nature14539.
- [34] Ş. Kayıkçı, "A convolutional neural network model implementation for speech recognition," *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, vol. 7, no. 3, pp. 1892-1898, 2019, doi: 10.29130/dubited.567828.
- [35] M. S. Başarslan and F. Kayaalp, "Performance Analysis Of Fuzzy Rough Set-Based And

- Correlation-Based Attribute Selection Methods On Detection Of Chronic Kidney Disease With Various Classifiers," *2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT)*, Istanbul, Turkey, 2019, pp. 1-5. doi: 10.1109/EBBT.2019.8741688.
- [36] K. Polat, and S. Güneş, "Breast cancer diagnosis using least square support vector machine," *Digital signal processing*, vol. 17, no. 4, pp. 694-701, 2007, doi: 10.1016/j.dsp.2006.10.008.

# Determination of The Critical Success Factors in Disaster Management Through The Text Mining Assisted Ahp Approach

 Halil İbrahim Cebeci<sup>1</sup>,  Yasemin Korkut<sup>2</sup>

<sup>1</sup>Sakarya University, hcebeci@sakarya.edu.tr,

<sup>2</sup> Corresponding Author, Yalova University, yasemin.korkut@yalova.edu.tr, +90 (226) 815 65 72,

Received 18 January 2021; Accepted 5 February 2021; Published online 16 February 2021

## Abstract

In the academic field and as well as the application field, substantial attention has been drawn to coping with disasters. Since natural dangers causing a large proportion of disasters cannot be avoided, attempts to combat disasters have centered on preventing hazards from evolving into disasters through measures and restructuring works taken before, during, and after the disaster. There are many players involved in the disaster management process and many factors are influential in the effectiveness of this process. Among these factors, deciding the critical ones offers significant advantages, particularly in terms of practical studies. Concentrating on a single stakeholder in deciding the factors crucial to the success of this management structure, which has many stakeholders, can cause to ignoring the significant viewpoints of other stakeholder groups. Accordingly, for the evaluation of several success factors achieved as a result of a thorough and systematic literature review, the purpose of our study is to develop a common critical success factor model that will represent both the viewpoints of operational experts and academic experts, who constitute the stakeholders of this domain. Analytical Hierarchy Process (AHP) is utilized to determine the opinions of field experts while the text mining method was used to determine the perspectives of academics. In the study, therefore, a new AHP model assisted by text mining is introduced. Socio-cultural factors were brought to light by the analysis results of the suggested model. It has been determined by the results of the study that these two perspectives are overlapped largely in the organizational field and relatively in socio-cultural, environmental, and legal fields.

**Keywords:** Disaster management, Critical success factors, AHP, Text mining

## 1. Introduction

Natural disasters that have inflicted numerous losses of life and material destruction that worth millions of liras throughout history, cannot be completely avoided or prevented even under today's technological means. Disaster is characterized as the results of natural, technological, or man-made events that lead individuals to endure physical, economic, social, and environmental losses, impact societies by preventing or disrupting normal lives and human activities, and cannot be resolved by the affected community members by using their own resources and methods [1].

The efficient implementation of proactive and reactive actions at different disaster phases will mitigate the harm due to natural disasters and avoid most disasters perceived to be man-made or technological from occurring. In this context, disaster management is a complete process of endeavor that must be carried out by the community to avoid and mitigate disasters, to respond promptly, efficiently, and effectively to the incidents that form the disaster, and to build a safer and more efficient living atmosphere for the people affected by the disaster [2]. In this direction, in terms of taking measures, with the effective management of disasters, it is possible to reduce human losses, environmental, social, and economic damages. The chaos generated by unregulated activities after the disaster can be avoided by carrying out various tasks in a prepared and organized way, by facilitating the return to normal life for people in the disaster zone or by providing a living environment for these individuals in better conditions than before the disaster.

A comprehensive approach that involves multiple stakeholders around the country could be more useful for effective disaster management, instead of concentrating on a single region or specific activities. In this context, it is important for effective disaster management to prioritize the variables and to disclose the crucial factors to enable the properly working of the integrated model, which involves several

variables. In the extensive literature review performed for the domain, it has been witnessed that studies focusing on critical success factors in disaster management have paid attention to a certain stage or area of disaster management [3-5] or the number of factors examined was rather limited in the studies that paid attention to the whole stages [6]. From this perspective, for the success of a critical area such as disaster management, a more detailed perspective is required.

This topic has been extensively investigated in the academic field since the consequences of disaster management throughout the years are very significant. The substantial accumulation of knowledge that has arisen in this domain should be considered. However, given the significance of the experiences obtained from the practices, a disaster management framework developed only as proposed by academic studies is not sufficient. In van Niekerk's [7] study, which explored whether academic discourse or practical reality must be at the center of disaster management, he claimed that a disaster management system that relies exclusively on the academic context, government, and international and regional organizations spend a substantial amount on activities that do not offer any value. He also claimed that a study would be insufficient to enhance disaster risk mitigation and management without addressing the roots of disaster studies and research in both social and natural sciences. We may conclude that disaster management practitioners are engaged in a complex and continually changing activity in a disaster management system which only depends on practices. Therefore, the result stating that disaster management activities require practical and scientific input is examined. Thus, it is aimed to create a new model in this study whose academic knowledge includes the opinions of the experts who have been working in this field in Turkey and have significant information. In this context, this research attempts to incorporate the knowledge obtained from a systematic and thorough literature review by introducing a model that integrates the perspectives of the field's operational experts, taking into account that an integrated approach that combines academic studies and the viewpoints of practitioners can be beneficial.

9 areas that directly influence the success of disaster management were reported as a result of the literature review undertaken to realize the purpose of the research. Factors indicated at the end of the screening are divided into sub-factor groups under the main factor groups for the scope of these 9 areas. Therefore, the need for managing a hierarchical structure composed of the main factor group, a sub-factor group, and the other factors necessitated a new model with a different viewpoint. AHP method has been used for main factor groups and sub-factor groups forming the first two levels of hierarchy and text mining method which will reflect the perspective of academics was applied for the factors that form the lowest level of the hierarchy. Finally, a Critical Success Score (CSS) was calculated for each factor by combining the results of these two analysis methods. The first 20 factors with the highest CSS were evaluated as critical success factors in disaster management.

This research is important in terms of benefiting from the views of both academics and field experts in the field of disaster management and also being a guideline for the translation of academic knowledge into practice by identifying. The study's extensive literature and the model proposal can be anticipated to lead further academic studies.

In the following part of the study, fundamental studies that have determined the main factor groups will be discussed. At the same time, success factors which are derived after the comprehensive literature review performed on main factor groups will be summarized in this part. Thereafter, the results obtained will be evaluated in light of the 3-stage integration model suggested in the study and findings of analyses, and in this context, the limitations of this study will be stated and suggestions for further studies will be presented.

## **2. Motivation and Previous Studies**

There are three main phases of activities in effective disaster management planning, namely pre-disaster, disaster response, and post-disaster [8]. These three main phases can be evaluated in five stages: planning and preparation, mitigation, response, recovery, and evaluation [8,9]. However, since each of these processes requires activities such as planning and risk mitigation measures, there is no requirement to pursue one another [10]. At the same time, though it is a matter of obscurity when disasters will occur,

these studies will not end. The stages of disaster management should therefore not be in a linear form, but in a cycle, as seen in Figure 1 [8-10].

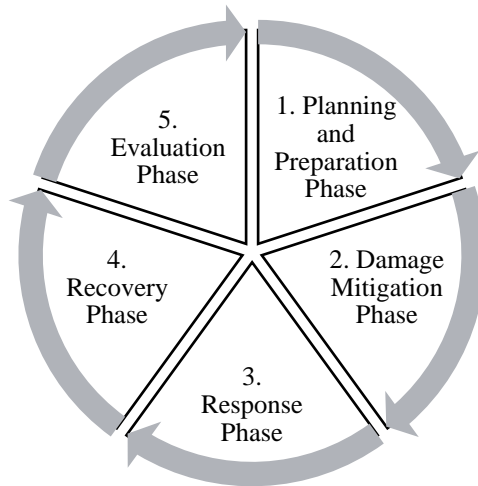


Figure 1 Disaster Management Cycle [8-10]

The effective factors are presented in achieving success at different phases of the cycle of disaster management. Since disaster management communicates with many areas, in terms of professional work in these areas, an area-based grouping is essential.

To improve the organizational effectiveness of management as a whole and to enable the entire management activity step by step, it is important to divide the disaster management process into meaningful elements and factors. It is necessary, however, to define success factors for disaster management to allow executives and decision-makers to concentrate on priority factors to enhance the process of disaster management [6]. Having considered these causes, several factors are key to the success of disaster management in various areas. While it is essential to consult with experts in various fields to evaluate these factors, identifying the factors in the literature called Critical Success Factors may guide to decide which topics regarding the combat against disasters should be used to prepare more dedicated studies. Therefore, in this study, previous research on this subject was examined to determine success factors in disaster management. The baseline studies were determined in this context, as shown in Table 1, and the main categories and factors determined by Ozceylan and Coskun [11] for a successful National Emergency Management Model were used to determine the study's route. After a literature review, three categories, environmental, legal, and operational, have been applied to these current categories. A change in the classification was made as socio-cultural and economic factors for the groups of cultural factors and socio-economic factors, based on other research shown in the literature review. With a comprehensive literature review, existing factors were expanded. The main factor groups were therefore gathered into 9 groups, as shown in Table 1.

Table 1 Literature Review Table of Main Factor Groups in Terms of Disaster Management Success

	Economic	Socio-Cultural	Technological	Environmental	Operational	Organizational	Political	Legal	Risk
<b>Ozceylan and Coskun [11]</b>	✓	✓	✓			✓	✓		✓
<b>Chou and Wu [12]</b>	✓	✓	✓						✓
<b>Pathirage et al. [13]</b>	✓	✓	✓	✓	✓	✓	✓	✓	
<b>Seneviratne et al. [14]</b>	✓	✓	✓	✓	✓	✓	✓	✓	
<b>Seneviratne et al. [15]</b>	✓	✓	✓	✓	✓	✓	✓	✓	
<b>Ahmed, Ahmad, and Zakaria [16]</b>			✓	✓		✓			

A literature review was performed for each of the 9 areas listed, as can be seen in Table 1, because of a systematic literature review. As a result of the analysis, 240 success factors were uncovered. Similar factors were incorporated as a result of interviews with field experts, academic experts, and academics who are specialists in factors, and certain ones were omitted from success factors because of their repetition. Thus, the number of factors has been decreased to 122. On the other hand, 122 factors were clustered as per the subjects to which they are relevant in nine areas and allocated to the corresponding 27 sub-factor groups. Sub-factor groups and factors are given in Table 2 with their corresponding academic references:

Table 2 Success Factors in Disaster Management

Main Factor Group	Sub Factor Group	Factor	Source
Economic	Pre-Disaster Preparation	Economic Planning	[6,11,13,14,15]
		Distribution of Disaster Prevention Resources	[12]
		Pre-Disaster Financial Instruments	[14,17]
		Sufficient Financial Support for Disaster Recovery Planning	[18]
	Post-Disaster Response	Restructuring Funds	[5,19]
		Investments for Mitigating Disaster Effect	[13]
	Macro-Economic Factors	Helpful Economic Environment	[5,20,21]
Economic Growth / Development		[11-13]	
Environmental	Environmental Disaster Prevention Activities	Use and Protection of Natural Barriers	[13,14,22]
		Using Man-Made Barriers	[13,14,23]
		Building and Urban Planning	[13,14]
		Land Use Planning	[13,22,24]
		Environmental Awareness and Education	[22]
		Addressing Environmental Issues	[22]
	Post-Disaster Environmental Responses	Environmental Management Systems	[22]
		Post-Disaster Waste Management	[13,23]
	Post-Disaster Life	Managing Chemical Hazard	[25]
		Disaster Waste Recycling Systems	[13,23]
		Criteria for environmental effects in Restructuring Projects	[22]
Socio-Cultural	Individual Factors (Qualifications and Skills)	Individual Attitudes and Characteristics	[16]
		Specialty Skills	[25,26]
		Civil and Occupational Responsibilities	[13]
		Interpersonal Trust and Justice	[11,16]
	Education	Prevention and Response Training	[6,11,13,27,28,29]
		Post-Disaster Response Drills	[6,11,28,29]
		Rescue and Healthcare Professional Training	[6,18,27]
		Educational Design (Education Quality, Training Content)	[29]
		Continuous Education	[29]
	Social Participation and Association	Society's Participation	[5,11,12,30]
		Participation of Media Channels	[24,25,31]
Participation of Military Units		[16,32,33]	



Table 2 Success Factors in Disaster Management (cont.)

		Private Sector Assistance	[12]
		Rehabilitation	[24]
		Participation of Civil Society Organizations	[16]
		Social Learning	[12]
		Disaster Culture	[5,11,12,13,14,19,24,25,32]
Technological	Disaster Management Support Systems	Emergency Aid Support (Information) System	[6,32,34]
		Projection and Early Warning Systems	[6,11,13,14,15,22,24,25,31,32]
		Communication Systems	[13,15,24,32]
		Geographic Information Systems	[14,31]
		Post-Disaster Response System	[34,35]
		Information Management System	[13,19,25,36,37]
		Efficient Material Supply System	[19,28]
		Equipment Management System for Disaster	[24]
	Effective Communication During and After Disaster	Communication Technologies	[16]
		Procedures for Effective Communication Mechanism	[32,37]
		Technical Support Units	[24,25]
	Information Management	Sharing Information	[11,16,24,31,37,38]
		On-Time (Real-Time) Information	[34,39]
		Disaster Records with Time-Dimension	[12]
		Information Quality	[31,40]
		Information Centers	[11,18,27,41]
		Technology Use	[13,32]
	Technological Infrastructure	Trusted Sources of Information	[12]
		Disaster Prevention Technology and Infrastructure	[11,12]
		Communication Network and Infrastructure	[11,24,31]
Information Update Mechanism		[42]	
Structural Measures		[15,43]	
		Logistics Technology	[6,28]
Operational	Pre-Disaster Planning Activities	Disaster Management Model	[42]
		Creating a Disaster and Emergency Plan	[13,18,19,25,27,32]
		Quality Control Activities	[19]
		Statistics of Previous Disasters and Analyzes	[2,44,45]
		Execution of Planning Documentation	[27]
		Continuous Assessment and Improvement of Disaster Management System	[6,19,32]
		Operational Consistency / Harmony	[25]

Table 2 Success Factors in Disaster Management (cont.)

	Disaster Response Systems and Activities	Access and Evacuation Channels for People Affected by Disaster	[5,24]
		Time Management in Crisis Time	[19]
		Taking Precautions of Preventive Health Measures	[24,25]
	Post-Disaster Recovery (Rescue) Operations	Assessment of Damage	[5,13]
		Prioritizing Activities of Improvement	[27]
		Search and Rescue Operations	[25]
		Assessing Disaster Effect	[24,25]
	Logistics Activities Before, During, and After Disaster	Restructuring Activities	[25]
		Logistics Planning and Management	[6,11,14,32,37]
		Resource Planning and Management	[12,24,30,32,35,37,46]
	Organizational	Organizational Structure	Security of Rescue Equipment
Organizational Culture			[11,16,29,35]
Transparency and Accountability			[5,19,35]
Organizational Design			[11]
Corporate Arrangement			[13,32,37]
Centralized Decision-Making Structure			[11]
Executive Support			[11,16,18,27]
Enterprise Integration			[14]
Unity of Purpose and Political Goals			[27,32,37]
Organizational Agility			[35]
Response Time		[6]	
Precise Job Description and Roles		[11,42]	
Inter-Organizational Collaboration and Participation		Inter-Organizational and External Communication	[11,13]
		Coordination and Collaboration	[5,11,12,19,24,25,28,31,32,35,36,37,42,47]
		Degree of Involvement in the Process of Decision Making	[11,18,27]
Corporate Disaster Management Plan		Developing the Master Plan	[13]
		Disaster Management System	[25]
		Disaster Management Strategy and Plan	[6,11,28,31,32,35,42]
		Planning the Rescue Needs	[6,28]
Individual Competencies	Taking Individual and Institutional Initiative	[30]	
	Leadership	[11,14]	
	Teamwork	[11,19]	
	Managers Staff and Team Member Competencies	[13,19,32,37]	
Political	Communication and Information Management	Quality of Government Sharing of Information	[36]
		Public Advice and Advisory Services	[25]

Table 2 Success Factors in Disaster Management (*cont.*)

		Network with NGOs and International Organizations	[12,34]
		Cooperation with Other Countries	[42]
	Operational Factors	Restructuring Support	[13,47]
		Including Disaster Management Contents in National Education Curriculum	[13]
		National Disaster Management Policy	[25]
		Health, Safety and Security Management	[19]
Legal	Restrictive Legal Regulations	Enactments and Laws	[11]
		Terms of References and Regulations	[6,11,28]
		Production Regulations	[13]
		Local Regulations	[11]
		Environmental Rules and Standards	[22]
	Factors Related to the Implementation of Laws	Consideration of Social Factors When Making Laws	[14]
		Continuous Legal Regulations Update	[14]
Risk	Pre-Disaster (Related to Disaster) Risks	Geographical Risks	[11,12]
		Political Risks	[11,19]
		Technological Risks	[11]
		Evacuation Risk	[48,49]
	Risk Factors Related to Disaster Management	Risk Evaluation	[12,13,14,18,22,25,27]
		Evaluation of Potential Vulnerability	[14,24,25]
		Experience Level Concerning Different Types of Disasters	[11]
		Infrastructure	[11]

While the literature review is outlined in Table 2, it also describes a hierarchical left-to-right model consisting of 9 main factor groups, 27 sub-factor groups, and 122 factors. For instance, since certain factors are formed as per the phases of disaster management, as a conclusion of the analysis performed for the main economic factor group, a grouping was therefore developed as Pre-Disaster Preparation and Post-Disaster Response, while the Supportive Economic Environment and Economic Growth / Development factors are grouped as Macroeconomic factors because they will provide large-scale precautions and response. Groupings were created for the other 8 main factor categories because of the shared features of the factors. A method for managing such an integrated structure was necessary due to the hierarchical structure of a table obtained in this way and a high number of factors. In this context, an integrated approach was applied to collect data for different layers of the hierarchy from various expert groups. In the next part, the model is illustrated.

### 3. Method

Critical Success Factors (CSF) are described as a specific number of areas that provide the company with a good competitive result if the effects are satisfactory for companies, and they are a few main areas that have to be properly implemented for the businesses to grow [50]. It is necessary for the short and long-term success of a project, organization, or initiative to acknowledge or assess the CSF in a management system [51]. To guarantee the success of a management process such as disaster

management, which involves multiple disciplines within its body, by specifying significant levels to different aspects, it is important to recognize the critical aspects and to conduct the necessary activities by concentrating on these factors. However, there are studies carried out to determine critical success factors through a limited number of factors [6] even they focus on certain areas of disaster management [3], certain stages [4, 5] or directly on disaster management. It can also be stated that it is of crucial importance to evaluate critical success factors in the domain of disaster management. In this study, the model in figure 2 was established to manage the hierarchical structure formed by many factors from the perspective of different expert groups.

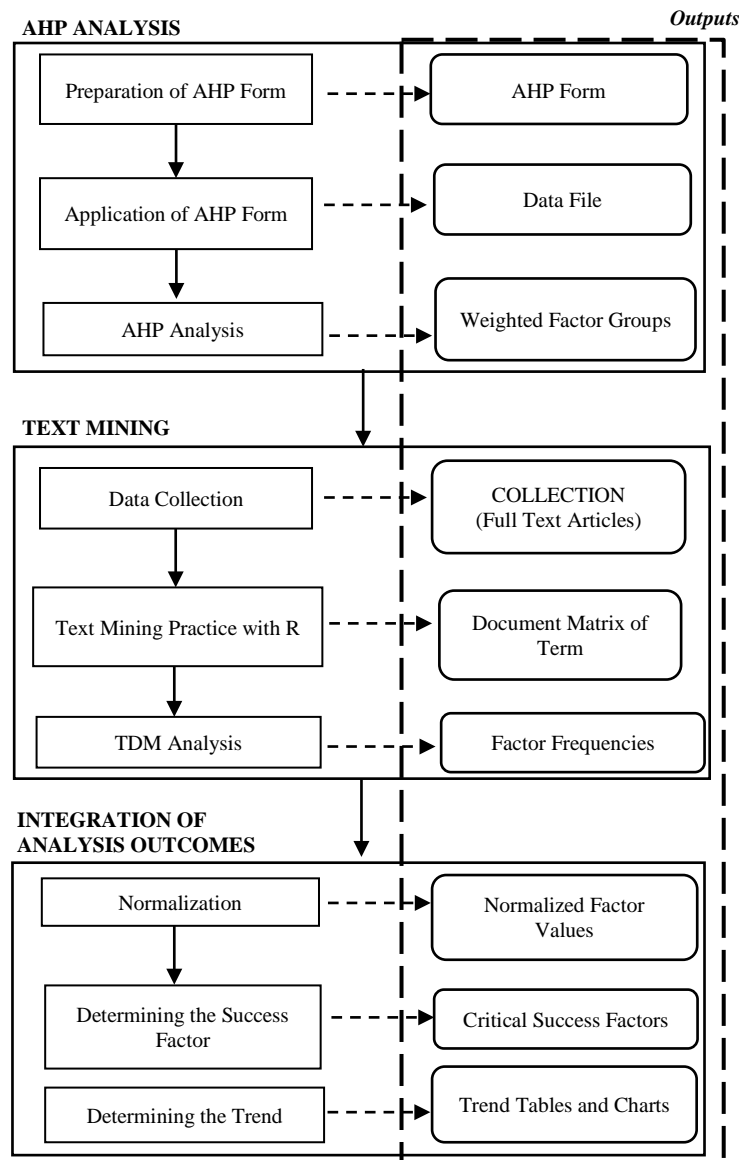


Figure 2 Critical Success Factor Model

It was determined to use the AHP method to collect the views of field experts on a successful disaster management system. The AHP approach helps decision-makers to analyze perceptions, senses, decisions, and experiences that impact their choices by comparing and evaluating them [52]. Thus, as it has a resolution structure that contains objective and subjective components, relative to many decision-making methods, it can be viewed as a more realistic solution approach [53]. Since these too many binary combinations that arise in the comparison of factors which are the lowest layer of the hierarchy will take too much time for the experts, they cannot make gathering data feasible. Therefore, this method could not be utilized for factors. As an alternative technique, the text mining method was included in the study in the evaluation of 122 factors, to capture and evaluate data relating to academic studies after

taking into account its capacity of including views of academics in the model. Thus, a method model was proposed by multiplying the normalized values of layers from top to bottom to reflect the evaluation of the last layer of academic experts and also the first two layers of field experts. Furthermore, by reflecting the data collected from academic studies up to the hierarchy, it was attempted to specify the areas and factors on which the studies concentrate on each layer of the hierarchy by years.

### 3.1 AHP Analysis

In this study, the AHP approach, which has a solution framework that includes objective and subjective aspects, was selected to evaluate the personal opinions of individuals who are accredited as disaster management experts. In this sense, by granting their significance, AHP allows for binary comparison of factors and prioritization. To compare the main factor groups with each other and the sub-factor groups among themselves within the main factor group to which they belong, the AHP form was applied to the experts in a structure designed in such a way that the main factor groups represent the first level of the hierarchy, the sub-factor groups form the second level of the hierarchy, and lastly, the factors constitute the third level of the hierarchy. The factors in the third level of the hierarchy were not included in the AHP application since too many binary comparison combinations they must make the application harder. Thus, the hierarchical structure of the AHP application consisting of the first two levels of the hierarchy can be seen in Figure 3:

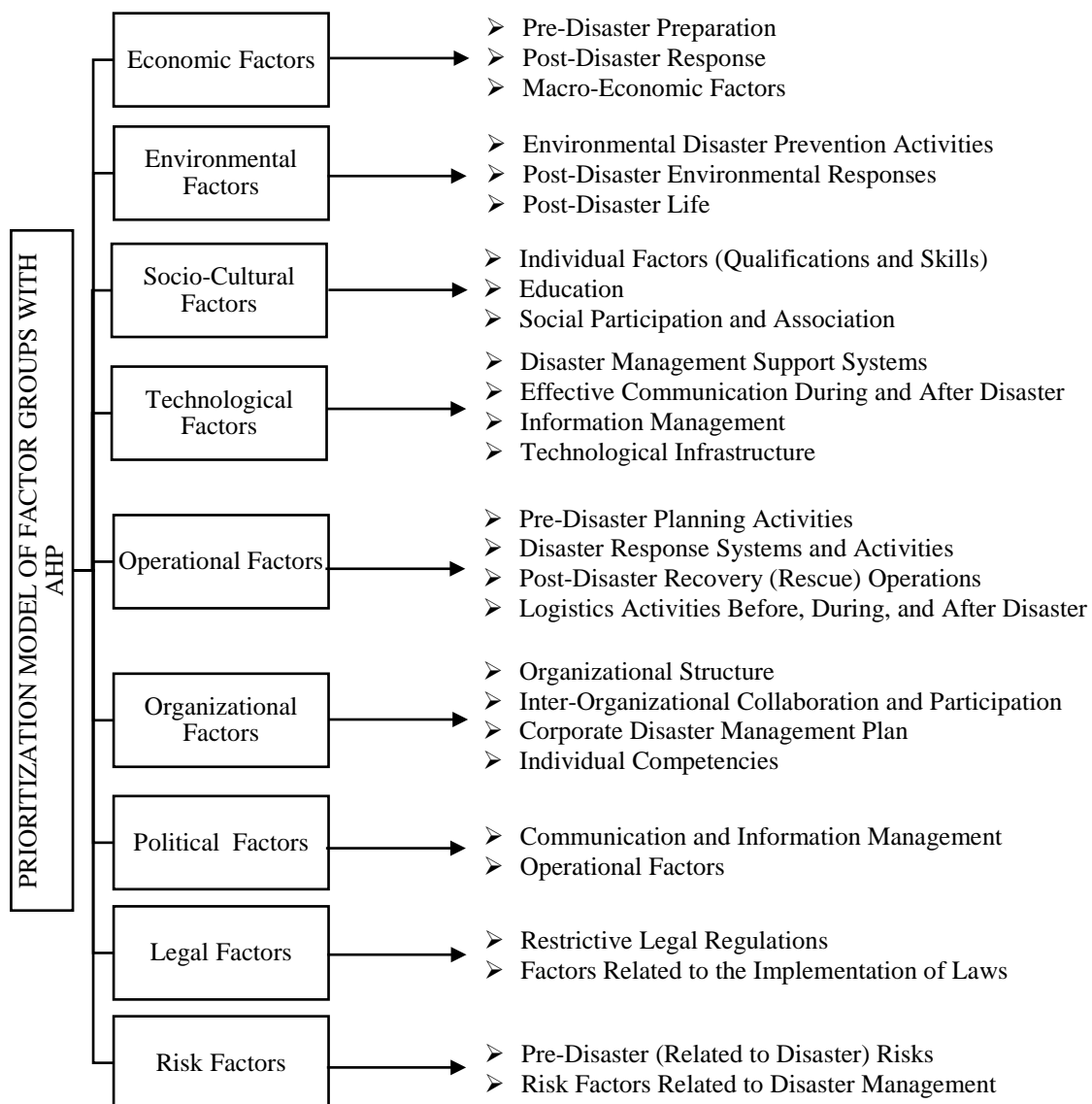


Figure 3 Prioritization Model of Factor Groups with AHP

To prioritize the groups seen in the hierarchy model from 1 to 9 in pairs, a 9-point AHP form was developed. A total of 20 individuals, including civil defense experts employed in different provinces and participating in the training program organized by the Ministry of Agriculture and Forestry of the Republic of Turkey in Antalya, disaster management experts from Ankara, Sakarya, and Yalova, and an academic with a Ph.D. in earthquake engineering, were administered to this form. With the data obtained, matrices were created, and the weights of the main factor groups derived from the solution of these matrices and the weights of the sub-factor groups belonging to these main factor groups were multiplied downwards. The real weights of the sub-factor groups were thus calculated and in other main factor groups, they were contrasted with the sub-factor groups.

### 3.2 Text Mining

The high number of disaster management studies and the fact that a total of 122 successful factors for disaster management will be analyzed from an academic perspective required the process to become semi-automated. For this purpose, in the second phase of the method model, the method of text mining, which can systematically analyze documents involving large numbers of unstructured data collected as a result of a comprehensive literature review to automate the process, was chosen. In this sense, the search for the "Scencedirect" academic publication database in 2017 using the keyword of "Disaster Management" was performed to access academic publications in the field of disaster management. 773 papers published between 2000 and 2016 were included in the study, as a result of the searching publications in the field.

To make them ready for the method, the articles were transformed into plain text (TXT) files, then the collections were cleaned with pre-processing and the quality of the data was improved. The size of the data set was decreased by removing low-frequency data with a repetition frequency of less than 1% from the study to handle the large matrix consisting of 160508 rows. Thus, the number of rows has reduced to 13332. A single word assessment would not be adequate due to the long texts of certain factors in disaster management, so phrases up to 7 words were included in the Term Document Matrix (TDM) with the N-Gram technique if they exceed the repetition frequency. A single TDM matrix consisting of 34043 rows and 786 columns was generated by merging the 7 TDMs provided.

It was considered that converting 13332 words or word groups to 122 factors with an automated topical modeling approach such as machine learning gave ineffective results, given the similar expressions and intersects of disaster management factors. Therefore, this converting procedure was conducted with the Excel search function and with binary cross-validation manual coding.

### 3.3 Integration of Analyses

The research integration was accomplished by vertically multiplying from top to bottom the weighted values obtained by the AHP method and the normalization results calculated by the text mining method in the hierarchy. To grasp the general structure, as shown in Figure 4, the CSS calculation model that will bring us to the aim of the research has been visualized. The first hierarchical level AHP weight value in CSF calculation is shown as  $W_i$  ( $i=1, 9$ ), the second hierarchical level AHP weight value is shown as  $W_{ij}$  ( $i=1, 9$ ), ( $j=1, 4$ ) and lastly, the third hierarchical level Text Mining weight value is shown as  $W_{ijz}$  ( $i=1, 9$ ), ( $j=1, 4$ ), ( $z=1, 11$ ). To calculate CSS, weighted values of Main Factor Groups and Sub-Factor Groups were first obtained. For text mining, which gives the values of the factors, by the Linear Normalization method, firstly the data is normalized. At this point, the frequencies derived concerning the number of each factor mentioned in all articles were divided into the total number of articles (773) and the normalization was carried out by multiplying these values by 100 to prevent them from being too small in the triple scalar product and to increase their value to 0-1. As a consequence, integrated weights of each factor (122) were obtained by multiplying the AHP weight values of the main factor groups vertically from top to bottom with the AHP weight values of the sub-factor groups of these main factor groups and the normalization values obtained by text mining analysis of these sub-factor groups. These integrated weights form the Critical Success Score (CSS) of these factors.

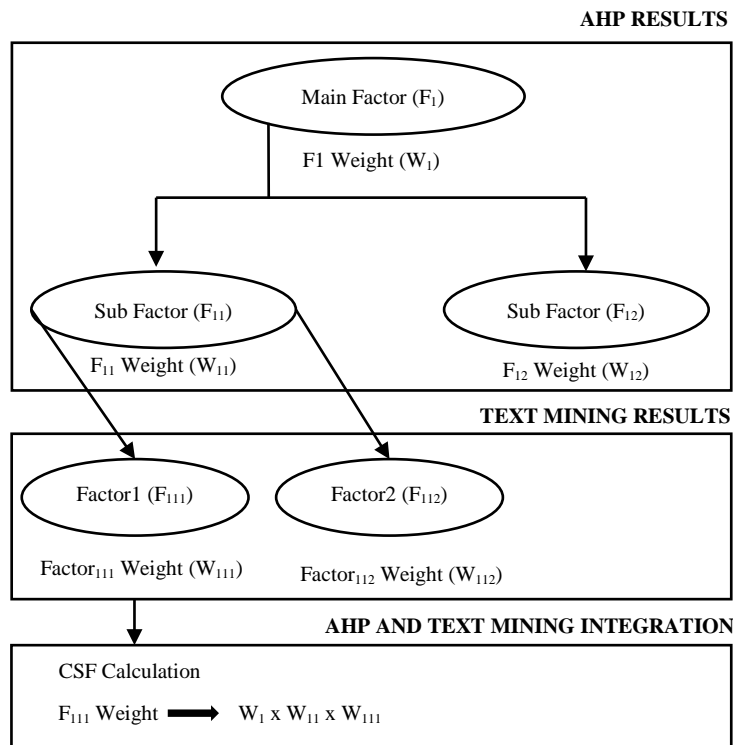


Figure 4 Critical Success Score Calculation Model

Since raw text files are recorded with year prefixes, as stated in text mining analyses, therefore the data provides an opportunity to show which areas the emphasis has shifted to regarding the success of disaster management. The results of "Trend Analysis" are also provided at the last stage of the study based on this situation. The values of linear regression slopes were analyzed in trend analysis by considering the values that state in how many articles each factor was listed.

#### 4. Findings

The AHP method and the text mining method will be presented separately before the results that are obtained by combining these two methods to evaluate the views of operational and academic experts separately. On the other hand, integrated results together include the views of operational and academic experts in assessing the critical success factors that are the aim of the study. Although there are several studies in this field, including the opinions of operational experts is of particular importance in terms of successful disaster management in Turkey.

##### 4.1 AHP Analysis Findings

By evaluating data obtained from the views of operational experts in disaster management, a prioritization was established among 9 main factor groups. This prioritization is shown in Table 3:

Table 3 Main Factor Weights

Main Factors	Weights
Legal Factors	0.19
Risk Factors	0.16
Socio-Cultural Factors	0.15
Political Factors	0.14
Organizational Factors	0.12
Operational Factors	0.09
Technological Factors	0.07
Environmental Factors	0.05
Economic Factors	0.04

The weight averages determined for the lower level of the AHP hierarchical model are given in Table 4 together with their consistency rates, including the weight values of the sub-factor groups based on the main factor group and the normalized weight values for comparison with other sub-factor groups.

Table 4 Normalized Weight Values for Sub-Factor Groups

Main Factor Groups	Sub Factor Groups	Factor Weights		
		Main Factor	Sub Factor Group	Normalized Factor
Economic Factors CR=0,01	Pre-Disaster Preparation	0.04	0.69	0.028
	Post-Disaster Response		0.11	0.005
	Macro-Economic Factors		0.20	0.008
Environmental Factors CR=0,01	Environmental Disaster Prevention Activities	0.05	0.74	0.039
	Environmental Response After Disasters		0.13	0.007
	Post-Disaster Life		0.14	0.007
Socio-Cultural Factors CR=0,02	Individual Factors (Qualifications and Skills)	0.15	0.11	0.015
	Education		0.50	0.074
	Social Participation and Association		0.39	0.058
Technological Factors CR=0,07	Disaster Management Support Systems	0.07	0.30	0.02
	Effective Communication During and After Disaster		0.19	0.013
	Information Management		0.22	0.015
	Technological Infrastructure		0.30	0.02
Operational Factors CR=0,07	Pre-Disaster Planning Activities	0.09	0.52	0.045
	Disaster Response Systems and Activities		0.19	0.016
	Post-Disaster Recovery (Rescue) Operations		0.16	0.013
	Logistics Activities Before, During, and After Disaster		0.13	0.012
Organizational Factors CR=0,01	Organizational Structure	0.12	0.19	0.023
	Inter-Organizational Collaboration and Participation		0.32	0.037
	Corporate Disaster Management Plan		0.34	0.04
	Individual Competencies		0.14	0.017
Political Factors CR=0	Communication and Information Management	0.14	0.45	0.064
	Operational Factors		0.55	0.077
Legal Factors CR=0	Restrictive Legal Regulations	0.19	0.19	0.035
	Factors Related to the Implementation of Laws		0.81	0.153
Risk Factors CR=0	Pre-Disaster (Related to Disaster) Risks	0.16	0.44	0.071
	Risk Factors Related to Disaster Management		0.56	0.089

As the consistency ratio (CR) of all matrices belonging to the sub-factors is less than 0.1, it is understood that the evaluations made for each group by the participants are consistent. When the sub-factor groups in Table 4 are evaluated, especially the "Factors Related to the Implementation of Laws" sub-factor group along with the "Pre-Disaster Preparation", "Environmental Disaster Prevention Activities", "Pre-Disaster Planning Activities" sub-factor groups have come to the fore due to their important weight values in the main factor group they belong to. This case demonstrates the importance of legal factors under main factor groups within the scope of disaster management studies in Turkey. However, the weight values of the other three sub-factor groups suggest that, relative to post-disaster studies, the factors related to the reduction of the impact of disasters prior to the disaster should be given more significance, in line with the views of the people work in this field.

The values in the right column of the table are obtained by multiplying the weights of the main factor groups by the weights of the sub-factor groups of those main factor groups so that the sub-factor groups can be compared with those of the sub-factor groups of the other main factor groups. When considering the values, the group of "Factors Related to the Implementation of Laws" was concluded as the most important sub-factor group with a weighted average of 0.153. It has been determined that there is a lack of direct law enforcement in Turkey on the grounds of this group, which is considered to be the most



important compared to all other sub-factors, and it can be concluded that the focus should be on the implementation of laws. Besides, both groups have come to the fore with significant values in the Risk Main Factor Group, and the importance of risk management is also recognized from the experts' perspective. This shows that significance should be given to studies within the framework of risk management. Again, the 'Operational Factors', 'Education', 'Communication and Information Management' sub-factor groups were considered essential and these observations revealed the legal, risk, and political main factor groups based on sub-factor groups.

#### 4.2 Text Mining Analysis Findings

Over 773 articles published between 2000 and 2016 text mining analysis was performed and revealed in how many articles each factor was mentioned. Frequencies belonging to sub-factor groups and main factor groups were established via the results obtained as per the factors. In this context, when assessed in terms of factors, the 20 factors which were the most mentioned ones by academic studies are classified in Table 5 according to the results of the text mining analysis.

Table 5 Number of Articles That Were Mentioned the Factors

Ordering	Main FG	Sub Factor Group	Factors	Number of Articles	Percentage (%)
1	Legal	Restrictive Legal Regulations	Terms of References and Regulations	258	33.38
2	Legal	Restrictive Legal Regulations	Enactments and Laws	229	29.62
3	Technological	Disaster Management Support Systems	Information Management System	206	26.65
4	Operational	Pre-Disaster Planning Activities	Creating a Disaster and Emergency Plan	197	25.49
5	Technological	Information Management	Sharing Information	193	24.97
6	Environmental	Environmental Disaster Prevention Activities	Use and Protection of Natural Barriers	180	23.29
7	Organizational	Corporate Disaster Management Plan	Disaster Management Strategy and Plan	178	23.03
8	Technological	Disaster Management Support Systems	Projection and Early Warning Systems	173	22.38
9	Organizational	Inter-Organizational Collaboration and Participation	Coordination and Collaboration	172	22.25
10	Technological	Disaster Management Support Systems	Emergency Aid Support (Information) System	158	20.44
11	Operational	Logistics Act. Before, During, and After Disaster	Resource Planning and Management	151	19.53
12	Risk	Risk Factors Related to Disaster Management	Risk Evaluation	150	19.40
13	Economic	Pre-Disaster Preparation	Pre-Disaster Financial Instruments	149	19.28
14	Technological	Information Management	Communication Content (Information) Quality	142	18.37
15	Legal	Restrictive Legal Regulations	Local Regulations	131	16.95
16	Operational	Pre-Disaster Planning Activities	Statistics and Analysis of Previous Disasters	128	16.56

Table 5 Number of Articles That Were Mentioned the Factors (cont.)

17	Technological	Technological Infrastructure	Logistics Technology	114	14.75
18	Operational	Logistics Act. Before, During, and After Disaster	Logistics Planning and Management	114	14.75
19	Organizational	Individual Competencies	Leadership	110	14.23
20	Operational	Post-Disaster Recovery (Rescue) Operations	Assessing Disaster Effect	108	13.97

Taking Table 5 into account, it is seen that the first 20 factors have so many technical factors. In line with the views of academic experts, the fact that there are 5 factors of operational factors highlights the importance of technology and operational studies in general. In this context, it can be concluded that, in practice, attention should be paid to these two areas. The number of main factor groups and sub-factor groups articles and rates generated by the number of articles considered on a factor basis are shown in Table 6 (Since the factor belonging to more than one group is mentioned in the same article, the total number is seen more than 773 articles.).

Table 6 Number of Publications Mentioning the Main Factor Groups and Sub-Factor Groups

Main Factors	Sub Factor Groups	Number of Articles	Percentage (%)
Economic Factors 269 (%34.80)	Pre-Disaster Preparation	225	29.11
	Post-Disaster Response	30	3.88
	Macro-Economic Factors	85	11
Environmental Factors 293 (%37.90)	Environmental Disaster Prevention Activities	253	32.73
	Environmental Response After Disasters	78	10.09
	Post-Disaster Life	33	4.27
Socio-Cultural Factors 317 (%41.01)	Individual Factors (Qualifications and Skills)	70	9.06
	Education	168	21.73
	Social Participation and Association	174	22.51
Technological Factors 587 (%75.94)	Disaster Management Support Systems	455	58.86
	Effective Communication During and After Disaster	165	21.35
	Information Management	356	46.05
	Technological Infrastructure	235	30.40
Operational Factors 538 (%69.60)	Pre-Disaster Planning Activities	368	47.61
	Disaster Response Systems and Activities	125	16.17
	Post-Disaster Recovery (Rescue) Operations	219	28.33
	Logistics Activities Before, During, and After Disaster	233	30.14
Organizational Factors 423 (%54.72)	Organizational Structure	215	27.81
	Inter-Organizational Collaboration and Participation	216	27.94
	Corporate Disaster Management Plan	233	30.14
	Individual Competencies	180	23.29
Political Factors 237 (%30.66)	Communication and Information Management	158	20.44
	Operational Factors	131	16.95
Legal Factors 439 (%56.79)	Restrictive Legal Regulations	439	56.79
	Factors Related to the Implementation of Laws	7	0.91
Risk Factors 233 (30.14)	Pre-Disaster (Related to Disaster) Risks	84	10.87
	Risk Factors Related to Disaster Management	207	26.78

The factors belonging to the sub-factor groups of "Disaster Management Support Systems" and "Restrictive Legal Regulations" are mostly mentioned ones in academic publications, and the factors of "Pre-Disaster Planning Activities" were also mentioned in a more significant ratio compared to other factor groups. Also, the ratio of 46.05% received by the "Information Management" sub-factor group revealed the importance of the "Disaster Management Support Systems" sub-factor group and the technological factors in academic studies based on the sub-factor group. The frequency values of the factors and the number of articles were also estimated, as well as the percentage values of the main factor groups, and these values can be seen in Table 6.

Technological factors were listed in the highest number of publications, according to the frequency values in Table 6. It can be seen in this sense that today's technical advances are or should be expressed in the domain of disaster management. Within the framework of academic studies, operational factors that can be regarded as tangible indicators of disaster management have been given prime attention. The two main factor groups that fell behind based on AHP results, should also be taking into consideration by disaster management actors of Turkey based on academic studies. This comparison is also visualized in Figure 5:

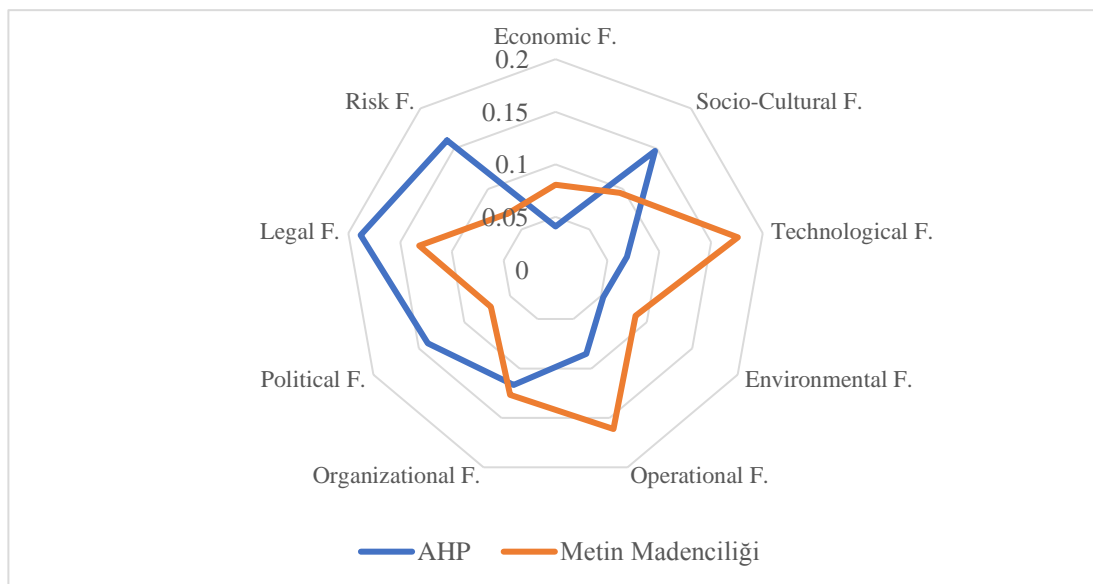


Figure 5 Comparison of Academic and Field Expert Perspectives Based on Main Factors

According to Figure 5, the perspectives of academic and operational experts are mostly coincided in the organizational field and relatively overlapped in economic, socio-cultural, environmental, and legal fields. Major differences of opinion in the areas of technology and risk were reported. The cause of the observed perspective differences in risk factors can be due to the common view of inadequate efforts made in this area and the necessity of their consideration by experts. It is seen that the AHP results reflect the disaster management structure in Turkey relatively and based on these results, the prominence of political factors expressed by experts in the field shows the importance of the role of political actors in the field of disaster management in our country. It can be suggested based on these results that political actors should exert their power in guiding activities in this field.

### 4.3 Integrated Analysis Findings

The critical success factors generated by integrating AHP with text mining results will be discussed in this part, thus the perspective of both operational experts and academic experts will be presented. Each factor's Critical Success Score (CSS) was calculated by multiplying the AHP weights of the main factor group and sub-factor groups with the normalized values derived from the frequency of the factors' publication number by text mining method. It was shown that the first 20 factors with the highest score were above 0.5 when the CSS values were evaluated. It can be concluded, based on this result, that the value of 0.5 can be taken as the threshold value for future studies. Given this situation, 20 factors were

identified as Critical Success Factors in Disaster Management within the framework of this study, because they have a value higher than 0.5. These factors are shown in Table 7.

Table 7 Critical Success Factors in Disaster Management

Order	Main Factor Group	Sub Factor Group	Factor	CSS
1	Risk	Risk Factors Related to Disaster Management	Risk Evaluation	1.727
2	Legal	Restrictive Legal Regulations	Terms of References and Regulations	1.183
3	Operational	Pre-Disaster Planning Activities	Creating a Disaster and Emergency Plan	1.135
4	Legal	Restrictive Legal Regulations	Enactments and Laws	1.05
5	Organizational	Corporate Disaster Management Plan	Disaster Management Strategy and Plan	0.913
6	Environmental	Environmental Disaster Prevention Activities	Use and Protection of Natural Barriers	0.9
7	Organizational	Inter-Organizational Collaboration and Participation	Coordination and Collaboration	0.834
8	Risk	Risk Factors Related to Disaster Management	Infrastructure	0.806
9	Operational	Pre-Disaster Planning Activities	Statistics and Analysis of Previous Disasters	0.737
10	Political	Communication and Information Management	Public Advice and Advisory Services	0.692
11	Political	Operational Factors	Health, Safety and Security Management	0.674
12	Socio-Cultural	Education	Educational Design (Education Quality, Training Content)	0.671
13	Socio-Cultural	Social Participation and Association	Rehabilitation	0.647
14	Socio-Cultural	Education	Post-Disaster Response Drills	0.643
15	Legal	Restrictive Legal Regulations	Local Regulations	0.601
16	Socio-Cultural	Education	Rescue and Healthcare Professional Training	0.595
17	Technological	Disaster Management Support Systems	Information Management System	0.545
18	Economic	Pre-Disaster Preparation	Pre-Disaster Financial Instruments	0.541
19	Risk	Pre-Disaster (Related to Disaster) Risks	Geographical Risks	0.507
20	Organizational	Inter-Organizational Collaboration and Participation	Inter-Organizational and External Communication	0.504

While risk evaluation is the most critical success factor in disaster management, if we look at the table, we see that it is notable that the factors are usually linked to prevention and planning. This proves the statement that pre-disaster prevention and mitigation measures which will mitigate the loss of life and property and minimize the potential impact of disasters are more critical than post-disaster response, rescue, and recovery activities. For this purpose, the need to prepare pre-disaster activities in our country can be suggested to institutions and organizations in the field of disaster management and to political actors who have power and influence in this domain. Table 8 was created to examine in a more general perspective, showing how many critical success factors belong to which main factor group.

Table 8 Number of CSF Belonging to Main Factor Groups

Main Factor Group	Number of CSF that inherits
Economic Factors	1
Socio-Cultural Factors	4
Technological Factors	1
Environmental Factors	1
Operational Factors	2
Organizational Factors	3
Political Factors	2
Legal Factors	3
Risk Factors	3

The fact that all nine main factor groups include critical success factors in disaster management is an indicator that these main factor groups are correctly determined in the first place and that all areas are relevant. The importance of education in disaster management is demonstrated by the fact that three of the socio-cultural factors belong to the education sub-factor group. For this reason, it can be inferred that disaster management education studies are not only productive in how we can act in times of danger, but also very significant in developing a culture of disaster. The significance of disaster culture education can also be observed in practice in Japan. The readiness culture formed for earthquakes that occur infrequent and unpredictable conditions forms the general culture of the people of Japan in this context. Schools, community centers, and workplaces are equipped with training in disaster management to implement such a culture [54]. Table 8 further highlights the need for disaster management research to have an organizational basis, the requirement to enhance them with legal factors, and to include risk management.

#### 4.4 Trend Analysis Findings

Regression analysis was carried out with text mining data to assess the factors in which the focus has shifted toward concerning the academic studies. Defining the focus in the academic field will ensure that this focus is pursued also in practical studies. In this context, while the values related to the number of articles that mentioned the factors suggest the dependent variable in the calculation of the regression slopes, the years are considered as the independent variable. Table 9 shows the top 20 factors with the highest slope value.

Table 9 The Trend of the Number of Publications Concerning the Factors by Years

Ordering	Main FG*	Sub Factor Group	Factors	Regression Slope
1	Legal	Restrictive Legal Regulations	Terms of References and Regulations	2.377
2	Legal	Restrictive Legal Regulations	Enactments and Laws	2.15
3	Technological	Information Management	Sharing Information	2.145
4	Technological	Disaster Management Support Systems	Information Management System	2.081
5	Environmental	Environmental Disaster Prevention Activities	Use and Protection of Natural Barriers	1.748
6	Technological	Disaster Management Support Systems	Emergency Aid Support (Information) System	1.62
7	Organizational	Inter-Organizational Collaboration and Participation	Coordination and Collaboration	1.6
8	Organizational	Corporate Disaster Management Plan	Disaster Management Strategy and Plan	1.566
9	Technological	Information Management	Communication Content (Information) Quality	1.556

Table 9 The Trend of the Number of Publications Concerning the Factors by Years (cont.)

10	Operational	Pre-Disaster Planning Activities	Creating a Disaster and Emergency Plan	1.52
11	Technological	Disaster Management Support Systems	Projection and Early Warning Systems	1.478
12	Operational	Pre-Disaster Planning Activities	Statistics and Analysis of Previous Disasters	1.439
13	Operational	Logistics Act.** Before, During and After Disaster	Resource Planning and Management	1.407
14	Economic	Pre-Disaster Preparation	Pre-Disaster Financial Instruments	1.324
15	Operational	Logistics Act.** Before, During and After Disaster	Logistics Planning and Management	1.248
16	Risk	Risk Factors Related to Disaster Management	Risk Evaluation	1.167
17	Technological	Technological Infrastructure	Logistics Technology	1.13
18	Legal	Restrictive Legal Regulations	Local Regulations	1.123
19	Organizational	Individual Competencies	Leadership	1.118
20	Technological	Disaster Management Support Systems	Post-Disaster Response System	1.105

\* Main Factor Group

\*\*Activities

According to the values in Table 9, the slope ratios of the factors of "Terms of References and Regulations", "Enactments and Laws", "Sharing Information" and "Information Management System" showed that the academic focus shifted to these factors. The finding that 7 of the first 20 factors with the highest slope value are technological factors illustrates the impact of technological advances on disaster management between 2000-2016. Table 10 displays the slope regression values of the main factor groups and sub-factor groups calculated by the frequency of the factors.

Table 10 The Slope of the Number of Publications in Main Factor Groups and Sub-Factor Groups by Years

Main Factors (Regression Slope)	Sub Factor Groups	Regression Slope
Economic Factors (2.618)	Pre-Disaster Preparation	2.125
	Post-Disaster Response	0.392
	Macro-Economic Factors	0.963
Environmental Factors (2.691)	Environmental Disaster Prevention Activities	2.495
	Environmental Response After Disasters	0.441
	Post-Disaster Life	0.373
Socio-Cultural Factors (2.674)	Individual Factors (Qualifications and Skills)	0.502
	Education	1.157
	Social Participation and Association	1.816
Technological Factors (5.336)	Disaster Management Support Systems	4.13
	Effective Communication During and After Disaster	1.505
	Information Management	3.542
	Technological Infrastructure	2.326
Operational Factors (5.086)	Pre-Disaster Planning Activities	3.402
	Disaster Response Systems and Activities	1.044
	Post-Disaster Recovery (Rescue) Operations	2.027
	Logistics Act. Before, During, and After Disaster	2.245
Organizational Factors (3.917)	Organizational Structure	2.005
	Inter-Organizational Collaboration and Participation	2.032
	Corporate Disaster Management Plan	2.115
	Individual Competencies	1.708

Table 10 The Slope of the Number of Publications in Main Factor Groups and Sub-Factor Groups by Years  
(cont.)

Political Factors (2.071)	Communication and Information Management	1.426
	Operational Factors	1.179
Legal Factors (3.998)	Restrictive Legal Regulations	3.998
	Factors Related to the Implementation of Laws	0.069
Risk Factors (2.105)	Pre-Disaster (Related to Disaster) Risks	0.645
	Risk Factors Related to Disaster Management	1.904

As seen in Table 10, The finding that technological and operational factors have the greatest slope value is an indicator of the need to concentrate on these two fields in practice. The observation that disaster management studies also pursue technological advances is not an unforeseen outcome. Academic articles draw attention to the preparation process to include the pre-disaster, during, and post-disaster phases and the execution of the practices in the light of these preparations. Again, the legal factors coincide with the results of the AHP analysis, with a very high slope ratio. Since the organizational factors have a similar ratio and high ranking, this indicates that disaster management should have an organizational base. These results may guide the implementation of disaster management practices in Turkey particularly. Furthermore, since there is a positive slope in all the main factor groups this suggests that disaster management research in these particular nine groups gained greater value over the years.

When evaluated on the basis of sub-factor group, the "Disaster Management Support Systems" sub-factor group belonging to the technological factors has the highest slope value. This situation shows those decision support systems, which have become important in every field as a result of today's technological developments, are also reflected in the field of disaster management within the scope of this study. As "Restrictive Legal Regulations," another sub-factor group stands out as having a high slope value. This may mean the need for legal regulations, such as the establishment of standards for structural or environmental problems that pose a threat to disaster management. In general, there is a positive slope for each sub-factor group.

The above findings provide thorough guidance for successful disaster management in Turkey and offer results that allow the introduction of helpful suggestions. These findings are obtained as a result of a study aimed at building a new model that would not overlook the opinions of operational experts in Turkey and as well as the academic studies in this area.

## 5. Results

In the area where they happen, disasters can cause loss of life and property and economic losses that extend beyond the boundaries of that region. Disaster management corresponds to an infinite process consisting of pre-disaster, during disaster and post-disaster activities to minimize or avoid such losses. This process is directly affected by activities such as the availability of the required resource requirements, the formation of a social system ready for a disaster in a socio-cultural context, and technological advances.

In this study carried out for determining the critical factors for having a successful disaster management structure, it has been concluded that 9 areas which are economic, environmental, socio-cultural, technological, operational, organizational, political, legal, and risk affect the success of disaster management. The literature review for each of these areas has demonstrated that the success of disaster management is influenced by several factors. Through putting together the identified factors that have similar characteristics in the main areas, a new grouping was formed. As shown in the proposed model, a hierarchy was created, with 9 main areas at the first level, sub-factor groups at the second level, and factors at the third level. AHP method was utilized for the first two levels of the hierarchy during the application of the study. For AHP management, data obtained from experts in the field of disaster management was used. For the third level, the text mining method was used to add academic knowledge to the model by analyzing the perspective of academic experts. Critical success factors have been

obtained as a result of the integration of these two methods, which incorporate the expertise of these two different disaster management expert groups into a single melting pot.

As a result of AHP analysis, the shining out of legal factors in main factor groups, this can be interpreted by stating that disaster management works in Turkey will be functional if a legal basis is provided. Therefore, at this point, devoted efforts in the field of disaster management should be made by legislators in Turkey. Since the legal factors have also been discussed by academic studies, this can be seen in conjunction with the results of the AHP and highlight the significance of disaster management legislation in many parts of the world, not just in Turkey. In particular, the views of field experts focused on the need for education suggest that educational research should be reinforced by legislation and it is an essential requirement to set up an education system that would make instruction in disaster management mandatory.

According to text mining research, when reviewing the advancements in the technology sector today, the prevalence of technological factors is not a surprising outcome. In particular, the attention shown on the Disaster Management Support Systems is an indicator of the possible contribution that decision-support systems can offer to the disaster management. Via text mining analyses, the operational factor group has also stepped forth. This outcome matches the operational system's continuous improvement factor, which is identified by Zhou, Huang, and Zhang [6] as one of the critical success factors in disaster management. Therefore, it can be suggested to disaster management institutions in the world and Turkey to plan the operations that take place at each stage of the disaster management cycle in a feasible manner. Besides, the rising importance of slope factors over the years suggests that the focus has shifted to these two main groups of factors, which are more measurable and manageable. In terms of converting theoretical knowledge into practice in disaster management, these conclusions should be kept in mind.

Determining the risk evaluation factor as the most critical success factor after integrating both analysis methods shows that the pre-disaster process is quite crucial. The results of this study also affirm that pre-disaster research should include activities of risk assessment. The obtained results can be outlined based on the findings of the study, that disaster management should be assisted by enactments and laws, the terms of references and regulations that promote the implementation of the laws should be taken into account, and that plans should be designed to be ready for disasters and emergencies and that all these risks evaluations should be included in the process.

Highlighting the coordination and sufficient financial support in disaster management by Ophiyandri et al. [5] and emphasize of the financing plan in the study by Liu, Scheepbouwer, and Giovinazzi [4], in which critical success factors for infrastructure recovery after disasters are determined, coincide with the critical success factors obtained as a result of this study. To facilitate the sharing of the collected data among planners, designers, operators, and decision-makers, Liu, Scheepbouwer, and Giovinazzi [4] also defined the standardization of the data management mechanism as a critical success factor. Likewise, the identification of an effective emergency information system by Zhou, Huang, and Zhang [6] as one of the five critical success factors consistent with the findings of this study, which shows the value of the information management system.

While the critical success factors for disaster management defined within the framework of the study may provide practical benefits to the further research in this area, the identification of prominent factors within the groups may guide the studies carried out on a group basis. Furthermore, with the guidance of the literature, the presence of critical success factors in each of the nine main factor groups confirms that an appropriate grouping has been achieved. The following suggestions can be proposed in this context to lead future studies:

- Through accessing studies in other databases, the examined publication sample may be extended.
- Perspectives of people exposed to disasters can also be included in the study model.
- It is possible to not completely grasp the factors interacting with each other in the hierarchical structure. In this context, with different analytical methods, such as ANP (Analytical Network Process), which takes interaction into account, the model can be extended.
- With distinct methods, such as NLP (Natural Language Processing), the text mining approach can be structured.






## References

- [1] O. Ergünay, "Afet Yönetimi: Genel İlkeler Tanımlar Kavramlar," first ed., Ankara, 2009.
- [2] Afet ve Acil Durum Yönetimi Başkanlığı, "Açıklamalı afet yönetimi terimleri sözlüğü," 2020. Available: <https://www.afad.gov.tr/aciklamali-afet-yonetimi-terimleri-sozlugu> [Accessed 10 July 2020].
- [3] B. Hidayat, C. Egbu, "Critical success factors associated with post-disaster reconstruction projects," in: *Proceedings of the 27th Annual Association of Researchers in Construction Management (ARCOM) Conference – Bristol, UK*, pp. 889-898, 2011.
- [4] M. Liu, E. Scheepbouwer, S. Giovinazzi, "Critical success factors for post-disaster infrastructure recovery: Learning from the Canterbury (NZ) Earthquake Recovery," *Disaster Prevention and Management*, vol. 25, no. 5, pp. 685-700, 2016. <https://doi.org/10.1108/DPM-01-2016-0006>.
- [5] T. Ophiyandri, D. Amaratunga, C. Pathirage, K. Keraminiyage, "Critical success factors for community-based post-disaster housing reconstruction projects in the pre-construction stage in Indonesia," *International Journal of Disaster Resilience in the Built Environment*, vol. 4, no. 2, pp. 236-249, 2013. <https://doi.org/10.1108/IJDRBE-03-2013-0005>.
- [6] Q. Zhou, W. Huang, Y. Zhang, "Identifying critical success factors in emergency management using a Fuzzy DEMATEL Method," *Safety Science*, vol. 49, no. 2, pp. 243-252, 2011. <https://doi.org/10.1016/j.ssci.2010.08.005>.
- [7] D. van Niekerk, "Disaster risk reduction, disaster risk management and disaster management: Academic rhetoric or practical reality?," *Disaster Management: South Africa*, vol. 4, no. 1, pp. 6-9, 2007.
- [8] National Institute of Disaster Management, "Understanding disasters," 2020. Available: [http://nidm.gov.in/PDF/Disaster\\_about.pdf](http://nidm.gov.in/PDF/Disaster_about.pdf) [Accessed 10 January 2020].
- [9] J. Herrmann, "Disaster response planning & preparedness: Phases of disaster," in: S. Harding (Eds.), *NYDIS Manual For New York City Religious Leaders: Spiritual Care and Mental Health for Disaster Response and Recovery*, New York Disaster Interfaith Services (NYDIS), New York, pp. 11-14, 2007.
- [10] M. Kadioğlu, "Afet Yönetimi: Beklenilmeyeni Beklemek En Kötüsünü Yönetmek," first ed., T.C. Marmara Belediyeler Birliği Yayın, İstanbul, 2011.
- [11] D. Ozceylan, E. Coskun, "Defining critical success factors for national emergency management model and supporting the model with information systems," in: *Proceedings of the 5th International ISCRAM Conference – Washington, DC, USA*, pp. 376-383, 2008.
- [12] J. S. Chou, J. H. Wu, "Success factors of enhanced disaster resilience in Urban Community," *Natural Hazards*, vol. 74, no. 2, pp. 661-686, 2014. <https://doi.org/10.1007/s11069-014-1206-4>.
- [13] C. Pathirage, K. Seneviratne, D. Amaratunga, R. Haigh, "Knowledge factors and associated challenges for successful disaster knowledge sharing," *Prepared for the Global Assessment Report on Disaster Risk Reduction 2015*, pp. 1-30, 2014.
- [14] K. Seneviratne, C. Pathirage, D. Amaratunga, R. Haigh, "Disaster knowledge factors: Benefits and challenges," in: *Proceedings of International Conference on Building Resilience 2011 – Kandalama, Sri Lanka*, 2011.
- [15] K. Seneviratne, D. Amaratunga, R. Haigh, C. Pathirage, "Knowledge management for disaster resilience: Identification of the key success factors," in: *Proceedings of CIB World Congress 2010 – Salford, United Kingdom*, 2010.
- [16] Y. A. Ahmed, M. N. Ahmad, N. H. Zakaria, "Knowledge sharing framework for disaster management," *Journal of Information Systems Research and Innovation*, vol. 9, no. 1, pp. 50-60, 2015.
- [17] A. Yavuz, S. Dikmen, "Doğal afetlerin zararlarının finansmanında kullanılan afet öncesi finansal araçlar," *Marmara University Journal of Political Science*, vol. 3, no. 2, pp. 303-322, 2015. <https://doi.org/10.14782/sbd.2015216101>.
- [18] S. Chow, "Success factors for IS disaster recovery planning in Hong Kong," *Information management & Computer Security*, vol. 8, no. 2, pp. 80-87, 2000. <https://doi.org/10.1108/09685220010321326>.

- [19] D. Ismail, T. A. Majid, R. Roosli, N. A. Samah, "Project management success for post-disaster reconstruction projects: International NGOs perspectives," *Procedia Economics and Finance*, vol. 18, pp. 120-127, 2014. [https://doi.org/10.1016/S2212-5671\(14\)00921-6](https://doi.org/10.1016/S2212-5671(14)00921-6).
- [20] J. Andersen, "Globalization and natural disasters: An integrative risk management perspective," in: A. Kreimer, M. Arnold, A. Carlin (Eds.), *Building Safer Cities: The Future of Disaster Risk*, World Bank Publications, Washington, D.C., pp. 57-74, 2003.
- [21] P. Boer, "The Real Options Solution: Finding Total Value in a High-Risk World", first ed., Wiley, New Jersey, 2002.
- [22] H. Srinivas, Y. Nakagawa, "Environmental implications for disaster preparedness: Lessons learnt from the Indian Ocean Tsunami," *Journal of Environmental Management*, vol. 89, no. 1, pp. 4-13, 2008. <https://doi.org/10.1016/j.jenvman.2007.01.054>.
- [23] S. Sonak, P. Pangam, A. Giriyan, "Green reconstruction of the tsunami-affected areas in India using the integrated coastal zone management concept," *Journal of Environmental Management*, vol. 89, no. 1, pp. 14-23, 2008. <https://doi.org/10.1016/j.jenvman.2007.01.052>.
- [24] T. O. Owolabi, C. O. Ekechi, "Communication as critical factor in disaster management and sustainable development in Nigeria," *International Journal of Development and Economic Sustainability*, vol. 2, no. 3, pp. 58-72, 2014.
- [25] N. Carter, "Disaster Management: A disaster manager's handbook," first ed., Asian Development Bank, Philippines, 2008.
- [26] International Strategy for Disaster Reduction, "Living With Risk: A Global Review of Disaster Reduction Initiatives," vol. 1, Geneva, 2004.
- [27] S. Chow, W. On Ha, "Determinants of the critical success factor of disaster recovery planning for information systems," *Information Management & Computer Security*, vol. 17, no. 3, pp. 248-275, 2009. <https://doi.org/10.1108/09685220910978103>.
- [28] Y. Li, Y. Hu, X. Zhang, Y. Deng, S. Mahadevan, "An evidential DEMATEL method to identify critical success factors in emergency management," *Applied Soft Computing*, vol. 22, pp. 504-510, 2014. <https://doi.org/10.1016/j.asoc.2014.03.042>.
- [29] N. Nazli, S. Sipon, A. R. Zumrah, S. Abdullah, "The factors that influence the transfer of training in disaster preparedness training: A review," *Procedia-Social and Behavioral Sciences*, vol. 192, pp. 54-58, 2015. <https://doi.org/10.1016/j.sbspro.2015.06.008>.
- [30] M. Col, "Managing disasters: The role of local government," *Public Administration Review*, vol. 67, pp. 114-124, 2007. <https://doi.org/10.1111/j.1540-6210.2007.00820.x>.
- [31] R. Dahlan, H. M. Dahan, Y. M. Saman, "The success factors for government information sharing (GIS) in natural disaster management and risk reduction," in: *Proceedings of the 5th International Conference on Information and Communication Technology for the Muslim World – Sarawak, Malaysia*, 2013. <https://doi.org/10.1109/ICT4M.2013.6518927>.
- [32] S. Donald, "Connecting to Disasters": The Critical Success Factors of Mobile Phone Utilisation within Disaster Management Operations: The Case of Vanuatu, Master's Theses at Victoria University of Wellington School of Geography, Environment and Earth Sciences – Wellington, New Zealand, 2012.
- [33] R. Oloruntoba, "An analysis of the cyclone larry emergency relief chain: Some key success factors," *International Journal of Production Economics*, vol. 126, no. 1, pp. 85-101, 2010. <https://doi.org/10.1016/j.ijpe.2009.10.013>.
- [34] F. Arain, "Knowledge-based approach for sustainable disaster management: Empowering emergency response management team," *Procedia Engineering*, vol. 118, pp. 232-239, 2015. <https://doi.org/10.1016/j.proeng.2015.08.422>.
- [35] J. R. Harrald, "Agility and discipline: Critical success factors for disaster response," *The Annals of the American Academy of Political and Social Science*, vol. 604, no. 1, pp. 256-272, 2006. <https://doi.org/10.1177/0002716205285404>.
- [36] R. Dahlan, H. M. Dahan, M. Y. Saman, "The success factors in government information sharing (GIS)—empirical findings from the Malaysians Natural Disaster Management Context," in: *Proceedings of the 5th International Conference on Information and Communication Technology for The Muslim World – Sarawak, Malaysia*, 2013. <https://doi.org/10.1109/ICT4M.2014.7020642>.

- [37] T. Lin Moe, P. Pathranarakul, "An integrated approach to natural disaster management: Public project management and its critical success factors," *Disaster Prevention and Management: An International Journal*, vol. 15, no. 3, pp. 396-413, 2006. <https://doi.org/10.1108/09653560610669882>.
- [38] International Strategy for Disaster Reduction, "Hyogo Framework for Action 2005-2015: Building the Resilience of Nations and Communities to Disasters," *World Conference on Disaster Reduction – Hyogo*, Japan, 2005.
- [39] C. Pathirage, K. Seneviratne, D. Amaratunga, R. Haigh, "Managing disaster knowledge: Identification of knowledge factors and challenges," *International Journal of Disaster Resilience in the Built Environment*, vol. 3, no. 3, pp. 237-252, 2012. <https://doi.org/10.1108/17595901211263620>.
- [40] M. Arain, "An information technology (IT) based approach for enhancing prompt and effective post-disaster reconstruction," *Business Review*, vol. 6, no. 2, pp. 67-79, 2011.
- [41] A. Arnell, "Handbook of Effective Disaster/Recovery Planning: A Seminar-Workshop Approach," first ed., McGraw-Hill, New York, 1990.
- [42] E. Caymaz, F. V. Akyon, F. Erenel, "A model proposal for efficient disaster management: The Turkish Sample," *Procedia-Social and Behavioral Sciences*, vol. 99, pp. 609-618, 2013. <https://doi.org/10.1016/j.sbspro.2013.10.531>.
- [43] B. Hansen, "Buildings: Simple, economical house design to resist future tsunamis," *Civil Engineering – ASCE*, vol. 78, no. 8, pp. 13-14, 2005.
- [44] D. Guha-Sapir, R. Below, "The quality and accuracy of disaster data: A comparative analyses of three global data sets," in: for *The Provention Consortium The Disaster Management Facility, The World Bank – Brussels*, Belgium, pp. 1-18, 2002.
- [45] O. Ergünay, P. Gülkan, H. Güler, "Afet yönetimi ile ilgili terimler açıklamalı sözlük," in: M. Kadioğlu, E. Özdamar (Eds.), *Afet Zararlarını Azaltmanın Temel İlkeleri, Japonya Uluslararası İşbirliği Ajansı Türkiye Ofisi*, Ankara, pp. 301-353, 2008.
- [46] R. Youker, "Managing international development projects: Lessons learned," *Project Management Journal*, vol. 30, no. 2, pp. 6-7, 1999. <https://doi.org/10.1177/875697289903000202>.
- [47] R. Oloruntoba, "A wave of destruction and the waves of relief: Issues, challenges and strategies," *Disaster Prevention and Management: An International Journal*, vol. 14, no. 4, pp. 506-521, 2005. <https://doi.org/10.1108/09653560510618348>.
- [48] X. Chen, M. P. Kwan, Q. Li, J. Chen, "A model for evacuation risk assessment with consideration of pre-and post-disaster factors," *Computers Environment and Urban Systems*, vol. 36, no. 3, pp. 207-217, 2011. <https://doi.org/10.1016/j.compenvurbsys.2011.11.002>.
- [49] M. Goerigk, K. Deghdak, P. Heßler, "A comprehensive evacuation planning model and genetic solution algorithm," *Transportation Research Part E: Logistics and Transportation Review*, vol. 71, pp. 82-97, 2014. <https://doi.org/10.1016/j.tre.2014.08.007>.
- [50] F. Rockart, "Chief executives define their own data needs," 1979. Available: <https://hbr.org/1979/03/chief-executives-define-their-own-data-needs> [Accessed 11 January 2020].
- [51] S. S. Rad, "Critical success factors (CSFs) in strategic planning for information systems," *Journal of Applied Environmental and Biological Sciences*, vol. 5, no. 6, pp. 334-339, 2015.
- [52] T. L. Saaty, "How to make a decision: The analytic hierarchy process," *Interfaces*, vol. 24, no. 6, pp. 19-43, 1994. <https://doi.org/10.1287/inte.24.6.19>.
- [53] M. Timor, "Analitik Hiyerarşi Prosesi," first ed., Türkmen Kitabevi, İstanbul, 2011.
- [54] S. Tripathi, "Disaster and its socio-cultural implication: A study with culture and personality approach of anthropology (With special reference to disaster in Japan)," *Asian Academic Research Journal of Social Sciences & Humanities*, vol. 1, no. 16, pp. 58-64, 2013.

# Performance Assessment of a Turn Around Ranging in Communication Satellite Orbit Determination

 İbrahim Öz<sup>1</sup>,  Ü. Cezmi Yılmaz<sup>2</sup>,  Ümit Güler<sup>3</sup>

<sup>1</sup>Corresponding Author; Turksat AS, Cevizlidere Cad. No:31 Ankara/Turkey; ibrahimoz@gazi.edu.tr; +903129252000

<sup>2</sup>Turksat Satellite Control Center, Yaglipinar Mh. No:1 Golbasi/Ankara/Turkey; cyilmaz@turksat.com.tr;

<sup>3</sup>Turksat Satellite Control Center, Yaglipinar Mh. No:1 Golbasi/Ankara/Turkey; uguler@turksat.com.tr;

Received 12 October 2020; Revised 24 January 2021; Accepted 5 February 2021; Published online 16 February 2021

## Abstract

Satellite operators utilize a two-stations turn around ranging (TAR) system to reduce the ground station measurement system's complexity and cost while having the same or better orbit determination accuracy for communication satellites orbit determination recently. This study investigates two stations' performance, four-way ranging on communication satellite orbit determination, operational conformance, and cost. The observation data sets are collected using traditional single station tracking (SST) and the new method TAR. The computed results using the Monte Carlo method encourage the satellite operators to use a four-way ranging system to observe and measure required data sets. TAR performance is evaluated, taking SST as a reference. The six classical orbital elements (a, e, i, RAAN, AoP, and TA) are compared for large numbers of observation data. The SST and TAR results are very close to each other. The worst-case calculated Euclidian distance between SST and TAR is 1.893 km at the epoch below the 6 km success criteria. The TAR observation method is appropriate to collect data sets for precise orbit determination. This work result indicates that satellite operators should consider deploying TAR stations to collect two-station range data sets and compute the orbit for nominal north-south station-keeping maneuvers (NSSK) and east-west station-keeping (EWSK) maneuver operations. The TAR method is superior to SST in terms of accuracy, operational conformance, and costs.

**Keywords:** orbit determination, turn around ranging, four way ranging, single station tracking, satellite orbit measured data set

## 1. Introduction

Satellite orbit is determined by the use of observation data obtained from ground-based or space-based systems. Observed and measured data sets are gathered from those stations. Technical properties and utilization type of the station influence orbit estimation accuracy. Satellite operators prefer reliable, high-performance, and cost-effective station solutions [1], [2].

Satellites in orbit are subject to external forces (such as sun, moon, non-uniform earth gravity), and those forces cause orbit perturbations. Satellite operators perform regular north-south station-keeping maneuvers (NSSK) and east-west station-keeping (EWSK) maneuvers to compensate perturbations and keep the satellite within a defined control window. The assigned window usually covers a range of  $\pm 0.10^\circ$  in longitude and latitude, which the satellite should not violate, to avoid signal interference (or even physical contact) with neighboring spacecraft. The orbit of a satellite must be known precisely to perform the required maneuver efficiently and obey co-location rules if required. So, precise orbit determination is a critical factor in successfully keeping a satellite at the desired orbital location [3], [4].

The orbit determination of communication (or GEO) satellites uses many types of observations, data, and data processing methods [2]. Satellite laser ranging system uses laser light for range measurement [5]. Precise orbit determination of GEO satellites during orbit Maneuvering [6], autonomous orbit determination and orbit control for GEO satellite-based on a neural network [7], orbit determination of geostationary satellite during maneuver [8], sequential orbit determination for geostationary satellites operations [9] are most utilized orbit determination methods.

Orbit determination is the process of obtaining values of orbital parameters that entirely specifies the motion of a satellite. An accurate orbit estimator takes the measurement noise into account and determines an orbit that provides a "best fit" to the collected data. These data sets are subject to the dynamics of a satellite's orbital motion during the collecting process [10]-[12].

Precise orbit determination is necessary for the planning of orbit maneuvers and anticipating events such as eclipses. GEO satellite uninterrupted communication performance requires to keep the satellite within control windows. Communication satellite service availability requires a well-controlled satellite inside the defined window.

Single station tracking is the most frequently used ground station system for orbit determination of a communication satellite. Satellite operators alternatively select other types of tracking systems, such as turn around range measurement or four-way range measurement systems for NSSK and EWSK operations [13]. When selecting a system, the satellite operator's target is to improve orbit determination accuracy and decrease the system's operational complexity and cost.

Single station tracking method (SST) uses angle measurement of antenna and station to satellite range with a time tag. The antenna generally has mono-pulse tracking equipment, and mono-pulse tracking techniques are utilized to collect azimuth and elevation angle. When collecting angular values, mono-pulse antenna points to the satellite, and the antenna control unit reads out resolver azimuth and elevation values. A ground station transmits a ranging signal to a satellite TCR sub-system or a transponder, and a satellite transmits the received ranging signal back to a ground station to obtain range data. In this system, antenna misalignment, angular measurement accuracy, temperature fluctuation in day and night, wind load, and mechanical precisions are the primary error sources. Those errors affect the calculated orbital parameter accuracy [14].

In the turnaround ranging system (TAR), a four-way station to satellite range with a time tag is measured. TAR does not need an angular measurement. This simplifies antenna structure, and many sources of errors disappear. In addition to those advantages, the cost of the antenna significantly decreases. The measurement principle of TAR is that the ranging signal emitted from the ground station and a satellite transponder receives the signal and re-transmits to the ground; simultaneously, the remote station receives the transmitted signal from a satellite transponder. This signal uplinked to a satellite transponder, and the satellite transmits this signal back like the first signal. The main station (station A) ground system receives both re-transmitted signal and turn around (round trip) signal and processes the signals to obtain range data. This method uses a cost-effective and straightforward system compared to SST, to gather necessary data for orbit determination.

### 1. 1. Related Works

In literature, there are different works on the orbit determination of objects. Measat operational experience shows that TAR orbit determination accuracy is similar to classical ranging station performance. TAR makes ground station operations simplified [1].

Traditional antenna angle tracking can be improved by using an interferometer. This configuration provides more accurate orbit determination for geostationary satellites [3].

Satellite laser ranging (SLR) is another method to determine satellite orbits. It is an accurate method that provides sub-centimeter level range measurement. SLR enables a user to predict orbits precisely [5].

There are small changes in azimuth, elevation, and range of communication satellites. Single station tracking for orbit determination measures those three values. The study shows that elevation angle affects mostly the X variances, and the azimuth angle is more dominant in the Y variance. The range measurement effect is minimal and has the weakest effects on variations [14].

SST has azimuth and elevation bias error. Single station azimuth bias can be fixed by applying an estimated azimuth bias periodically. Three-sigma position accuracy of approximately 1.5 km can be achieved with the corrected azimuth bias for communication satellites [15]. In a satellite laser ranging, the received energy, the number of returned photons, the number of photoelectrons, and the Time-Of-Flight "TOF" affect the orbit determination accuracy [16]. The ground tracking and inter-satellite link is

another method of orbit determination for the GEO satellite. Inter satellite link geometry and ground station clock errors affect calculated error in the cross-track or along-track direction [17]. Precise measurement and high precision time synchronization are necessary between ground stations to determine satellite orbits. Two-way satellite time and frequency transfer (TWSTFT) technology is commonly used among various time synchronization methods because of very high time transmission accuracy [18].

## 1.2. Ground-Based Satellite Observations

Satellite observation can be classified as optical observations, radio observations, and radio interferometry [2, 19]. Keplerian (classical) six independent elements describe a satellite's motion in space entirely. A tracking station aims to make observations from which these six motion elements can be deduced and an orbit computed.

Satellite orbit determination requires an input measurement data related to satellite position and/or velocity. Those observation data are obtainable from the ground-based tracking system or the sensors onboard the satellite. The transmitter and the receiver may be satellite onboard or ground station equipment.

Table 1 shows the most common ground-based observation and tracking methods and their measured values [19]. The following acronyms are defined; Deep Space Network (DSN), Satellite Laser Ranging (SLR), Total Count Phase (TCP), Time Difference of Arrival (TDOA), Frequency Difference of Arrival (FDOA), the time derivative of TDOA (TDOA dot), single differencing (SD), double differencing (DD), Tracking Data and Relay System (TDRS), Bilateral Ranging Transponder System (BRTS), and Dual Frequency (DF).

Table 1 Most common ground-based measurement for orbit determination

Method	Measured Data Sets
Traditional	Azimuth/Elevation, Right Ascension/Declination, Bistatic range, 2-way range,
DSN	3-way doppler, 3-way TCP, Doppler, TCP, Sequential range
SLR	Normal pointing range
Geolocation	TDOA, FDOA, TDOA dot, SD TDOA, SD FDOA, Ground TDOA, Ground
TDRS	BRTS Range, BRTS Doppler

Table 2 shows the most common space-based object (satellite) observation and tracking methods and their measured values. In this method, GNSS (Global Navigation Satellite System) measurements in the form of pseudo-range and doppler phase-count measurements from various GNSS constellations (GPS, Glonass, Galileo, QZSS, and Beidou) are processed to generate orbits.

Table 2 Most common space-based measurements for orbit determination

Method	Measured Data Sets
GNSS, GPS etc	Pseudorange (CA, L1, L2), SD and DD, Phase (CA, L1, L2, LA), SD and DD, CA and DF navigation solution (X, Y, Z)
TDRS	4-way range, 5-way doppler, 3-way return-link doppler
Space to Space	Range, Azimuth / Elevation, Right Ascension / Declination
Ephemeris	Position (X, Y, Z), Velocity (X dot, Y dot, Z dot)

Satellite operators (or orbit determiner) specifies the available measurement data sets of satellites depending on tracking systems. Satellites or objects may have a predefined set of allowable measurement types due to the ability of an observation system.

## 2. Methods of Observation and Data Collection

All orbit determination methods have their advantages and some shortages compared to others, as expected. The operators' trade-off among them, and select the best method and relevant system according to the needs and aims.

We use two types of observation (measurement) in this study. The first one is traditional angles and range observation called SST, and the second one has recently developed two stations, four-way range observation called TAR.

The measurement data are collected by a tracking system by means of electromagnetic wave propagation in this study because of existing ground system properties. Satellite measurement data has been collected in two different ways. The first one is classic traditional (conventional) topocentric coordinate azimuth elevation and range measurement, as shown in Figure 1(a). Large size antennas with a mono-pulse tracking system and ground-based measurement equipment gather necessary data precisely. This study utilized three mono-pulse tracking earth stations (station A, B, and C) and obtained azimuth elevation and range data for SST.

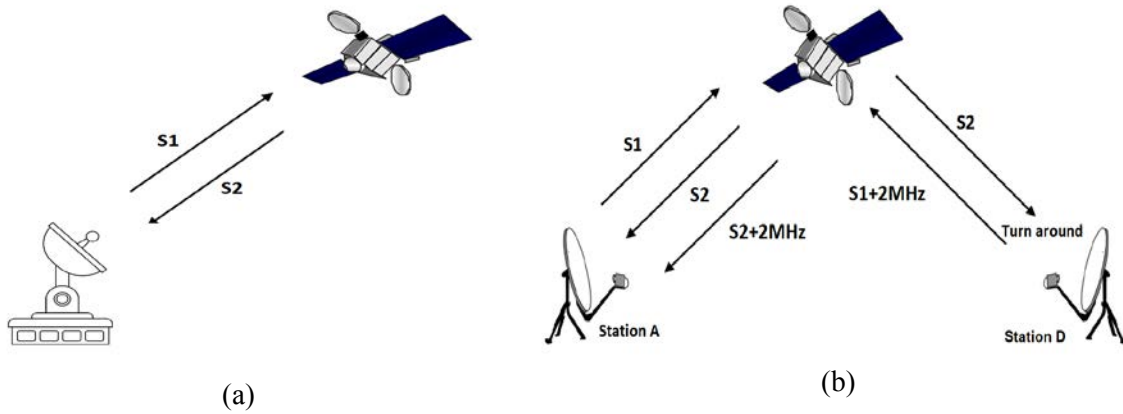


Figure 1 representation of (a) SST ground station and (b) TAR ground station

The second method is a four-way turnaround, only range measurement. In this method, two stations operate simultaneously. Generally, station D is an unmanned remote station. Both stations have 1.8 m or 2.4 m Vsat type cost-effective antennas in this method. These stations' costs are generally about 1/10 of SST type mono-pulse antenna system as of today's market conditions. Remote unmanned TAR site operational expenses are very low, and maintenance requires less effort than SST station since TAR does not need satellite tracking equipment and associated systems, large reflector, etc. In the TAR method to measure range, station A emits ranging signals S1 and the transponder onboard a GEO satellite transmits S2 (downconverted S1 signal), and the transmitted signals are received by the original tracking station to realize the distance measurement as shown in Figure 1(b).

Similarly, station D (remote station) receives the S2 signal and converts it to S1+2MHz and uplinks to the satellite. The satellite transponder onboard receives the signal and down-converts and re-transmits it as signal S2+2MHz. The ground station instrument processes both S2 and S2+2MHz signals and obtains station A to satellite range and station A to station D via satellite range data.

We utilize six earth stations (station D, E, F, G, H, J) to collect TAR data sets and evaluate two-station methods.

Both observation methods measure data every hour for two days. These duration (48 hours) is two periods of a geostationary satellite. TAR method collects 48 independently measured data sets at the end of the ranging campaign. Angles, azimuth, elevation, and range data sets are obtained for the SST method. Two station range data sets are obtained for the TAR method.

The distance from a ground station to a satellite can be defined in the following Equation 1.

$$\rho = |R_{SAT} - R_{GS}| + c \cdot \tau_{delay} + 2\Delta D_{trop} + 2\Delta D_{ion} + \varepsilon \quad (1)$$

where;  $\rho$ : station to satellite distance,  $R_{SAT}$ : satellite position vector,  $R_{GS}$ : ground station position vector,  $c$ : speed of the light,  $\tau$ : ground station and transponder time delay,  $\Delta D_{trop}$ : tropospheric delay,  $\Delta D_{ion}$ : ionospheric delay,  $\varepsilon$ : other errors

The instantaneous observation from a ground station to a satellite can be expressed in the following Equation 2 and 3.

$$t = (t_i + t_k) / 2 \tag{2}$$

$$\rho = (t_k - t_i) * c / 2 \tag{3}$$

Where  $t_i$  : time signal emitted from the ground station,  $t^k$  : time signal received at the satellite transponder,  $c$  is the speed of the light.

The antenna control unit directly reads angles (azimuth and elevation).

The following Equations 4, 5 and 6 are used to calculate satellite coordinates [13];

$$x = \rho \cos \beta \cos \alpha \tag{4}$$

$$y = \rho \cos \beta \sin \alpha \tag{5}$$

$$z = \rho \sin \beta \tag{6}$$

where,  $\rho$  is the range of the satellite,  $\alpha$ : azimuth angle  $\beta$ : elevation angle,  $x, y, z$  : coordinate of the range between station and the satellite

Figure 2(a) presents azimuth and elevation measurement of a satellite, the left vertical axis shows elevation angle data, and the right vertical axis shows azimuth angle data; Figure 2(b) shows station to satellite range and station to station range via satellite measurement of a satellite, the left vertical axis provides station to satellite range in km, and the right vertical axis provides station to station via satellite range in km. The horizontal axis is the observation number from one to forty-eight. The graphs show one complete observation that takes 48 hours and contains 48 data sets.

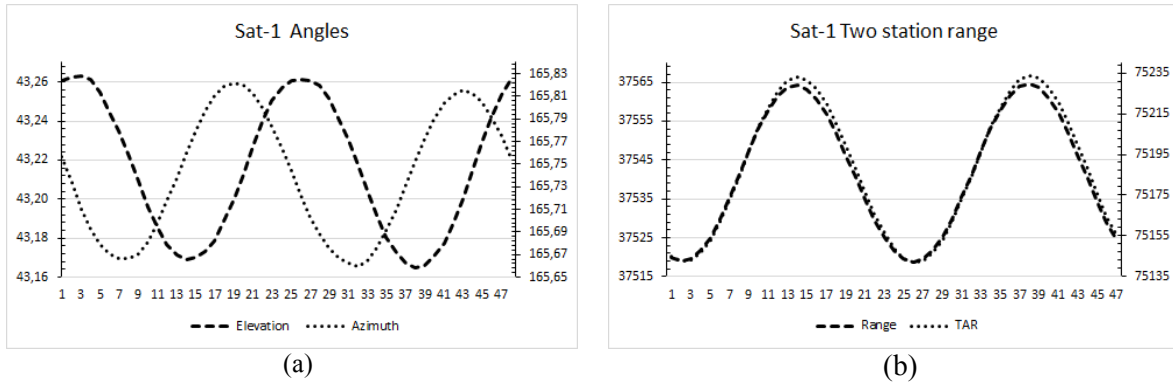


Figure 2 (a) Azimuth and elevation angle data from the mono-pulse antenna in degree (b) Station to satellite range and station to station via satellite range (TAR) in km

In this work, the overall orbit control strategy is based on a 14 days cycle. Initially, we performed a 48 hours ranging campaign to collect observation data sets, and then the NSSK maneuver is performed according to calculated orbit. After that, a 48 hours ranging campaign is performed to have the result of NSSK and prepare maneuver for EWSK. EWSK maneuver is performed after the next 48 hours, as it is necessary to wait the optimum E/W maneuver time. A 48 hours ranging campaign is then performed to confirm the orbit and provide accurate orbit parameters until the end of the cycle, as shown in Figure 3.

For one maneuver cycle of SST, three times 48 azimuth, elevation, and range measurements are gathered for a satellite. These data sets make  $3 * 48 = 144$  measurements to process for orbit determination for one cycle. Similarly, two stationTAR measurements produce 144 data sets. This work performed for two methods, three satellites and three cycles for  $2 * 3 * 3 * 144 = 2592$  measured values to process and evaluate orbits. The total duration of the work is 45 days, including data collection, processing, and maneuvering.



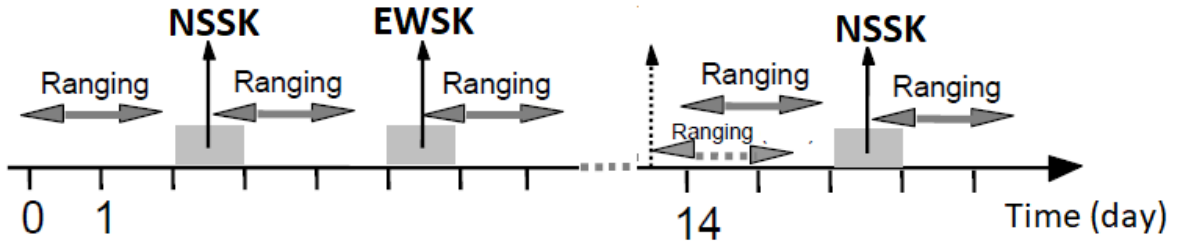


Figure 3 Geo satellite observation data collection and maneuver planning schedule

Those collected data are processed to determine the satellite orbits. This study utilizes focusgeo orbit determination software. The Orbit Determination software estimates orbital parameters from observed tracking (azimuth, elevation, range, and turn-around range) data collected from one or more tracking sites. The software is able to update the current orbit estimate based on a single new observation or process all observations.

The Orbit Determination software is able to predict the current orbit by using turn-around measurements.

The software proposes to accurately propagate an orbit into the future from a set of initial observations taking the various perturbations, as well as instrumentation errors into account. The orbit determination software estimates the six orbital elements which uniquely define the orbit, as well as maneuver delta-velocity components. In addition to the orbital elements and maneuver delta-velocity components, the software is capable of estimating tracking antenna azimuth and elevation biases, turn-around range bias, the range bias corrections to the solar force model, and maneuver performance calibrations. The software functions properly even during the absence of azimuth and elevation observations, assuming that turn-around range or range from a second, geographically remote site is available. The orbit determination software reads the initial orbit conditions from a data file specified by the user. The orbit propagator calculates the initial orbit to the time of each observation. The software provides orbit determination by using Monte Carlo methods. Using the iterative Monte Carlo method, the software outputs a summary to the user's, which indicates the level of convergence (e.g., residual, orbital element changes, etc.). The software calculates measurement residuals and orbit's determination residuals.

### 2.1. RMSE Method

A four-way TAR raging was conducted to explore the accuracy of communication satellite orbit by having range-range measurements. We propose to use the root mean square error (RMSE) method to analyze the performance of TAR measurement in this study. The RMSE compares the differences between values predicted by a model and the values computed by the other method. The RMSE is a measure of accuracy, to compare errors of different models for a particular dataset as shown in the following Equation 7.

$$RMSE = \sqrt{\frac{\sum_{t=1}^T (y1_t - y2_t)^2}{T}} \tag{7}$$

where  $y1_t$ : calculated parameters using SST,  $y2_t$ : calculated parameters using TAR,  $T$ : number of calculated results.

In this study, classical orbital elements (Keplerian parameters) of a geostationary satellite orbit are calculated using focus geo software. The first and the reference calculation approach is obtaining orbital parameters using azimuth elevation and range data with the Monte Carlo method. The second approach calculates the same epoch orbital parameters using two-station range data with the same Monte Carlo method. The epoch date and equations solving method Monte Carlo kept the same to compare the orbital parameters of two different observation methods. So, the difference in orbital parameters provides information about SST and TAR measurement model approach. Safe satellite operation requires a minimum 6 km inter-satellite distance in co-located satellites for the three-sigma separation confidence level. This 6 km requirement is selected as the success criteria of TAR orbit determination.

### 3. Results and Discussion

We have evaluated classical orbital elements of nine observation data sets from nine stations for every three satellites by using the SST and TAR method. We use the Monte Carlo method to solve the set of equations. The determined orbit results are in two formats, the first one classical orbital elements format and the second one earth-centered inertial (ECI) cartesian coordinate system format. Table 3 shows the orbital parameters of Sat-1 according to the first measurement of the cycle for SST and TAR measurement. Semimajor axis (a), eccentricity (e), and inclination (i) pairs' values are quite close to each other. Right ascension of ascending node (RAAN), argument of perigee (AoP) and true anomaly (TA) pairs' values have small differences. In Table 3, 4, and 5, a represents semimajor axis in km, e eccentricity unitless, i inclination in degree, RAAN right ascension of ascending node in degree, AoP is the argument of perigee in degree, and TA true anomaly in degree.

Table 3 Sat-1 initial orbital parameters

Classic orbital elements	SST		TAR		ECI (Cartesian)	
	SST	TAR	SST	TAR	SST	TAR
a (km)	42164.986	42164.9723	x (m)	-42155779.12	-42155979.71	
e	0.000459	0.000453	y (m)	1396566.3770	1395363.0210	
inc (deg)	0.049100	0.052900	z (m)	-17066.93322	-18001.92038	
RAAN (deg)	136.0224	137.5415	Vx (m/s)	-102.7475618	-102.6198870	
AoP (deg)	72.15160	70.01420	Vy (m/s)	-307.897254	-307.889126	
TA (deg)	329.9286	330.4004	Vz (m/s)	-2.322385740	-2.14385019	

SST method and TAR method computed a, e, i, orbital elements are shown in Table 4 for nine observations of Sat-1. It is recognized that the results are very close to each other. Similarly, Table 5 provides the results of the remaining orbital elements, RAAN, AoP, and TA. However, according to orbital parameters such as semimajor axis, inclination, etc. the variation is different.

Table 4 Sat-1 computed three orbital parameters pairs with SST and TAR observation for nine measurements. Station A for SST and station D & station E for TAR measurements

Obs #	a-SST	a-TAR	i-SST	i-TAR	e-SST	e-TAR
1	42164.98596	42164.97230	0.04910234	0.05286016	0.00045902	0.00045333
2	42165.26394	42165.26632	0.03666194	0.03887000	0.00045837	0.00045469
3	42166.78175	42166.78487	0.03892321	0.04268257	0.00054942	0.00054351
4	42164.85940	42164.83790	0.05037842	0.05065172	0.00045653	0.00045551
5	42164.81078	42164.81450	0.02874066	0.03054285	0.00045366	0.00045185
6	42166.34659	42166.34626	0.03056887	0.03293101	0.00045213	0.00044899
7	42166.22994	42166.23123	0.04153524	0.04233847	0.00036831	0.00036720
8	42165.44997	42165.45371	0.02146868	0.02114429	0.00041670	0.00041762
9	42167.20587	42167.20546	0.02554608	0.02653001	0.00049724	0.00049570

Longitude is not one of the six classical orbital elements, but it is calculated using some elements and provides information about satellite orbital location. Satellite operators keep a satellite in a defined longitudinal window, that's why longitude values of SST and TAR method are added to Table 5's column.

Table 5 Sat-1 computed four orbital parameters pairs with SST and TAR observation for nine measurements. Station A for SST and station D & station E for TAR measurements

Obs #	RAAN-SST	RAAN-TAR	AoP-SST	AoP-TAR	TA-SST	TA-TAR	Long-SST	Long-TAR
1	136.022429	137.541499	72.151565	70.014227	219.848164	220.254944	41.968486	41.970130
2	166.797524	166.036758	48.624112	49.110470	213.485476	214.114985	42.046363	42.045992
3	159.473316	158.897754	51.964912	52.119508	237.321082	237.742659	41.977561	41.978172
4	124.713056	126.761488	102.261324	99.807798	219.848164	220.254944	41.963148	41.964834
5	158.628444	162.762408	74.730430	69.967004	213.485476	214.114985	42.033573	42.033620
6	149.192599	152.585861	76.339560	72.339674	222.800639	223.407696	41.986209	41.986642
7	119.697845	118.149078	101.599432	103.519587	226.873536	226.502180	41.940869	41.940901
8	164.420003	165.718663	71.227446	69.842861	211.824073	211.909868	42.021552	42.021422
9	149.334633	150.615965	78.773232	77.314953	221.070292	221.247426	41.972284	41.972471

The difference in orbital parameters calculated by SST and TAR are provided in Table 6. The semimajor axis maximum difference is 0.4160 m in observation number 9, and the minimum difference is 0.0013 m in observation number 7. These values are very small and acceptable. The eccentricity difference is  $5.91 \times 10^{-6}$  and  $9.2 \times 10^{-7}$  for the maximum and the minimum values, respectively. Inclination maximum difference value is  $0.00376^\circ$ , and the minimum value is  $0.00027^\circ$ . RAAN and AoP differences are in the order of  $1^\circ$  to  $5^\circ$  and TA difference  $0.6199^\circ$  and  $0.0858^\circ$  for the maximum and the minimum, respectively. Those values are very close to each other and acceptable.

Table 6 Sat-1 calculated six orbital elements and their associated longitude difference for nine observations.

Obs #	$\Delta a(\text{km})$	$\Delta e$	$\Delta inc(\text{deg})$	$\Delta RAAN$	$\Delta AoP$	$\Delta TA$	$\Delta Long$
1	0.0136610	5.690E-06	-0.00376	-1.51907	2.137340	-0.61991	-0.00164
2	-0.002381	3.680E-06	-0.00221	0.760770	-0.48636	-0.27404	0.000370
3	-0.003123	5.910E-06	-0.00376	0.575560	-0.15460	-0.42158	-0.00061
4	0.0214990	1.020E-06	-0.00027	-2.04843	2.453530	-0.40678	-0.00169
5	-0.003723	1.810E-06	-0.00180	-4.13396	4.763430	-0.62951	-0.00005
6	0.0003330	3.140E-06	-0.00236	-3.39326	3.999890	-0.60706	-0.00043
7	-0.001282	1.110E-06	-0.00080	1.548770	-1.92016	0.371360	-0.00003
8	-0.003742	-9.200E-07	0.000320	-1.29866	1.384590	-0.08579	0.000130
9	0.0004160	1.540E-06	-0.00098	-1.28133	1.458280	-0.17713	-0.00019

Sat-1 calculated orbital elements with SST and TAR observation, and their differences are shown in Figure 4 - Figure 6. The left vertical axis shows the calculated value of the orbital element in two different colors. The right vertical axis shows the difference in the result. The horizontal axis is the observation number from one to nine. It is recognized that the magnitude of differences varies according to orbital elements. However, orbital element values are all within acceptable limits.

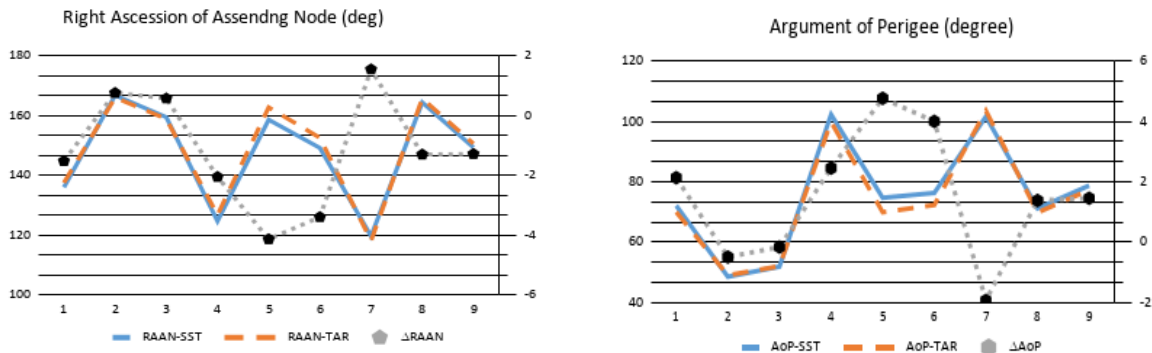


Figure 4 Calculated right ascension of ascending node, argument of perigee, eccentricity values from SST and TAR observation data and their difference for Sat-1 communication satellite.

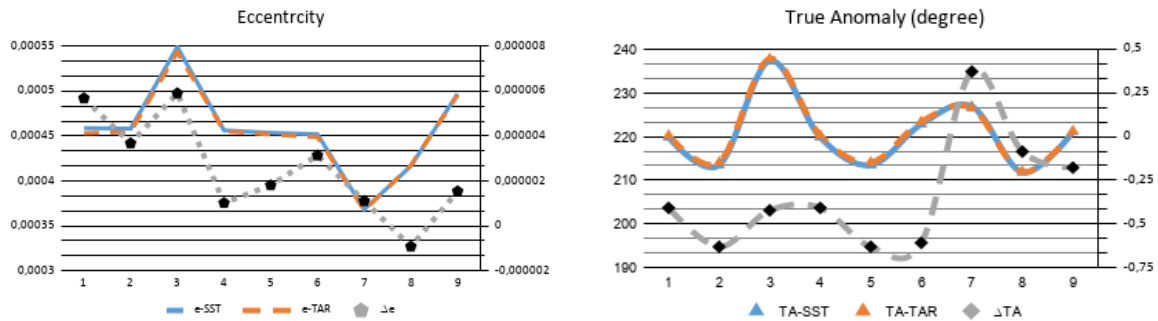


Figure 5 Calculated eccentricity, and true anomaly values from SST and TAR observation data and their difference for Sat-1 communication satellite.

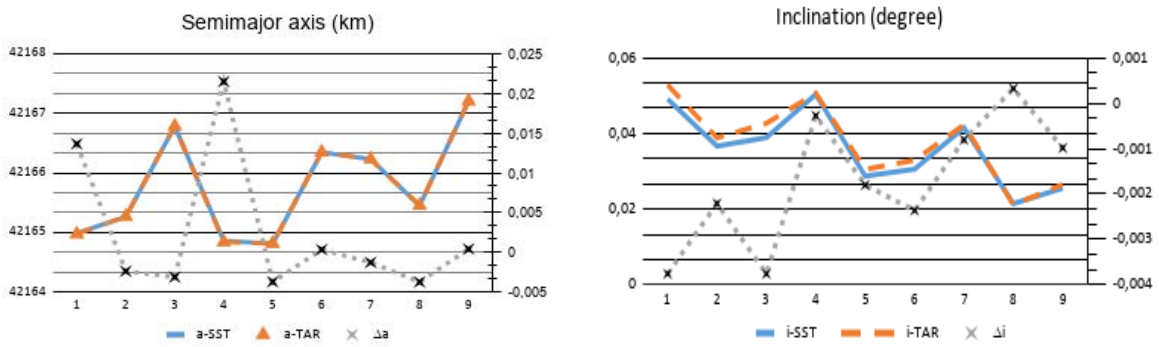


Figure 6 Calculated semimajor axis and inclination values from SST and TAR observation data and their difference for the Sat-1 communication satellite.

Table 7 shows SST and TAR observation Eucliden 3-D distance difference. This table provides results for three satellites and nine measurements. The maximum distance between SST and TAR orbit determination result is 1.893 km, and the minimum Euclidean distance is 0.2723 km. The velocity difference is 0.2307 m/s and 0.0247 m/s for the maximum and minimum value, respectively.

Table 7 Calculated Euclidean distance of three satellites for nine different SST and TAR observation data.

Obs#	Sat-1		Sat-2		Sat-3	
	$\Delta\rho$ [m]	$\Delta v$ [m/s]	$\Delta\rho$ [m]	$\Delta v$ [m/s]	$\Delta\rho$ [m]	$\Delta v$ [m/s]
1	1537.043471	0.230717612	970.5935432	0.063792428	706.5908515	0.046583117
2	898.7939172	0.109295987	847.0846436	0.061467485	1026.124267	0.109127211
3	1361.035126	0.183722471	1111.038423	0.061053572	902.7798117	0.266153135
4	1342.749503	0.130285556	1274.088327	0.055727644	1506.889572	0.051774291
5	995.2406176	0.125742438	975.3496304	0.065700717	967.8471576	0.110689447
6	1188.324237	0.146028127	1342.10072	0.06924296	1098.628561	0.174532104
7	666.7779358	0.037654242	1696.844302	0.054849738	1893.410309	0.094463516
8	272.2957814	0.024368393	1159.273708	0.069173446	1539.522977	0.129045018
9	547.2337698	0.058411687	1296.835335	0.063771096	966.2989378	0.064260646

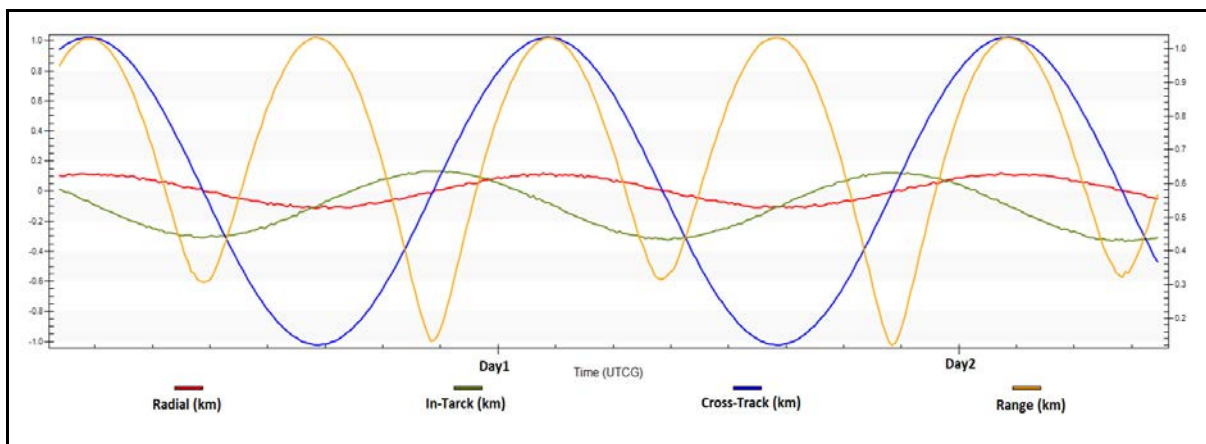


Figure 7 Cross-track, in-track, and radial position vector of Sat-1.

The position vector accuracy investigation in Figure 7 provides information that the maximum difference is in cross-track, which is about 2 km. The in-track difference is noticeably small and about 0.4 km. The radial distance is very low and less than 0.2 km.

Table 8 RMSE summary of six classical data for three satellites calculated by using SST and TAR measurement data

Satellites	$\Delta a(\text{km})$	$\Delta e$	$\Delta \text{inc}(\text{deg})$	$\Delta \text{RAAN}$	$\Delta \text{AoP}$	$\Delta \text{TA}$
Sat-1	0.008781565	3.32454E-06	0.002204812	2.153668135	2.522529844	0.440205331
Sat-2	0.001183016	1.35181E-06	0.001524819	1.555173108	1.171455943	0.415063107
Sat-3	0.000328508	2.21273E-06	0.001540499	7.182120424	6.107330600	1.177065880

Overall, SST and TAR orbital elements differences obtained using RMSE are shown in Table 8. Semimajor axis difference is 8.78 m, eccentricity difference is  $3.24 \times 10^{-6}$ , inclination difference is  $0.022^\circ$ , RAAN is  $7.18^\circ$ , AoP is  $6.107^\circ$ , and TA is  $1.777^\circ$  for the worst case. So the values are very close to each other.

Table 9 RMSE summary of position difference and velocity difference for three satellites

Satellites	$\Delta \rho$ [m]	$\Delta V$ (m/S)	$\Delta \text{Long}(\text{degree})$
Sat-1	1056.542946	0.132691565	0.000836586
Sat-2	1209.904997	0.062939317	0.000306716
Sat-3	1232.455060	0.133339923	0.000581840

Similarly, the result of all calculations in Table 9 shows that the worst distance between SST and TAR orbit determination is 1.056 km and the worst velocity is 0.1333 m/s. The longitudinal difference is  $0.00084^\circ$ . The computed orbital elements values for SST and TAR observation are quite similar.

#### 4. Conclusion

GEO satellite orbit determination is conducted to assess the proposed method TAR performance. Classical orbital element calculations are carried out using real data from the tracking systems. The orbital element values are evaluated and compared for the SST and TAR methods. The results can be concluded as follows:

The position vector between SST and TAR as a Euclidean distance is about 1.056 km, for the worst case. The 48-h prediction in-track difference is about 0.4 km. The radial distance is noticeably very low and 0.2 km maximum.

Our study indicates that the new proposed method TAR results are usable, reliable, and acceptable for orbit determination operations. Satellite operators should consider TAR methods to collect the data. Consequently, satellite operators should direct more attention to deploy TAR stations.

Orbit determination using TAR measurement data is reliable and useful for maneuver operations and other calculations such as eclipses.

In summary, the TAR method is presented in this work, which solves operational complexity, multiple error source, and an investment and operation cost problem of the SST method.

This work may be extended to evaluate the TAR performance of different orbit determination methods and different filters.


#### Acknowledgments

We would like to thank Turksat AS for its invaluable support.

## References

- [1] M. Possner, F. F. Martinez , A. Agueda Mate, , G. Garcia, , C. N.Ping Choon, , W. Hasnibi, "Operational and performance aspects of a turn-around tracking system," *In SpaceOps 2010 Conference Delivering on the Dream Hosted by NASA Marshall Space Flight Center and Organized by AIAA*, pp. 2374, 2010.
- [2] O. Montenbruck, E. Gill, , F. Lutze, "Satellite orbits: models, methods, and applications," *Appl. Mech. Rev.*, 55(2), B27-B28, pp. 193-222, 2002.
- [3] Soop, E. M., "Angular tracking for geostationary orbits," *IFAC Proceedings*, Volumes: 27, pp.181-186., 1994.
- [4] I. Oz, , U. C. Yilmaz , "Determination of Coverage Oscillation for Inclined Communication Satellite," *Sakarya Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 24(5), pp. 963-973, 2020.
- [5] G. Bury, K. Sośnica, R. Zajdel, "Multi-GNSS orbit determination using satellite laser ranging," *Journal of Geodesy*, 93(12), pp.2447-2463, 2019.
- [6] Z. Qin, , G. Huang, Q.Zhang, L. Wang, X. Yan, , S. Xie, X. Wang, "Precise Orbit Determination for BeiDou GEO/IGSO Satellites during Orbit Maneuvering with Pseudo-Stochastic Pulses," *Remote Sensing*, 11(21), 2587, 2019.
- [7] Y. Gao, Z.You, , B. Xu, "Integrated Design of Autonomous Orbit Determination and Orbit Control for GEO Satellite Based on Neural Network," *International Journal of Aerospace Engineering*, 2020.
- [8] M. Li, J. Geng, C. Shi, Q. Zhao, "Orbit determination of geostationary satellite during maneuvers," *In International Conference on Earth Observation Data Processing and Analysis (ICEODPA)*, vol. 7285, pp. 728520, 2008.
- [9] A. Águeda Maté, L. Strippoli, , J. Cuesta Cabanás, F. Martínez Fadrique, "SEGORD: Sequential Orbit Determination Tool for Geostationary Satellites Operations," *In SpaceOps 2008 Conference*, pp. 3206, 2008.
- [10] M. Ibrahim, M. Zahara, A.Emam, M. A. Elghany, "Evaluation of Orbit Determination Using Dual Ranging Method," *In WSEAS International Conferences*, 2007.
- [11] I. Oz, "Evaluation of station location for orbit determination of geo satellites at different slots," *In 2017 8th International Conference on Recent Advances in Space Technologies (RAST)*, IEEE, pp. 375-379, 2017.
- [12] H. D. Curtis, *Orbital mechanics for engineering students*. Butterworth-Heinemann., pp. 193- 254, 2013.
- [13] M. Possner, F. Martinez Fadrique, A. Agueda Mate, G. Garcia, C. N. Ping Choon, W. Hasnibi, "Operational and performance aspects of a turn-around tracking system," *In SpaceOps 2010 Conference Delivering on the Dream Hosted by NASA Marshall Space Flight Center and Organized by AIAA*, pp. 2374), 2010.
- [14] C. Hajiyev, M. Ata, , "Error analysis of orbit determination for the geostationary satellite with single station antenna tracking data," *Positioning*, vol:2(4), pp.135-144, 2011.
- [15] Y. Hwang, B. S. Lee, H. Y. Kim, H. Kim, and J. Kim "Orbit determination accuracy improvement for geostationary satellite with single station antenna tracking data", *ETRI journal*, 30(6), pp.774-782., 2008.
- [16] M. Ibrahim, A. M. Abd El-Hameed, Z. Liang, H. Xingwei, "Analytical investigation for the satellite ranging data of changchun-SLR station", *NRIAG Journal of Astronomy and Geophysics*, 9(1), 321-329, 2020.
- [17] Y. Lv, T. Geng, Q. Zhao, X. Xie, F. Zhang, and X. Wang, "Evaluation of BDS-3 Orbit Determination Strategies Using Ground-Tracking and Inter-Satellite Link Observation", *Remote Sensing*, 12(16), 2647, 2020.
- [18] W. Wang, X. Yang, W. Li, Y. Ge, L. Chen, F. Cao, P. Wei, P. Liu and Z. Li, "Research on Evaluation Method of Time Transfer Performance Between Ground Stations in Two-Way Satellite Comparison Network", *IEEE Access*, 9, 14038-14047, 2021.
- [19] D. Vallado, "Orbit determination using ODTK version 6," *European Space Astronomy Centre (ESA/ESAC)*, Madrid, 2010.

# A Conjoint Analysis of Propellant Budget and Maneuver Life for a Communication Satellite

 Ibrahim ÖZ<sup>1</sup>

<sup>1</sup>Corresponding Author; Turksat AS, Cevizlidere Cad. No:31 Ankara/Turkey; ioz@turksat.com.tr; +90 312 925 20 00

Received 23 December 2020; Revised 2 February 2021; Accepted 17 February 2021; Published online 25 February 2021

## Abstract

Maneuvers require velocity augmentation to control a satellite at the defined orbit. The velocity augmentation provides achieving geostationary orbit, compensating orbital perturbation, orbit dispersion correction, and any other maneuver's operations for a communication satellite. All maneuvers and propellant consumption must be taken into account in the propellant budget for successful mission management. In this study, a straightforward method was proposed to calculate satellite maneuver life or associated propellant budget for general purposes. The method provides enough accuracy for general mission planning. However, communication satellite accurate end of life estimation, especially in the last three months is vitally important and depends on many factors. According to the performance requirement of procurement's standard, the propellant budget and associated satellite maneuver's life are calculated based on the worst-case or adverse three-sigma. The worst-case calculations include allocations for inefficiencies, velocity uncertainties, dispersions resulting from thruster firings, propellant residuals, the selected thrusting, and maneuver strategies' performance. High accuracy remaining propellant estimation is necessary for a successful end of life operation and decommissioning. The cost of early deorbit because of propellant misestimation is millions of dollars. Accurate remaining propellant and associated maneuver life analysis can be performed in different methods.

The most common three methods are pressure, volume, temperature (PVT), bookkeeping (BK), and thermal propellant gauging (TPG). The propellant accuracy analysis shows that the propagation of uncertainties is related to system design, tank fill ratio, propellant load accuracy, orbital maneuver's inefficiency, pressure and temperature sensors, transducers, telemetry resolution, and error in equipment test data. Comparing the methods, PVT provides accuracy between  $\pm 27.81$  kg to  $\pm 38.93$  kg, depending on equipment size and accuracy. BK currently provides the best estimation and the highest gauging accuracy between  $\pm 9.83$  kg to  $\pm 13.76$  kg. TPG provides accuracy between  $\pm 10.52$  to  $\pm 14.73$  kg for some cases. However, the satellite operators request  $\pm 1$  kg estimation of the remaining propellant to extend the lifetime and reduce costs. The satellite manufacturers should optimize propulsion and attitude control subsystem design and manufacturing, including propellant management device performance, applied sensors reliability and accuracy, and tank expansion performance over a mission life.

**Keywords:** satellite maneuver life, propellant budget, pvt, bookkeeping, pulse counting, thermal gauging

## 1. Introduction

The satellite operators maneuver the satellites to control them at the desired orbit by changing their velocity. The velocity changes require force to apply to the satellites. Applying force is possible by using the thrusters. The thrusters need a propellant (fuel and oxidizer) to generate a force. Satellite fuel and oxidizer tanks are loaded according to mission requirements and the capability of the propulsion system. The satellites consume the fuel and oxidizer starting from transfer orbit operations. The propellant consumption continues with the geosynchronous orbit (GEO) operations and finishes with a de-orbit operation. Satellite design life and service life are not significant factors to estimate in orbit operation duration. The fuel is a crucial factor to determine a satellite life in space. Theoretical maneuver life can be computed using the classical Tsiolkovsky rocket equation. However, maneuver life parameters have uncertainties. More accurate maneuver calculations become very complicated by considering uncertainties associated with the different error sources. The satellite end of life is calculated by estimation of remaining fuel after each usage. The fuel estimation algorithms/methods provide the remaining fuel in the onboard propellant tank(s) [1].



Currently, there are three common propellant gauging methods in use. Those are pressure, volume, temperature (PVT), ideal gas law, Bookkeeping (BK), propellant flow integration, and thermal propellant gauging (TPG).

The PVT method relies on the ideal gas law  $PV=nRT$ . The tanks' pressure and temperature data are received via telemetry, and then the propellant volume and mass are obtained from pressuring gas (Helium) volume. The method's accuracy depends on the pressure transducer's accuracy, telemetry data resolution, and tank fill ratio. The accuracy decreases when the propellant amount decreases, but there is no accumulative error [2]. The accuracy of PVT depends on equipment measurement accuracy precision and size of the tanks.

The BK, bookkeeping (or pulse counting) method is based on thruster flow rate prediction. In this method, all thrusters on-times, pulse widths, and flow rates are recorded. The uncertainty of the flow rate during apogee kick engine (AKE) and thruster firings affect the method's accuracy directly. The propellant flow rate depends on many parameters like thruster temperature, feed-pressure, and duty-cycle. The BK method has a typically consumed propellant accuracy in between  $\pm 0.5\%$  and  $3.5\%$ . Thrusters parameters and efficiency of firing mode (pulse or continues) may have different values; however, these values are taken as constant. An accumulated error occurs when time increases [5].

The TPG (thermal propellant gauging) method relies on the tank's thermal response filled with liquid propellant and pressuring gas helium. In the TPG test, the resulting temperature increase with time due to the known amount of energy applied to the propellant tank by the heaters, are recorded. Temperature versus heating time curves for different propellants are utilized to calculate in orbit propellant estimation using on-orbit TPG data. The TPG method is superior to the Bookkeeping and PVT methods [3]. The accuracy of the Bookkeeping and the PVT methods decrease due to an accumulation of error with time. The TPG method has increasing accuracy with time [3]. However, in the ABS-1A case, the TPG method provided the best result. The remaining propellant has been predicted for PVT, BK, and TPG. The satellite has been decommissioned, and all tanks fully depleted. The actual tank remaining propellant is estimated, and the methods are compared. The results showed that TPG prediction was very accurate and less than 1 kg. In case of a pressure transducer failure, PVT and BK methods become unavailable, but the TPG can provide data about the remaining fuel. It is an additional capability for propellant estimation. [4]. TPG method propellant error estimation is inversely proportional to the heating time [7]. TPG method application on a real in orbit satellite is shown in Figure 1 and Figure 2. Figure 1(a) shows Sat-1 finite element tank model grids developed for TPG.

The development of the finite element model of the propellant tank is a complex process. Figure 1(b) shows temperature distribution over the tank's surface for 20, 50, and 80 hours of heating [7].

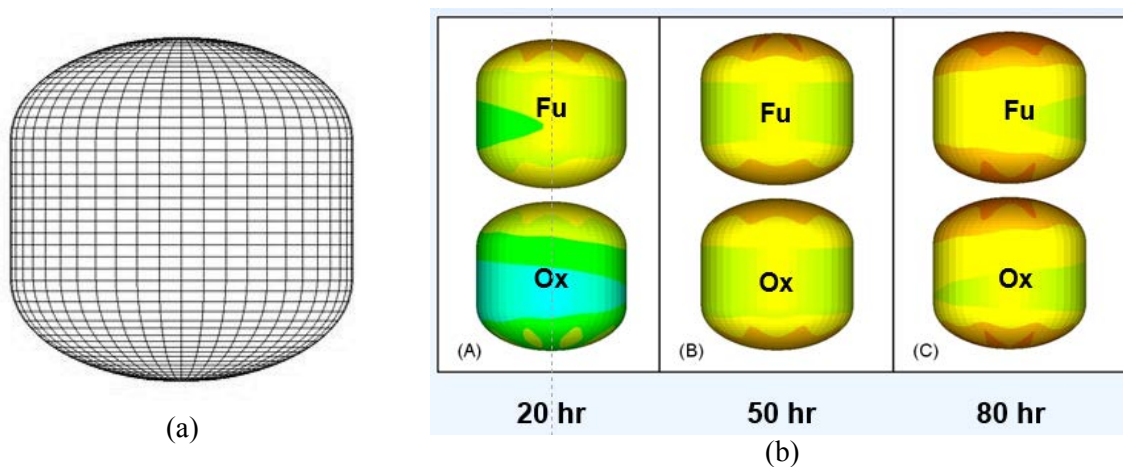


Figure 1 Sat-1 tank thermal model (a) Finite Element model (b) temperature distribution



Figure 2(a) shows the remaining fuel estimation based on simulation results. The solid line shows simulation results, and markers show flight data temperature sensor readings. Tank heaters were turned ON at  $t=0$ . Figure 2 (b) shows TPG operations, response to heaters. The temperature versus time data of top and bottom oxidizer and fuel tanks. Figure 2 (a) curves were obtained using TPG test data and their several times' calculated simulation results.

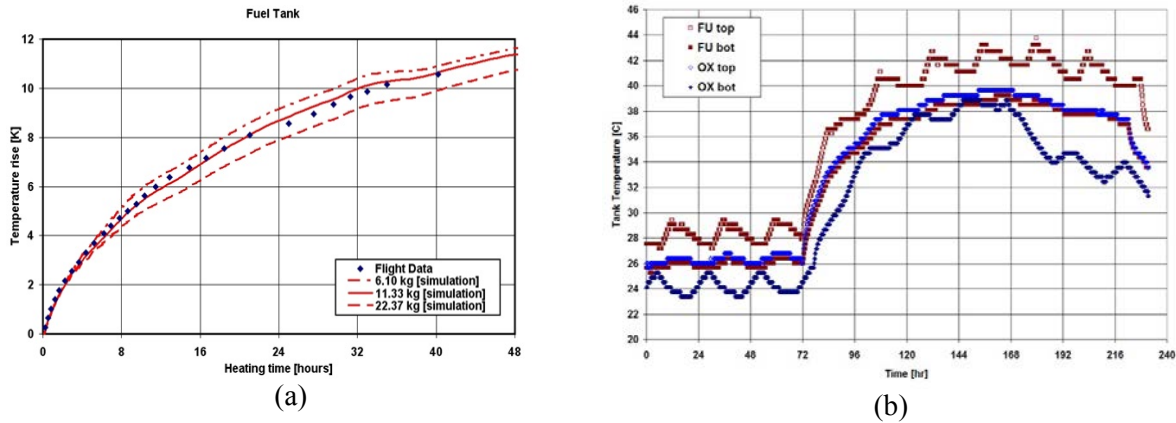


Figure 2 (a) Results of TPG estimation for fuel tanks. (b) TPG operations, response to heater.

All three-gauging methods have differences between them. System design and types of equipment highly affect the precision of the gauging systems.

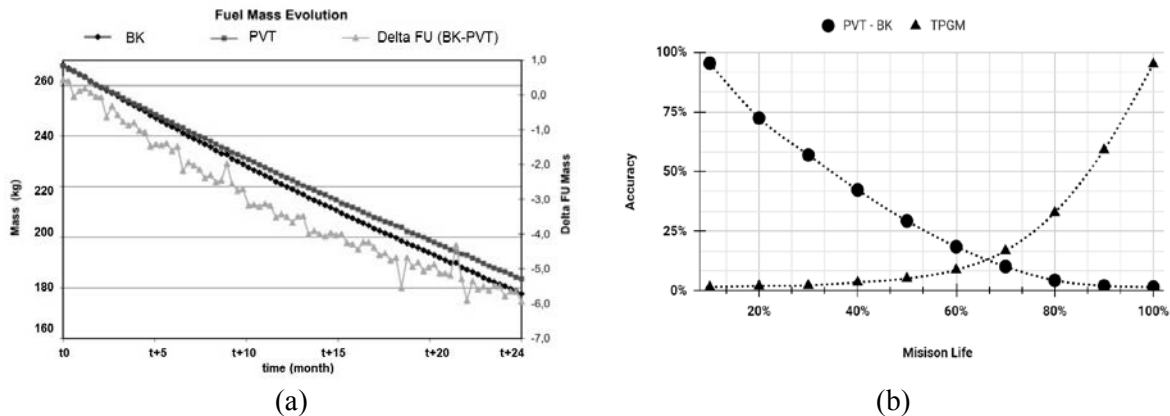


Figure 3 (a) PVT and BK comparison with in-orbit data (b) Schematic representation of propellant gauging systems error built-up

Figure 3(a) shows PVT vs. BK fuel evaluation of Sat-A in-orbit data. The remaining propellant was calculated after each use. The difference between BK and PVT methods are shown on the left vertical axis.

Figure 3 (b) shows the general trend of PVT-BK versus the TPGM method for estimation of remaining propellant performance. The PVT and BK have better accuracy than the TPGM method at the beginning of satellite life. The accuracies of all methods are comparable in the middle of life. However, the TPGM method has better accuracy than PVT and BK at the end of satellite life.

Table 1 provides currently available the most popular propellant gauging system, their description, and sensors [1]. The sensor's accuracy is a major contributor to propellant gauging and calculations.

Table 1 Typical parameters and sensors for three gauging systems

Method	Description	Sensors
PVT	Pressure, volume, temperature (gas law) based on measurement of absolute pressure and temperature.	pressure transducers, temperature sensors
BK	Bookkeeping, based on counting firings and flow of propellant.	thrust, flow-meter, temperature sensors, pressure sensors,
TPG	Thermal propellant gauging; based on the measurement of the temperature response of the fuel tank after applying heat via tank heaters.	temperature sensors, power measurements

Commercial communication satellite's replacement is very expensive and puts pressure on the improvement of end-of-life gauging accuracy for lifetime extension. The communication satellite lifetime can be extended by performing only the east-west station-keeping (EWSK) maneuver to use the remaining limited amount of fuel more efficiently. The satellite operators decide to stop the north-south station-keeping (NSSK) maneuvers of a communication satellite to extend the lifetime for several years. In this case, satellites become operated in an inclined geosynchronous orbit (IGSO), and only EWSK maneuvers are performed. EWSK maneuver consumes much less propellant than NSSK maneuvers. It is about 20-24 times less propellant consumption than NSSK maneuvers [8]. The satellite operators may provide special services and protect their orbital rights with few amounts of fuel in inclined geosynchronous orbit (IGSO) for many years. In this case, a small amount of propellant becomes very important and may lead to saving millions of dollars. A few amounts of uncertainty in the fuel may correspond to several years of in-orbit operations with high revenues [9]. The problem of the estimation of remaining propellant on geo satellites is a nightmare for the manufacturer and the operators. Early deorbit of a satellite because of misestimation causes a huge commercial loss. In general, the manufacturer prefers the worst-case end of life approach to deorbit the satellite and have a new satellite order while the satellite operators try to continue operation as long as possible. Early deorbit of a satellite not only causes the revenue loss in inclined services but also may cause to rent an interim satellite to protect orbital rights. The cost of orbital rights protection is a million of dollars debate.

Various satellite operators have different regulations, priority, and approaches to satellite end of life topics. Some operators never perform a deorbit maneuver and use every drop of propellant for nominal maneuvers, but ITU-R recommends moving the satellite graveyard. As a result, the process to obtain common decisions is long and complicated [9].

The satellite operators are crucial to have accurate propellant gauging systems onboard communication satellites for cost-effective decommissioning. However, currently, the best gauging systems in use typically have an End-of-life (EOL) prediction of  $\pm 4-6$  months for a geostationary satellite because of a combination of sensor accuracies, design, and cost limitations. In order to guarantee a 20-30 year of continuous operation, satellite builders usually take appropriate margins in the amount of propellant onboard. EOL accuracy is typically in between  $\pm 3$  to  $\pm 15$  months, depending on the gauging method and satellite system's parameters currently. More precise monitoring of the remaining propellant could extend the operational life of a satellite. The improvement of gauging accuracy is linked with the system design, and sensor accuracies. A precise comparison of all gauging methods is hard to perform [3], [10].

Tank volume and pressure variation uncertainty considerably impact the compressed propellant mass gauging. Tank volume error increases with a higher fill ratio, but in contrast, pressure variation error decreases with a high fill ratio [11]. The gauging methods have different error characteristics of prediction data due to the calculation method principle. The uncertainty of each measurement method should be estimated based on this approach [13]. The remaining propellant quantity is highly sensitive to the propellant tank pressure sensor and pressurant tank temperature sensor. This error causes a residual (ullage) amount over-prediction in the tank in general [14].

## 2. Straightforward Maneuver Life Calculation Methods

The propellant budgets are calculated according to mission requirements and associated maneuver life. The satellites need velocity change during their mission life. This velocity change ( $\Delta V$ ) can be achieved by using thrusters. The spacecraft life cycle  $\Delta V_{\text{Total}}$  can be expressed in the following Equation 1,

$$\Delta V_{\text{Total}} = \Delta v_{\text{GTO-to-GEO}} + \Delta v_{\text{NSSK}} + \Delta v_{\text{EWSK}} + \Delta v_{\text{AOCS}} + \Delta v_{\text{re-loc}} + \Delta v_{\text{disp}} + \Delta v_{\text{de-orbit}} \quad (1)$$

where:

$\Delta v_{\text{GTO-to-GEO}}$  necessary velocity augmentation to rise from geo transfer orbit (GTO) to GEO m/s,

$\Delta v_{\text{NSSK}}$  north-south station keeping velocity augmentation m/s,

$\Delta v_{\text{EWSK}}$  east-west station keeping velocity augmentation m/s,

$\Delta v_{\text{AOCS}}$  attitude control velocity augmentation m/s,

$\Delta v_{\text{re-loc}}$  station relocation velocity augmentation m/s,

$\Delta v_{\text{disp}}$  dispersion of launcher, AKE and thrusters,

$\Delta v_{\text{de-orbit}}$  maneuver to rise orbit to graveyard velocity augmentation m/s.

Each  $\Delta V$  of Equation 1 can be calculated or estimated in different methods. The first term  $\Delta v_{\text{GTO-to-GEO}}$  represents the launch and early orbit phase maneuvers. The launch vehicles generally inject the spacecraft into the transfer orbit. Launch vehicle performance and spacecraft mass affect the injected orbit parameters such as semimajor axis, inclination, and eccentricity. The spacecraft needs  $\Delta V$  to climb from transfer orbit to geostationary orbit. The following Equation 2 expresses the necessary  $\Delta v_{\text{GTO-to-GEO}}$  to perform combined inclination change and orbit raising.

$$\Delta v_{\text{GTO-to-GEO}} = \sqrt{v_{\text{ap}}^2 + v_{\text{GEO}}^2 - 2v_{\text{ap}}v_{\text{GEO}}\cos\Delta i} \quad (2)$$

where;  $v_{\text{ap}}$ , velocity at apogee m/s,  $v_{\text{GEO}}$ , velocity at GEO m/s, and  $\Delta i$  inclination angle change

Table 2 provides information about currently available the most common commercial launch vehicle performance and necessary  $\Delta v_{\text{GTO-to-GEO}}$  for a satellite to reach the final geostationary orbit. The results were obtained by using Equation 2.

Table 2 launch vehicle's typical performance and necessary  $\Delta V$  to rise orbit from GTO to GEO

Launcher	Inclination (deg)	Perigee (km)	Apogee (km)	$\Delta V$ to geo (m/s)
Falcon-9	28.5	185	35786	1837.45
Ariane-5	6.0	250	35786	1490.27
Ariane-5	2.0	250	35786	1470.10
Proton	12.0	9800	35786	961.15

The satellites in orbit are under perturbing forces such as the sun, the moon, the non-uniform gravitational force of the earth, and solar pressure, etc. The satellite oscillates in a north-south direction and drifts in the east or west direction due to these forces. The satellite operators perform north-south station-keeping (NSSK) and east-west station-keeping (EWSK) maneuvers regularly to compensate for the perturbation in orbit. Figure 3(a) shows annual inclination increment and necessary  $\Delta v_{\text{NSSK}}$  starting from 2020 for 30 years [8]. It is seen that the  $\Delta V$  requirement varies with time. The EWSK maneuver  $\Delta v_{\text{EWSK}}$  requirement depends on mainly the orbital slot of a satellite as shown in figure 3b [15], [16].

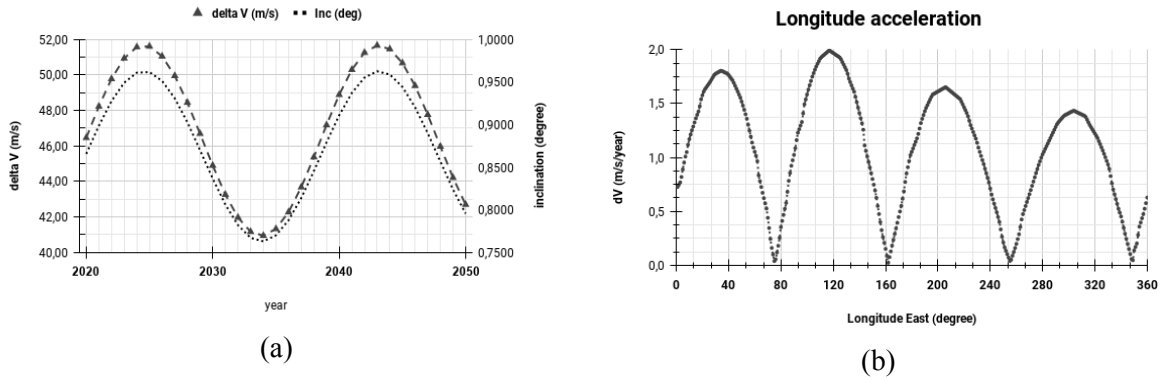


Figure 4 (a) Annual  $\Delta V$  of GEO satellite for NSSK (b) Annual  $\Delta V$  of GEO satellite for EWSSK

Satellite  $\Delta V$  capacity identifies the maneuver life of a satellite. Satellite dry mass and on-board loaded propellants and thrusters' specific impulses are key factors to amount off  $\Delta V$ . The Tsilknovsky rocket equation provides an amount of  $\Delta V$  value from the given parameters (satellite wet mass  $m_0$  and dry mass  $m_1$ ) as shown in Equation 3.

$$\Delta v = I_{sp} g_0 \ln\left(\frac{m_0}{m_1}\right) \quad (3)$$

Satellite mass change for the required  $\Delta V$  can be calculated by using the following Equation 4 derived from Equation 3.

$$m_1 = m_0 e^{\frac{-\Delta v}{I_{sp} g_0 \eta}} \quad (4)$$

where;  $m$  : satellite mass before the maneuver,  $I_{sp}$  : specific impulse of the thruster(s) used  
 $\eta$  : maneuver efficiency,  $\Delta V$  : velocity increment.

Satellite maneuver lifetime can be computed by using Equation 1. Total  $\Delta V$  requirement can be calculated by satellite propellant budget analysis. The primary parameters and conditions to execute propellant budget analysis are the following items; launch date, based on the worst-case or adverse three-sigma performance from the thrusters' datasheet, satellite dry mass, thruster efficiencies (e.g., thruster canting and plume impingement), other inefficiencies in propellant residuals, velocity uncertainties and dispersions resulting from thruster firings. The  $\Delta V$  requirements come from orbital mechanics. GEO orbit insertion, NSSK, and EWSK, and propellant budget approach identify the usage of  $\Delta V$  and associated propellant.

Velocity change ( $\Delta V$ ) optimization is necessary for all maneuvers.  $\Delta V_{GTO-to-GEO}$  should be optimized with respect to the launcher and apogee kick engine (AKE) performances, depending on the dry mass, filling ratio, and level of confidence to be reached in orbit insertion. This  $\Delta V$  is provided by the 400N AKE (sometimes called apogee boost motor ABM or liquid apogee engine LAE). The total  $\Delta V_{gto-to-geo}$  requirement is the nominal  $\Delta V$  for raising the perigee to GEO altitude from transfer orbit perigee altitude while also removing transfer orbit inclination.  $\Delta V_{GTO-to-GEO}$  is 1470.1 m/s for Ariane 5, 250 km perigee altitude with  $2^\circ$  inclination.

The cumulative NSSK  $\Delta V$  budget from Figure 2 is 875.84 m/s for 20 years from the beginning of January 2020 in the GEO orbit. Similarly, the NSSK  $\Delta V$  requirement is 1461.30 m/s for 30 years starting from the beginning of January 2020.  $\Delta V_{EWSK}$  is 1.82 m/s/year, averaged between BOL and EOL for when controlled inside a  $\pm 0.1^\circ$  window at  $31^\circ E$  from Figure 4 (b).

Longitude repositioning requires  $\Delta V_{re-loc}$  equals to 5.7 m/s, corresponding to one longitude shift of  $1^\circ/day$  and End-of-life re-orbiting requires  $\Delta V_{de-orb}$  equals 12.8 m/s to reach the circular graveyard orbit 350 km above the GEO from Hohman transfer.

The propellant mass requirements for attitude control and wheel unloading is 3 kg for a synchronous GTO injection with three AKE firings. Normal mode and station-keeping mode requires  $\Delta V_{AOCS}$  equivalent to 0.420 kg/year, for wheel unloading and attitude control.

Most of these values are based upon typical flight experience and generic analysis of a communication satellite.

Table 3 shows the total  $\Delta V$  for a satellite to operate 30 years in orbit starting from 2020. It can be recognized that EWSK  $\Delta V_{EWSK}$  depends on the satellite operational orbital location. Other  $\Delta V$  columns are the sum of attitude control (including wheel unloading), orbital relocation (5.68 m/s for 1° /day), and de-orbit  $\Delta V_{de-orb}$  (12.76 m/s for 350km).

Table 3 Satellites in geosynchronous orbit  $\Delta V$  requirement for 30 and 20 years from 2020 to 2050.

Longitude	30 year lifetime				20 year lifetime			
	NSSK	EWSK	Others	Total $\Delta V$ m/s	NSSK	EWSK	Others	Total $\Delta V$ m/s
8.5° E	1461.30	32.70	29.00	1523.00	975.84	18.17	24.20	1018.21
31.0° E	1461.30	55.20	29.00	1545.50	975.84	30.67	24.20	1030.71
42.0° E	1461.30	53.40	29.00	1543.70	975.84	29.67	24.20	1029.71
50.0° E	1461.30	46.20	29.00	1536.50	975.84	25.67	24.20	1025.71
75.1° E	1461.30	0.00	29.00	1490.30	975.84	0.00	24.20	1000.04

The dispersion budget is calculated based on typical communication satellite analyses and flight experiences. It includes on a worst-case basis (three-sigma) the contributions of launch vehicle dispersions, GEO rising maneuvers dispersions, orbit maneuvers dispersions, attitude control dispersions, mixture ratio dispersions.

Typical thrusters, either for the apogee engine or for the control thrusters, are given in Table 4. The specific impulses for the AKE and the thrusters have been computed using the qualification test reports. The truster efficiencies are extracted from the datasheet in order to take into account the thrust arc losses.

Table 4 Typical Thrusters specific impulse and efficiency data from manufacturer data sheet

Maneuvers	Efficiency (%)	Isp (sec)	Dispersion (sec)
AKE (400N)	99.6	320.7	±1.1
EWSK (average)	90.9	289.5	±3.0
NSSK (average)	93.1	287.5	±3.5
Re-location (MOL)	90.7	288.1	±3.5
De-orbiting (EOL)	90.5	286.2	±3.5
Wheel unloading	90.5	286.1	±3.5

The propellant non-usable quantity includes the static and dynamic residuals, which correspond to non-exPELLABLE propellants due to subsystem design. The static residuals include propellant liquid remaining in lines and propellant liquid and vapor remaining in the tank due to tank expulsion efficiency. Dynamic residuals are due to mixture ratio shifts through satellite life and are mainly influenced by propellant tank temperature and pressure variations. Typically, static residuals are ±0.1% of consumed propellant, and dynamic residuals are 0.8% of the propellant consumed mass. Those values highly depend on design and manufacturing parameters; for more accurate data, someone should refer to equipment data sheets and system design parameters from the manufacturers. Concerning loading uncertainty, we assume a typical value of ± 0.14% kg of total loaded propellant.

## 2.1 Calculation of Uncertainties

The calculation of uncertainties in the remaining propellant measurement is obtained by the PVT, BK, and TPG methods. As mentioned above, the PVT method is based on the determination, through the use of the on-board pressure and temperature telemetry data, of the remaining liquid volume in the propellants tanks. BK is based on the determination, through the use of the number of pulses, accumulated ON time, and tank pressure telemetry data, of the AKE motor and thrusters consumption. TPG is based on tank thermal response to applied known heater energy.

The calculation of an error function for three gauging methods is performed based on the volume, pressure, temperature, and power and their relevant sensors precision.

$\beta_M$  is defined to analyze the system linearly. The relative system function or the propellant load factor  $\beta_M$  is the ratio between the propellant mass and the tank fill mass during assembly, integration, and test (AIT) of a satellite.  $\beta_M$  varies over satellite life between  $\beta=1$  at AIT and  $\beta=0$  at EOL [1]. This ratio is obtained from the system function  $f$  as a function of all relevant system parameters  $x_i$ , as shown in Equation 5.

$$\beta_M = \frac{M_L}{M_{AIT}} = f(x_1, x_2, x_3, \dots, x_i) \quad (5)$$

In the PVT method, the propellant volume  $V_L$  is estimated from a subtraction of the known tank volume  $V_T$  and the residuals (ullage) volume  $V_U$  using Equation 6.

$$V_L = V_T - V_U \quad (6)$$

The propellant mass  $M_L$  is calculated from propellant density as a function of temperature, as shown in Equation 7.

$$M_L = V_L \rho T_L = (V_T - V_U) \rho T_L \quad (7)$$

So that the EOL gauging accuracy is related to the error in the tank and ullage volume during AIT and the in-flight ullage pressure and temperature sensor accuracies for the PVT method. The momentary propellant mass  $M_L$  can be calculated using Equation 8.

$$M_L = M_{AIT} \beta_M \quad (8)$$

BK method essential parameters for the propellant system function is the predicted total number of burns  $N$  needed for EOL operations to deplete the propellant tank.  $N$  is related to the predicted number of burns per day, and the mission duration  $D_m$  (days), as shown in Equation 9.

$$N = \frac{\text{burns}}{\text{day}} D_m \quad (9)$$

However, on average, the propellant mass  $m_n$  expelled per burn for station keeping should be like in Equation 10.

$$\bar{m}_n = \frac{M_{L,BOL}}{N} = \frac{M_{AIT}}{N} \beta_{M,BOL} \quad (10)$$

BK method's EOL gauging accuracy is related to the error in the tank mass during AIT and the relative error of the flow meter per burn, as shown above.

TPG method uncertainty can be calculated as the temperature response as function  $T(t)$  when applying a power ( $Pe$ ) to a propulsion system with heat capacity ( $H_s$ ), and effective conduction to the satellite  $C$  can be calculated with the following differential Equation 11.

$$Pe - H_s \frac{dT}{dt} - CdT = 0 \quad (11)$$

The system heat-capacity  $H_s$ , which is the sum of the heat capacity of the propellant  $H_L$  and the tank  $H_T$  is estimated from the system temperature response. Equation 12 provides heat response to propellant mass uncertainty.

$$\frac{Pe}{C_{PL} M_{L-AIT}} \approx \frac{\Delta T}{\Delta t} \frac{H_s}{H_{L-AITs}} \approx \frac{\Delta T}{\Delta t} \beta_M \quad (12)$$

TPG EOL gauging accuracy is related to the error in the tank mass and heat capacity. These two parameters can be measured on the ground. The uncertainty analysis of the TPG method considers two sources, the uncertainty of the curve fit associated with propellant load estimation and uncertainties of specific model parameters.

### 3. Results and Conclusions

In this study, we propose a straightforward method to calculate the maneuver life and associated propellant budget. The proposed method uses Equation 1 and calculates the velocity augmentation by using Equation 2 and Equation 3. As an illustrative example, we calculated two satellites' propellant budgets and associated maneuver life. Sat-A has 3500 kg mass at launch and 1400 kg dry mass, Sat-B has 4800 kg launch mass and 1570 kg dry mass. Assuming both satellites launched on January 1st, 2020, and operated at 31° E. We follow the steps below to apply the proposed method and calculate the propellant budget of two satellites;

Step 1. Calculate the  $\Delta V$  amount using Equation 3. The result is 2690.70 m/s.

Step 2. Calculate necessary  $\Delta V$  from GTO to reach GEO by using Equation 2 combine apogee firing and inclination correction (suppose Ariane launch vehicle utilized for launching with 250 km perigee height and 2° inclination on January 1st, 2020). Necessary  $\Delta V$  is 1470.1 m/s to reach geo orbit

Step 3. Calculate satellite mass change because of expelling the propellant by using Equation 4. The beginning of life at geo satellite mass is calculated as 2129.63 kg assuming apogee kick engine  $I_{sp}$ : 321 s and efficiency 99%.

Step 4. Calculate remaining available  $\Delta V$  by using Equation 4 and propellant residuals (ullage), 23.5kg. We obtained 1042.08m/s  $\Delta V$  for operations.

Step 5. Calculate  $\Delta V$  available for NSSK and EWSK. The satellite has 1012.64 m/s  $\Delta V$  for NSSK and EWSK maneuver. Satellite orbital maneuver NSSK accumulated  $\Delta V$  of 975.84 m/s makes 20.0 year satellite lifetime from Figure 4 (a) and Figure 4 (b).

The summary of the above-mentioned steps is shown in Table 5. Table 5 provides the necessary  $\Delta V$  of the proposed straightforward method for different launch weights. In reverse, the propellant budget and associated maneuver life can be calculated by applying the method.

Table 5 Sat-A maneuver life calculation using the proposed straightforward method

	$\Delta V$ (m/s)	$I_{sp}$ (s)	Efficiency	Consumed propellant(kg)	Resulting weight (kg)
Satellite total mass (kg)					3500,00
GTO to GEO	1470.10	321	94.00%	1370.37	2129.63
EWSK	36.80	288	90.00%	30.61	2099.02
Orbital relocation	5.68	288	85.00%	4.96	2094.06
Attitude control, wheel unloading		288	85.00%	9.57	2084.48
NSSK	975.84	291	91.00%	652.95	1431.54
De-orbiting	12.76	288	85.00%	7.59	1423.95
Propellant residuals				23.95	1400.00
Satellite dry mass (kg)					1400.00

Falcon 9 and Proton launch vehicle performances are different from Ariane 5 so the maneuver lifetime shown on the table 5 is subject to change. Similarly, the proposed method provides the propellant budget shown in Table 6 for 30 years maneuver life satellite.

Table 6 Sat-B maneuver life calculation using the proposed straightforward method

	$\Delta V$ (m/s)	$I_{sp}$ (s)	Efficiency	Consumed propellant (kg)	Resulting weight (kg)
Satellite total mass (kg)					4800,00
GTO to GEO	1470.10	321	94.00%	1879.37	2920.63
EWSK	55.20	288	90.00%	62.74	2857.89
Orbital relocation	5.68	288	85.00%	6.75	2851.14
Attitude control, wheel unloading		288	85.00%	19.53	2831.61
NSSK	1461.30	291	91.00%	1218.55	1613.06
De-orbiting	12.76	288	85.00%	8.55	1604.51
Propellant residuals and He				34.51	1570.00
Satellite dry mass (kg)					1570.00

The above algorithm is a straightforward approach and provides useful results to estimate the maneuver life of a satellite. However, if the satellite operators need more accurate remaining propellant information, then the situation becomes complex, and some error sources associated with input parameters must be considered.

The uncertainties from dispersions should be taken into account for a more accurate propellant budget. Table 7 summarizes the dispersions and their translation into the equivalent  $\Delta V$  associated with the  $I_{sp}$  and efficiency.

Table 7 Sat-A and Sat-B dispersion calculation based on satellite and equipment manufacturer data

Dispersions	$\Delta V$ (m/s)	$I_{sp}$ (sec)	Efficiency	20 year $\Delta V$ (m/s)	30 year $\Delta V$ (m/s)
Launch dispersions corrected with ABM	0.524	321.00	0.993	0.470	0.470
Launch dispersions corrected with Thruster	4.321	288.00	0.860	4.990	4.990
AKE pointing	4.147	321.00	0.993	3.721	3.721
AKE thrust level	0.104	321.00	0.993	0.093	0.093
AKE specific impulse	4.665	321.00	0.993	4.186	4.186
Thruster pointing	0.648	286.00	0.920	0.844	1.312
Thruster $I_{sp}$ specific impulse	9.190	286.00	0.920	11.975	18.613
Impact of N/S control on orbit longitude	7.287	286.00	0.860	10.157	15.789
Attitude control uncertainty				5.821	9.048
Residuals uncertainty (mixture ratio)				30.079	41.219
<b>Total (RSS) m/s</b>				35.247	49.34
<b>Total (RSS) kg</b>				18.912	30.08

Communication satellite detailed propellant budget can be calculated by using the same equations, but more detailed parameters should be taken into account as shown in Table 8. Table 8 provides propellant budget details of Sat-A communication satellite. In this case, 20.413 year maneuver life is calculated. There is some increase in maneuver life, but this does not mean that the detailed calculations always provide higher results.

Table 8 Sat-A maneuver life calculation considering uncertainties

Maneuvers	$\Delta V$ (m/sec)	$I_{sp}$	Efficiency	Consumed Propellant (kg)	Resulting weight (kg)
Satellite total mass (kg)					3500.00
Transfer orbit attitude control				3.00	3497.00
AKE 1	570.00	320.95	99.40	582.46	2914.54
AKE 2	794.20	320.95	99.30	654.01	2260.54
AKE 3	105.90	320.95	99.50	75.16	2185.37
Post-apogee maneuver	4.00	289.35	85.00	3.62	2181.75
IOT reposition. Maneuver	5.68	289.35	85.00	5.13	2176.62
Attitude and wheel unloading: 1		288	90	4.70	2171.92
NSSK	1013.14	291.50	91.10	699.99	1471.93
EWSK	36.40	278.50	90.20	21.59	1450.34
Attitude and wheel unloading: 2	5.5	290	100	2.80	1447.54
Station repositioning	5.680	289.35	90.20	3.21	1444.33
On-orbit raising	12.760	288.10	90.20	7.21	1437.11
Dispersion corrections	35.050	282.00	100.0	18.10	1419.01
Propellant residual		285	100	14.81	1404.20
Pressurant (Helium)				4.20	
Satellite dry mass (kg)					1400.00

Communication satellite Sat-B, detailed propellant analysis, is shown in Table 9. Table 9 provides a detailed lifetime analysis. Orbit raising details,  $I_{sp}$ , and efficiency of the thrusters are taken into account



more realistically. The result is more accurate than the straightforward method. However, the difference is only 1%.

Table 9 Sat-B maneuver life calculation considering uncertainties

Maneuvers	$\Delta V$ (m/sec)	Isp	Efficiency	Consumed Propellant (kg)	Resulting weight (kg)
Satellite total mass (kg)					4800.00
Transfer orbit att.				3.00	4797.00
AKE 1	570.00	320.95	99.40	798.98	3.998.02
AKE 2	794.20	320.95	99.30	897.13	3.100.89
AKE 3	105.90	320.95	99.50	103.10	2.997.78
Post-apogee maneuver	4.00	289.35	85.00	4.97	2.992.81
IOT reposition maneuver	5.68	289.35	85.00	7.04	2.985.77
Attitude and wheel unloading: 1		288	90	8.80	2976.98
NSKK	1.485.73	291.50	91.10	1294.27	1682.70
EWSK	54.60	278.50	90.20	36.88	1645.82
Attitude and wheel unloading: 2		8	290	4.62	1641.20
Station repositioning	5.680	289.35	90.20	3.64	1637.56
On-orbit raising	12.760	288.10	90.20	8.18	1629.38
Dispersion corrections	49.340	282.00	100.0	28.81	1600.57
Propellant residual	45.000	285	100	25.56	1575.00
Pressurant (Helium)				5.00	
Satellite dry mass (kg)					1570.00

The proposed method provides enough accuracy to estimate maneuver life based on the propellant budget. The accuracy of the remaining propellant at the end of life is calculated using three methods. PVT method provides  $\pm 27.3$  kg and  $\pm 38.93$  kg remaining propellant for Sat-A and Sat-B respectively. BK method remaining propellant estimation is  $\pm 9.83$  kg and  $\pm 13.76$  kg, and TPG remaining propellant assessment is  $\pm 10.52$  and  $\pm 14.73.85$  for Sat-A and Sat-B. Due to the high uncertainty of the PVT method, it cannot reliably be used for the EOL determination. Instead, it will be used in comparison with the pulse counting method to detect any anomaly. The accuracy analysis shows that the propagation of uncertainties is related to the propellant load, system design, ground filling, orbital operations, and applied sensors. It was found that BK currently provides the highest gauging accuracy, as shown in Table 10.

Table 10 Sat-A and Sat-B remaining propellant estimation for three gauging methods and associated IGSO maneuver life

Methods	Remaining propellant (kg)	
	Sat-A	Sat-B
PVT method	$\pm 27.81$	$\pm 38.93$
Bookkeeping method	$\pm 9.83$	$\pm 13.76$
Thermal Propellant Gauging method	$\pm 10.52$	$\pm 14.73$

The proposed methods can calculate a propellant budget from a required maneuver life. However, the satellite operators need a more accurate remaining propellant amount, especially close to satellite EOL. The most common three methods are discussed, and their performances are evaluated. The BK provides the best results currently, and TPG is the second accurate method. PVT provides coarse results and can be utilized for checking purposes. The results are based on typical communication satellite system data and equipment characteristics. Changes in system design and equipment characteristics affect the estimated accuracy of the methods. The satellite operators request to improve the gauging accuracy up to  $\pm 1$  months for new satellites. This goal can be achieved with significant improvement of the system design and analysis, ground operations, and in-flight sensor technology.

## Acknowledgments

I would like to thank Turksat AS for its invaluable support.

## References

- [1] R. C. Benthem *et al.*, “Accuracy analysis of propellant gauging systems”, *43rd International Conference on Environmental Systems*, pp. 3300, 2013.
- [2] R. Nariyoshi, S. Chernikov, and B. S.Yendler, “Prediction of Spacecraft Remaining Life-Challenges and Achievements”, *49th AIAA/ASME/SAE/ASEE Joint Propulsion Conference*, pp. 4162, 2013.
- [3] J. W. Eun, “A Study on Fuel Estimation Algorithms for a Geostationary Communication & Broadcasting Satellite”, *Journal of Astronomy and Space Sciences*, 17(2), pp.249-256., 2000.
- [4] B. S. Yendler, M. Myers, N. Chilelli, S. Chernikov, J. Wang, A. Djamshidpour, “Implementation of Thermal Gauging Method for ABS 1A (LM 3000) satellite”, *14th International Conference on Space Operations*, (p. 2458)., 2016.
- [5] B. S. Yendler, J. Molinsky, S. Chernikov, D. Guadagnoli, “Comparison of gauging methods for Orbital’s GEOStarTM 1 Satellites”, *SpaceOps 2014 Conference*, pp.1810, 2014.
- [6] B. Yendler, “Review of propellant gauging methods”, *44th AIAA aerospace sciences meeting and exhibit*, p. 939, 2006.
- [7] I. Oz, L. Pelenc, B. Yendler, “Thermal Propellant Gauging, SpaceBus 2000 (Turksat 1C) Implementation”, *AIAA 2008-7697, San Diego, California*, pp.7697, 2008.
- [8] I. Oz and Ü. C. Yılmaz, “Determination of Coverage Oscillation for Inclined Communication Satellite”, *Sakarya Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 24(5), pp. 963-973., 2020.
- [9] B. Cabrières, F. Alby, C. Cazaux, “Satellite end of life constraints: Technical and organisational solutions”, *Acta Astronautica*, vol.73, pp. 212-220, 2012.
- [10] J. Fu, X. Chen, Y. Huang, “Uncertainty Analysis of Propellant Compression Mass Gauge for Spacecraft”, *Procedia Engineering*, 31, pp.122-127, 2012.
- [11] S. Côté, S., K. Srivastava, P. Le Dantec, R. Hawkins, K. Murnaghan, ” Anik E Spacecraft Life Extension”, *Space OPS 2004 Conference*, pp. 208., 2004.
- [12] A. Aparicio, B. Yendler, “Thermal propellant gauging at EOL, Telstar 11 implementation” , *SpaceOps 2008 Conference*, pp. 3375, 2008.
- [13] W. Yin, Z. Cao, Z. Lin, “Research on Combination of Multiple Methods for Spacecraft Propellant Consumption Prediction”, *IOP Conference Series: Materials Science and Engineering*, Vol. 887, No. 1, pp. 012039 IOP Publishing., 2020.
- [14] A. Lal, B. N. Raghunandan, “Uncertainty analysis of propellant gauging system for spacecraft”, *Journal of Spacecraft and Rockets*, 42(5), pp. 943-946., 2005.
- [15] E. M Soop, W. R. Burke, “Introduction to geostationary orbits”, *STIN*, 84, 21590., 1983.
- [16] A. Grise, T. Douglas, “Maximization of satellite lifetime: Telesat Canada's experience”, *SpaceOps 2006 Conference*, pp. 5906, 2006.

# Electrical-Thermal-Mechanical Analysis of Cable Connection with Screw-Connected Terminal Strips Using Finite Element Method

 Mustafa TOSUN<sup>1</sup>,  Hüseyin AKSOY<sup>2</sup>

<sup>1</sup>Corresponding Author; Department of Electrical and Electronics Engineering, Simav Technology Faculty, Kutahya Dumlupinar University, Kutahya, Turkey; mustafa.tosun@dpu.edu.tr; +90 533 520 40 06

<sup>2</sup>Department of Electrical and Electronics Engineering, Simav Technology Faculty, Kutahya Dumlupinar University, Kutahya, Turkey; haksoy43@gmail.com

Received 2 July 2020; Revised 21 December 2020; Accepted 25 February 2021; Published online 11 March 2021

## Abstract

Electric energy passes through many stages from production to usage, and many electrical connections take place at these stages. Electrical connections are quite important for system safety and electrical energy quality. In this study, current-temperature relationship in the cable was investigated in cable joints made with screw connection terminals, depending on torque applied to screw. In numerical solution of the problem, Comsol Multiphysics program based on Finite Element Method was used first, the screw-conductor interface was investigated based on applied torque, and then electrical-thermal analysis was performed on this geometry. Experimental studies were carried out to demonstrate accuracy of digital model, and conductors and terminals used in household installations were taken into account in these studies. Amount of deformation of conductors depending on torque applied to screw and its effect to current carrying capacity of these terminals having screw connection were investigated. In this study, the appropriate torque value is 0.4 Nm, and the maximum temperature value is 45 °C on contact surface. Also, it has been shown that the optimum torque value to be applied in the copper conductor cable joint with 1.5 mm<sup>2</sup> cross section area is 0.4 Nm.

**Keywords:** Connector, COMSOL Multiphysics, Conductor junction, Electrical Analysis, Finite Element Method, Screw-connection terminal strips, Thermal Analysis

## 1. Introduction

Today, development level of countries is measured by energy consumption [1]. Electricity consumption in the world has increased day by day [2]. Cables used in transmission of electricity are connected by different joints methods. Connection elements used in electrical energy transmission systems are very important for these systems [3]. In addition, proper electrical connections are required to ensure both systems safety and transmission of electrical energy with less loss [4]. In studies of building fires occurred in Turkey was determined to be caused by electrical installations of 19% [5]. In electrical installations, the riskiest parts of cables are cable joint points [6]. Some of faults in cable joints are due to operators. At the beginning of these faults, screw-type joint method involves over-tightening or loosening the screw.

Cable joints are among the most important factors affecting the maximum current carrying value of cables [7]. In order to place transmission lines, which are used especially in high voltage, under ground cable parameters [8] and environmental parameters are determined [9]. In addition, electrical field control was made at connection points and ends of these cables [10, 11]. Finite element method is widely used for static electrical field analysis in cable joints [11,12]. Shazly Jehan H. et al. [7] Have used COMSOL Multiphysics software to evaluate three-phase underground cable, steady state and transient thermal performance. Both analytical and numerical methods have been developed in cables to calculate distribution of temperature caused by current. Analytical methods can be applied only in homogeneous ambient conditions and in simple geometries. Therefore, various numerical models have been developed to examine current carrying capacity of cables in various mounting situations [13,14,28]. COMSOL is a suite of finite element modeling software with a predefined set of physics interfaces. This software facilitates development of numerical models by solving partial differential equations [11,15].

In this study, electrical-thermal analysis of cable joints made with screw connection terminal blocks produced with COMSOL software for use in 220-volt voltage values in house electrical installations with electrical and thermal-mechanical analysis was performed. In the first phase of the study, different values of torque forces were applied to the screw by the experimental setup. The amount of crushing (deformation) in the conductor due to the applied torque forces was measured. In order to examine effects of deformation of the conductor on current carrying capacity of the conductor, temperature values occurring on conductive surface depending on conductor flow were measured. In the second phase of the study, screw connection terminal model is modeled by using COMSOL multiphysics software. Then, electrical, thermal and mechanical analysis were performed on this model. In the mechanical analysis of the designed model, the screw was tightened with forces determined in specified values and stress-deformation analysis on the conductor surface in this contact area was performed. After contact was formed, current density analysis and potential difference analysis were performed by applying 14 Amperes current to cable joint area for electrical analysis of the model. In the thermal analysis section, temperature changes due to current and contact pressure were analyzed. As a result of the analyzes, the optimum torque value for the copper conductor cable joint with a cross sectional area of 1.5 mm<sup>2</sup> was found to be 0.4Nm.

## 2. Material and Method

Insulating parts of terminal blocks are made of plastic, porcelain or bakelite according to the purpose of use. Rice, nickel and steel materials are mostly used in the conductor interior. Screw connection terminals have insulation material, terminal and screw. The insulation material is made of polyethylene (PE) material to prevent contact of conductive surfaces with each other and to prevent short circuit. The tightening torque applied to the screw creates a torsional shear stress on the screw. The torque applied to object is calculated by following Equation 1 [16].

$$\theta = r F \sin\theta \quad (1)$$

Here  $\theta$ ; the angle  $r$  between  $R$  and  $F$ ; The distance of the force from the axis of rotation (m) is  $F$ ; The applied force (N) is  $\theta$ , Torque (SI): Nm. When a material is subjected to a tensile load along a single axis, the relationship between stress and strain occurring in the material is determined by Hooke's Law [17,18] which is given in Equation 2.

$$\text{Coefficient} = \text{Stress}/\text{Strain} \quad (2)$$

The coefficient constant for normal stress along the X axis is given in Equation 3.

$$E = \sigma_x / \epsilon_x \quad (3)$$

The relationship between tensile stress and axial stress in the linear elastic region of a material is calculated by Equation 3. In the Equation 3,  $\epsilon_x$  expresses proportional deformation while  $\sigma_x$  refers to force per unit area. These two ratios allow to obtain Young's modulus (E). [19,20,21]. Conductivity is one of the main characteristics that separates copper from other metals. Pure copper, aluminum and silver are conductive materials commonly used in electricity production. Electrical conductivity of materials is given in Equation 4 [22].

$$\sigma = n \cdot q \cdot \mu \quad (4)$$

Here,  $\sigma$  is the electrical conductivity, the amount of load passing through the conductor, the charge quantity  $q$ , and the  $\mu$  load bearing constant.

### 2.1. Experimental Measurement Board

In the experimental part of this study, the connection between current and temperature in the conductor was investigated depending on the torque applied to the screw in the cable joints using screw connection terminals. For this process, the experimental setup shown in Figure 1 was formed. In this board; torque screwdriver, screw connection terminal block, H07V (NYA) type used in electrical installations, PVC insulated, without sheath, single-core 1,5 mm<sup>2</sup> copper conductor cable, arc welding machine with

transformer as power supply, measuring the temperature in terminals and conductors with current passing through the conductor, one multimeter pliers' ammeter was used.



Figure 1 Image of the experiment setup

The screw connection terminal block used was made of special C-45 steel. Insulation part of terminal block was made of Polyamide 6.6 that does not melt until 140 Celsius degrees. In screw connection terminals, conductor-terminal connection was made with a torque-adjustable screwdriver as shown in Figure 2.



Figure 2 Torque Screwdriver

Torque values of 0.3 Nm, 0.4 Nm and 0.5 Nm were applied to the screw with a screwdriver. Deformation values of these forces applied on the cable conductor were then measured. Temperature values of cable connections with 1,5 mm<sup>2</sup> cross-sectional area were measured by applying a constant current of 14 A to cable connections formed by applying torque forces mentioned above.

## 2.2. Modeling with COMSOL Software

The finite element method is a method used to solve complex structural physical problems in engineering analysis and design [23,24]. In this method, object to be modeled is divided into small and simple finite elements [25] and mathematical solutions of each element are obtained Figure 3 [26,27].

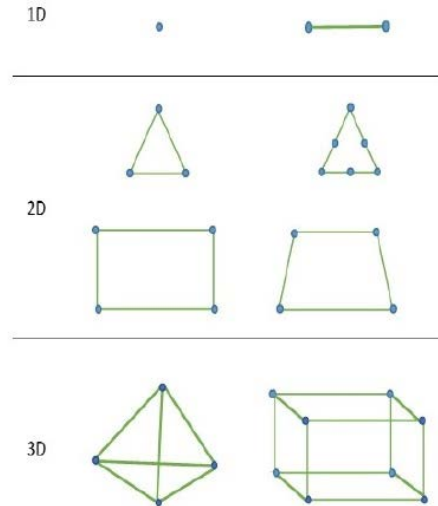
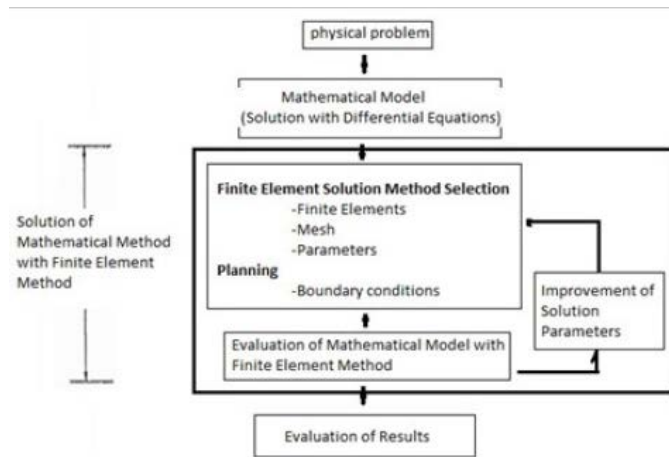


Figure 3 Shapes of elements of basic structure for finite element method [23]



(a)

Material Contents					
Property	Name	Value	Unit	Property group	
<input checked="" type="checkbox"/> Density	rho	8960[kg/...	kg/m <sup>3</sup>	Basic	
<input checked="" type="checkbox"/> Young's modulus	E	110e9[Pa]	Pa	Young's modulus and Pois...	
<input checked="" type="checkbox"/> Poisson's ratio	nu	0.35	1	Young's modulus and Pois...	
Relative permeability	mur	1	1	Basic	
Electrical conductivity	sigma	5.998e7[...	S/m	Basic	
Coefficient of thermal expansion	alpha	17e-6[1/K]	1/K	Basic	
Heat capacity at constant pressure	Cp	385[J/(kg...]	J/(kg·K)	Basic	
Relative permittivity	epsilon	1	1	Basic	
Thermal conductivity	k	400[W/(...	W/(m·K)	Basic	
Reference resistivity	rho0	1.72e-8[...	Ω·m	Linearized resistivity	
Resistivity temperature coefficient	alpha	0.0039[1/...	1/K	Linearized resistivity	
Reference temperature	Tref	298[K]	K	Linearized resistivity	

(b)

Figure 4 a) Finite element method process b) Finite element method conditions

Figure 4a shows the finite element analysis process. Physical problems (pre-processes -Modeling):

- Determination of problem type (electric-thermal-mechanical)
- Establishment of mesh structure
- Determination of material properties (hardness, conductivity, heat, Young's Module etc.).

Mathematical model solution:

- Apply loads and boundary conditions
- Let computer do numerical calculations
- Restart from pre-processing if error occurs

Improving Solution Parameters:

- Display Results in tables or graphs
- Review if result is reasonable
- If not, restart from pre-processing

Electrical analysis modeling equations are based on Maxwell's equations. AC/DC interfaces formulate the differential equations of Maxwell's equations at initial and boundary conditions. The relevant load and boundary conditions are presented in Figure 4b.

In this study, numerical modeling of screw-connection terminal blocks was used in COMSOL Multiphysics program. The screw connection terminal dimensions used in the test were measured with a digital caliper and modeled according to these measurements. The model with screw connection terminals used for analysis is shown in Figure 5.

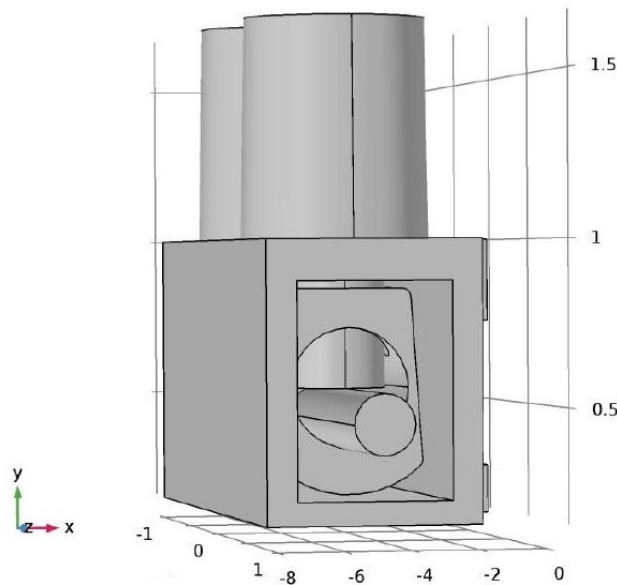


Figure 5 Installation of 1,5 mm<sup>2</sup> copper cable to the model

Mesh settings of Comsol Multiphysics program and the screw connection terminal model were divided into small sections and the model's finite element network was formed. The wiring of the screw connection strips model consisting of 37174 finite elements is shown in Figure 6. In the study, the mesh structure was made by default in physics-controlled-mesh COMSOL program instead of user-controlled mesh.

### 3. Analysis and Findings

#### 3.1. Experimental Findings

In the experimental set-up shown in Figure 1, different torque was applied to screw connections. The amount of change in diameter of the conductor depending on the torque applied to the screw is given in Figure 7.

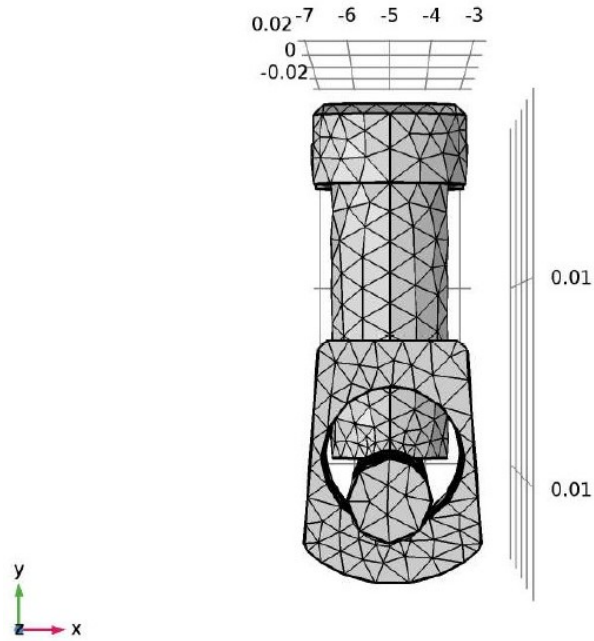


Figure 6 Mesh structure of model

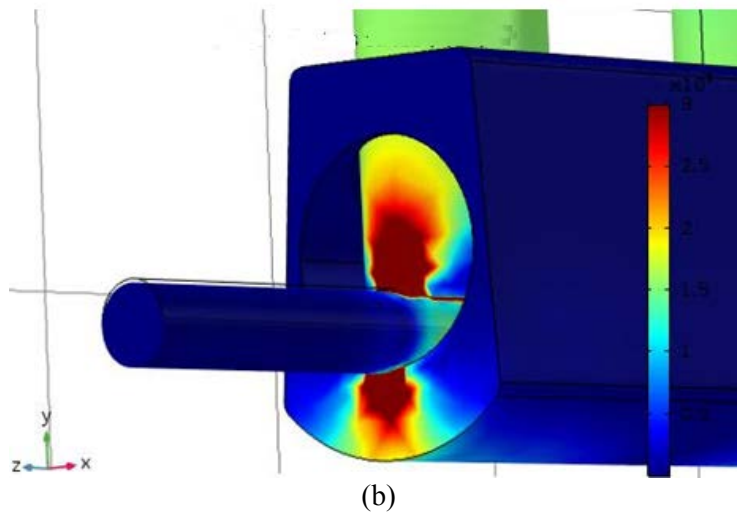
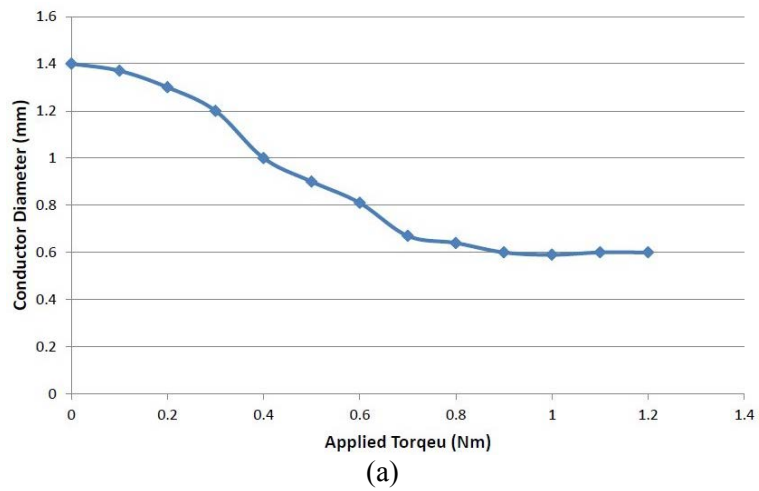


Figure 7 a) The amount of change in the conductor diameter depending on the applied torque b) Deformation of the cable on the consol



When constant 14 A current is applied to 1.5 mm<sup>2</sup> cable conductor insert, which is applied to the screw connection terminal with 0.3 Nm, 0.4 Nm and 0.5 Nm torque, respectively, temperature change values measured on the cable joint surface are shown graphically in Figure 8.

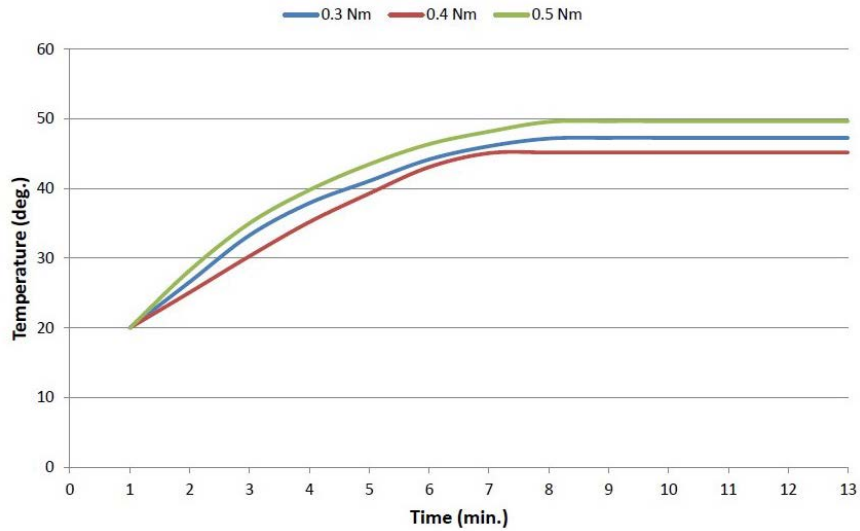


Figure 8 0.3 Nm, 0.4 Nm, 0.5 Nm Torque values applied to the cable joint surface temperature change

### 3.2. Mechanical Analysis and Findings

In the structural mechanical analysis module of COMSOL software, solid mechanics solutions were realized depending on torque values applied to the screw with its solid mechanics. The terminal connection consisting of a contact with a non-prestressed screw was subjected to a load group consisting of an axial force and a torque. Effective stress values on the screw top surface and the cable joint is also shown in Figure 9 when the torque force of 0.3 Nm is applied to the screw. The pressure change graph applied to the cable conductor contact surface of 0.3 Nm torque force is shown in Figure 10a. Arc length is the pressure that occurs between the screw axis and the contact point of the cable conductor and its unit is meter as shown in Figure 10b.

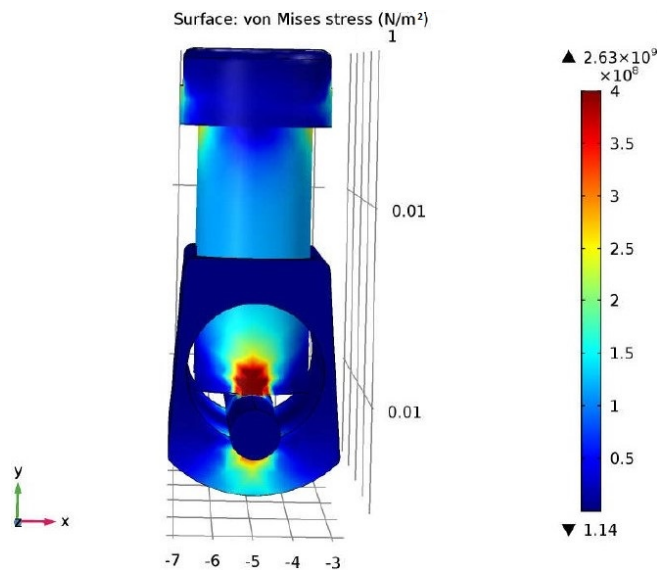
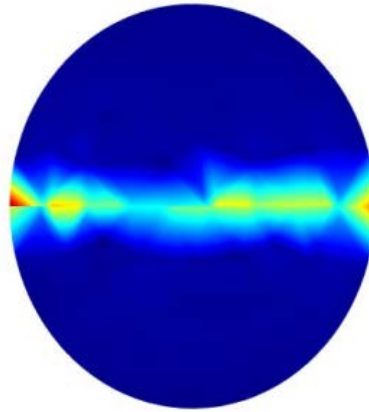
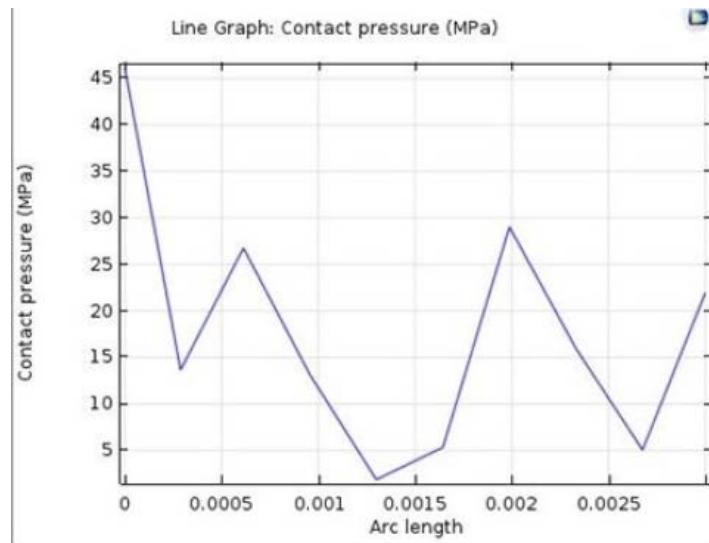


Figure 9 Effective stress at 0.3 Nm torque applied screw



(a)



(b)

Figure 10 a) Pressure on screw axis and cable conductor contact surface, b) Change of contact pressure according to arc length (meter)

### 3.3. Electrical Analysis and Findings

Results of electrical potential analysis for the model generated by the finite element method are shown in Figure 11. Analysis of current densities for the model generated by the finite element method are shown in Figure 12 and Figure 13. The cable current density-temperature relationship of the contact zone is shown in Figure 14. The terminal strips upper zone current density-temperature change graph is shown in Figure 15. The higher the temperature, the higher the conductor's resistivity. Therefore, the current density, which increased to a certain degree, fell after a certain temperature value [29]. This situation can be seen in Figure 15. The change in current density along the conductive surface is given in Figure 16. This situation refers to the current change of the conductor on the upper part of the conductor, which is screwed with a screw on the surface of the conductor. It was observed that there was more current change in the junction region.

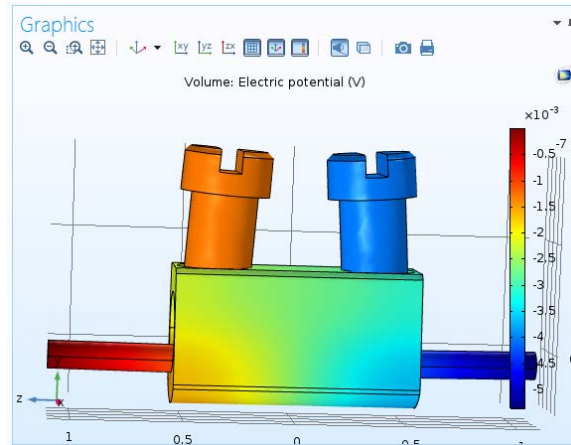


Figure 11 Electrical potential distribution for 14 A current value

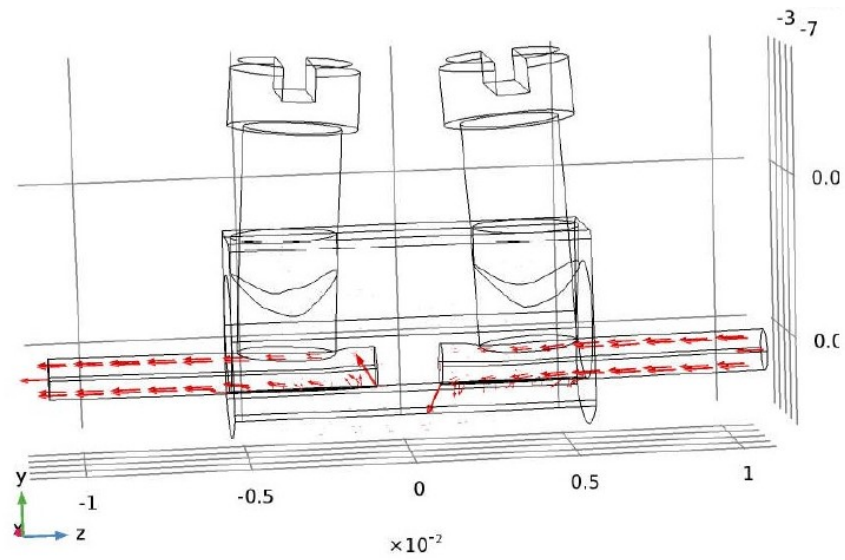


Figure 12 Distribution of current density ( $A/m^2$ ) on the model

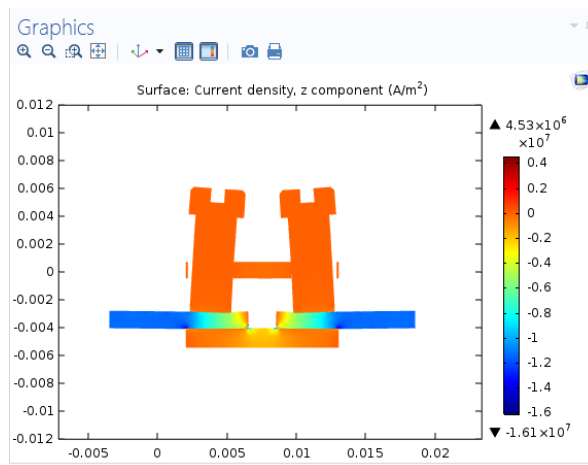


Figure 13 Analysis of the current density in the additional area of the model conductor

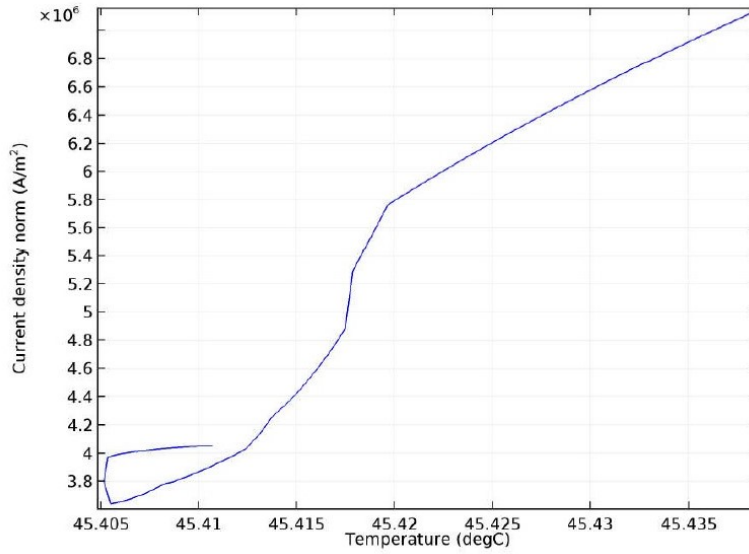


Figure 14 The relationship between the cable current density and temperature in the contact zone

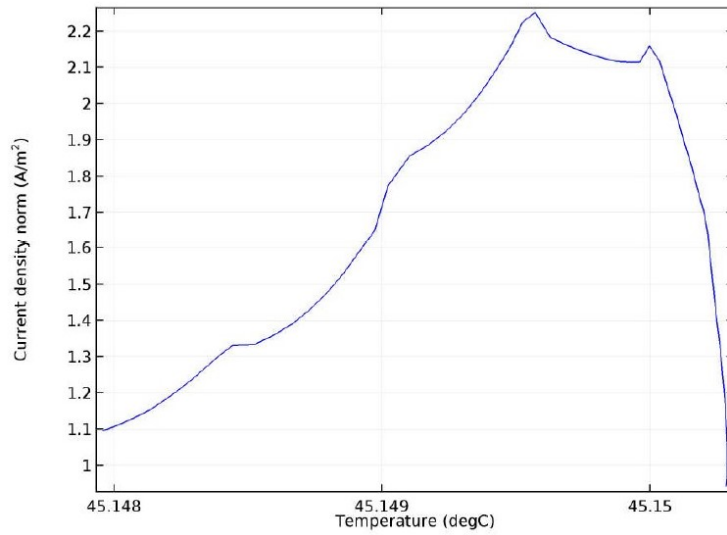


Figure 15 Screw top zone current density-temperature change graph

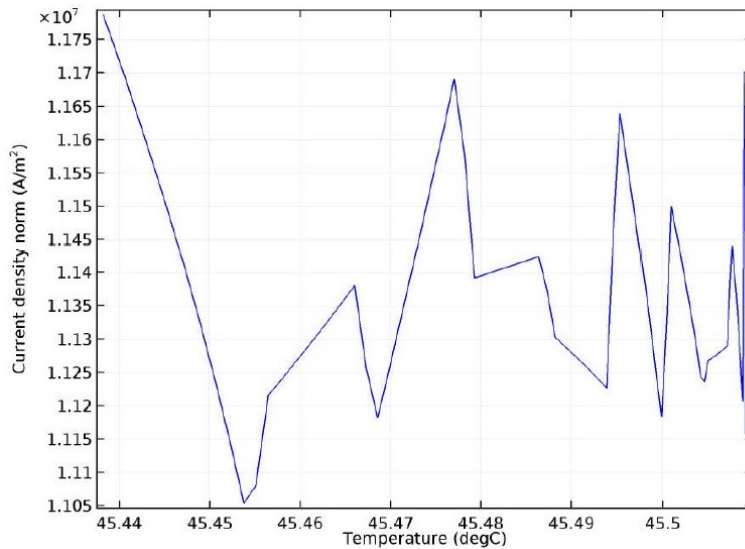


Figure 16 Change of current density along the conductor surface

### 3.4. Thermal Analysis and Findings

In the thermal analysis, electrical current applied to the model and temperature changes in the contact surface were examined. The temperature changes of the contact surface, cable conductor and screw surfaces at the 0.3Nm, 0.4Nm and 0.5Nm torque forces applied to the screw connection terminal at 14-amp constant current are shown in Figure 17, Figure 18 and Figure 19, respectively.

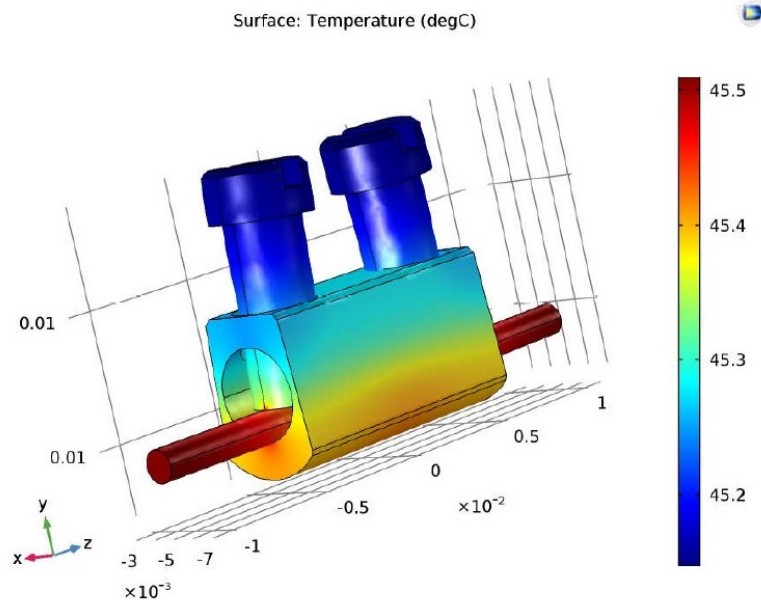


Figure 17 Temperature distribution for 0.3 Nm and 14 A status

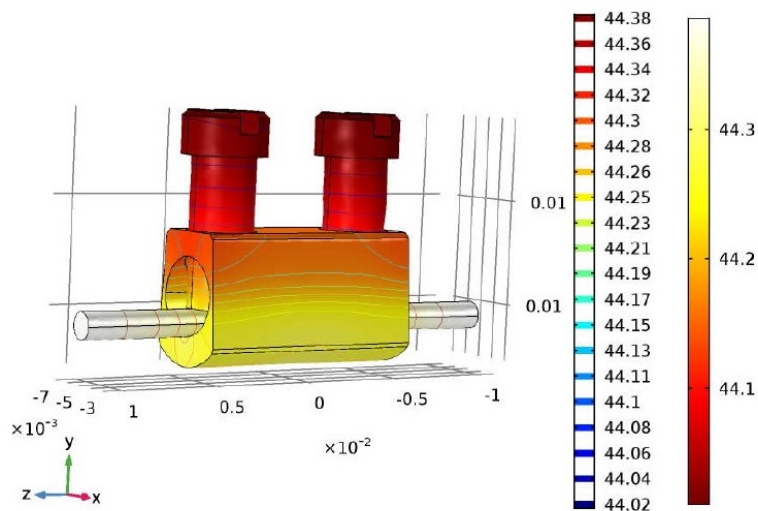


Figure 18 Temperature distribution for 0,4 Nm and 14 A status

## 4. Discussion and Conclusions

Unlike the electrical, thermal analysis studies performed in previous high voltage and current values [6,7,11], electrical, thermal and mechanical analysis of the cable joints made with screw connection terminals produced for use in 220-volt voltage values have been performed in this study. When the results obtained from the numerical model and the experimental study are compared, the experimental values of temperature changes in cable joints made with screws tightened by applying the 0.3 Nm, 0.4 Nm, 0.5 Nm force to copper conductor cable having 1.5 mm<sup>2</sup> cross-sectional area were same with the

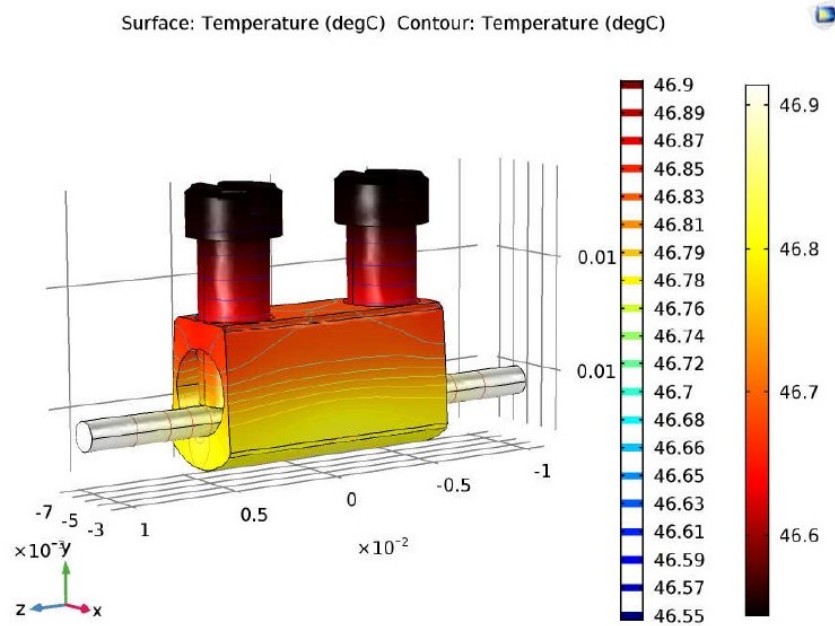


Figure 19 Temperature distribution for 0.5 Nm and 14 A status

temperature values obtained in the numerical values. A similar situation was observed in electrical and mechanical analyses (Figure 8, Figure 17, Figure 18, Figure 19.)

In order to achieve the minimum cable section deformation and the minimum temperature increase with the numerical model, the optimum torque value applied to the screw were found as 0.4 Nm. In the connection made by applying the torque value of 0.3 Nm, the deformation was less but the contact surface was small and the temperature increase value was higher than the torque value of 0.4Nm. In the connection made by applying the torque value of 0.5Nm, the contact surface was large but since the amount of deformation in the cable was large, the temperature increase was the highest due to shrinkage of cable cross-section.

According to the results of the electrical potential analyses for the model generated by the Finite element method shown in Figure 11, the difference in the electrical potential from the point where voltage source is connected to soil decreased. In Figure 17 and Figure 18, it was found that the current was more intense in the conductive cross-sectional area than in the additional region. These results were similar to those results obtained by Ruan [6] and colleagues. Figure 12 shows that the current density increases on the outer surface of the cable. Shazly Jehan H. et al. [7] studied three-phase, XLPE-insulated 220 kV, 340 MVA, single-core copper power underground cable components and the highest values of temperature distributions in the surrounding environment occur with the highest temperature seen in the regions where the work occurs. Zhou et al. [11] reported that the electric field concentration of the electric field was higher than the cable joint but the electric field concentration was found tended to intensify in the small cavities in the inserts.

In our study, it was shown that small cavities in the cable contact points at the copper cable ends in the connection zone increased the current density concentration (Figure 13). In additional applications done without the use of a Torque Screwdriver, different deformations may occur in the conductor cable according to physical characteristics of apply personnel. In this study, the importance of the use of tools that can apply standard forces to the screw such as torque screwdriver has implementation of explanation, while cable joints are made. In this study, it was specified that electrical and thermal analyzes of cable joints in high voltage transmission lines are important and it shows that it is important to make these analyzes for inserts made at 220 Volt voltage values. In this study, a 1.5 mm<sup>2</sup> cross-sectional copper cable was also analyzed. Analyzes will also be carried out for copper cable joints with different cross sections, which are commonly used in residential electrical installations, in our future

studies. In addition, same analysis can be performed for aluminum cables whose usage increases due to increase in copper conductor prices.

## References

- [1] I. Vera and L. Langlois, "Energy indicators for sustainable development," *Energy*, vol. 32, pp. 875-882, 2007.
- [2] M. Melikoglu, and M. Geothermal, "Energy in Turkey and around the World: A review of the literature and an analysis based on Turkey's Vision 2023 energy targets," *Renewable and Sustainable Energy Reviews*, vol. 76, pp. 485-492, 2017.
- [3] A.K. Rajak, and S.D. Kore, "Experimental investigation of aluminium–copper wire crimping with electromagnetic process: Its advantages over conventional process", *Journal of Manufacturing Processes*, vol. 26, pp. 57-66, 2017.
- [4] S. Madhusoodhanan, K. Mainali, A.K. Tripathi, A. Kadavelugu, D. Patel, and S. Bhattacharya, "Power loss analysis of medium-voltage three-phase converters using 15-kV/40-A SiC N-IGBT," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 4, pp. 902-917, 2016.
- [5] İ. B. Kara, "2013-2017 yılları arasında artvin il merkezinde meydana gelen bina yangınlarının incelenmesi", *Doğ. Afet Çev. Dergisi*, vol. 4, pp.105-114, 2018.
- [6] J. Ruan, Q. Zhan, L. Tang, and K. Tang, "Real-time temperature estimation of three-core medium-voltage cable joint based on support vector regression," *Energies*, vol. 11, no. 6, pp. 1405, 2018.
- [7] J.H. Shazly, M.A. Mostafa, D.K. Ibrahim, and E.E.A. El Zahab, "Thermal analysis of high-voltage cables with several types of insulation for different configurations in the presence of harmonics," *IET Generation, Transmission Distribution*, vol.11, pp.3439-3448, 2017.
- [8] P.Y. Wang, H. Ma, G. Liu, Z.Z. Han, D.M. Guo, T. Xu, and L.Y. Kang, "Dynamic thermal analysis of high-voltage power cable insulation for cable dynamic thermal rating," *IEEE Access*, vol. 7, pp. 56095-56106, 2019.
- [9] Q. Mei, W. Schoenmaker, S.H. Weng, H. Zhuang, C.K. Cheng, and Q. Chen, "An efficient transient electro-thermal simulation framework for power integrated circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, pp. 832-843, 2016.
- [10] D. Xiao, C. Zhou, Q. Ma, J. Lei, and X. Du, "Wearable Intelligent Warning System for Approaching High-voltage Electrical Equipment," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 12, pp. 9389-9397, 2020.
- [11] X. Zhou, J. Cao, S. Wang, Y. Jiang, T. Li, and Y. Zou, "Simulation of electric field around typical defects in 110kV XLPE power cable joints," *2017 International Conference on Circuits, Devices and Systems (ICCCDS)*, 2017.



- [12] B. Ren, C.T. Wu, and D. Lyu, "An h-adaptive meshfree-enriched finite element method based on convex approximations for the three-dimensional ductile crack propagation simulation," *Computer Aided Geometric Design*, vol. 76, pp. 101795, 2020.
- [13] E.A. Rodrigues, O.L. Manzoli, L.A. Bitencourt, T.N. Bittencourt, and M. Sánchez, "An adaptive concurrent multiscale model for concrete based on coupling finite elements," *Computer Methods in Applied Mechanics and Engineering*, vol. 328, pp. 26-46, 2018.
- [14] X. Song, V. Pickert, B. Ji, R.T. Naayagi, C. Wang, and Y. Yerasimou, "Questionnaire-Based Discussion of Finite Element Multiphysics Simulation Software in Power Electronics," *IEEE Transactions on Power Electronics*, vol.18, pp. 7010-7020, 2018.
- [15] G. Dagastine, "Numerical simulation-based topology optimization leads to better cooling of electronic components in toyota hybrid vehicles," *Special Advertising Section to IEEE Spectrum: Multiphysics Simulation*, 2012.
- [16] M.F. Moulki, and M. Khashab, "Haptic Servo System", *Dissertation*, 2015.
- [17] A. Hassani, M. Veyskarami, A. Al-Ajmi, and M. Masihi, "A modified method for predicting the stresses around producing boreholes in an isotropic in-situ stress field," *International Journal of Rock Mechanics and Mining Sciences*, vol. 96, pp. 85-93, 2017.
- [18] Y.L. Hao, D.L. Gong, T. Li, H.L. Wang, J.M. Cairney, Y.D. Wang, and R. Yang, "Continuous and reversible atomic rearrangement in a multifunctional titanium alloy," *Materialia*, vol. 2, pp. 1-8, 2018.
- [19] R.A. Heindl, and L.E. Mong, "Young's Modulus of Elasticity, Strength and Extensibility of Refractories in Tension," *Part of Journal of Research of the National Bureau of Standards*, vol. 17, pp.469-470, 1936.
- [20] K. Kese, P.A. Olsson, and A. Holston, "Broitman, E. High temperature nanoindentation hardness and Young's modulus measurement in a neutron-irradiated fuel cladding material," *Journal of Nuclear Materials*, vol. 487, pp. 113-120, 2017.
- [21] I. Zlotnikov, E. Zolotoyabko, and P. Fratzl, "Nano-scale modulus mapping of biological composite materials: Theory and practice." *Progress in Materials Science*, vol. 87, pp. 292-320, 2017.
- [22] X. Zhao, C. Liu, C. Xu, G. Xu, Y. Zhang, S. Tan, and Y. Han, "The effects of Ca<sup>2+</sup> and Y<sup>3+</sup> ions co-doping on reducing infrared emissivity of ceria at high temperature," *Infrared Physics Technology*, vol. 92, pp.454-458, 2018.
- [23] S. Çevikalp, "Yüksek Basıncılı Sistemlerde Cıvata Üzerindeki Gerilmenin Burç Boyuna Göre Analizi ve Optimum Burç Boyunun Analiz Edilmesi," *Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, İstanbul*, 2016.
- [24] S.S. Rao, "The finite element method in engineering", *Butterworth-heinemann*, 2017.
- [25] M.N. Özişik, H.R. Orlande, M.J. Colaço, and R.M. Cotta, "Finite difference methods in heat



transfer”, *CRC press*, 2017.

- [26] D. Pepper and J. Heinrich, “The Finite Element Method: Basic Concepts and Applications with MATLAB,” *MAPLE, and COMSOL*, *CRC Press*, 2017.
- [27] L. Monforte, M. Arroyo, J.M. Carbonell, and A. Gens, “Numerical simulation of undrained insertion problems in geotechnical engineering with the Particle Finite Element Method (PFEM),” *Computers and Geotechnics* vol. 82, pp. 144-156, 2017.
- [28] L.A. Bitencourt, O.L. Manzoli, P.G. Prazeres, E.A. Rodrigues, and T.N. Bittencourt, “A coupling technique for non-matching finite element meshes,” *Computer Methods in Applied Mechanics and Engineering*, vol. 290, pp. 19-44, 2015.
- [29] A. Dalcı, O. Çetin, C. Ocak, and F. Temurtaş, “Bir Elektromanyetik Fırlatıcı Bobininde Mermiye Etkiyen Kuvvetin Çok Katmanlı Sinir Ağı ile Kestirimi,” *Sakarya University Journal of Computer and Information Sciences*, vol. 1, no. 3, pp. 1-10, 2018.

# A V-Model Software Development Application for Sustainable and Smart Campus Analytics Domain

 Onur Dogan<sup>1</sup>,  Semih Bitim<sup>2</sup>,  Abdulkadir Hizirolu<sup>3</sup>

<sup>1</sup>Corresponding Author; Izmir Bakircay University, Department of Industrial Engineering, 35665, Izmir, Turkey; onur.dogan@bakircay.edu.tr; +905324915885

<sup>2</sup>Izmir Bakircay University, Department of Management Information Systems, 35665, Izmir, Turkey; semih.bitim@bakircay.edu.tr

<sup>3</sup>Izmir Bakircay University, Department of Management Information Systems, 35665, Izmir, Turkey; kadir.hizirolu@bakircay.edu.tr

Received 14 February 2021; Revised 24 February 2021; Accepted 3 March 2021; Published online 16 March 2021

## Abstract

As small cities, university campuses contain many opportunities for smart city applications to increase service quality and efficient use of public resources. Enabling technologies for Industry 4.0 play an important role in the goal of building a smart campus. An earlier work of the authors proposed a framework that was proposed for the development of a smart campus applications. It was the digital transformation process of Izmir Bakircay University which is a newly established university in Turkey. This study is related to the final part of the developed framework. It aims to systematically develop a software for a sustainable and smart campus. V-model software development methodology was followed in the study. The methodology was applied for the corresponding stage which mainly includes real-time analytics, monitoring, reporting and performance management. The data flow diagrams were presented at three levels, a context diagram for a basic form of the system and parent diagram for the detailed software modules, and a child diagram for a selected module. This study can guide to the following researches to create a smart campus framework and a real-time analytics software.

**Keywords:** real-time analytics, smart campus, software development methodology, data flow diagram, context diagram

## 1. Introduction

Many cities around the world have adapted the concept of the smart city to improve the quality of life of people, to ensure energy efficiency, and to improve management services. The smart city uses advanced technologies to provide a variety of services. The campuses, which can be seen as small cities, are also open to many improvements. Previous studies focused on the application of high-level smart skills to apply the concept of smart campus. Sari et al. [1] described the design of an IoT-based smart campus program that focuses on smart parking, smart room, and smart education. An integrated platform was presented for this service. Here, Wi-Fi was used to connect different sensors and cameras associated with the platform. However, applications related to learning predictions were mainly aimed at distance learning. Majeed and Ali [2] represented a range of opportunities to include the smart concept on campuses, especially parking, security, classroom support, and education. Some studies aimed at an intelligent analysis of teaching activities using game-based approaches [3], multimedia conferences [4] and smartphone apps [5]. In these studies, various smart approaches (by applying IoT and data processing) were developed for different purposes compatible with classical smart city applications. In this area, Alvarez-Campana et al. [6] developed the IoT platform on a university campus to monitor the environment and people. In terms of smart mobility, Toutouh et al. [7] developed a mobility forecasting mechanism at the University of Malaga campus, and Hannan et al. [8] focused on IoT-supported disaster management for smart campuses. Although many related studies applied various smart methods on campuses, putting the real-time analytics solutions in the center still requires a scientific-based system development methodology. Therefore, in this study, a three-stage framework shown in Figure 1 was developed that includes improvements in various application areas for efficient consumption of public resources. The first stage (single facility) and second stage (extended facility and environment) are an infrastructure projects and involve intelligent applications based on the Internet of things (IoT) to be

built across campus. The third stage contains real-time data analytics, monitoring, reporting, and performance measurement elements.

For the sustainability of the smart campus, the third stage of the framework is supported by a real-time data analytics, monitoring, reporting, and performance measurement module rather than merely developing a smart system. A real-time system is mainly built for safety issues or saving of resources (time, money, etc.) [10]. It is commonly described as a system in which the accuracy of output depends on both the precision of the logical outcomes and the point in time at which the outcomes are performed [11]. The design of a real-time system is pretty compelling and it is considerably different from non-real time system design. Real-time systems require to meet both the functional necessities and several performance needs such as timeliness, availability and fault tolerance [12]. Because defects in a real-time system cause a catastrophic effect, these systems are expected to be extremely dependable. Dependability can be embedded in the developed system by following both hardware design procedures and software development life cycle methodology (SDLC). This study focuses on software development for the third stage of the proposed framework given in Figure 1. For the main purpose of this study, a software is needed for the sustainable and smart campus framework to analyze the usage of public resources via the collected data at the first two stages across all application areas, which serves as a basis for the main purpose.

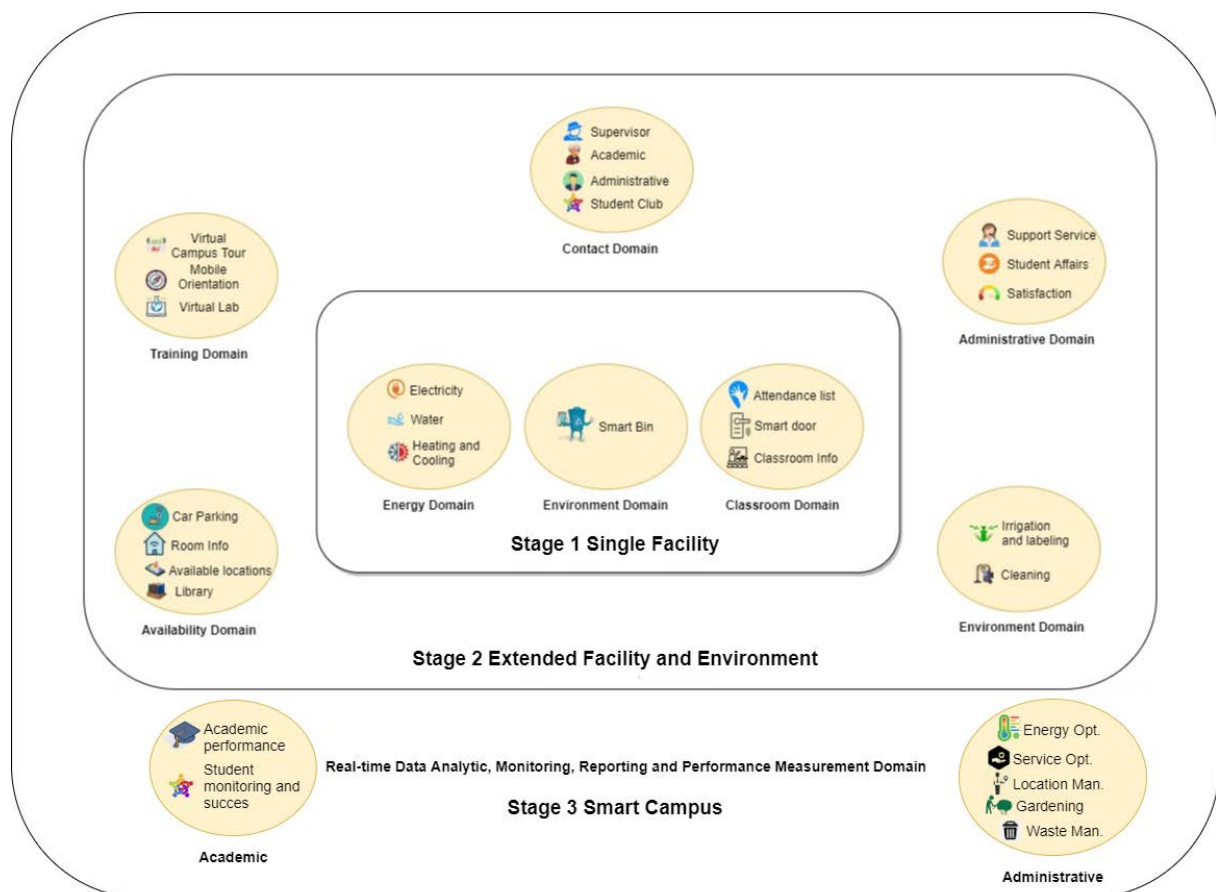


Figure 1 Sustainable and smart campus framework [9]

## 2. Selection of Software Development Methodology

A software development methodology (SDM) is typically a sequence of phases that present a model for software development and life cycle [13]. An SDM process aims to create high-quality, effective and affordable software. Waterfall and Agile models are two different types of SDM. The Waterfall model starts with well-known guidelines and specified system elements whereas the Agile model starts with a less convincing model and then performs modifications as required during the process. The advantages

of the waterfall design are high understandability, easy handling. Besides, it is more powerful than the agile model when quality issues are more significant than cost. The waterfall model can be adapted when specifications are clear and the product description is permanent [10]. V model, which is a kind of waterfall model, was adapted in this study. It highlights on verification and validation (V&V) of the software at each phase. This model is used when i) The whole specifications are ready before beginning the project, ii) Systems need profoundly reliable software, and iii) Technology and alternative solutions are identified. All steps to be performed while developing new software are designed within the frame of the V-model was shown in Figure 2. The figure defines the order of implemented steps and the results presented during software development. In contrast to using a linear design, the process levels are bent upwards after the software implementation (coding) phase to appear the V-shape. The V-model illustrates the connections between each step of the development cycle and its linked testing step. Whereas the left part of the 'V' outlines the exploration of software specifications and detailed design, the right part of the 'V' describes the combination and validation of the parts. Therefore, the V-model is also named as a verification and validation model.

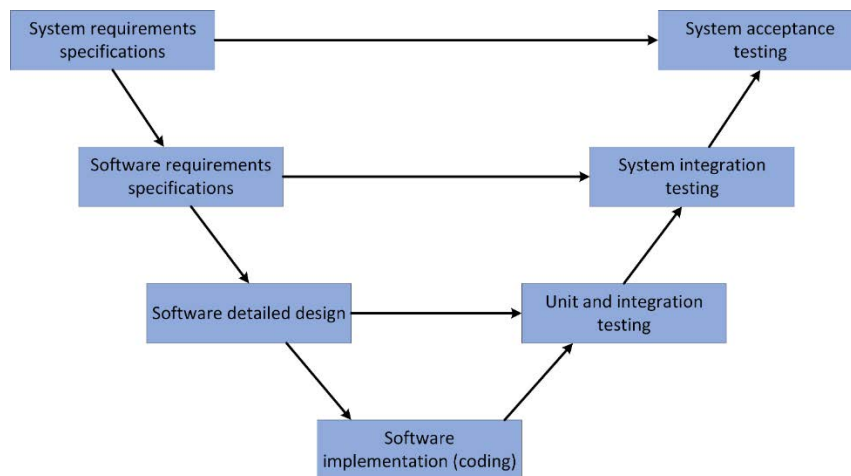


Figure 2 V-model [14].

### 3. Adaptation of the Select Methodology

Many real-time systems consist of various elements such as sensors, IT systems, and actuators [12]. In the real-time analytics framework given in Figure 3, the first phase consists of collecting relevant data from external units, taking the system data into its database and backing up this data against any negative situation. In the second phase, the real-time data analytics using the collected data in the servers are included. The third phase provides outputs to various users in the form of reports, dashboards and maps prepared as a result of these analytics efforts. In the light of these outputs, users can direct to the relevant response units.

Although V-model is the guide for this study to develop the software, only the first two steps (system requirements specification and software requirements specification) and their tests pertaining to these steps (System acceptance testing and System integration testing) were covered in the study. In the System Requirements Specification step, user expectations and corresponding requirements were determined. It should be maintained by communicating closely with expert users and end-users. This phase is also called Requirement Gathering. With consecutive meetings, surveys, areas where system users need functionalities were revealed. The data to be obtained from the relevant academic and administrative units were provided and the advantages of the included fields were specified. Thanks to

the interviews conducted with expert system users, all needs were ranked according to the advantages to be obtained and the ones that had high priority were included in the analytics framework.

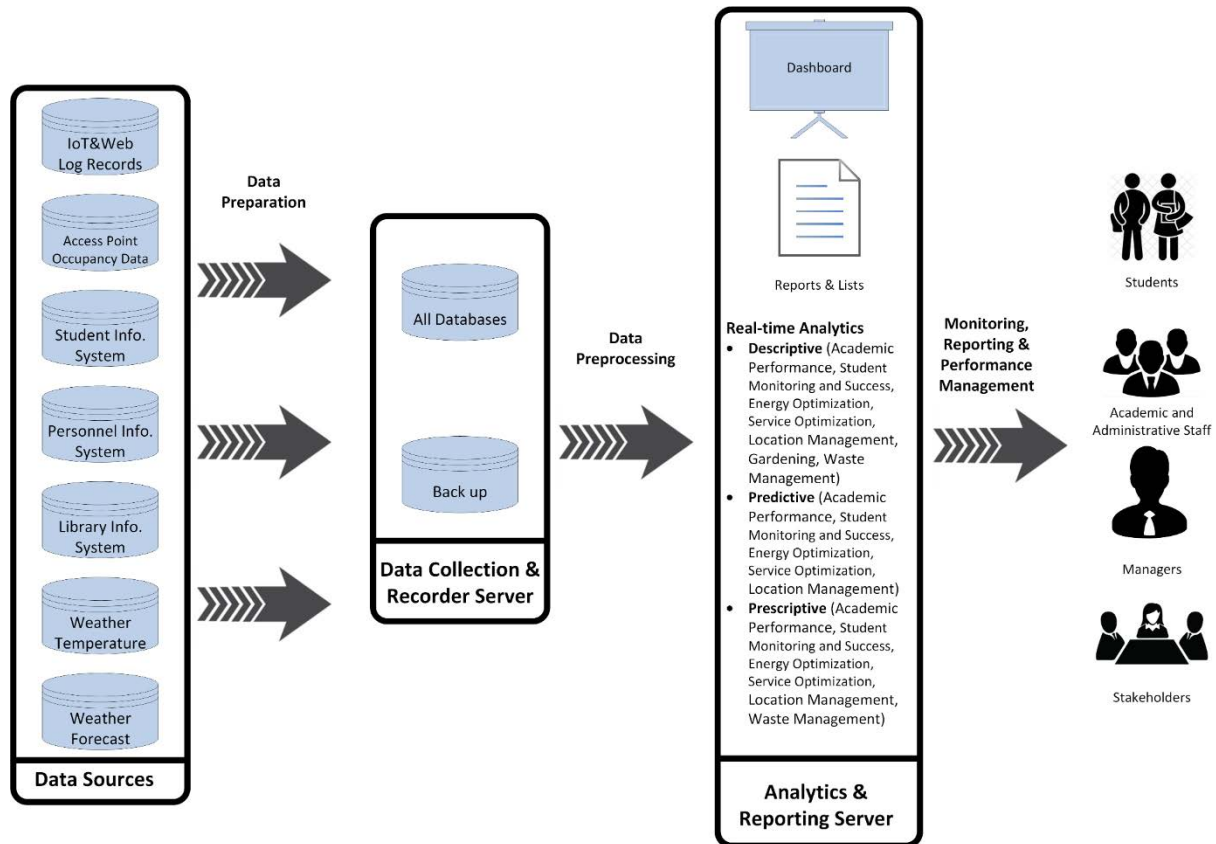


Figure 3 Real-time analytics framework

The System Acceptance Testing step must be performed before the next step. The requirements of a real-time system for the sustainable and smart campus were grouped into two categories, functional and technical requirements. Whether the functional requirements determined in the System Requirement Specifications step were met or not was tested in a pilot study with the participation of end-users. For technical requirements, it was tried to determine whether the scale that can meet the needs on the virtually created infrastructures with the sample datasets of the related fields were determined correctly or not. Also, the technical equipment was tested on different scales.

The Software Requirement Specifications step determines which software technologies the system will be prepared to use. Since there is a need to collect and process data from many different types of devices and systems, hybrid software technologies were such as integrated into the system to work in harmony with each other.

System Integration Testing step was carried out via prepared figures and tables whether the areas included in the system were designed to include all functionalities or not. A series of data flow diagrams were created to test system integration. It has been observed that the communication technologies with the other systems were selected correctly and can be provided without any problems.

Data flow diagrams model at least 3 levels, which are context, parent (Level 0) and child (Level n). The context diagram represents the most basic form of the system. There is information about which data included in the system are obtained and to whom the data received is processed and presented as output. In the context diagram, all the resources that provide data to the system are firstly specified, and what kind of data will be obtained from these resources (Ex: Access Point Occupancy Data - Spatiotemporal Data). Graphs, Alerts, Reports, Lists, etc. for all domains required by the end-users of the information to be generated from this data. Figure 4 presents the context diagram modeling.

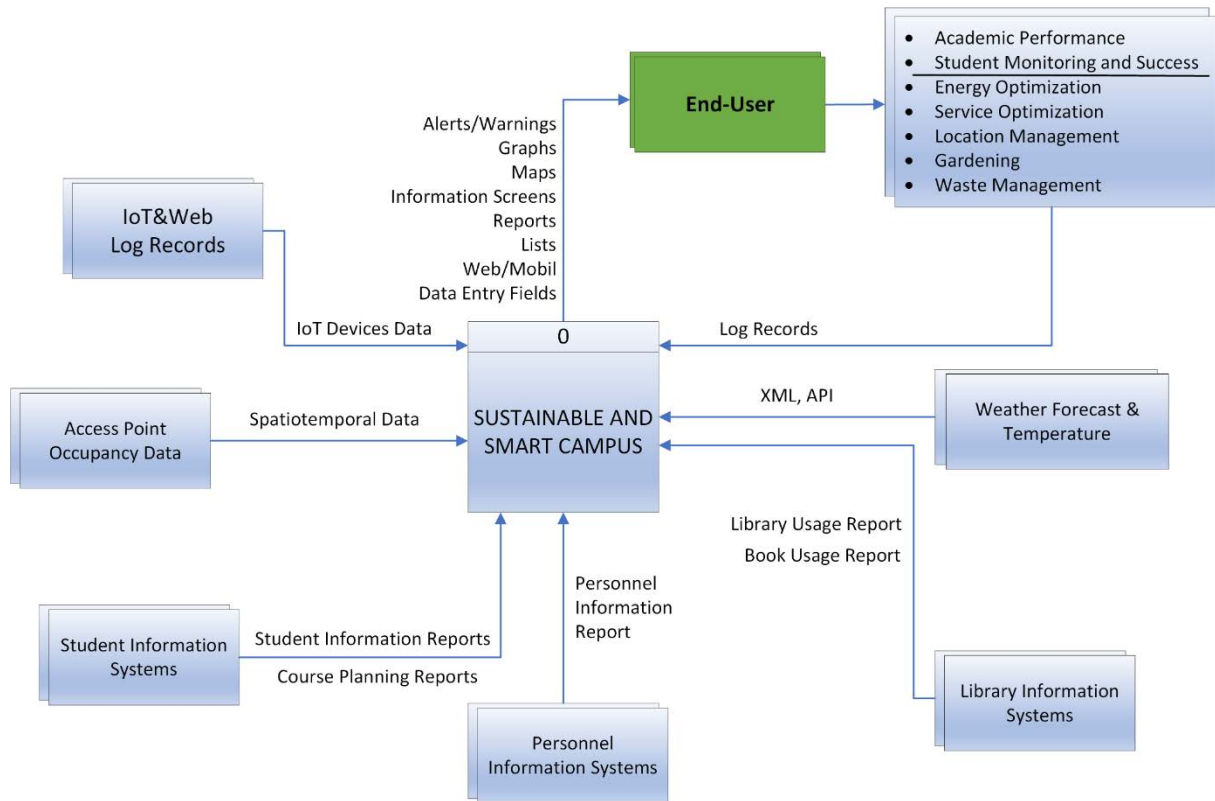


Figure 4 Context diagram smart campus

With regards to potential reports or results that an end-user may require, two main categories of these outputs were specified, namely academic and administrative domains. Table 1 presents the types of analytics that an output can be belong to and some potential examples pertaining to academic and administrative domains.

When Table 1 is examined, one can easily see that various analytics instruments can be applicable for different application domains. For instance, under the energy optimization sub-domain, usage reports of heating/cooling and lighting systems (Descriptive), prediction for the energy consumption of the next period (Predictive), achievement an objective under certain constraints to find the optimum working time (Prescriptive) can be applied. Also, location management may include a prediction analysis to forecast the length of the catering queue with respect to the type of food (Predictive). A stochastic model can be studied to reduce catering queue (Prescriptive). Additionally, under the waste management topic, a prescriptive analysis can be done. When more than one bin is full at the same time, route planning according to the location of the cleaning person. In summary, descriptive type of analytics include various form of static and dynamic reports as well as outputs in multi-dimensional form and context-dependent visual analytics displays using customized dashboards and alerts. While predictive analytics cover several applications by the use of different data mining methods, optimization techniques as well as simulation can be utilize within the context of prescriptive analytics.

The Parent Diagram of the sustainable and smart campus system was presented in Figure 5. Each module in the smart campus model was conceptually designed and the connections between these modules were presented without contradicting the context diagram. The software has eight basic modules. In the parent diagram, the interaction of each module with other modules is presented together with the data tables (data warehouse) designed. Link, End-User and User/Group/Role modules are essential elements in any software that processes real-time data and where communication among users is imperative.

Data pool, data processing and analytics, Data manipulation, Dashboards, Reports/Lists are modules that are offered to the real-time data analytics stage for a smart campus system. Data pool is the module where the data collected from all data sources are collected in tables after the data preparation process



Table 1 Analytics in context diagram

Domain	Sub-domain	Analytics Type	Example
Academic	Academic Performance	Descriptive	Academic resume report
		Predictive	Prediction of university's academic success for next year
		Prescriptive	Matches for multidisciplinary academic researches
	Student Monitoring and Success	Descriptive	Attendance list report, course success graph
		Predictive	Student success/fail prediction
		Prescriptive	Course selection suggestions
Administrative	Energy Optimization	Descriptive	Reports of heating/cooling and lighting
		Predictive	Prediction for the energy consumption of the next period
		Prescriptive	Achievement an objective under certain constraints to find the op-timum working time
	Service Optimization	Descriptive	Service location usage report, satisfaction alerts
		Predictive	Prediction of supply needs
		Prescriptive	A stochastic model can be studied to reduce catering queue
	Location Management	Descriptive	Availability reports, occupancy alerts
		Predictive	Prediction of the length of the catering queue with respect to the type of food and the curriculum
		Prescriptive	Assignment courses to classes
	Gardening	Descriptive	Reports for water consumption amount reports, visualization and alerts
	Waste Management	Descriptive	Waste amount reports, visualization and alerts
		Prescriptive	When more than one bin is full at the same time, route planning according to the location of the cleaning person

is passed. Dashboard is the front-end module where all system users access the interfaces according to their authorizations. User/Group/Role is the module where system users and credentials, as well as the groups that these users are connected to and role definitions that determine which interfaces this person or groups will access. Data Manipulation is the module that allows editing of missing, excessive or wrong data in the data pool. It is an accessible module only by users with special administrative privileges. Link is the module where all data sent by the users of the system to be recorded to the system are taken as input and sent to the data pool (Feedback, Alerts, suggestions, questions, etc.). Data Processing&Analytics is the module that the information requested by the end users and that can be obtained through various processes and calculations are produced in the system and that allows the results to be reported on the dashboard. Reports&Lists is the module where the information coming from the Data Processing&Analytics module is presented to users in grouped reports on a domain basis, and the data in the data pool is grouped according to various features.

Figure 6 depicts the child diagram of the data pool process as an example of detailed parent diagram. The main task of the data pool module is to send data from external sources to operational systems and archive this data for the use of end-users. With this module, all external data required for managing a smart university are provided. IoT device data, spatiotemporal data, student and course information, personnel information, library and book usage data, and meteorological records are the main external data sources. The end-user can view the data with various filters from the data pool processes. Considering that external data is very important especially for decision support systems [15], it can be said that the data pool module is one of the most critical elements of the real-time analytics system for the smart campus.

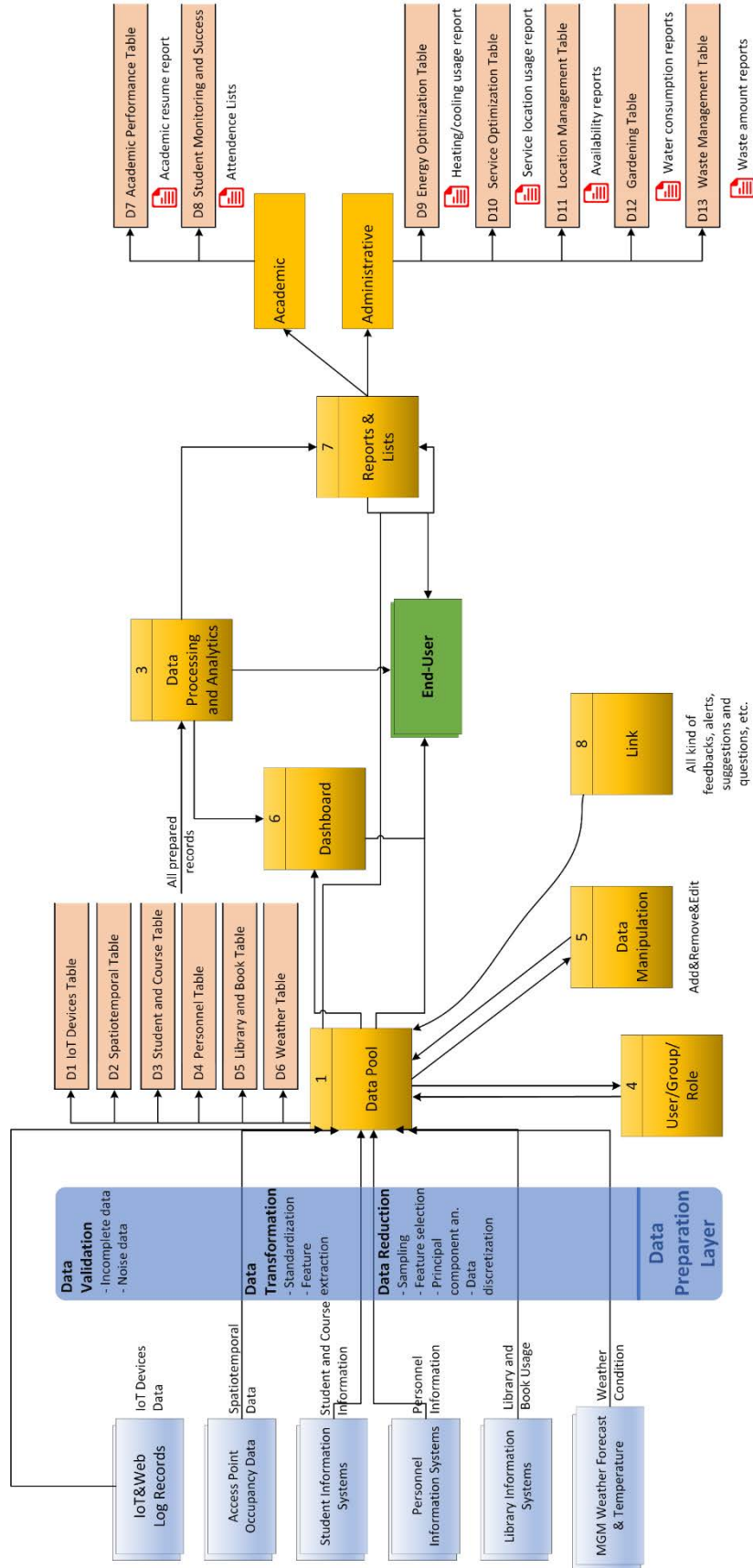


Figure 5 Parent diagram for smart campus



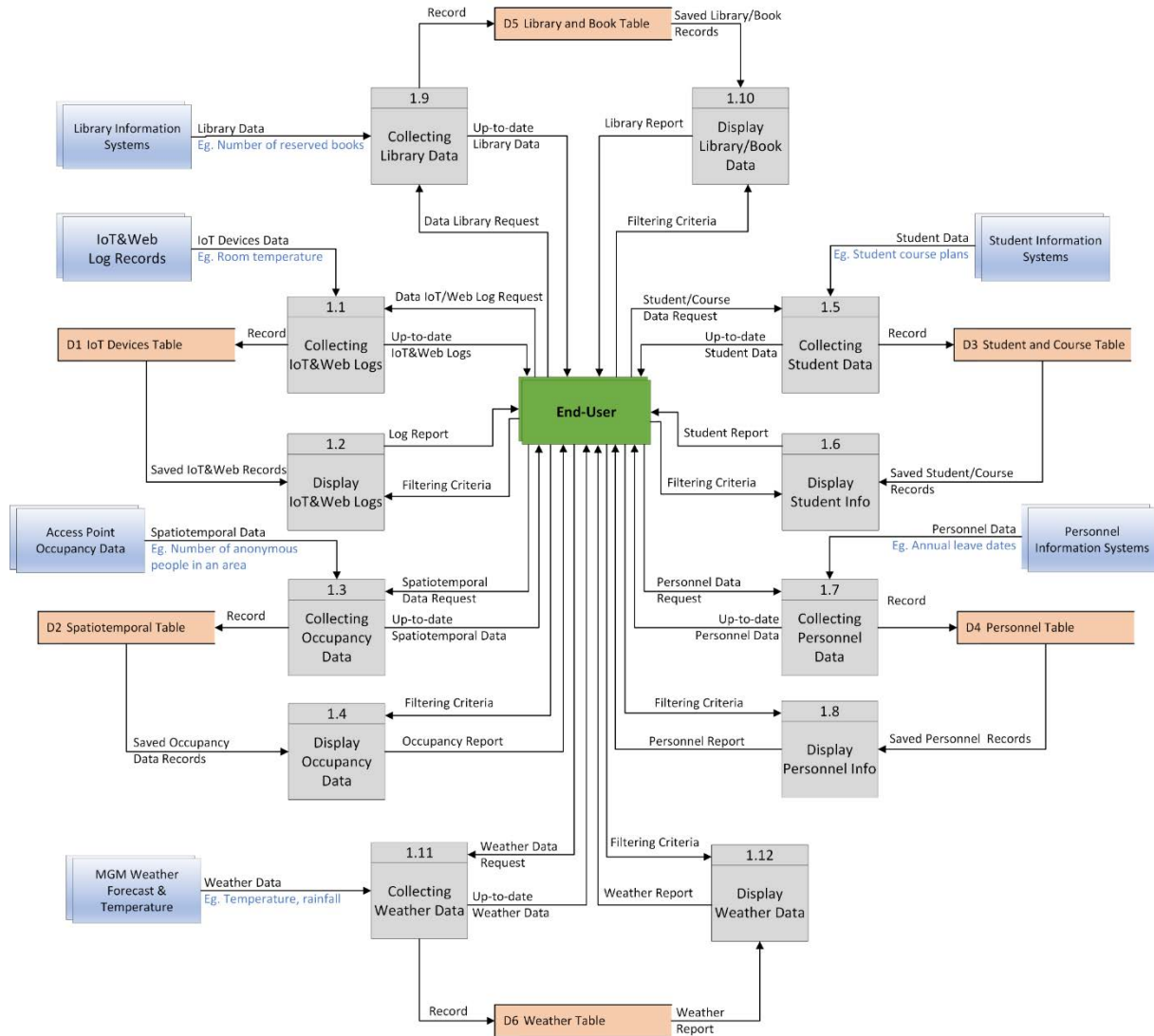


Figure 6 Child diagram for data pool module

#### 4. Conclusions



In this work, a software development methodology (SDM) that belongs to the academic and administrative domains was applied for a smart campus real-time analytics framework. After introducing software development methodologies, V-model was chosen because it was more suitable for the implementation area. The first two steps, which are system requirements specification and software requirements specification and their tests relate to these steps (System acceptance testing and System integration testing) are the focus of this study. This study can guide to the following researches to create a smart campus framework and a real-time analytics software. Within the scope of the study three-level data flow diagrams were developed for the analytics framework. Also, variety of analytics examples were presented for the use of the main application domains/sub-domains related to the presented framework. This study can be a starting guidance for the applications and developers of such framework within the context of a smart campus real-time analytics software implementation.

Upcoming research will have a detailed focus on the remaining steps of the V-model on the application context through carrying and the work related to software design and implementation (coding) phase.

## References

- [1] M. W. Sari, P. W. Ciptadi, and R. H. Hardyanto, "Study of smart campus development using internet of things technology," *IOP Conference Series: Materials Science and Engineering*, vol. 190, no. 1, p. 012032, 2017.
- [2] A. Majeed and M. Ali, "How internet-of-things (iot) making the university campuses smart? QA higher education(QAHE) perspective," in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 646–648, IEEE, 2018.
- [3] X. Zhai, Y. Dong, and J. Yuan, "Investigating learners' technology engagement-a perspective from ubiquitous game-based learning in smart campus," *IEEE Access*, vol. 6, pp. 10279–10287, 2018.
- [4] W. Zhang, X. Zhang, and H. Shi, "MMCSACC: A multi-source multimedia conference system assisted by cloud computing for smart campus," *IEEE Access*, vol. 6, pp. 35879–35889, 2018.
- [5] Y.-B. Lin, L.-K. Chen, M.-Z. Shieh, Y.-W. Lin, and T.-H. Yen, "Campustalk: IOT devices and their interesting features on campus applications," *IEEE Access*, vol. 6, pp. 26036–26046, 2018.
- [6] M. Alvarez-Campana, G. L'opez, E. V'azquez, V. Villagr'a, and J. Berrocal, "Smart cei moncloa: An iot-based platform for people flow and environmental monitoring on a smart university campus," *Sensors*, vol. 17, no. 12, p. 2856, 2017.
- [7] J. Toutouh, J. Arellano, and E. Alba, "Bipred: a bilevel evolutionary algorithm for prediction in smart mobility," *Sensors*, vol. 18, no. 12, p. 4123, 2018.
- [8] A. Hannan, S. Arshad, M. A. Azam, J. Loo, S. H. Ahmed, M. F. Majeed, and S. C. Shah, "Disaster management system aided by named data network of things: Architecture, design, and analysis," *Sensors*, vol. 18, no. 8, p. 2431, 2018.
- [9] Z. N. Kostepen, E. Akkol, O. Dogan, S. Bitim, and A. Hiziroglu, "A framework for sustainable and data-driven smart campus," in *ICEIS*, pp. 746–753, 2020.
- [10] M. Manimaran, A. Shanmugam, P. Parimalam, N. Murali, and S. S. Murty, "Software development methodology for computer based I&C systems of prototype fast breeder reactor," *Nuclear Engineering and Design*, vol. 292, pp. 46–56, 2015.
- [11] M. Sanfridson, "Timing problems in distributed real-time computer control systems," tech. rep., Mechatronics Lab, Dept. of Machine Design, Royal Inst. of Technology, Stockholm, 2000.
- [12] L. Wang, "Get real: Real time software design for safety-and mission-critical systems with high dependability," *IEEE Industrial Electronics Magazine*, vol. 2, no. 1, pp. 31–40, 2008.
- [13] A. M. Davis, E. H. Bersoff, and E. R. Comer, "A strategy for comparing alternative software development lifecycle models," *IEEE Transactions on Software Engineering*, vol. 14, no. 10, pp. 1453–1461, 1988.
- [14] R. S. Pressman, *Software engineering: a practitioner's approach*. Palgrave macmillan, 2005.
- [15] K. C. Laudon, J. P. Laudon, et al., *Management information systems*. Pearson Upper Saddle River, 2015.

## Effect of the Chaotic Crossover Operator on Breeding Swarms Algorithm

 Hüseyin Demirci<sup>1</sup>,  Nilüfer Yurtay<sup>2</sup>

<sup>1</sup>Corresponding Author; Sakarya University, Faculty of Computer and Information Sciences; huseyind@sakarya.edu.tr; +90 264 295 69 14

<sup>2</sup> Sakarya University, Faculty of Computer and Information Sciences; nyurtay@sakarya.edu.tr;

Received 18 September 2020; Revised 9 November 2020; Accepted 12 March 2021; Published online 18 March 2021

### Abstract

In this paper we present effect of the chaotic crossover operator with different chaotic maps on the metaheuristic search algorithm Breeding Swarms algorithm which is the Particle Swarm Optimization's one of the genetic algorithm hybrid form. Some of the many optimization problems could have too many local extrema. Most of the time optimization algorithms could stuck on these extrema therefore these algorithms could have trouble with finding global extremum. To avoiding local extrema and conduct better search on search space, a chaotic number generator is used on Breeding Swarms algorithm's most of the random procedures. To test efficiency and randomness of the chaotic crossover operator, different chaotic maps are used on the Breeding Swarm algorithm. Test and performance evaluations are conducted on Multimodal and unimodal benchmark functions. This new approach showed us that modified Breeding Swarms algorithm yielded slightly better results than Particle Swarm Optimization and original Breeding Swarms algorithms on tested benchmark functions.

**Keywords:** optimization, chaos, particle swarm optimization, hybrid algorithm

### 1. Introduction

Optimization process is the selection of a value obtained from a dataset that its principles are defined before the process. Optimization problems can be solved by running many algorithms. No matter how on fast computers, these algorithms spend too much time to reach the exact solutions. Because of that, nature inspired meta-heuristic algorithms were developed. These algorithms generally try to find approximate solution rather than exact solution and run faster than the other algorithms. However, they have still many deficiencies. To solve these downsides, algorithms are combined with other algorithms that leads to the development of hybrid algorithms. Meta-heuristic algorithms are used in many optimization problem types such as single objective, multi objective and multi objective discrete problems. A helpful review about selecting proper algorithm for these problem types were made by Qi Liu et al. [1].

Particle Swarm Optimization (PSO) is a swarm based optimization algorithm which developed by Eberhart and Kennedy in 1995 [2]. This algorithm is very easy to implement and to code with any kind of programming language. PSO particles have its own memory. It can remember past experiences and share this experiences with other particles. With shared experience all particles towards to best solution and search other possibilities through that way. PSO and GA algorithms used in many real-world problems such as wireless sensor optimization and electrical load forecasting [3],[4].

This procedure may seem to be efficient but some bad particles could stuck on local extrema and waste computational time. To handle this problem, PSO has been hybridized with Evolutionary Algorithms such as Genetic Algorithm (GA) and Differential Evolution algorithm.

One of these hybridized algorithm is Breeding Swarms (BS) algorithm [5]. This algorithm combines GA with PSO and proposes new crossover operator Velocity Propelled Averaged Crossover (VPAC). PSO and GA uses aimed random search for optimization. While the algorithm's GA part conducts global search, PSO part makes the local search. Thus, the algorithm could escape local extrema. The lack of the algorithm is that GA part can stuck on local extrema.

In this paper we present Velocity Propelled Chaotic Crossover operator (VPCC) for improvement of this algorithm's GA part. GA and chaotic maps are used with together in many applications such as encrypting [6]. This crossover operator takes advantage of the unpredictability and randomness of the chaotic systems. With these new attributes BS algorithm escapes all of the local extrema and gets better results than the original algorithm.

## 2. Background and Related Works

### 2.1 Particle Swarm Optimization

Particle Swarm Optimization is a swarm based optimization algorithm [2]. This algorithm inspired from bird's behaviour of search for food. Main idea of the algorithm is that birds in the swarm inform the found food resource to each other and then the birds move toward to best food resource in the found resources. While birds moving to best resource, they search the search space for new food resources so that most of the search space will be scanned for alternate resources and this may lead to discovery of approximate best solution. Mathematical expression of particles movement given by equation (1)

$$x_i(t+1) = x_i(t) + v_i(t) \quad (1)$$

where  $x_i$  is the current value of  $i^{\text{th}}$  particle optimization variable,  $v_i$  is the velocity vector,  $t$  is the current iteration.

In original PSO algorithm velocity vector calculated as given in equation (2)

$$v_{ij}(t+1) = v_{ij}(t) + c_1 r_{1j}(t)[y_{ij}(t) - x_{ij}(t)] + c_2 r_{2j}(t)[\hat{y}_j(t) - x_{ij}(t)] \quad (2)$$

where  $v_{ij}(t)$  is the velocity vector of the  $i^{\text{th}}$  particle at dimension  $j=1\dots n$  in  $t^{\text{th}}$  iteration,  $x_{ij}(t)$  is the position of the  $i^{\text{th}}$  particle at  $j^{\text{th}}$  dimension in  $t^{\text{th}}$  iteration,  $y_{ij}(t)$  is the best solution's position of the  $i^{\text{th}}$  particle's at  $j^{\text{th}}$  dimension,  $\hat{y}_j(t)$  is the best solution's position of all particles at  $j^{\text{th}}$  dimension,  $c_1$  and  $c_2$  are the personal and social learning coefficient respectively and the  $r_{1j}$  and  $r_{2j}$  are the random numbers generated in the range  $[0.0,1.0]$ . According to equation (2) particles use the knowledge of global best solution and personal best solution to move through the search space's corresponding dimension. The performance of the best solution is depends on  $c_1$  and  $c_2$  coefficients. If  $c_1$  is bigger than  $c_2$ , particles conduct local search. If  $c_2$  is bigger than  $c_1$ , particles conduct global search. Usually these parameters initially set equal to each other. Eberhart and Shi showed that  $c_1$  and  $c_2$  parameters can be set 1.494 [7].

General steps of PSO algorithm is shown below;

1. Generate the population. Each particle's velocity vector and variables are generated randomly
2. Calculate the fitness value. Each particle's fitness value calculated according to given fitness function
3. Determination of particle's best solution. Fitness value calculated in the previous step is compared with the best fitness value which is in the memory of the particle. If this value is better than the one in the memory, the value will be updated with better one
4. Determination of global best solution. Each particle's fitness value calculated in the second step compared with the global best fitness value. If this value is better than the one in the memory, the value will be updated with better one
5. Movement of the particle. Each particle's velocity vector will be calculated as stated in equation (2) and their positions will be calculated as stated in equation (1)
6. Until the stopping criterion to be achieved, procedures through 2 to 5 will be repeated

To determine the stopping criterion, two important issues must be considered. Firstly, the stopping criterion never has to cause the algorithm's early convergence. In this situation the algorithm makes only the local search, and this leads to stuck on the local extrema. Secondly, if the stopping criterion increases the costs of calculation, the algorithm cannot converge to the global minimum. Some of general stopping criteria are shown as below;

- Reaching the maximum iteration number,
- Reaching desired solution,
- The slowdown or lack of improvement

PSO algorithm has been changed through the time. Many researchers worked on the algorithm and developed some new features such as inertia weight [8], hybridization features and more.

First hybridization made by Angeline in 1998. This approach changes the bad particles velocity and position values by principle of tournament selection [9]. Another hybrid algorithm is NichePSO [10], this hybrid was developed using CGPSO [11]. This approach uses GA's techniques and divide population into sub populations and trains them with Kennedy's "cognition only model" [12]. Another hybrid algorithm proposes made by Løvebjerg in 2002. In this approach, GA, PSO and hill climbing algorithm were used. Particles select one of these algorithms for the optimization process. This model was called life cycle [13]. In 2003, Higashi and Iba combined Gaussian Mutation with PSO's velocity vector and position update formulations [14],[15].

## 2.2 Genetic Algorithm

Genetic algorithm was introduced first in 1975 by Holland [16]. Genetic algorithm is an evolutionary algorithm that inspired from biological mechanism. In this algorithm, each individual is called gene and the collection of these genes are called population. This algorithm contains two special steps. First one is crossover step that two genes are combined to generate a new one. Second one is mutation step that updates different parts of the generated new gene randomly. With these unique steps, randomly generated first population may convergence to good solution. Since GA's introduce, this algorithm is very successful for optimization problems. Like PSO, genetic algorithm was also changed through the time. Gene selection mechanism [17], crossover operators [18]-[20] and mutation operators [21] were introduced [22]-[25].

## 2.3 Breeding Swarms Algorithm

This algorithm proposed by M. Settles and T. Soule in 2005 [5]. In this algorithm, the population divided by predefined a constant value of breeding ratio. First part of population works exactly like PSO the other part of the population is discarded and replaced by GA. The population handling mechanism makes this algorithm is more robust to stuck on the local extrema. In this algorithm crossover operator called Velocity Propelled Averaged Crossover (VPAC) operator, Gaussian Mutation operator and Tournament Selection scheme have been used. Mathematical expression of VPAC given in equation (3)

$$\begin{aligned} c_1(x_i) &= \frac{p_1(x_i) + p_2(x_i)}{2.0} - \varphi_1 p_1(v_i) \\ c_2(x_i) &= \frac{p_1(x_i) + p_2(x_i)}{2.0} - \varphi_2 p_2(v_i) \end{aligned} \quad (3)$$

where,  $c_1(x_i)$  and  $c_2(x_i)$  are the positions of child 1 and child 2 at  $i^{\text{th}}$  dimension.  $p_1(x_i)$  and  $p_2(x_i)$  are the positions of the parent 1 and parent 2 at  $i^{\text{th}}$  dimension.  $\varphi$  is a random value generated in the range [0.0,1.0].  $p_1(v_i)$  and  $p_2(v_i)$  are the velocity vectors of parent 1 and parent 2 at  $i^{\text{th}}$  dimension. Newly generated particles inherit their parent's velocity vectors and personal best values. According to [5], proposed work yielded slightly better performance and this algorithm is faster than GA and PSO.

## 2.4 Chaotic Systems

Chaos theory is a study field in mathematics. It is based on unpredictability of nature. In this theory, small changes in the initial conditions of dynamic system can make a very different outcome. It is called butterfly effect and this theory was summarized by Edward Lorenz.

In the recent studies, random number sequences are created by chaotic number generator and in some occasions it yielded better results. In [26] chaotic sequences were used in evolutionary algorithms for

performance improvement. Chaotic sequences in genetic algorithms were introduced in [27] and chaos embedded particle swarm algorithms were studied in [28].

### 3. Proposed Work

In related works hybrid versions of PSO gets better results than original PSO and the algorithm that hybridized. The efficiency of PSO depends on aimed random search mechanism. Aiming part of the mechanism can be achieved by personal best and global best values in equation (2) and random search part can be achieved by random values in equation (2).

PSO is hybridized with GA to improve the efficiency of PSO so that Breeding Swarm (BS) model were introduced [4], however they have some deficiencies. In genetics, to generate a new individual at least two parents are required. These two parents combine their DNA's and generate a new individual. In the combining step, half of the DNA from each parent combine with each other so that new individual inherits both parents' features. In real world, almost all the time one parent become dominant and other become recessive in this step Because of that new born individual generated from same parents were dissimilar. But in the VPAC operator's equation, this diversity cannot be modelled. Hence newly generated two child have little difference to each other. To simulate this diversity, we changed VPAC operator to VPCC operator. Mathematical expressions of VPCC operator is given in equation (4)

$$\begin{aligned}c_1(x_i) &= m_1 * p_1(x_i) + (1 - m_1) * p_2(x_i) - m_2 p_1(v_i) \\c_2(x_i) &= m_3 * p_1(x_i) + (1 - m_3) * p_2(x_i) - m_4 p_2(v_i)\end{aligned}\quad (4)$$

where  $m_1, m_2, m_3, m_4$  are the random numbers generated from chaotic random number generator, the other parameters are same as VPAC parameters.

In chaotic number generator, we used logistic map because of its simplicity. Definition of the logistic map is given in equation (5)

$$\begin{aligned}map &= 4 * z * (1 - z) \\m_i &= (0.5 * r_1) + (0.5 * map)\end{aligned}\quad (5)$$

where  $m_i$  is the  $i^{\text{th}}$  random value in equation (4),  $z$  and  $r_1$  are the random value generated by compiler's random number generator.

Scheme of the BS algorithm's creating new children is very simple. In equation (3), simply adds parent particles' values and divides to two after that subtracts this result from parent particles' velocity. In late stages of original BS algorithm, all particles' personal best values would be similar to each other also their velocity would be similar because they tend to go same point which is global best value.

Our approach in equation (4) uses four different random number created from equation (5) which is the equation of logistic map. This approach divides parent particles' values randomly and adds to each other, after that it subtracts this result from randomly multiplied parent particle's velocity. Hence the children created from four different chaotic random number, even the velocity of parents are very similar, created  $c_1$  and  $c_2$  would be very different from each other. By mean of this diversity, particles can free themselves from local optimum points. Because of the algorithm's PSO part does the global search and the GA part does the local search, proposed approach only affects algorithm local search capability. Our approach changes only crossover part of the BS algorithm. It benefits from randomness of chaotic systems and combines this randomness with crossover operation. Pseudo code of the original BS algorithm has given in Algorithm 1.

Since the only thing change was the crossover operation, original algorithm and modified algorithm have similar working time. There are only a few milliseconds difference between two algorithms.

The BS algorithm with new proposed crossover operator VPCC were tested on multimodal and unimodal benchmark functions. Social parameters  $c_1$  and  $c_2$  are set to 2.0, population size set to 125. We used Gaussian Mutation in both PSO and BS algorithms. Inertia weight is linearly decreased from 0.9 to 0.2, mutation rate was set to 1/dimension. All used parameters are shown in Table 1.

## Algorithm 1 Pseudo Code of BS Algorithm

1	Initialize N population
2	Evaluate fitness value for each particle Sort the particles best to worst according to their fitness value
3	Discard worst N*breeding ratio particles Use PSO velocity updates on not discarded particles
4	Generate new particles by crossover and mutation from updated particles to replace discarded particles
5	Use PSO position update on not discarded particles
6	Use PSO position update on not discarded particles
7	Until stopping criteria go to line 2

Table 1 Parameters of PSO and BS

Parameter	PSO	BS
Population Size	125	125
Selection Scheme	-	Tournament
Tournament Size	-	3
Mutation Rate	-	1/Dimension
Mutation Variance	-	1.0 to 0.1
Social Parameters ( $c_1$ & $c_2$ )	2	2
Inertia Weight	0.9 to 0.2	0.9 to 0.2

## 4. Results

### 4.1 Test Problems

We used one unimodal and three multimodal benchmark functions for tests. These benchmark functions are popular and used in several studies [5],[9],[11],[14],[28].

First function is Ellipsoidal function which is unimodal. This function is described in equation (6).

$$f_1(x) = \sum_{i=1}^n ix_i^2 \quad (6)$$

Second function is Rosenbrock function which is multimodal. This function is described in equation (7).

$$f_2(x) = \sum_{i=1}^n (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2) \quad (7)$$

Third function is Griewank function which is multimodal. This function is described in equation (8).

$$f_3(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos \cos \left( \frac{x_i}{\sqrt{i}} \right) + 1 \quad (8)$$

Fourth function is Ackley function which is multimodal. This function is described in equation (9).

$$f_4(x) = 20 + e - 20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^n x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^n \cos(2\pi x_i)} \quad (9)$$

These functions initialization ranges are shown in Table 2.

Table 2 Initialization Range of Benchmark Functions

Function	Initialization Range
Ellipsoidal Function	(-100 to 100)
Rosenbrock Function	(-30 to 30)
Griewank Function	(-600 to 600)
Ackley Function	(-32,768 to 32,768)

All test functions were run with 10 and 20 dimension settings. For 10 dimension algorithms were run for 1000 iteration and for 20 dimensions algorithms were run for 2000 iteration. Each function was run for 15 times.

#### 4.2 Results

PSO, BS and BS with VPCC algorithms results on benchmark functions are given in Table 3.

Table 3 Test Results of PSO, BS and BS with VPCC

Function	Dimension	PSO Mean Best (Std-Dev)	BS Mean Best (Std-Dev)	BS with VPCC Mean Best (Std-Dev)
Ellipsoidal Function	10	0,125176 (0,39708)	6,33E-08 (-2,26E-07)	2,42E-11 (8,98925E-11)
	20	6,115002 (9,104639)	0,744034 (0,789478)	4,2238E-05 (6,65888E-05)
Rosenbrock Function	10	4,778433 (2,487662)	2,110912 (2,352057)	0,80326 (1,111899)
	20	13,09686727 (13,8583444)	10,65285311 (2,23640593)	9,502027 (5,706486)
Griewank Function	10	0,004367103 (0,005704754)	0,063425016 (0,098349987)	0,02798 (0,057941)
	20	0,183087907 (0,229020138)	0,197518884 (0,191952133)	0,182558 (0,185198)
Ackley Function	10	1,325490902 (0,914628641)	0,822298525 (0,921644178)	2,3E-07 (7,17E-07)
	20	4,34005 (1,172947)	1,755973 (0,736546)	1,691745 (0,836093)

In every test cases the BS with VPCC algorithm reached the near global minimum point in a majority of trials. For the minority of the trials BS with VPCC algorithm has stuck on local minimums. In the other hand, most of the trials PSO and BS has stuck on local minimums.

Table 3 shows that Breeding Swarm algorithm with VPCC yielded better results than original PSO algorithm and BS algorithm. In 20 dimension problems BS with VPCC yielded slightly better performance. But in 10 dimension problems BS with VPCC yielded clearly better performance except in the Griewank function.

Figures 1, 2, 3, 4 shows the results of each algorithms performance on problems with 10 dimensions and 5, 6, 7, 8 shows the performance on the problems with 20 dimensions through the iterations.

In general, VPCC added version of BS algorithm outperformed the original BS algorithm in every test cases and outperformed the original PSO algorithm in most of the cases.



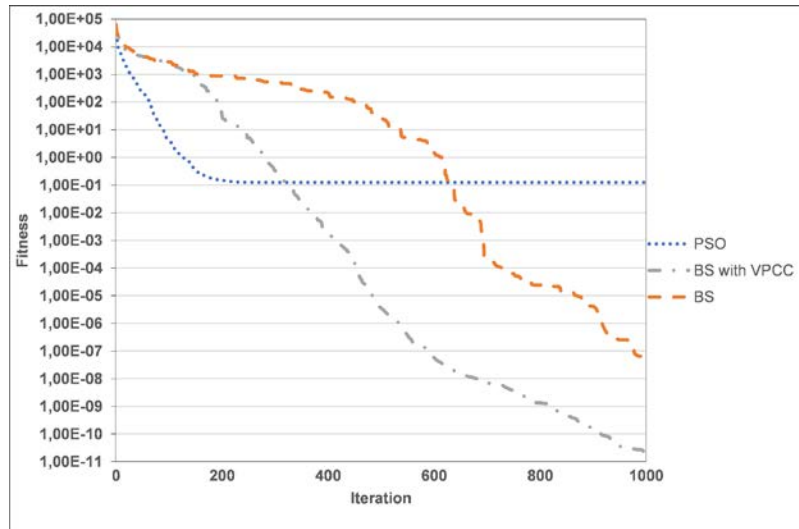


Figure 1 Ellipsoidal Function 10 Dimension

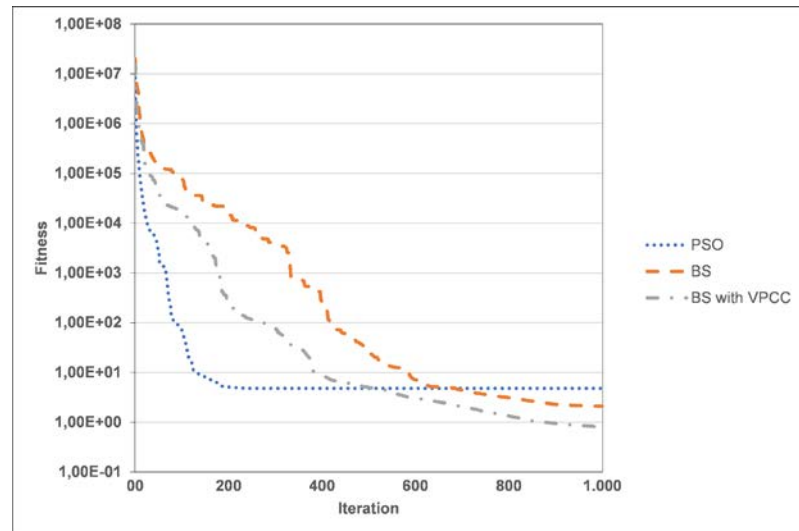


Figure 2 Rosenbrock Function 10 Dimension

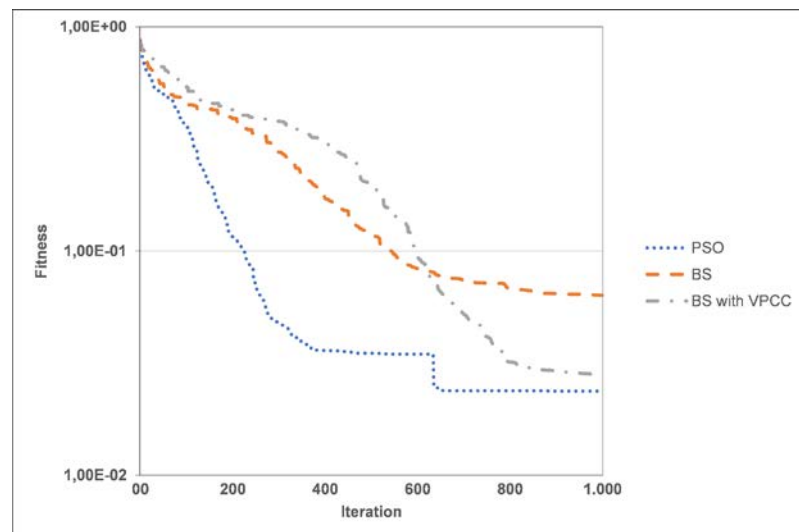


Figure 3 Griewank Function 10 Dimension

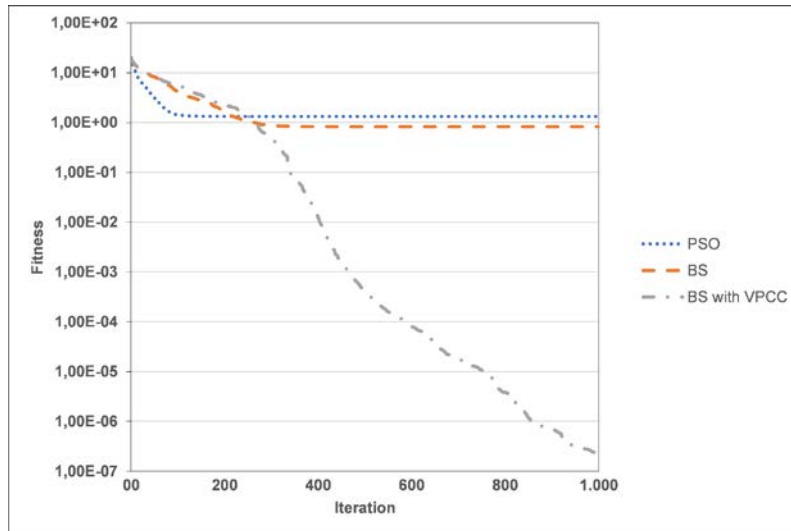


Figure 4 Ackley Function 10 Dimension

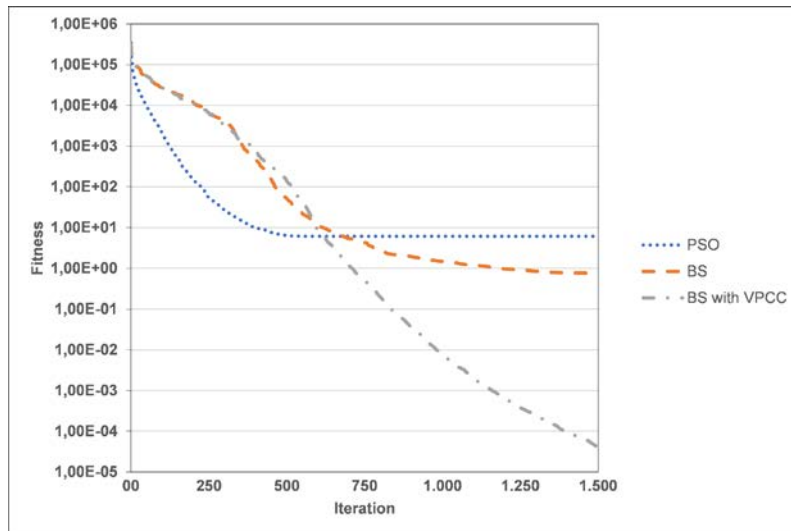


Figure 5 Ellipsoidal Function 20 Dimension

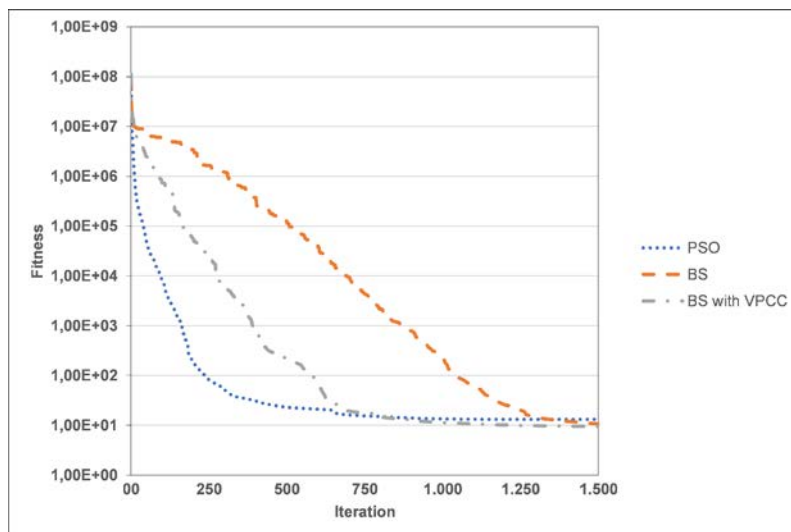


Figure 6 Rosenbrock Function 20 Dimension

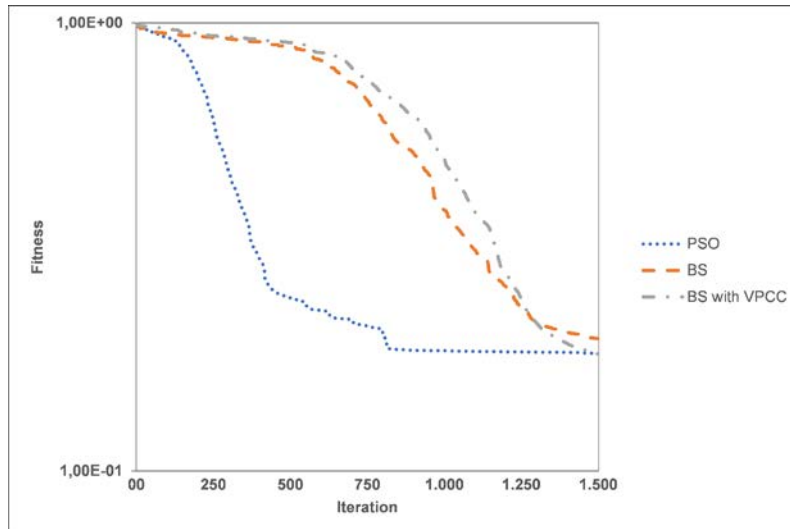


Figure 7 Griewank Function 20 Dimension

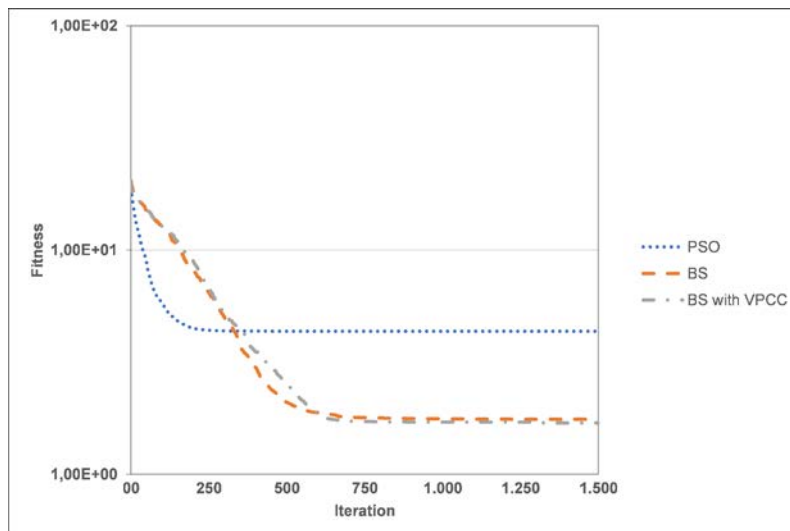


Figure 8 Ackley Function 20 Dimension

## 5. Conclusion

According to results, BS with VPCC algorithm is competitive with both PSO and BS algorithms. With the power of VPCC operator BS algorithm has overcome the early convergence problem and searched the entire search space. Because of the chaotic random number generator's effect on the crossover operator, the diversity of the population is improved. Therefore, the probability of finding the global best solution increases. In our research we used logistic map function for chaotic map for its simplicity. More complex chaotic maps might improve the randomness and gave better results but since the map complexity increases working time of the algorithm might be increase.

For future research, the VPCC operator which studied here or other operators or the other processes of PSO or GA can be combined with chaotic number generators. Different chaotic maps could be investigated because different maps might give different result with same algorithm.




## References

- [1] Q. Liu, X. Li, H. Liu and Z. Guo, "Multi-objective metaheuristics for discrete optimization problems: A review of the state-of-the-art," *Appl Soft Comput.*, vol. 93, 2020.

- [2] J. Kennedy and R.C. Eberhart, "Particle Swarm Optimization," *IEEE Int. Conf. Neural Networks*, pp. 1942-1948, 1995.
- [3] W. Yanmin, "Optimization of Wireless Sensor Network for Dairy Cow Breeding Based on Particle Swarm Optimization," *Int. Conf. Intell. Trans. Big Data & Smart City (ICITBS)*, pp. 524-527, 2020.
- [4] Y. Özger, M. Akpınar, Z. Musayev and M. Yaz, "Electrical Load Forecasting Using Genetic Algorithm Based Holt-Winters Exponential Smoothing Method," *Sakarya University Journal of Computer and Information Sciences.*, vol. 3, no. 2, pp.108-123, 2019.
- [5] M. Settles and T. Soule, "Breeding swarms: a GA/PSO hybrid," *ACM Conf. Genetic and Evol. Comput. (GECCO '05)*, pp. 161-168, 2005.
- [6] H. R. Vanamala and D. Nandur, "Genetic Algorithm and Chaotic Maps based Visually Meaningful Image Encryption," *TENCON 2019 - 2019 IEEE Region 10 Conf. (TENCON)*, pp. 892-896, 2019.
- [7] R.C. Eberhart and Y. Shi, "Comparing inertia weights and constriction factors in particle swarm optimization," *IEEE Congr. Evol. Comput.*, pp. 84-88, 2000.
- [8] R. Eberhart and Y. Shi, "A Modified Particle Swarm Optimizer," *IEEE World Cong. Comput. Intel.*, pp. 69-73, 1998.
- [9] P.J. Angeline, "Using selection to improve particle swarm optimization," *IEEE World Cong. Comp. Intel.*, pp. 84-89, 1998.
- [10] R. Brits, A.P. Engelbrecht and F. van den Bergh, "A niching particle swarm optimizer," *Proc. Sim. Evol. Learn. SEAL.*, 2002.
- [11] F. van den Bergh and A.P. Engelbrecht, "A new locally convergent particle swarm optimizer," *IEEE Int. Conf. Syst., Man and Cyb.*, vol. 3, 2002.
- [12] J. Kennedy, "The particle swarm: social adaptation of knowledge," *Proc. of 1997 IEEE Int. Conf. Evol. Comput.*, pp. 303-308, 1997.
- [13] T. Krink and M. Løvebjerg, "The lifecycle model: combining particle swarm optimization, genetic algorithms and hillclimbers," *Conf. Parallel Probl. Solving Nat., 7th — PPSN VII*, pp. 621-630, 1997.
- [14] N. Higashi and H. Iba, "Particle swarm optimization with Gaussian mutation," *IEEE Swarm Intel. Symp. (SIS)*, pp. 72-79, 2003.
- [15] A. Banks, J. Vincent and C. Anyakoha, "A review of particle swarm optimization. Part II: hybridisation, combinatorial, multicriteria and constrained optimization, and indicative applications," *Nat. Comput.*, vol. 7, pp. 109-124, 2007.
- [16] J. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence :2<sup>nd</sup> ed.* University of Michigan Press, 1992.
- [17] A. Lipowski and D. Lipowska, "Roulette-wheel selection via stochastic acceptance," *Phys. A (Amsterdam, Neth.)*, vol. 391, pp. 2193-2196, 2012.
- [18] Y. Kaya, M. Uyar and R. Tekin, "A Novel Crossover Operator for Genetic Algorithms: Ring Crossover," *arXiv preprint arXiv:1105.0355*, 2011.
- [19] M.J. Varnamkhasti, L.S. Lee, M.R.A. Bakar and W.J. Leong, "A Genetic Algorithm with Fuzzy Crossover Operator and Probability," *Adv. Oper. Res.*, vol. 2012, Article ID 956498, 2012.
- [20] D. Vrajitoru, "Crossover improvement for the genetic algorithm in information retrieval," *Inf. Process. Manage.*, vol. 34, pp. 405-415, 1998.
- [21] M. Srinivas, "Adaptive probabilities of crossover and mutation in genetic algorithms", *IEEE Int. Conf. Syst. Man. Cyb.*, vol. 24, pp. 656-667, 1994.
- [22] I. Abuiziah and N. Shakarneh, "A Review of Genetic Algorithm Optimization: Operations and Applications to Water Pipeline," *Int. J. of Math. Comput. Phys. Quan. Eng.*, vol. 7, pp.136-147, 2013.
- [23] R. Sivaraj and T. Ravichandran, "A Review of Selection Methods in Genetic Algorithm," *Int. J. Eng. Sci. Technol. (IJEST)*, vol. 3, pp. 3792-3797, 2011.
- [24] N. M. Razali and J. Geraghty, "Genetic Algorithm Performance with Different Selection Strategies in Solving TSP," *Proc. World Cong. Eng.*, vol. 2, 2011.

- [25] O. Abdoun and J. Abouchabaka, "A Review of Selection Methods in Genetic Algorithm," *Int. J. Comp. App. (IJCA)*, vol. 31, pp. 49-57, 2011.
- [26] R. Caponetto, L. Fortuna, S.Fazzino and M.G. Xibilia, "Chaotic sequences to improve the performance of evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 7, pp. 289-304, 2003.
- [27] L. J. Yang and T. L. Chen, "Application of Chaos in Genetic Algorithms," *Commun. Theor. Phys.*, vol. 38, pp. 168-172, 2002.
- [28] B. Alatas, E. Akin and A.B. Ozer, "Chaos embedded particle swarm optimization algorithms," *Chaos, Solitons Fractals*, vol. 40, pp. 1715-1734, 2009.

# Deep Neural Networks Based on Transfer Learning Approaches to Classification of Gun and Knife Images

 Mehmet Tevfik Ağdaş<sup>1</sup>,  Muammer Türkoğlu<sup>2</sup>,  Sevinç Gülseçen<sup>3</sup>

<sup>1</sup>Department of Electronic and Automation, Artvin Çoruh University, Artvin, Turkey;  
mtevfikagdas@artvin.edu.tr;

<sup>2</sup>Corresponding Author; Department of Computer Engineering, Bingöl University, Bingöl, Turkey;  
mturkoglu@bingol.edu.tr;

<sup>3</sup>Informatics Department, Istanbul University, Istanbul, Turkey; gulseccen@istanbul.edu.tr

Received 4 March 2021; Revised 15 March 2021; Accepted 16 March 2021; Published online 25 March 2021

## Abstract

Most of the criminal acts are performed using criminal tools. One of the most effective ways of preventing crime is to observe and detect offensive weapons by security camera systems. Deep learning techniques can show very high-performance in observing and perceiving objects. In the current study, the performances of the pre-trained AlexNet, VGG16, and VGG19 models based on convolutional neural networks, were tested for the detection and classification of criminal tools such as guns and knives. In the study, the training process was carried out using transfer learning approaches such as Fine-tuning and Training from scratch based on deep architectures. To test the deep architectures used in the proposed study, the gun and knife datasets frequently used in the literature were collected and combined with new datasets obtained originally from search engines and videos, and then their performances were tested. In the experimental results, the VGG16 model based on fine-tuning for the two and three classes achieved the highest accuracy in detecting criminal devices with a rate of 99.73% and 99.67%, respectively. As a result, the study has observed that offensive weapons could be detected with security cameras using learned weights of deep architectures.

**Keywords:** Pattern recognition, Gun and Knife Detection, Deep Learning, Offensive weapons, Convolutional Neural Network

## 1. Introduction

Globally the number of violent incidents committed by individuals with their weapons is increasing day by day. In the 2019 report named the Map of the Armed Violence in Turkey released by Umut Foundation, an institution standing against the individual armament in Turkey, it has been stated that the number of violent events committed by using personal arms is arising every passing year in Turkey. The report has declared that individuals involving criminal acts in 2019 took their places in the media for 3623 disagreements. Out of these events, firearms were used in 2867 conflicts, and all kinds of sharp and piercing tools were used in the remaining 756 incidents. Besides, according to the foundation's research results obtained from other sources, it has been stated that 2,211 people died, and 3,736 people were injured in the same year resulting from firearms and sharp objects utilization [1].

The Umut Foundation report also includes a comparative regional armed violence report for the years 2015-2019. As seen in Figure 1, at least a 46% increase in the numbers of violent crimes and crime tools is observed in every region of Turkey between 2015 and 2019 years [1]. That people openly carry offensive instruments is a sign of possible acts of violence. Detection of crime weapons may prevent violent incidents. Governments prohibit carrying offensive weapons with sanctions, and besides, assign law enforcement officers to detect. Law enforcement officers make observations directly in public areas and also with security camera systems through computers.

Object detection and recognition from images or videos have recently become one of the most popular research/application topics. Various applications such as recognitions and detections of face, vehicle, plant, license plate, item are performed automatically. The security sector is one of the most prominent areas that object detection or recognition applications can have significant effects. Today, security cameras are used in many living areas, and images taken from these security cameras play a primary

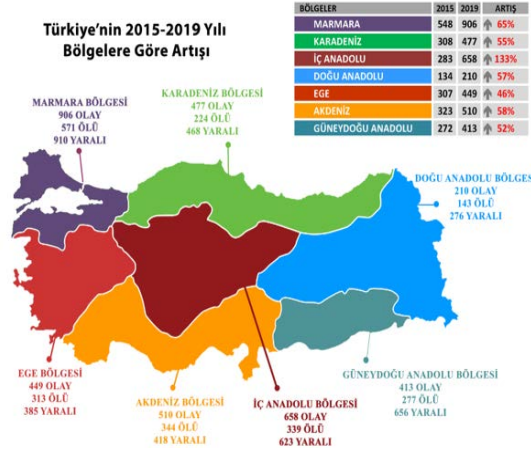


Figure 1. Umut Foundation 2015-2019 Armed Violence by Regions [1].

role in solving many incidents and providing information for forensic channels [2]. Usually, the usage of security cameras and getting crucial information for public safety are conducted by law enforcement. However, using computer vision systems for object perception or detection in today's technology will ensure reliability in the detection and reduce the law enforcement officers' workloads. Besides, since the computers carry out the automatic object detection and recognition processes, the law enforcement officers can get the information required more quickly and accurately. Research has shown that law enforcement officers' prompt intervention in preventing crime is of great importance [3].

In the method's selection for this study, some studies on gun and knife detection and perception in the literature were examined. In their studies, Garega et al. performed feature extraction from MPEG7 videos and made classifications using SVM algorithms to detect guns and knives through security cameras [4]. Gonzalez et al. conducted a real-time gun detection study in the closed-circuit security camera (CCTV) system. They used R-CNN based pre-trained ResNet-50 architecture [5]. Verma and Dhillon conducted a study for the detection of handheld guns by using Faster R-CNN, which is one of the deep learning techniques. They used Deep Convolution Network (DCN) for automatic gun detection from scattered scenes [6]. Tiwari and Verma carried out a computer vision-based visual gun detection study using a Harris point of interest detector. They have extracted the unrelated object from the images using the K-Means algorithm. They made use of the color-based sections in the appearances [7]. Kmiec et al. presented a new application of Active View Patterns from Computer vision techniques to detect the blades in images. In the examination, they sought an answer to whether there was a knife in the security camera images [8]. Castillo et al. made a preprocessing through the brightness guidance for automatic cold steel gun detection in security camera surveillance videos featured with deep learning. They used CNN-based ResNet architectures for detection. They also used the DaCoLT (Darkening and Contrast in Learning and Testing stages) technique for brightness-guided preprocessing [9]. Carrobles et al. conducted a Faster R-CNN-based gun and knife detection study for video surveillance. They used R-CNN based SqueezeNet architecture and GoogleNet architecture in their studies. In test measurements of the detection process, they achieved a performance score of 46.68% with GoogleNet architecture and 85.44% with SqueezeNet architecture [10]. Jain et al. conducted a gun detection study using artificial intelligence and deep learning techniques for security applications. They used the R-CNN neural network SSD VGG-16 architecture in their work and achieved an average performance score of 73.8% [11]. Olmos et al. made automatic gun detection in videos using deep learning approaches. They used an open-access dataset. In their studies, they performed the training process with the CNN based R-CNN model [12].

In this study, the pre-trained deep architecture performance was evaluated to classify guns, knives, and regular images. In this direction, approaches such as transfer learning and scratch were used for pre-trained deep architectures. In the experimental studies, a comprehensive dataset containing guns, knives, and ordinary images was used to test the proposed system. In the experimental results, VGG16 architecture achieved a criminal devices recognition performance of 99.73%.

The present study was carried out to automatically detect guns and knives that could be used in violent incidents. For this purpose, deep learning techniques were suggested. The measurements were made on high-performance GPU supported computers with AlexNet, Vgg16, and Vgg19 architectures previously trained with deep learning techniques. Previous object detection studies in the literature were examined, and the existing datasets were analyzed. To increase the dataset reliability, original and new gun and knife data were collected from internet pages, combined with the existing dataset, and tested with architectures. The architectures of the dataset used, and the test performances have been compared with the table. From the current study results in the literature, methods with better performance have been recommended for applications that require this type of perception.

The main contributions of this study are given below:

- In experimental studies, it has been determined that the learned weights of pre-trained Deep architectures provide high performance in detecting crime instruments.
- In this study, the performances of deep architectures such as AlexNet, VGG16, and VGG19 were evaluated, and their strengths were demonstrated.
- The proposed study can be used in simple and real-time applications.

## 2. Material and Method

In this paper, we evaluated performances of deep models based on transfer learning approaches to the classification of gun and knife images. The general flow diagram of the proposed study is given in Figure 2.

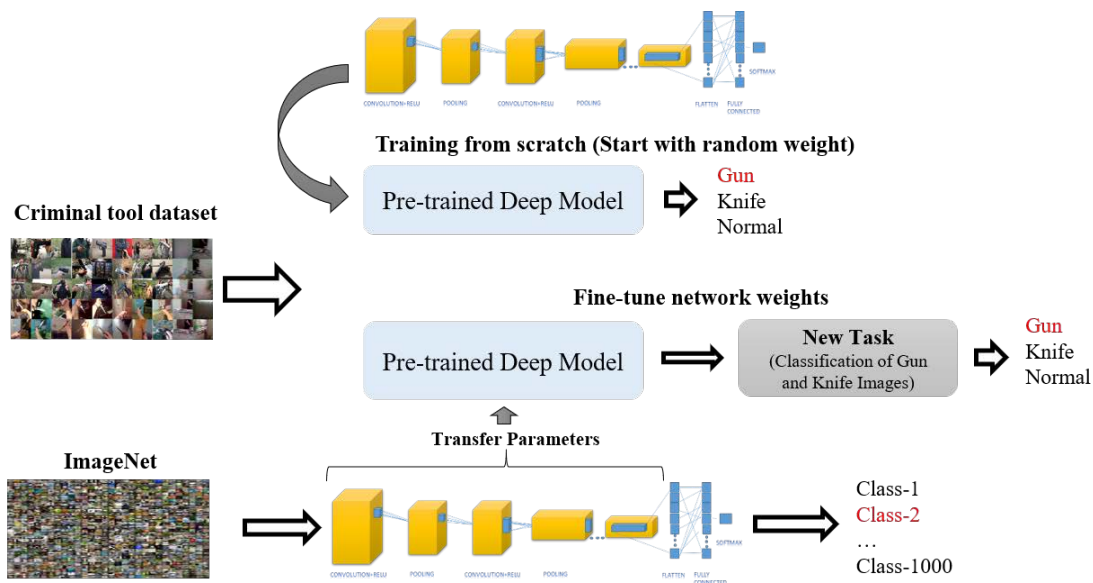


Figure 2. The general structure of the proposed study

### 2.1. Dataset

In this study, a data set suitable for the purposes was created by collecting images from open access data sets [9, 12-14], which have repeatedly been the source of many previous studies in the literature and original internet browsers and video pages [4, 15]. This dataset has 16000 images containing 9500 knives, 3500 guns, and 3000 ordinary pictures. Sample images belonging to these classes are given in Figure 3.





Figure 3. Data set example, a) Gun images, b) Knife images, c) Normal images

## 2.2 Convolutional Neural Networks

A convolutional neural network (CNN) is a neural network that has matrix multiplication and convolution system in at least one of its layers. CNNs must have at least one or more of the convolution layer, non-linearity layer, pooling layer, fully connected layer, and hidden layers (Figure 4).

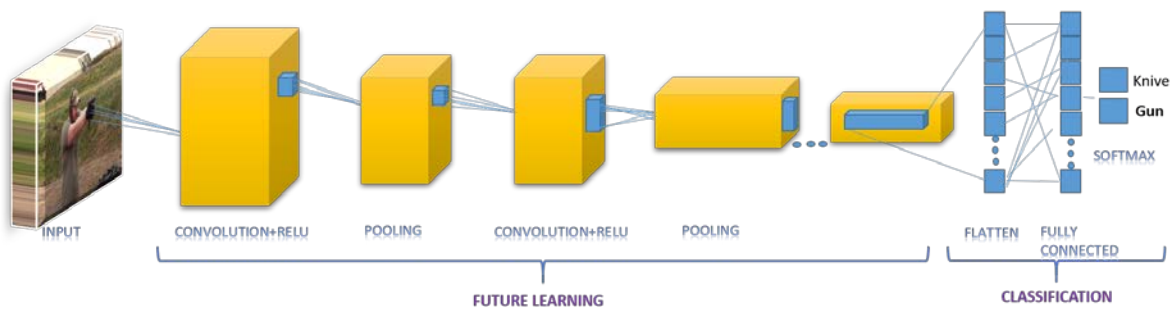


Figure 4. Convolutional Neural Network Layers

The layers given in Figure 4 can be briefly summarized as follows [16-19]:

- **Convolutional Layer-** The layer used to detect properties,
- **Non-Linearity Layer-** The layer used to make the system nonlinear,
- **Pooling Layer-** The layer controlling the number of weights and over-fitting,
- **Flattening Layer-** Data preparation layer for classical neural network,
- **Fully Connected Layer-** Standard neural network layer used for classification.

CNNs are inspired by people's ability to comprehend events. It is in the deep feedforward artificial neural networks class to analyze images. CNN uses 2D and 3D images when generating spatial and configuration information. Convolutional neural networks use three receptive fields, shared weights, and sub-sampling mechanisms to increase the models' working areas. CNNs are generally used to recognize patterns in images. For this reason, features within images can be encoded into architectures, thus making them more suitable for vision network-oriented tasks while further reducing the parameters required for model building. CNN applications are widely used for object detection, prediction, text detection, image detection, etc. [18-24].

### 2.3 Deep Architectures

In this study, the convolutional neural network models AlexNet, VGG-16, and VGG-19 architectures, which were previously trained on over one million images, were used, and the comparison results were obtained.

#### 2.3.1 AlexNet Architecture

AlexNet [25] is a convolutional neural network architecture created in 2012. It has 60 million parameters and 650 million neurons. AlexNet comprises five Convolutional Layers and three Fully Connected Layers. The Multiple Convolutional Kernels layer (Filters layer) discerns unusual features in an image. There are usually plenty of same-size cores in a single convolution layer. For example, AlexNet's first Conv Layer contains 96 cores at  $11 \times 11 \times 3$ . The width and height of the cores are usually the same sizes. The Max Pooling layer comes after the first two-convolutional layers. The third, fourth and fifth convolutional layers directly connect. The Overlapping Max Pooling layer, whose output goes to a series of two fully connected layers, follows the fifth convolutional layer. The fully connected second layer is followed by a 1000 class labeled Softmax classifier [26-27]. For the Alexnet architecture, images should be specified as  $256 \times 256$  inputs. The structure of the AlexNet architecture given in Figure 5.

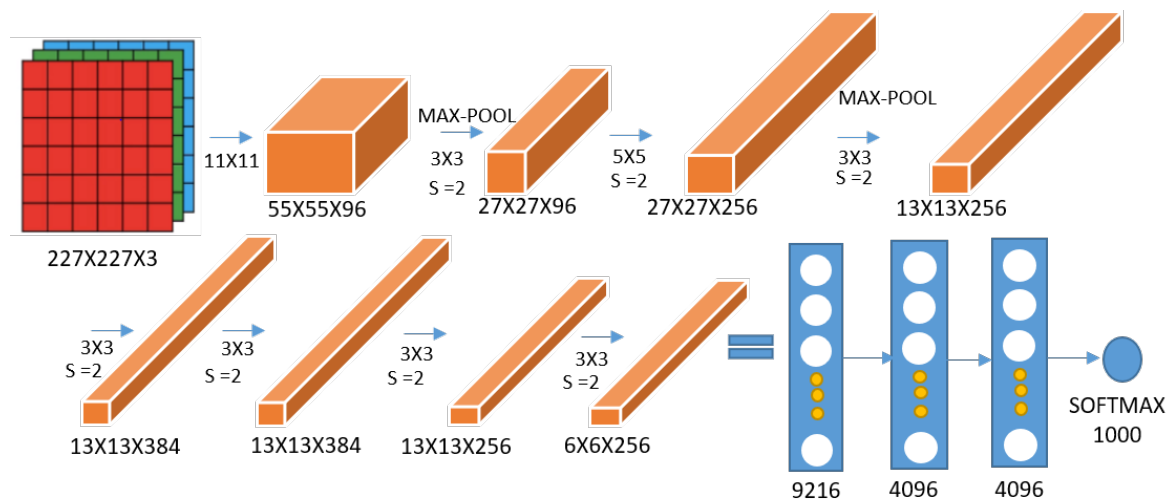


Figure 5. AlexNet Architecture

#### 2.3.2 VGG Architecture

The VGG-16 architecture [28], including 21 main layers, consists of convolutional, pooling, and fully connected layers. Its architecture has an increasing network structure. The image input resolution must be  $224 \times 224$  pixels in size. In this architecture, the last layers comprise fully connected layers used for feature extraction [29-30]. The structure of the VGG16 architecture given in Figure 6.



Figure 6. VGG16 Architecture

VGG-19 architecture [28] with 24 main layers consists of 16 convolutional, five pooling, and three fully connected layers. Since VGG-19 has a deep network, the filters are used in the convolution layer to reduce the parameters' number. In this architecture, the image input size is 3x3 pixels. The VGG-19 architecture contains nearly 138 million parameters [29-30]. The structure of the VGG19 architecture given in Figure 7.



Figure 7. VGG19 Architecture

## 2.4 Transfer learning approaches

Pre-trained deep models contain parameters that have been trained using a large multi-class data set. Two transfer learning approaches, Fine-tuning and Training from scratch are available to use these deep models in other data sets. The fine-tuning method uses the learned weights of pre-trained deep models. This approach is based on adapting deep architectures by replacing the last three layers with new three layers to solve another classification problem. On the other hand, the Training from scratch approach uses the pre-trained model architecture to train the dataset accordingly. The starting weights of the model begin randomly [21, 31].

## 3. Experimental Works

The experimental studies were performed using MATLAB (R2020a) software on a computer equipped with RTX 2080 GPU card, 32 GB Ram, and Intel Core i7. In experimental studies, 2-class and 3-class data sets were used. These data sets were split into training and test sets, at the ratios of 80-20%, 60-40%, 40-60%, and 20-80%, and a comprehensive experimental study was conducted for each. Besides, the network parameters used for training the pre-trained ESA architectures used in this study are given in Table 1.

Table 1 The deep network parameters used in the current study.

Mini-batch size	8-16
Maximum epoch number	5-20
Weight decay factor	0.01
Initial learning rate	0.0001
Optimization method	SGDM (Stochastic Gradient Descent with Momentum)

Accuracy scores obtained in the experimental study performed for the 2-class (knife and normal) data set are given in Table 2.

Table 2 Accuracy scores (%) for the 2-class data set

Approaches	Models	80-20	60-40	40-60	20-80	Average
Fine-tuning	Alexnet	98.80	98.31	97.39	96.48	97.74
	VGG16	99.73	99.65	99.42	98.77	99.38
	VGG19	99.65	99.53	99.25	98.66	99.27
Training from scratch	Alexnet	87.87	87.09	84.05	79.40	84.58
	VGG16	84.73	83.57	80.23	74.38	80.72
	VGG19	83.76	79.46	77.46	75.40	79.02

In Table 2, separation values for four different training/test clusters are used, and separate results are given for each. According to these results, among deep models based on the fine-tuning approach, the VGG16 architecture achieved the highest accuracy on average, while the AlexNet architecture showed the lowest performance. On the other hand, among deep models based on the training from scratch approach, the Alexnet architecture obtained the highest accuracy on average, while the VGG19 architecture achieved the lowest performance. Additionally, the VGG16 architecture based on the Fine-tuning approach obtained the highest accuracy score among deep models with a 99.73% success for separating training/testing clusters as 80-20, while an 87.87% score was obtained with the AlexNet architecture based on the training from scratch approach. The confusion matrices of the highest performance based on these fine-tuning and the training from scratch approaches are given in Figure 8.

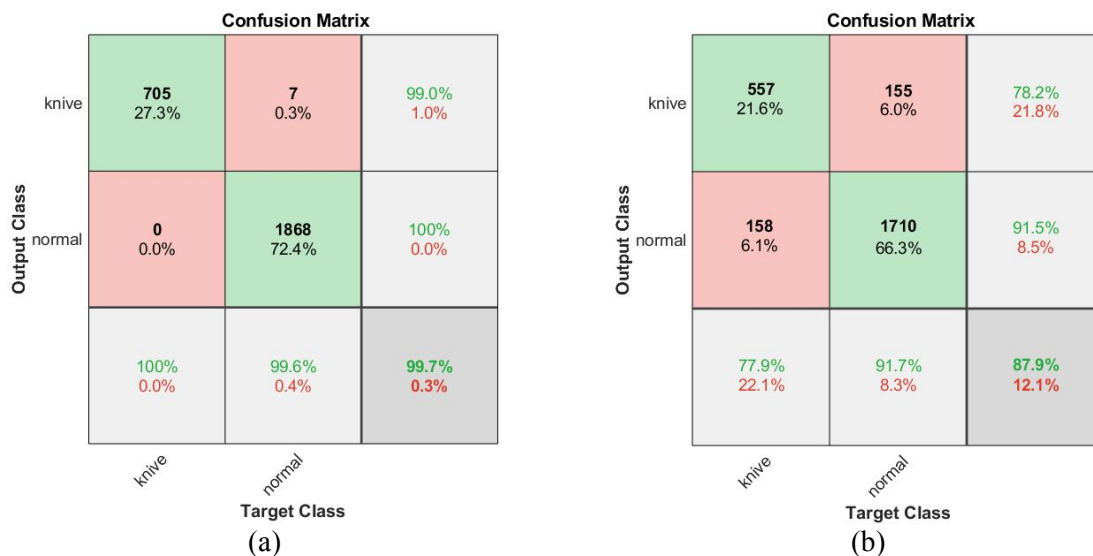


Figure 8. Confusion Matrices of proposed deep model based approaches to 2-class, a) Fine-tuning, b) Training from scratch.

The accuracy scores obtained in the experimental studies performed for the 3-class (knife, gun, and ordinary) data set are given in Table 3.

Table 3 Accuracy scores (%) for the 3-class data set

Approaches	Models	80-20	60-40	40-60	20-80	Average
Fine-tuning	Alexnet	98.86	98.68	97.82	96.72	98.02
	VGG16	99.62	99.41	99.29	98.64	99.24
	VGG19	99.59	99.27	99.15	98.59	99.15
Training from scratch	Alexnet	86.77	85.25	80.48	72.86	81.34
	VGG16	85.74	83.31	78.18	70.03	79.31
	VGG19	83.67	79.69	74.04	64.64	75.51

As seen from Table 7, the average highest accuracy among deep models based on the Fine-tuning approach was achieved as 99.24% with the VGG16 model, while the AlexNet model using the training

from the scratch approach achieved an 81.34% success rate. In terms of the 80-20 separation value of training/test clusters, the highest accuracy score among the deep models was achieved with a 99.62% rate by the VGG16 architecture based on the fine-tuning approach. Confusion matrices of the highest performance among deep models based on the fine-tuning and training from the scratch approaches are given in Figure 9.



Figure 9. Confusion Matrices of proposed deep model based approaches to 3-class, a) Fine-tuning, b) Training from scratch.

As a result, it has been observed that the fine-tuning approach provides higher performance than the training from the scratch approach for pre-trained deep models. Besides, according to the results obtained from all experimental studies, the highest accuracy among deep models based on the fine-tuning method was obtained with the VGG16 model, while the AlexNet model got the highest score by using the training from scratch approach.

#### 4. Discussion

Nowadays, most offensive actions are committed with criminal tools. One of the most effective ways of preventing malignant acts is to observe and detect the weapons in advance through security cameras. Many studies in the literature deal with the classification of criminal instruments such as guns and knives utilizing visuals containing offensive images. These studies are detailed in Table 4.

Table 4 Comparison of the proposed approach with previous studies

References	Methods	Number of Classes	Number of Images	Accuracy (%)
[32]	Fine-tuning based LeNet	3	12,000	99
[33]	Visual vocabularies and deep features	4	1,950	95 and above
[2]	Alexnet+SVM	2	2,000	95
[15]	Fuzzy classification model	2	12,899	86
[34]	VGGNet	3	5,504	98.41
The current study	Fine-tuning based VGG16	2	12,899	99.73
		3	14,481	99.62

In Table 4, the accuracy scores of previous studies using knives, guns, and ordinary images are given. In these researches, generally, deep-learning-based models were used. In Table 4, it is observed that the



current study has used more visuals than other works. As a result, it has been observed that the proposed research achieved a higher accuracy score than previous studies.

## 5. Conclusion

This study assessed the performances of deep convolutional neural networks based on different approaches for the classification of criminal tools. For this purpose, pre-trained AlexNet, VGG16, and VGG19 models were used. The performances of these models were calculated using Fine-tuning and Training from scratch approaches. As a result of extensive experimental studies, it was clearly observed that nearly 100% accuracy was obtained using the learned weights of deep architectures in the detection of crime tools. As a result, the study has observed that offensive weapons could be detected with security cameras using learned weights of deep architectures. In future studies, we are planning to detect video-based criminal devices using a deep neural network.

## References

- [1] “Umut Vakfi - Türkiye Silahlı Şiddet Haritası 2019”. [Online]. Available: <http://umut.org.tr/umut-vakfi-turkiye-silahli-siddet-haritasi-2019/> [Accessed: 02-January-2021].
- [2] S. B. Kibria and M. S. Hasan, “An analysis of Feature extraction and Classification Algorithms for Dangerous Object Detection,” *In 2017 2nd International Conference on Electrical & Electronic Engineering (ICEEE)*, pp. 1-4, 2017.
- [3] R. Kanehisa and A. Neto, “Firearm Detection using Convolutional Neural Networks”, [Online]. Available: <https://www.scitepress.org/Link.aspx?doi=10.5220/0007397707070714/> [Accessed: 24-January-2021].
- [4] M. Grega, A. Matiolański, P. Guzik, and M. Leszczuk, “Automated Detection of Firearms and Knives in a CCTV Image,” *Sensors*, vol. 16, no 1, pp. 47, 2016.
- [5] J. L. Salazar González, C. Zaccaro, J. A. Álvarez-García, L. M. Soria Morillo, and F. Sancho Caparrini, “Real-time gun detection in CCTV: An open problem,” *Neural networks*, vol. 132, pp. 297-308, 2020.
- [6] G. K. Verma and A. Dhillon, “A Handheld Gun Detection using Faster R-CNN Deep Learning,” *Proceedings of the 7th International Conference on Computer and Communication Technology*, New York, NY, USA, pp. 84-88, 2017.
- [7] R. K. Tiwari, and G. K. Verma, “A computer vision based framework for visual gun detection using harris interest point detector,” *Procedia Computer Science*, vol. 54, pp. 703-712, 2015.
- [8] M. Kmieć, A. Głowacz, and A. Dziech, “Towards robust visual knife detection in images: active appearance models initialised with shape-specific interest points,” *In International Conference on Multimedia Communications, Services and Security*, Berlin, Heidelberg, 2012, pp. 148-158, doi: 10.1007/978-3-642-30721-8\_15.
- [9] Castillo, S. Tabik, F. Pérez, R. Olmos, and F. Herrera, “Brightness guided preprocessing for automatic cold steel weapon detection in surveillance videos with deep learning,” *Neurocomputing*, vol. 330, pp. 151-161, 2019.
- [10] M. M. Fernandez-Carrobles, O. Deniz, and F. Maroto, “Gun and Knife Detection Based on Faster R-CNN for Video Surveillance,” *Pattern Recognition and Image Analysis*, Cham, pp. 441-452, 2019.
- [11] H. Jain, A. Vikram, Mohana, A. Kashyap, and A. Jain, “Weapon Detection using Artificial Intelligence and Deep Learning for Security Applications,” *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pp. 193-198, 2020.
- [12] R. Olmos, S. Tabik, and F. Herrera, “Automatic handgun detection alarm in videos using deep learning,” *Neurocomputing*, vol. 275, pp. 66-72, 2018.
- [13] R. Olmos, S. Tabik, A. Lamas, F. Pérez-Hernández, and F. Herrera, “A binocular image fusion approach for minimizing false positives in handgun detection with deep learning,” *Inf. Fusion*, no. 49, pp. 271-280, 2019.

- [14] F. Pérez-Hernández, S. Tabik, A. Lamas, R. Olmos, H. Fujita, and F. Herrera, "Object Detection Binary Classifiers methodology based on deep learning to identify small objects handled similarly: Application in video surveillance," *Knowl.-Based Syst.*, vol. 194, pp. 105590, 2020.
- [15] A. Matiolański, A. Maksimova, and A. Dziech, "CCTV object detection with fuzzy classification and image enhancement," *Multimed. Tools Appl.*, vol. 75, no. 17, pp. 10513-10528, 2016.
- [16] M. Türkoğlu, K. Hanbay, I. S. Sivrikaya, and D. Hanbay, "Derin Evrişimsel Sinir Ağı Kullanılarak Kayısı Hastalıklarının Sınıflandırılması," *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, vol. 9, no. 1, pp. 334-345, 2021.
- [17] "Introduction to Convolutional Neural Networks", 2018. [Online]. Available: <https://rubikscodex.net/2018/02/26/introduction-to-convolutional-neural-networks/> [Accessed: 03-January-2021].
- [18] M. Turkoglu, "COVIDetectioNet: COVID-19 diagnosis system based on X-ray images using features selected from pre-learned deep features ensemble," *Applied Intelligence*, pp. 1-14, 2020.
- [19] K. Kayaalp, and S. Metlek, "Classification of robust and rotten apples by deep learning algorithm," *Sakarya University Journal of Computer and Information Sciences*, vol. 3, no. 2, pp. 112-120, 2020.
- [20] I. Ozsahin, and D. U. Ozsahin, "Neural network applications in medicine," In *Biomedical Signal Processing and Artificial Intelligence in Healthcare*, Academic Press, pp. 183-206, 2020.
- [21] M. Türkoğlu, and D. Hanbay, "Plant disease and pest detection using deep learning-based features," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 27, no. 3, pp. 1636-1651, 2019.
- [22] H. Ahmetoğlu, and R. Daş, "Derin Öğrenme ile Büyük Veri Kumelerinden Saldırı Türlerinin Sınıflandırılması," In *2019 International Artificial Intelligence and Data Processing Symposium (IDAP)*, pp. 1-9, 2019,
- [23] R. Daş, B. Polat, and G. Tuna, "Derin Öğrenme ile Resim ve Videolarda Nesnelerin Tanınması ve Takibi," *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, vol. 31, no. 2, pp. 571-581, 2019.
- [24] D. Şengür, and S. Siuly, "Efficient approach for EEG-based emotion recognition," *Electronics Letters*, vol. 56, no. 25, pp.1361-1364, 2020.
- [25] A. Krizhevsk, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," In: *Advances in Neural Information Processing Systems*, pp. 1097-1105, 2012.
- [26] F. Doğan, and I. Türkoğlu, "Comparison of Leaf Classification Performance of Deep Learning Algorithms," *Sakarya University Journal of Computer and Information Sciences*, vol. 1, pp. 10-21, 2018.
- [27] M. Uçar, and E. Uçar, "Computer-aided detection of lung nodules in chest X-rays using deep convolutional neural networks," *Sakarya University Journal of Computer and Information Sciences*, vol. 2, no. 1, pp. 41-52, 2019.
- [28] K. Simonyan, and A. Zisserman "Very deep convolutional networks for large-scale image recognition," *arXiv preprint*, arXiv:1409.1556, 2014.
- [29] D. Akgün, "An Evaluation of VGG16 Binary Classifier Deep Neural Network for Noise and Blur Corrupted Images," *Sakarya University Journal of Computer and Information Sciences*, vol. 3, no. 3, pp. 264-271, 2020.
- [30] E. Erdem, and T. Aydın, "Detection of Pneumonia with a Novel CNN-based Approach," *Sakarya University Journal of Computer and Information Sciences*, vol. 4, no. 1, pp. 26-34, 2021.
- [31] M. Turkoglu, O. F. Alcin, M. Aslan, A. Al-Zebari, and A. Sengur, "Deep rhythm and long short term memory-based drowsiness detection.," *Biomedical Signal Processing and Control*, vol. 65, pp. 102364, 2021.
- [32] M. U. Salur, İ. Aydın, and M. Karaköse, "Gömülü Derin Öğrenme ile Tehdit İçeren Nesnelerin Gerçek Zamanda Tespiti," *Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi*, vol. 10, no. 2, pp. 497-509, 2019.
- [33] D. Mery, E. Svec, M. Arias, V. Rizzo, J. M. Saavedra, and S. Banerjee, "Modern computer vision techniques for x-ray testing in baggage inspection," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 4, pp. 682-692, 2016.

- [34] N. Dwivedi, D. K. Singh, and D. S. Kushwaha, "Weapon Classification using Deep Convolutional Neural Network," *In 2019 IEEE Conference on Information and Communication Technology*, pp. 1-5, 2019.



# Deep Learning Methods for Autism Spectrum Disorder Diagnosis Based on fMRI Images

 Muhammed Ali Bayram<sup>1</sup>,  İlyas Özer<sup>2</sup>,  Feyzullah Temurtaş<sup>3</sup>,

<sup>1</sup>Corresponding Author; Electrical-Electronics Engineering Department, Bandırma Onyedi Eylül University;  
muhammedbayram@ogr.bandirma.edu.tr

<sup>2</sup>Computer Engineering Department, Bandırma Onyedi Eylül University;  
iozer@bandirma.edu.tr

<sup>3</sup>Electrical-Electronics Engineering Department, Bandırma Onyedi Eylül University;  
ftemurtas@bandirma.edu.tr

Received 13 February 2021; Revised 06 March 2021; Accepted 23 March 2021; Published online 02 April 2021

## Abstract

Brain injuries are significant disorders affecting human life. Some of these damages can be completely eliminated by methods such as drug therapy. On the other hand, there is no known permanent treatment for damages caused by diseases such as Alzheimer, Autism spectrum disorder (ASD), Multiple sclerosis and Parkinson. Treatments aimed at slowing the progression of the disease are generally applied in these types of disorders. For this reason, essential to diagnose the disease at an early phase before behavioral disorders occur. In this study, a study is presented to detect ASD through resting-state functional magnetic resonance imaging rs-fMRI. However, fMRI data are highly complex data. Within the study's scope, ASD and healthy individuals were distinguished on 871 samples obtained from the ABIDE I data set. The long short-term memory network (LSTM), convolutional neural network (CNN), and hybrid models are used together for the classification process. The results obtained are promising for the detection of ASD on fMRI.

**Keywords:** Deep learning, RNN, rs-fMRI, Autism Spectrum Disorder

## 1. Introduction

The brain is the core member of the nervous system and one of the human body's essential organs [1]. For this reason, the functioning of the brain has always been an area of interest for researchers. Brain functions may be partially or wholly impaired over time for many reasons, such as living conditions and genetic factors. These impairments in brain functions lead to the emergence of some diseases. Multiple sclerosis (MS), Attention deficit hyperactivity disorder (ADHD), Parkinson's, ASD and depression are some of these diseases. Depression, one of the disorders, can be treated entirely with drugs [1].

On the other hand, there is no known definitive treatment for diseases such as MS, Parkinson's, and ASD [1]. Therefore, the progression of the disease is controlled rather than curing it completely. However, in most cases, diseases can only be diagnosed after complications have occurred. In this situation, patients are exposed to irreversible damage [1]. In this respect, diagnosing diseases at an early stage is one of the critical issues.

Many different methods are used to detect impairments in brain functions. Methods such as magnetoencephalography (MEG) and positron emission tomography (PET) are some of them. Similarly, Electroencephalography (EEG) and rs-fMRI are some of the methods used. Compared to other methods, rs-fMRI provides much better spatial resolution [2]. Besides, it provides a temporal resolution of brain functions [2]. In this noninvasive method, activities in the brain are evaluated by following the movements in oxygen levels in the blood [2,3]. Two different methods are used for fMRI [1]. These are called task-based fMRI and resting-state fMRI [1]. It is possible to diagnose ASD using the rs-fMRI method [4].

ASD is an extremely complicated neurodevelopmental disorder and it is getting more common day by day [1,4]. Many comorbid disorders such as seizures, anxiety, and intellectual impairment can occur in

this disease [5,6]. According to a report prepared in 2013, it appears that one in 55 children between the ages of 6 and 17 is diagnosed with ASD [7]. Mild or severe disorders that affect social interaction, communication, behavior, and imagination can be seen in patients with ASD [8-11]. Until today, diagnoses of ASD in childhood have generally been made based on the data obtained from clinical interviews and behavioral observations [1]. One of the most significant weaknesses in traditional diagnostic techniques is that the diagnosis can be made after substantial behavioral disorders occur [1]. On the other hand, it is crucial to diagnose these disorders early before behavioral disorders arise [1]. It is vital to improving diagnostic accuracy based on medical imaging techniques such as fMRI [12,13].

In recent years, studies on the use of machine learning (ML) techniques have gained momentum to increase the accuracy of diagnosis in neurological disorders and reduce physicians' workload [1,4]. ML are artificial intelligence applications that can automatically learn and detect patterns over a group of data [14,15]. ML has been used successfully in many tasks, where using traditional algorithms is difficult or not possible. One of the essential advantages of machine learning is making consistent and high-performance predictions using complex and nonlinear relationships between features [16]. In this direction, they can identify complex relationships that people cannot see directly. Accordingly, ML algorithms have been used successfully in different tasks such as disease diagnosis [17-20] and evaluation of EEG data [19,21].

With the Autism Brain Imaging Data Exchange I (ABIDE I) [22] data set, studies for detecting ASD on fMRI images gained speed [1]. In this direction, in the study conducted in [23], Various algorithms such as Linear Discriminant Analysis (LDA), Random Forest (RF) and Support Vector Machine (SVM) were used in the ABIDE data set. In addition, neural network algorithms such as Multi-Layer Perceptron (MLP) are also used for ASD detection in the ABIDE data set. ASD detection has been carried out. The tests performed achieved the highest performance by using MLP with 56.26% [23]. In another study performed using a CNN on ABIDE I data set, a classification success of 72.73% was obtained [24]. In another study, only NYU Langone Medical Center data were used instead of the entire ABIDE I data set [4]. A success of 90.39% was achieved in the tests performed using the deep neural network. On the other hand, using data from only one clinic instead of the entire data set stands out as an essential deficiency. When the entire whole ABIDE set is evaluated together, it is seen that the data is obtained from different brand devices in different locations and under different settings. In this case, it stands out as a factor that significantly affects the classification accuracy.

In this study, hybrid approaches using CNN, LSTM and both are suggested for ASD detection on fMRI images. While CNNs are very successful in detecting spatial dependencies, LSTMs are very successful in detecting temporal dependencies. Accordingly, it is evaluated that hybrid models formed by CNN and LSTM models on fMRI data with temporal and spatial dependencies can provide more successful results by detecting both spatial and temporal dependencies. The results obtained are promising in terms of detecting ASD on medical images.

## 2. Materials and Methods

### 2.1 Datasets

Used the ABIDE 1 dataset for making our study comparable with existing studies [22]. The dataset includes rs-fMRI brain images, phenotypic information and structural MRI (T1-weighted) of 1112 subjects, composed of 2 different groups, 539 ASD and 573 typical controls (TC). The data are gathered from 17 various institutions and brought together and made available to all researchers for use in scientific research. Studying on the ABIDE 1 data set is difficult due to the heterogeneity of the subjects, large number of diversity, and the lack of some data sets within the dataset. ABIDE 1 dataset is accessible on various websites of organizations operating in this field. Since these websites belonging to independent communities from each other receive the data independently, the differences between the data on the websites are very critical. To make our study consistent, we used the same 871 samples from the ABIDE 1 data set [23], [24], [25]. Of the 871 subjects, 403 patients were ASD and 468 were TC patients.

## 2.2 Brain Network

As is known, a network consists of edges and nodes. Creating a network representing a brain with functional MR data is quite complex, as it is difficult to detect intricate nodes and edges [26], [27]. If edges and nodes are misidentified, the network that studying can become extremely difficult to analyze [28]. The two most commonly used methods to describe a network are ROI-based [24] and voxel-based [29] parcellation schemes. In voxel-based approaches, each voxel in the MRI data is defined as a node. In addition, the connection between each voxel is expressed as an edge [30]. In this approach, ROIs are not considered to have the main effect on brain function [31]. Additionally, in ROI parcellation schemes, the data of the human brain is split into several different ROIs. Here all ROIs are defined as nodes. There are also connections between these ROIs. These connections are called endpoints. It is possible that ROIs can be expressed anatomically. This means that ROIs correspond to several anatomical parts of the brain, namely the hippocampus, perirhinal cortex, fissure, pons, and the like [31]. Anatomical atlases in which the cortex is separated according to anatomical features are widely used in neuroimage studies.

In this study, we used multiscale functional brain parcels. Brain parcels were created using a method known as the bootstrap analysis of stable clusters called BASC [32]. Scales were selected using a data-based method called MSTEPS [33]. There are different scales in this method. In this study, the scale is set to 122. An edge in a network of brain implies a basic connection between two nodes. An edge between a pair of ROIs is usually weighted with the Pearson correlation coefficient (PCC). For this process, time-series of the brain network obtained from rs-fMRI measurements are used. PCC  $r_{xy}$  calculated as follows:

$$r_{xy} = \frac{\sum_{b=1}^s (x_b - x_m)(y_b - y_m)}{\sqrt{\sum_{b=1}^s (x_b - x_m)^2} \sqrt{\sum_{b=1}^s (y_b - y_m)^2}} \quad (1)$$

Where,  $x$  and  $y$  is time series,  $s$  is length,  $x_b$  and  $y_b$  are the  $b$ -th components of time series, also  $x_m$  and  $y_m$  are means of the time series  $x$  and  $y$ . In addition, the pre-processed version of the ABIDE 1 data set was used in this study. Information of the pre-processing phases are given in study [34].

## 2.3 Recurrent Neural Network

Artificial neural networks have a variety of implementation fields. Over time, various versions of artificial neural networks have been specialized in analyzing different types of data. For example, CNNs are specialized for the processing of matrix-type data such as images, whereas recurrent neural networks (RNNs) have also been developed for the processing of sequence data [35]. Traditional feed-forward neural networks (FFNNs) take only existing samples to which they are exposed as input. Differently, RNNs apply what they perceive over time as input as well as existing samples.

The input sequence is given as  $[x_1, x_2, \dots, x_k]$  and  $x_i \in \mathbb{R}^d$  is allowed. Here the value of  $k$  can vary depending on the length of the sequences. For each step of the RNN model, it generates a hidden state of the sequence  $[h_1, h_2, \dots, h_k]$ . Where,  $h_{t-1}$  is the previous hidden state and  $x_t$  is the current input, the activation of the hidden state at time  $t$  can be written as:

$$h_t = f(x_t, h_{t-1}) \quad (2)$$

Unlike traditional FFNNs, RNNs have a repeating layer. Through this layer, the state information generated by the FFNN is stored and reapplied to the network along with the input information. In other words, RNNs have a memory that keeps what has been calculated until now [36]. Figure 1 shows the representation of the RNN.

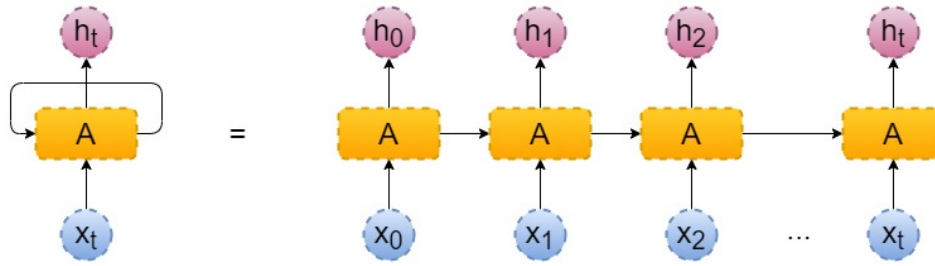


Figure 1 Recurrent neural network representation.

## 2.4 Long Short-Term Memory Network

LSTM is a special form of RNN. It can learn long term dependencies. This model was first proposed in the mid-90s [36] and is widely used today [37]. As mentioned in the above sections, RNNs aim to store and transfer the state information of the artificial neural network while working on arrays. However, it is often not possible to transfer it without breaking long-term dependencies due to the constant processing and transmission of state information.

Although it aims to store and transfer the state information of the artificial neural network while operating on the arrays in RNNs, it is not possible to transfer the status information without breaking the long-term dependencies due to the continuous processing and transfer of the state information. This means the short-term dependencies in the sequence can be effectively transferred. However, there is a problem in the transfer of long-term dependencies. LSTMs are a special model designed to solve this problem.

All RNN-based networks consist of repetitive structures as a chain. In standard RNNs, each of these structures usually consists of a tanh layer or a similar single layer. Figure 2 shows the internal structure of a standard RNN. The feature that distinguishes LSTMs from standard RNNs is that their internal structure is different, as seen in Figure 3.

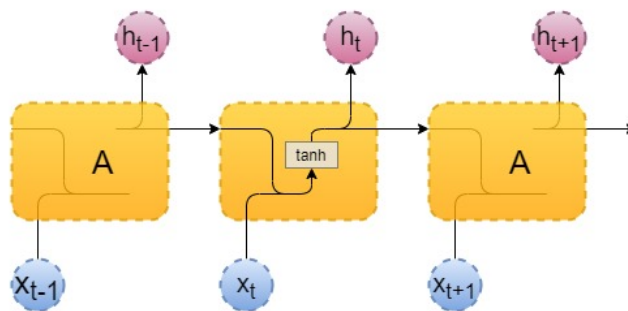


Figure 2 Standard RNN module structure with a single layer.

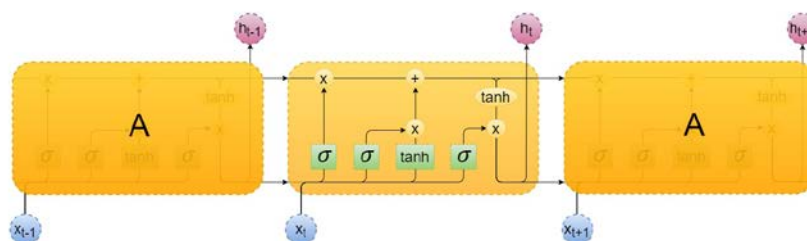


Figure 3 Recurring LSTM module structure with four interactive layers.

An LSTM module consists of input, forget and output gates. The forget gate uses a sigmoid function. This function takes a value between 0 and 1. At state 0, no information is transmitted. 1 means that all must be transmitted. The mathematical model of this gate can be expressed as:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (3)$$

Then another sigmoid function decides what information needs to be updated. In addition, a list of candidate values is created and the two operations are combined. Where  $\tilde{C}_t$  is a list of candidate values, these two operations can be expressed mathematically as follows:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (4)$$

$$\tilde{C} = \tanh(W_C[h_{t-1}, x_t] + b_C) \quad (5)$$

The new status information of the memory cell can be calculated as follows:

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t \quad (6)$$

Finally, the output of the system  $h_t$  is calculated as follows:

$$h_t = o_t \tanh(C_t) \quad (7)$$

## 2.5 Bidirectional RNN Models

In the standard RNN and LSTM models, the representations of previous time steps are learned. In some of its applications, the content may need to be learned in its future representations in order to better grasp the content and remove ambiguities. It is performed in both processes in bidirectional models. The main difference is that the forward propagation process is calculated in two steps. First of all, values are calculated from the first time step to the last time step. Then, starting from the last time step, the values are calculated up to the first time step.

## 2.6 Convolutional Neural Networks

CNN, a type of multiple layered neural network model, was first recommended for computer based vision problems [35,38] and has been used successfully in many image-related applications. CNN's are largely similar to feed-forward neural networks but show differences in the connection between neurons in neighbor layers. CNN models are commonly used by convolution, fully connected and pooling layers.

The size of filter and the generated quantity of maps are used to identify the convolution layer. This layer is the most basic and important unit that forms the CNN. The convolution layer neurons are linked to just a miniature piece of the inlet and extend across the whole measurement of the data applied to the inlet. Then, in the forward propagation phase, dot multiplication is performed between the input data and the filters and a 2-dimension of activation map is created. However, the information required to learn the CNN model is generally nonlinear, but the convolution process is a method involving linear operations such as matrix multiplication and others. Therefore, the non-linearity of the CNN model is improved by applying ReLU (Rectified Linear Units), which is an unsaturated activation function. ReLU can be expressed as:

$$f(x) = \begin{cases} x, & x \geq 0 \\ 0, & otherwise \end{cases} \quad (8)$$

Also, another important layer is the pooling layer in which down-sampling operations are performed. There are many pooling methods used in the literature, but the two most commonly used methods are maximum pooling and average pooling. In this study, maximum pooling was used. With this method, the input split into non-overlapping rectangles, and only the maximum value is taken from each sub-part. The neurons belonging to the fully connected layer are connected to every neuron in the layers before and after the layer, as in traditional artificial neural networks, and may use in the output layer with softmax or another classifier.

### 3. Model Setup & Evaluation Metrics

#### 3.1 Model Parameters

In order to compare our study with existing studies, 871 samples in the ABIDE I data set were used [23–25, 39]. While 403 records are associated with patients diagnosed with ASD, the remaining 468 samples constitute the control group.

In this study, 7 different models including standard RNN, LSTM, CNN and these were used for ASD detection. A total of 7381 features were obtained as described in the sections above. Features were applied by adjusting to 61x121 dimensions in all models.

The proposed CNN model consists of 4 learnable layers. The first layer contains a convolution layer of 32 filters of 3x3 dimensions. In this layer, ReLU is used as the activation function. Next is a dropout layer with a value of 0.2. After the dropout layer, there is a 2x2 max-pooling layer. After these layers, there are convolution, dropout, and max-pooling layers, respectively. All these layers have the same property values as the first layers. This layer's output is implemented as an entry into a fully connected layer consisting of 150 nodes. The activation function of the fully connected layer is sigmoid. After this stage, there is another dropout layer, which is the value of 0.2. At the last stage, there is a fully connected layer consisting of 1 node and having a sigmoid activation function.

In the CNN-RNN model, unlike the CNN model given above, instead of a fully connected layer consisting of 150 nodes, there is an LSTM layer consisting of 150 nodes to perceive temporal dependencies. All other values are the same as for the CNN model. The CNN-BiRNN model also has a BiRNN layer instead of the RNN layer.

The first layers of the RNN and LSTM models are respectively RNN and LSTM layers, consisting of 150 units. The activation function of the RNN and LSTM layers is the Scaled Exponential Linear Unit (SELU). With  $\lambda$  and  $\alpha$  constant values, the SELU function can be expressed as:

$$selu(x) = \lambda \begin{cases} x & \text{if } x > 0 \\ \alpha e^x - \alpha & \text{if } x \leq 0 \end{cases} \quad (9)$$

After this layer, there is a dropout layer of 0.2. The last layer contains a fully connected layer consisting of 1 node. Detailed parameters of models are given in Table 1.

#### 3.2 Performance Evaluation Metrics

The rs-fMRI images used in our study consist of two separate classes; these are the patients diagnosed with ASD and the control group. Therefore, a binary classification process is carried out. Several different metrics were used to evaluate the performance of models. The performance evaluation criteria used in this study 10-fold cross-validation, classification accuracy (ACC), sensitivity (SENS) and specificity (SPEC) analysis.

Tests on the ABIDE I data set were carried out using the 10-fold cross-validation method. The main purpose of this process is to use each data for both training and testing.

The formulas used to calculate the classification accuracy can be expressed as follows, where  $n$  is the test data set, "cn" is the class of  $n$  value, Estimate ( $n$ )  $n$  is the result of the classification process, and  $k$  is the total number of groups in the data set.

$$Accuracy(N) = \frac{\sum_{i=1}^{|N|} estimate(n_i)}{|N|}, \quad n_i \in N \quad (10)$$

$$Estimate(n) = \begin{cases} 1, & \text{if } estimate(n) = cn \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

$$\text{Classification Accuracy}(ML) = \frac{\sum_{i=1}^{|k|} \text{Accuracy}(N_i)}{|k|} \quad (12)$$

ACC alone may not provide enough information to perform the performance evaluation of the classification process.

Therefore, additional evaluations, such as SENS and SPEC, can help evaluate the final performance. Here, the SENS term gives information about the proportion of correctly predicted positive classes, while the SPEC term provides information about the correctly predicted negative classes.

Here, true positive (TP), ASD decision should be made while ASD diagnosis should be made, true negative (TN), control decision should be true negative (TN), false positive (FP), ASD decision when control sample should be, and false negative (FN) SENS and SPEC values can be expressed as follows, including those who were mistakenly decided to control while ASD should be diagnosed. Sensitivity and specificity can be expressed as follows:

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (13)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (14)$$

Table 1 Detailed scanning parameters of models

	CNN	CNN-RNN	RNN	LSTM
1. Layer	Convolutional Layer Filter Count: 32, Filter Size: 3x3 Activation: ReLU	Convolutional Layer Filter Count: 32, Filter Size: 3x3 Activation: ReLU	RNN Layer Node Count: 150 Activation: SELU	LSTM Layer Node Count: 150 Activation: SELU
2. Layer	Dropout Layer Value:0.2	Dropout Layer Value:0.2	Dropout Layer Value:0.2	Dropout Layer Value:0.2
3. Layer	Max-pooling Size: 2x2	Max-pooling Size: 2x2	Fully Connected Layer Node Count: 1 Activation: Sigmoid	Fully Connected Layer Node Count: 1 Activation: Sigmoid
4. Layer	Convolutional Layer Filter Count: 32, Filter Size: 3x3 Activation: ReLU	Convolutional Layer Filter Count: 32, Filter Size: 3x3 Activation: ReLU		
5. Layer	Dropout Layer Value:0.2	Dropout Layer Value:0.2		
6. Layer	Max-pooling Layer Size: 2x2	Max-pooling Layer Size: 2x2		
7. Layer	Fully Connected Layer Node Count: 150 Activation: Sigmoid	RNN Layer Node Count: 150 Activation: SELU		
8. Layer	Dropout Layer Value:0.2	Dropout Layer Value:0.2		
9. Layer	Fully Connected Layer Node Count: 1 Activation: Sigmoid	Fully Connected Layer Node Count: 1 Activation: Sigmoid		

#### 4. Results and Discussion

Early diagnosis of ASD is very important in keeping the progression of the disease under control. On the other hand, today's diagnostic methods are based on observing behavioral disorders caused by the disease. At this stage, the disease has already caused permanent, irreversible damage. In this respect, it is very important to diagnose ASD at an early stage using medical imaging techniques instead of following behavioral disorders [12,13]. In this study, a study on ASD detection on rs-fMRI data is presented. Compared to many techniques such as rs-fMRI, MEG, PET and EEG, it provides much better spatial resolution and an acceptable temporal resolution.

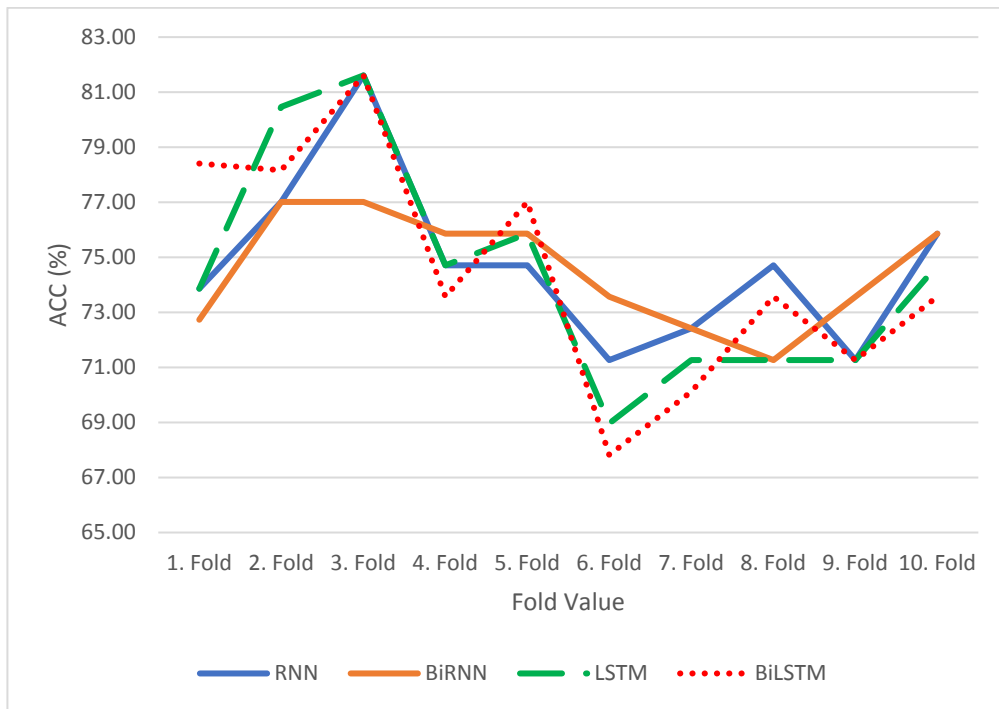
Table 2 Average classification performances

METHOD	ACC	SENS	SPEC
RNN	<b>74.74</b>	<b>72.95</b>	76.28
BiRNN	74.51	68.73	79.49
LSTM	74.4	66.25	<b>81.41</b>
BiLSTM	74.51	72.21	76.5
CNN	70.26	66.01	73.93
CNN-RNN	67.28	57.57	75.64
CNN-BiRNN	67.51	65.51	69.23

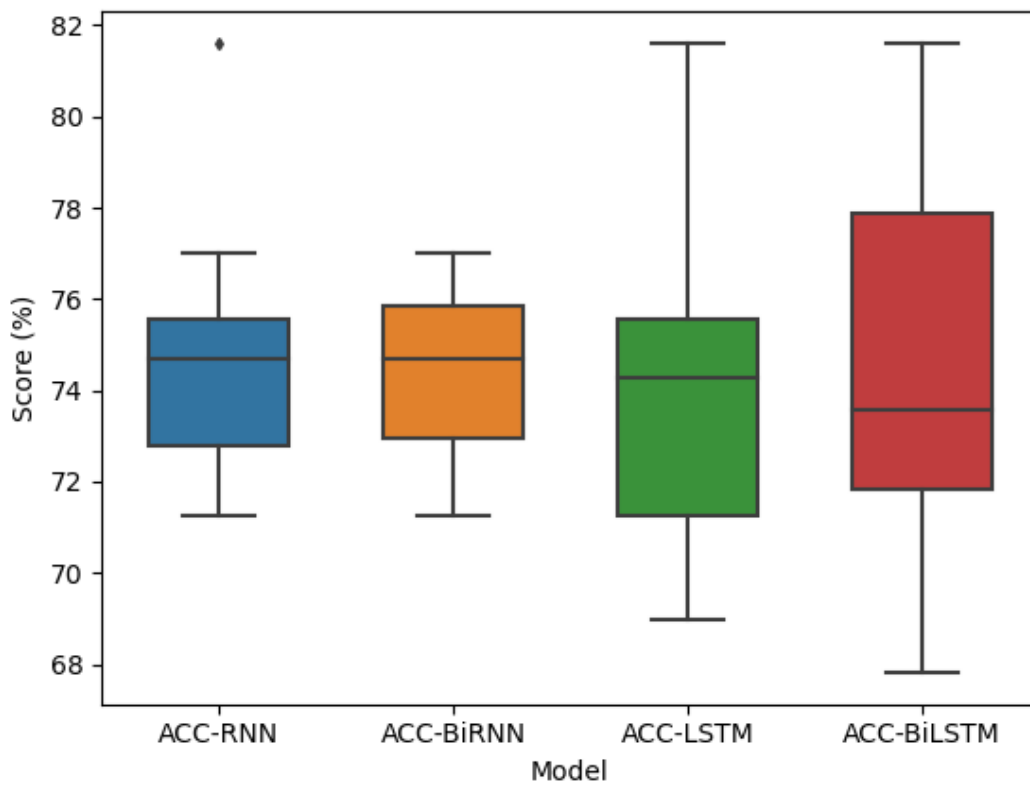
Table 2 shows the average classification performances obtained by 10-fold cross-validation tests. When the classification results are examined, it is seen that the highest performance value is provided by RNN with 74.74%. Similarly, the best score with 72.95% in the SENS value was again provided by RNN. On the other hand, in SPEC value, the best score belongs to LSTM model. The SPEC value for LSTM was 81.41%. In general, it is seen that the classification performance of RNN and LSTM based models are very close to each other. On the other hand, the performance of CNN-based models is significantly lower than other models. The classification success for CNN, CNN-RNN and CNN-BiRNN models were 70.26%, 67.28% and 67.51%, respectively. While CNNs are very good at capturing spatial dependencies, RNN-based networks are good at capturing temporal dependencies [30]. Depending on the brain network used here, it can be evaluated that temporal dependencies are significantly more dominant. On the other hand, it is seen that the performances of all LSTM and RNN models are close to each other in terms of classification performance. As mentioned in the above sections, LSTM is essentially a special type of standard RNN model. It is successful in eliminating the disadvantages of standard RNN models in capturing long-term dependencies. When the current results obtained with RNN and LSTM models are evaluated, it can be evaluated that long-term dependencies do not affect the classification performance too much. In addition to these, it is seen that the BiRNN and BiLSTM models do not change their classification performance much. However, it appears to affect SENS and SPEC values.

As seen in Graphic 1 and 2, the classification performance of RNN, BiRNN, LSTM and BiLSTM models are very close to each other. However, the SENS values of RNN and BiLSTM models are significantly better than other models. In Graphic 3, box plot representations of SENS values of 4 different models can be seen. Here, the SENS value will decrease if a person who needs to be diagnosed with the disease is mistakenly called healthy. That is, the decrease in the number of FNs is more critical in terms of preventing permanent body damage that may be caused by the disease at an early stage. Graph 1 shows the comparison of ACC values on a fold basis. Also, in Graph 2, boxplot representation of ACC values obtained from RNN, BiRNN, LSTM and BiLSTM models can be seen. When the results are evaluated, it is seen that the RNN model provides stable results in a narrower range. When the values given in Graph 2 are examined, it is seen that the behaviors of the SENS values of RNN and BiLSTM models are close to each other. However, it is seen that the peak values obtained for SENS in the RNN model are better. As a result, it can be evaluated that long-term dependencies do not have a significant effect on performance for the current features used.

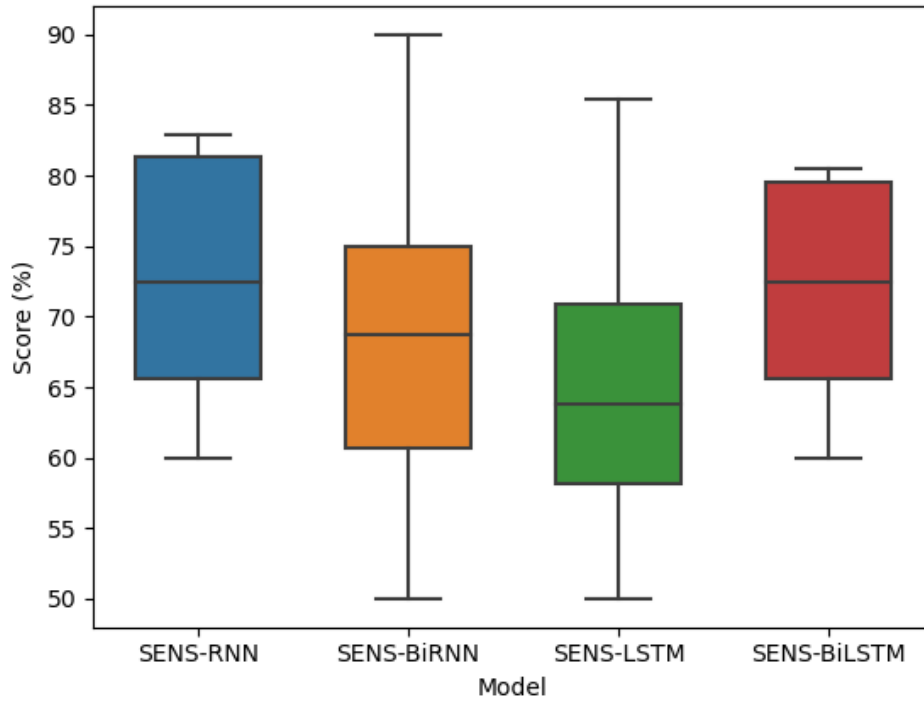




Graph 1 Comparison of ACC values on a fold basis

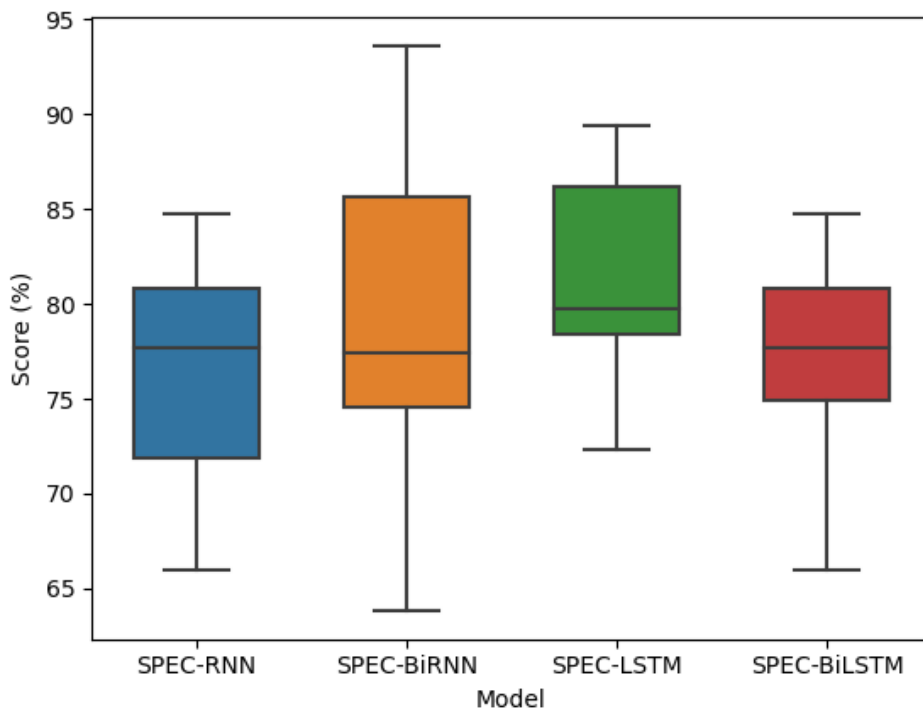


Graph 2 Box plot representation of ACC values on a fold basis



Graph 3 Box plot representation of SENS values on a fold basis

Graph 4 shows a comparison of SPEC values for four models. The highest performance at the average SPEC value belongs to the LSTM model. In the RNN, LSTM, and BiLSTM models, values closer to each other were obtained on the basis of each fold. In the BiRNN model, the lowest and highest values were realized in a wider range than other models. In addition, the SPEC values of all models are higher than SENS values. It is considered that the reason for this is that machine learning models are generally prone to the majority class.



Graph 4 Box plot representation of SPEC values on a fold basis

Finally, we compare our study with other studies in the literature. Table 3 shows the comparison of the models used in this study with other studies in the literature. There are a total of 1112 samples in the ABIDE 1 data set. However, instead of using the entire data set, studies are carried out using different numbers of samples in line with certain criteria. In addition, the properties and classification methods used may differ. In order to make a healthy comparison, we list the sections mentioned above in Table 3. In addition to these, different validation methods can be used in ASD studies. All of the studies shown in Table 3 used 10-fold cross-validation. When the results given in Table 3 are evaluated, it is seen that the highest performance value was obtained in the study performed using power atlas and linear discriminant analysis (LDA) [39]. In the proposed method, a power brain atlas was applied to the pre-processed data. Then, correlation matrices were created using PCC. The feature selection process was applied to the 264x264 dimensional representations created and finally classified with LDA. Here, the performance value was obtained as 77.7% [39]. The lowest performance value given in Table 3 is Abraham et al. It was obtained in the study performed by [24]. They used more than one brain atlas for ASD detection [24]. As a result, they reached a performance value of 66.8% in the study [24]. In this study, the second-highest value among the results given in Table 3 was obtained by using a total of 7381 features in 61x121 dimensions. These results confirm the success of RNN-based networks in capturing temporal correlations. It is better than this study with the value obtained in [39]. However, it is evaluated here that the main factor affecting the performance is the extracted features rather than the classification algorithm. In future studies, it is planned to test the models used in this study on different features.

Table 3: Comparison of ACC values with other studies in the literature

Authors	Features	Classification Method	Subject Number	Acc. (%)
Wong et al. [23]	Harvard-Oxford Atlas / Riemannian and Correlation Matrix	Logistic Regression- $l_2$	871	71.7
Dvornek et al. [40]	Craddock 200 Atlas / Phenotypic Data	LSTM	1100	70.1
Heinsfeld et al. [41]	Craddock 200 Atlas / PCC	Denoising Autoencoder, Deep Neural Network	1035	70
Abraham et al. [24]	Multiple Atlas (Harvard-Oxford, Structural, Yeo Functional, Craddock 200 etc.)	Support Vector Classifier- $l_2$	871	66.8
Khosla et al. [43]	Multiple Atlas (Harvard-Oxford, Craddock 200, Craddock 400, Dosenbach 160, Eickhoff-Zilles etc.)	3D CNN	774	73.3
Parisot et al. [25]	Harvard-Oxford Atlas / PCC	CNN	871	69.5
Mostafa et al. [39]	Power Atlas / PCC, Sequential Feature Selection	LDA	871	<b>77.7</b>
In this study	BASC Atlas, MSTEPS	Standart RNN	871	74.7
In this study	BASC Atlas, MSTEPS	BiRNN	871	74.5
In this study	BASC Atlas, MSTEPS	LSTM	871	74.4
In this study	BASC Atlas, MSTEPS	BiLSTM	871	74.5

## 5. Conclusion

In this study, an approach is presented to detect ASD on rs-fMRI data. Within the scope of the study, 871 samples in the ABIDE I data set were used. While 403 of these samples contain ASD, 468 are control samples.

rs-fMRI data provide information about brain functions in temporal and spatial domains. Accordingly, the performances of RNN-based models, which are very successful in detecting temporal relations between data, and CNN models, which are successful in capturing spatial dependencies, have been compared in this study. The highest ACC and SENS values were obtained with the standard RNN model. ACC and SENS values are 74.74% and 72.95%, respectively. The classification performance of LSTM models is also very similar to the standard RNN model. However, the performance values of the CNN model are lower compared to these models. In the CNN model, the performance value was obtained at 70.26%.

Classification performance largely depends on extracted features as well as ML algorithms. In future studies, using different brain atlases, their effects on classification performance will be evaluated.

## Acknowledgments

This study has supported by the Scientific Research Project (BAP) Coordinatorship of Bandirma Onyedü Eylül University under grant number BAP-21-1003-003.

## References

- [1] M. A. Aghdam, A. Sharifi, and M. M. Pedram, "Diagnosis of Autism Spectrum Disorders in Young Children Based on Resting-State Functional Magnetic Resonance Imaging Data Using Convolutional Neural Networks," *Journal of Digital Imaging*, vol. 32, no. 6, pp. 899-918, 2019.
- [2] B. Crosson *et al.*, "Functional Imaging and Related Techniques: An Introduction for Rehabilitation Researchers," *Journal of Rehabilitation Research and Development*, vol. 47, no. 2, pp. vii-xxxiv, 2010.
- [3] S. Sarraf and J. Sun, "Functional Brain Imaging: A Comprehensive Survey," *ArXiv Preprint*, arXiv:1602.02225, 2016.
- [4] Y. Kong, J. Gao, Y. Xu, Y. Pan, J. Wang and J. Liu, "Classification of Autism Spectrum Disorder by Combining Brain Connectivity and Deep Neural Network Classifier," *Neurocomputing*, vol. 324, pp. 63-68, 2019.
- [5] D. G. Amaral, C. M. Schumann, and C. W. Nordahl, "Neuroanatomy of Autism," *Trends in Neurosciences*, vol. 31, no. 3, pp. 137-145, 2008.
- [6] K. C. Turner, L. Frost, D. Linsenbardt, J. R. McIlroy and R. Müller, "Atypically Diffuse Functional Connectivity Between Caudate Nuclei and Cerebral Cortex in Autism," *Behavioral and Brain Functions*, vol. 2, no. 1, p. 34, 2006.
- [7] S. J. Blumberg, M. D. Bramlett, M. D. Kogan, L. A. Schieve, J. R. Jones and M. C. Lu, "Changes in prevalence of parent-reported autism spectrum disorder in school-aged US children: 2007 to 2011-2012," *National Center for Health Statistics*, no. 65, pp. 1-11, 2013.
- [8] M. Langen, S. Durston, W. G. Staal, S. J.M.C.Palmen and H. V. Engeland, "Caudate Nucleus Is Enlarged in High-Functioning Medication-Naive Subjects with Autism," *Biological psychiatry*, vol. 62, no. 3, pp. 262-266, 2007.
- [9] M. Coleman and C. Gillberg, "The Autisms," *OUP USA*, 2012.
- [10] L. Waterhouse, "Rethinking Autism: Variation and Complexity," *Academic Press*, 2013.
- [11] E. Fernell, M. A. Eriksson, and C. Gillberg, "Early Diagnosis of Autism and Impact on Prognosis: a Narrative Review," *Clinical Epidemiology*, vol. 5, pp. 33-43, 2013.
- [12] B. E. Yerys and B. F. Pennington, "How do we establish a biological marker for a behaviorally defined disorder? Autism as a test case," *Autism Research*, vol. 4, no. 4, pp. 239-241, 2011.
- [13] M. Plitt, K. A. Barnes, and A. Martin. "Functional connectivity classification of autism identifies highly predictive brain features but falls short of biomarker standards," *NeuroImage: Clinical*,

- vol. 7, pp. 359-366, 2015.
- [14] C. M. Bishop, "Pattern Recognition and Machine Learning," *Springer-Verlag New York Inc.*, Secaucus, NJ, USA, 2006.
- [15] A. L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers," *IBM J. Res. Dev.*, vol. 3, no. 3, pp. 210-229, Jul. 1959.
- [16] T. Hastie, R. Tibshirani, and J. Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction," *Springer Series in Statistic*, 2009.
- [17] M. Uçar and E. Uçar, "Computer-aided detection of lung nodules in chest X-rays using deep convolutional neural networks," *Sakarya University Journal of Computer and Information Sciences*, vol. 2, no. 1, pp. 1-8, 2019.
- [18] Y. Alakoç, V. Akdoğan, M. Korkmaz and O. Er, "Pre-Diagnosis of Osteoporosis Using Probabilistic Neural Networks," *Sakarya University Journal of Computer and Information Sciences*, vol. 1, no. 3, pp. 1-6, 2018.
- [19] E. Erdem and T. Aydin, "Detection of Pneumonia with a Novel CNN-based Approach," *Sakarya University Journal of Computer and Information Sciences*, vol. 4, no. 1, pp. 26-34, 2021.
- [20] D. B. Aydin and O. Er, "A new proposal for early stage diagnosis of urinary tract infection using computers aid systems," *Sakarya University Journal of Computer and Information Sciences*, vol. 1, no. 1, pp. 1-9, 2018.
- [21] G. Ozen, R. Sultanov, Y. Özen and Z. Y. Güneş, "A Convolutional Neural Network Based on Raw Single Channel EEG for Automatic Sleep Staging," *Sakarya University Journal of Computer and Information Sciences*, vol. 3, no. 2, pp. 149-158, 2020.
- [22] A. Di Martino *et al.*, "The Autism Brain Imaging Data Exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism," *Molecular Psychiatry*, vol. 19, no. 6, pp. 659-667, 2014.
- [23] E. Wong, J. S. Anderson, B. A. Zielinski and P. T. Fletcher, "Riemannian regression and classification models of brain networks applied to autism," *International Workshop on Connectomics in Neuroimaging*, Springer, Cham, 2018.
- [24] A. Abraham, M. Milham, A. D. Martino, R. C. Craddock, D. Samaras, B. Thirion and G. Varoquaux, "Deriving reproducible biomarkers from multi-site resting-state data: An Autism-based example," *NeuroImage*, vol. 147, pp. 736-745, 2017.
- [25] S. Parisot, S. I. Ktena, E. Ferrante, M. Lee, R. G. Moreno, B. Glocker and D. Rueckert, "Spectral graph convolutions for population-based disease prediction," *International conference on medical image computing and computer-assisted intervention*, Springer, Cham, 2017.
- [26] J. Wang *et al.*, "Parcellation-dependent small-world brain functional networks: A resting-state fMRI study," *Human brain mapping*, vol. 30, no. 5, pp. 1511-1523, 2009.
- [27] A. Zalesky, A. Fornito, I. H. Harding, L. Cocchi, M. Yücel, C. Pantelis and E. T. Bullmore, "Whole-brain anatomical networks: does the choice of nodes matter?," *Neuroimage*, vol. 50, no. 3, pp. 970-983, 2010.
- [28] C. T. Butts, "Revisiting the foundations of network analysis," *Science*, vol. 325, no. 5939, pp. 414-416, 2009.
- [29] M. Jenkinson, P. Bannister, M. Brady and S. Smith, "Improved optimization for the robust and accurate linear registration and motion correction of brain images," *Neuroimage*, vol. 17, no. 2, pp. 825-841, 2002.
- [30] Z. Long, X. Duan, D. Mantini, and H. Chen, "Alteration of functional connectivity in autism spectrum disorder: effect of age and anatomical distance," *Scientific Reports*, vol. 6, no. 1, pp. 1-8, 2016.
- [31] P. Fransson, U. Aden, M. Blennow and H. Lagercrantz, "The functional architecture of the infant brain as revealed by resting-state fMRI," *Cerebral Cortex*, vol. 21, no. 1, pp. 145-154, 2011.
- [32] P. Bellec, P. Rosa-Neto, O. C. Lyttelton, H. Benali, A. C. Evans, "Multi-level bootstrap analysis of stable clusters in resting-state fMRI," *Neuroimage*, vol. 51, no. 3, pp. 1126-1139, 2010.
- [33] P. Bellec, "Mining the hierarchy of resting-state brain networks: selection of representative clusters in a multiscale structure," *International Workshop on Pattern Recognition in Neuroimaging, IEEE*, 2013.
- [34] C. Craddock *et al.*, "The neuro bureau preprocessing initiative: open sharing of preprocessed

- neuroimaging data and derivatives," *Frontiers in Neuroinformatics*, 2013.
- [35] A. T. Kabakuş, "A Comparison of the State-of-the-Art Deep Learning Platforms: An Experimental Study," *Sakarya University Journal of Computer and Information Sciences*, vol. 3, no. 3, pp. 169-182, 2020.
- [36] Z. Özer, "The Effect of Normalization on the Classification of Traffic Comments," *Ph. D. Thesis*, Karabük Univ. Grad. Sch. of Nat. and App. Sci, Dept. of Computer Engineering, Karabük, Turkey, 2019.
- [37] A. A. Müngen, İ. Aygün, and M. Kaya, "News and Social Media Users Emotions in the COVID-19 Process," *Sakarya University Journal of Computer and Information Sciences*, vol. 3, no. 3, pp. 250-263, 2020.
- [38] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998.
- [39] S. Mostafa, L. Tang, and F-X. Wu, "Diagnosis of Autism Spectrum Disorder Based on Eigenvalues of Brain Networks," *IEEE Access*, vol. 7, pp. 128474-128486, 2019.
- [40] N. C. Dvornek, P. Ventola, and J. S. Duncan, "Combining phenotypic and resting-state fMRI data for autism classification with recurrent neural networks," *IEEE 2018 15th International Symposium on Biomedical Imaging, IEEE*, 2018.
- [41] A. S. Heinsfeld, A. R. Franco, R. C. Craddock, A. Buchweitz and F. Meneguzzi "Identification of autism spectrum disorder using deep learning and the ABIDE dataset," *NeuroImage: Clinical*, vol. 17, pp. 16-23, 2017.
- [42] H. Sharif and R. A. Khan, "A novel framework for automatic detection of autism: A study on corpus callosum and intracranial brain volume," *arXiv preprint*, arXiv:1903.11323, 2019.
- [43] M. Khosla, K. Jamison, A. Kuceyeski, and M. R. Sabuncu, "3D convolutional neural networks for classification of functional connectomes," *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, Springer, Cham, pp. 137-145, 2018.

# Designing a Data Warehouse for Earthquake Risk Assessment of Buildings: A Case Study for Healthcare Facilities

 Mert Özcan<sup>1</sup>,  Serhat Peker<sup>2</sup>

<sup>1</sup>Corresponding Author; Izmir Bakircay University, Department of Management Information Systems; mert.ozcan@saglik.gov.tr; +90 312 4718350

<sup>2</sup>Izmir Bakircay University, Department of Management Information Systems; serhat.peker@bakircay.edu.tr

Received 1 February 2021; Revised 16 March 2021 Accepted 24 March 2021; Published online 26 April 2021

## Abstract

Since earthquake is one of the most dangerous natural phenomena, it is necessary to be prepared for the negative consequences of the earthquake in advance. It is very important that healthcare facilities must continue to provide service during and after an earthquake. Therefore, this study focuses on designing a data warehouse model for the earthquake risk assessment of healthcare facilities which are needed much more than other public buildings physically. The proposed design utilizes a fact constellation schema model and take a public legislation containing principles regarding identification of risky buildings. This solution can provide a repository for data regarding earthquake risk assessment from different operational systems and play a key role in supporting critical decision-making process.

**Keywords:** Data warehouse, data management, multidimensional data model, data mining, decision support systems, business intelligence and analytics

## 1. Introduction

Earthquake is one of the most dangerous natural disasters which causes not only damages to buildings but also a great number of deaths and injuries because of building collapses. Although it is not possible to prevent this natural phenomenon itself, various technological developments and approaches show that many of deaths and injuries are preventable. Turkey has been exposed to major and destructive earthquakes for centuries [1]. 92% of the country is under the risk of earthquake, thus, the earthquake risks may affect the 95% of the population in the country. Although public authorities make legislative efforts, the quality of building stock in Turkey is not at the desired level for such disasters yet [2]. Because of that, earthquake is a critical threat for people living in this country. It is clear that, healthcare facilities have higher risks than other types of buildings due to their high importance [3]. Thus, assessing healthcare facilities is quite necessary for disaster preparedness process.

Turkey is among the top 10 OECD countries in terms of total physicians, hospitals number and hospital beds, etc. [4]. These rankings indicate that, Turkey has an important health infrastructure among the developing and developed countries. Health services are currently provided by over 65.000 healthcare facilities, including hospitals, family practice centers, clinics, laboratories, emergency medical service stations, etc. These healthcare facilities are constructed by Ministry of Health or used by renting. For instance, there are over 40.000 private clinics and family practice centers in Turkey, however the most of them are not constructed by Ministry of Health itself. On the other hand, approximately 650.000 healthcare professionals serve healthcare at such a large number of healthcare facilities [5]. Together with patients, millions of people are constantly served by these personnel in the healthcare facilities. However, there is insufficient evidence for the overall systematic assessment of these facilities in terms of earthquake preparedness. Therefore, a systematical perspective is considered to be needed for earthquake risk assessment of the healthcare facilities.

Risk assessment of buildings is a complex process which relies on a series of evaluation criteria with multiple parameters. To manage this process with massive amounts of data from various sources, a

comprehensive data warehouse model is needed. Data Warehouse (DW) term can be defined as multi-dimensional databases that involve large amount of data [6]. Data warehouses have major features such as integrated, subject-oriented, time variant and nonvolatile support for decision making [7]. It provides architectural tools for decision makers in order to organize, understand and use their data to make strategic decisions [8]. In addition to these, many enterprise information systems utilize data warehouses to enhance business intelligence and analytics by implementing diverse data mining techniques [9].

Based on the aforementioned benefits, data warehouses can become the repository for complex process of earthquake risk assessment for buildings. Therefore decision makers can make faster and better decisions on urgent improvement works for earthquake countermeasures. In this context, the study aims to design a data warehouse model for earthquake risk assessment of healthcare facilities in Turkey. A fact constellation schema model is devised based on a public legislation.

The rest of the paper is organized as follows: Section 2 provides an overview of the background of data warehouses and related works. Section 3 presents the proposed data warehouse design with details of the methodology and sample queries. Finally, Section 4 presents concluding remarks and future work directions.

## 2. Background

### 2.1. Data Warehouse

Data warehouses have been almost indispensable for decision support systems, data mining, business analysis, forecasting and business intelligence since “Business Data Warehouse” term took part in the literature in 1988 [10]. Data warehouses are data management systems that are designed for specific purposes. By using specific queries and analyzing tools in a more efficient way, data warehouses provide great convenience both for IT professionals and end users. One of the most important features of data warehouses for business intelligence that integrated and purposive data are stored rather than detailed and individual ones [11]. Therefore, it is possible to have higher performance, safer, more reliable, more retrievable and more manageable systems.

Success of the selected data warehouses approach relates how data modeling techniques are applied. As a logical design approach, dimensional models are usually chosen for data warehouses. As an alternative approach to traditional entity relationship model, dimensional models have more advantages for decision supports. In this approach, there are fact tables and dimension tables. Fact tables have numerical and additive measurements for specific requirements of a business. Dimension tables are created for defining business entities. Dimension tables are linked with fact table with their primary keys [12]. By this approach, it is aimed to obtain easier structures for end users and to have more efficient queries. For this reason, less tables and relations are preferred with less joins in queries [13].

The most demanded and well-known methods for applying multidimensional modeling are star schema, snowflake schema and fact constellation schema. Star schema is created by linking a series of dimensional tables which generally consist of embedded hierarchies around a fact table. Star schema has one large central table which is called fact table. The fact table is linked with smaller dimension tables. The dimension tables surround the fact table. The fact table may be linked with numerous dimension tables by one to many relationships in star schema [13]. In snowflake schema, dimension tables are linked to another dimension table without being linked directly to the fact table [14], [15]. Thus, snowflake schema differs from star schema with including more hierarchies rather than only one hierarchy. The third dimensional model which has more than one fact table is called fact constellation schema. This schema can be considered as a combination of two or more star schemas. One of the typical features of the fact constellation schema is flexibility. The fact constellation schema has more than one fact tables. These fact tables share several dimension tables. Although modeling process is more complex than the other models, the complexity brings to ability to have more accurate outputs [16]. This model is also called galaxy model in the literature.



## 2.2. Related Work

Before starting the design process, different dimensional data warehouses and their design aspects were reviewed. The review on current literature has showed that the data warehouses are raising their own significance by creating more effective decision-making tools in order to support end user queries.

Different solutions have been proposed so far for data warehouses, most of them based on disaster management. Asghar et al. [17] introduced a business intelligence system that links dimensions of business intelligence and processes together, for providing a good decision support, for disaster management organization as a case study. Their data warehouse model has three main fact tables and two dimensions tables. According to the validated experimental results of the study, proposed business intelligence system performs for decisions on disaster management efficiently.

Panrungsri and Sangiamkul [18] applied a business intelligence model for disaster management area by applying a case study. They design and develop a data warehouse with multidimensional model for severity analysis disasters in risk areas. The aim of the study is to have a business intelligence concept by improving data quality and exposing to better decision making for disaster management. It is mentioned that the proposed data model with fact constellation schema can enable agility capacity for business intelligence technology in disaster management processes.

Permana et al. [19] has recently proposed a data warehouse for disaster management. The proposed data warehouse in this research was designed as a snowflake schema based on information about kind of disaster, place and the effect of disaster. The feature of Online Analytical Processing (OLAP) for disaster report can be a significant tool for decision makers in order to manage future risk of the disasters. In another recent study, a web-based assessment software has been developed in order to describe risk priorities of concrete buildings [20]. In this study, a rapid assessment method issued by the Ministry of Environment and Urbanization was employed.

Sarı and Türk [21] has built a database to investigate building damages caused by earthquakes in Turkey. They measured the building qualities and illustrated in proposed Geographical Information System (GIS). The aim of this study was to determine the earthquake risk levels of buildings in selected area. Building data were created in accordance with the Federal Emergency Management Agency (FEMA) approach in this research. The data were obtained from the municipality and transferred to GIS environment. Then the vulnerability and quality levels of buildings were determined in Sivas Municipality City Information System in a possible earthquake.

Considering the studies mentioned above, although especially in recent years, several research studies have relied on many different data models in disaster management, there has not been any prior study that mainly focuses on building a comprehensive data warehouse model for earthquake risk assessment. In this regard, designing a data warehouse focusing on earthquake risk assessment of buildings can significantly contribute to relevant literature.

## 3. Building Data Warehouse

This study is based on principles of identification of risky buildings which are published by Turkish Environment and Urban Ministry [22]. This legislation includes calculations to classify buildings in terms of earthquake risk levels by considering certain parameters. These parameters affecting the risk assessment of buildings are as follows:

- Structural system type;
- Number of floors;
- Visual quality;
- Soft storey / weak storey;
- Vertical irregularity;
- Heavy overhang;
- Torsion effect;
- Short column effect;
- Construction Regularity;

- Ground slope;
- Earthquake danger zone.

Each building is assessed according to three sub assessment calculations which are called as negativity, ground system and structural system. For negativity calculation, each building is given a value according to certain eight parameters as shown in Table 1. Buildings are given 0, 1 or 2 value according to the negativity parameter detection case. Supposing that; if the visual quality of the building is determined as 'good', then the negativity value of "Visual Quality" parameter will be 0. If it is determined as 'moderate' or 'poor', then the parameter value will be 1 or 2, respectively. As a result of negativity parameter assessment, each building will have eight parameter values for eight negativity parameters on aggregate.

Table 1 Negativity Parameter Values (Ni)

No	Negativity Parameter	Case 1		Case 2	
		Parameter Detection	Parameter Value ( O <sub>i</sub> )	Parameter Detection	Parameter Value ( O <sub>i</sub> )
1	Visual Quality	Good	0	Moderate (Poor)	1 (2)
2	Soft Storey	No	0	Yes	1
3	Vertical Irregularity	No	0	Yes	1
4	Heavy Overhang	No	0	Yes	1
5	Torsion Effect	No	0	Yes	1
6	Short Column Effect	No	0	Yes	1
7	Construction Regularity	Detached	0	Attached	1
8	Ground Slope	No	0	Yes	1

Each negativity parameter has negativity parameter score according to the number of floors of the building as shown in Table 2. For instance, if a building has 5 floors, then the negativity score of visual quality will be measured as -25 for that building. To put it another example; if a building has 7 floors, then the negativity score of torsion effect will be -10 for that building.

Table 2 Negativity Parameter Scores (NPi)

Number of Floors	Visual Quality	Soft Storey	Vertical Irregularity	Heavy Overhang	Torsion Effect	Short Column Effect	Construction Regularity				Ground Slope
							Same		Different		
							Center	Side	Center	Side	
1,2	-10	-10	-5	-10	-5	-5	0	-10	-5	-15	-3
3	-10	-20	-10	-20	-10	-5	0	-10	-5	-15	-3
4	-15	-30	-15	-30	-10	-5	0	-10	-5	-15	-3
5	-25	-30	-15	-30	-10	-5	0	-10	-5	-15	-3
6,7	-30	-30	-15	-30	-10	-5	0	-10	-5	-15	-3

Overall, Ground System Score (GSS) and Structural System Score (SSS) are measured for each building, as shown in Table 3. Supposing that, a building has 3 floors, and it was built on a ground which is defined as danger zone 1. Then the GSS of that building will be measured as 80. Similarly, if the same building was built as reinforced concrete shear wall frame, then SSS of that building will be measured as 85. Otherwise, SSS will be measured as 0 for that building.

Table 3 Ground and Structural System Scores

Number of Floors	Ground System Score (GSS)				Structural System Score (SSS)	
	Danger Zone				Reinforced Concrete Frame (RCF)	Reinforced Concrete Shear Wall Frame (RCSWF)
	I	II	III	IV		
1,2	90	120	160	195	0	100
3	80	100	140	170	0	85
4	70	90	130	160	0	75
5	60	80	110	135	0	65
6,7	50	65	90	110	0	55

Total Score: Total Score indicates what the final assessment result of specific building is and gives information about calculated performance point. The collected data is evaluated, and a total score (TS) is calculated for each building. Total scores are used to determine the risk priorities of the buildings as well. Thus, Total Score in our model is calculated for each building as seen in Equation 1:

$$\text{Total Score} = \text{GSS} + \sum_{i=1}^8 (\text{Ni} * \text{NPi}) + \text{SSS} \quad (1)$$

where GSS represents ground system score of building, SSS reflects the structural system score of building, Ni denotes the value corresponding to the  $i^{\text{th}}$  negativity parameter of building and NPi refers to the score corresponding to the  $i^{\text{th}}$  negativity parameter of building. In this equation,  $i$  is set from 1 to 8 reflecting negativity parameters that are shown in Table 1.

The main aim of this method is to have general overview for the buildings as rapid as possible, therefore broader and more detailed assessment methods can be applied based on data that are gathered from this method if needed.

The next sub-sections describe the methodology of the proposed informative database along with its architecture for building data warehouse, logical model schema and sample SQL queries.

### 3.1. Methodology

There are four architectural processing stages as illustrated in Figure 1. In the first stage, data extracted from various databases are loaded into the staging area [11]. Then data are stored in data staging area for the purpose of cleansing, scrubbing, fixing data errors and transforming into a more normalized standard. In this process, it is necessary for dropping and merging tables, creating or removing columns. The third processing stage is data transferring into the data warehouse. [23] In the last stage, stored data in data warehouse are analyzed and then used for purpose of query, reporting, data mining and decision support to feed end users.

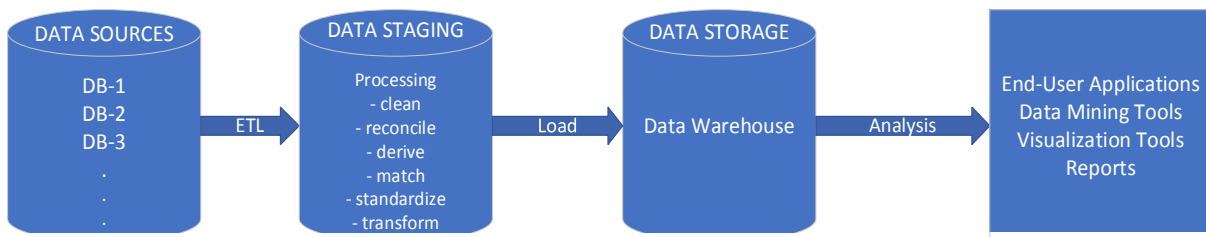


Figure 1 Architecture for Data Warehouse Framework

When designing data warehouses, multidimensional data modeling approaches have been used more often. These approaches enable to store data in a more systematical way. Moreover, they also make the data more useful for operations such as business intelligence, OLAP (On Line Analytical Processing) and data mining processes [15].

Based on the aforementioned issues, the proposed data warehouse is constructed by integrating multiple sources such as relational database systems of Ministry of Health and other public database systems. Finally, the architecture of data warehouse is designed to provide information from historic perspective and nonvolatility.

### 3.2. Data Warehouse Design

The proposed data warehouse has been designed in MS SQL database system. It contains 10 tables and 11 relationships. It contains two facts, Fact\_Risk\_Assessment and Fact\_Facility\_Negative\_Parameter\_Evaluation tables. Each fact table has primary keys and foreign keys for connecting to the dimension tables and a number of measures. The proposed fact constellation schema for “Earthquake Risk Assessment Data Warehouse” is shown in Fig-2.

In the model, it is assumed that all data are cleared, converted to the appropriate format and normalized during the implementation phase. Attributes of the fact and dimension tables are indicative. More attributes can be included in the data warehouse, containing appropriate amount of data for decision-making. Therefore, they are not included in this work for simplification purposes.

In the given dimensional table, it is aimed to collect to identify all healthcare facilities in "Dim\_Facility" table so as to analyze the facilities. 'FacilityID' serves as the primary key of the "Dim\_Facility" table, which is the unique ID that is provided for each facility and it includes all other information related to the facility. In the "Dim\_Facility" table, there are many attributes such as 'Facility\_Name', 'Address\_ID', 'Facility\_Type', 'Number\_of\_Floors', 'Indoor\_Area', 'Number\_of\_Beds', 'Number\_of\_Emergency\_Beds', 'Care\_Level', 'Building\_Ownership', 'Establishment\_Year', 'Danger\_Zone' and 'Service\_Status' that identifies healthcare facilities separately. The "Dim\_Facility" table is also referencing to "Dim\_Address" table which stores specific data of related facilities such as address name, district name, city name, seismic zone, population, etc.

"Fact\_Facility\_Negative\_Parameter\_Evaluation" represents a part of assessment results and it is directly related to principles published by Ministry of Environment and Urbanization. 'Facility\_ID', 'Date\_ID', 'Auditor\_ID', 'Negative\_Parameter\_ID' and 'Negative\_Parameter\_Value' are primary keys of the fact table. It has many to one relationship to two dimension tables; "Dim\_Negative\_Parameters" and "Dim\_Negative\_Parameter\_Value" dimensions. These two dimensions store various assessment measures and possible values for each parameter that are published by Ministry of Environment and Urbanization. It also refers to 'Dim\_Date', 'Dim\_Auditor' and 'Dim\_Facility' dimension tables in order to answer of the following questions such as 'When did the assessment occurred for each facility?', 'Who did assess for each facility?' and 'Which buildings were assessed?'. Therefore, "Fact\_Facility\_Negative\_Parameter\_Evaluation" table is an assessment table in which each result is related to a specific facility. Moreover, the fact here is negative parameter score by date, by auditor, by negative parameter and by negative parameter value.

The second fact is "Fact\_Risk\_Assessment" table. 'Facility\_ID', 'Date\_ID' and 'Auditor\_ID' are primary keys of the fact table. It is connected to three dimension tables; "Dim\_Date", "Dim\_Auditor" and "Dim\_Facility". The fact table depicts the assessment scores which would gather all related earthquake risk assessment parameters of healthcare facilities by facility, by date and by auditor. It may be possible not only to obtain total risk assessment scores of the healthcare facilities, but also to carry out wide range of analyzes on related data in the model by using business intelligence, data mining, integration techniques and various queries.

### 3.3. Sample Queries

This section includes sample queries that would run on the proposed data warehouse. Sample queries are presented in Queries 1-5. These queries illustrate how simple extracting statistical information from the proposed data warehouse model is. The sample queries can be performed for various decision processes and they are correspondingly extendable.

a) How is the distribution of healthcare facilities which are in the first-degree seismic zone according to the risk assessment score countrywide?

This query combines data from buildings where in the most dangerous areas of the country. The query is significant for having statistical data analysis based on facility, city and district.

b) How is the distribution of the health facilities in İstanbul according to the risk assessment score in the last ten years?

This query can be used to extract assessment score of buildings where in selected city and selected time period. Even if the query seems to have complexity, it is easy to express by performing five join operations in the fact constellation schema.

c) How many people will not be able to benefited from the healthcare facility which has the highest risk assessment score if it is damaged by an earthquake in the district?

In this query, a possible situation based on a scenario is considered. The query mentions the possible threat if the building which has the highest risk score cannot be used owing to an earthquake by showing affected population.

d) How is the distribution of attached and rented healthcare facilities which are in the first-degree seismic zone according to negative parameter score?

Earthquake threat may be higher for attached buildings in the most dangerous zone. Therefore, in this query, it is aimed to obtain assessment scores ordered by negative parameter scores and grouped by facility name.

e) What is the average number of risk assessment scores changes for number of floors of healthcare facilities countrywide?

This query considers the average risk assessment scores grouped by number of floors. It has an importance for having a general idea about the relationship between number of floors and risk level of the buildings. This query is thought as significant from the point of prioritizing risky buildings.

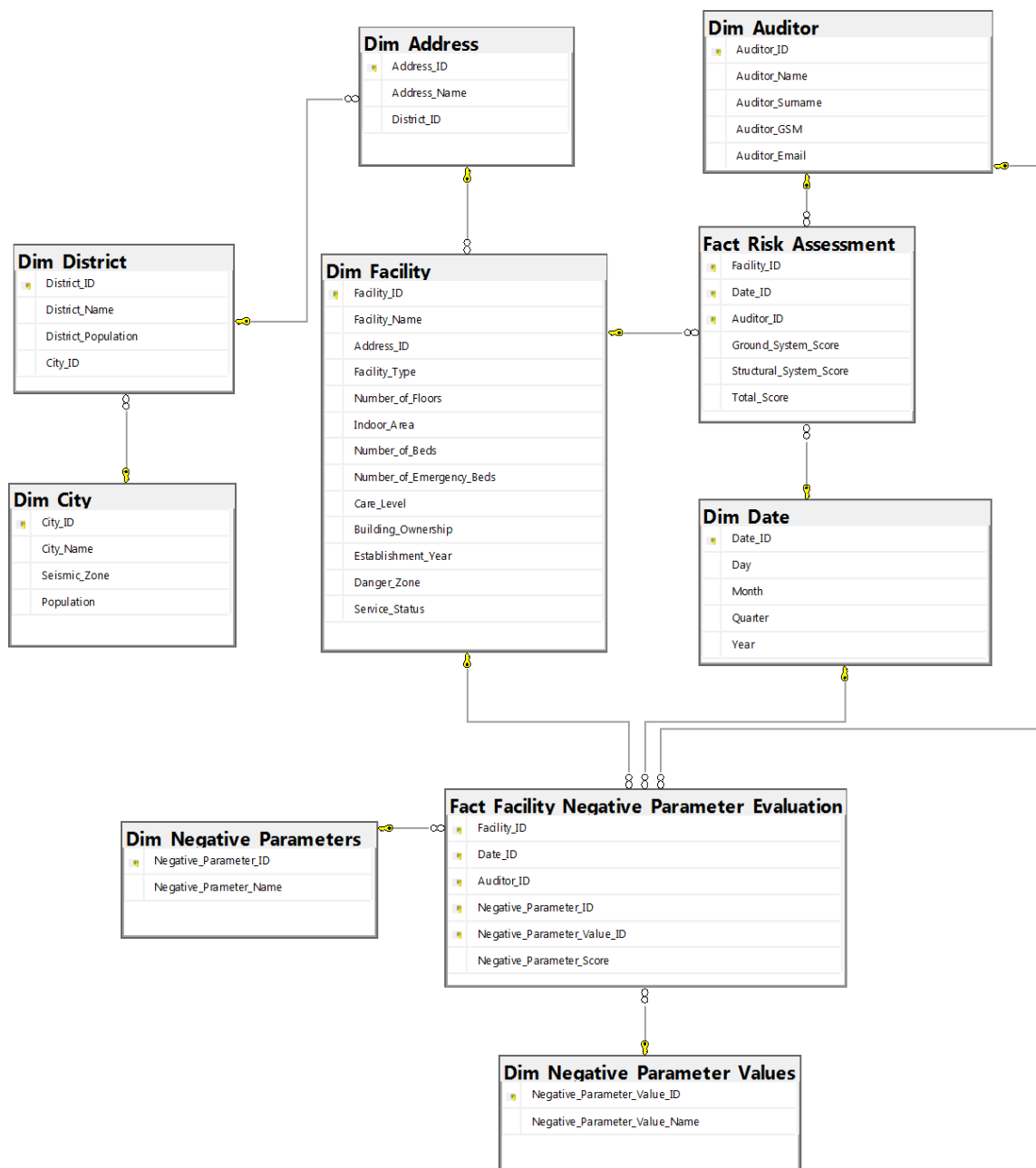


Figure 2 Proposed Fact Constellation Schema

## Query 1

```

1 SELECT Facility_Name AS 'Facility Name', City_Name AS 'City',
2 District_Name AS 'District', Total_Score 'Risk Assessment Score' FROM
3 Fact_Risk_Assessment
4 INNER JOIN Dim_Facility ON
5 Fact_Risk_Assessment.Facility_ID=Dim_Facility.Facility_ID
6 INNER JOIN Dim_Address ON
7 Dim_Facility.Address_ID=Dim_Address.Address_ID
8 INNER JOIN Dim_District ON
9 Dim_Address.District_ID=Dim_District.District_ID
10 INNER JOIN Dim_City ON Dim_District.City_ID=Dim_City.City_ID
11 WHERE Seismic_Zone=1 ORDER BY Total_Score DESC

```

## Query 2

```

1 SELECT Facility_Name AS 'Facility Name', Total_Score AS 'Risk
2 Assessment Score' FROM Fact_Risk_Assessment
3 INNER JOIN Dim_Facility ON
4 Fact_Risk_Assessment.Facility_ID=Dim_Facility.Facility_ID
5 INNER JOIN Dim_Date ON Dim_Date.Date_ID=Fact_Risk_Assessment.Date_ID
6 INNER JOIN Dim_Address ON
7 Dim_Facility.Address_ID=Dim_Address.Address_ID
8 INNER JOIN Dim_District ON
9 Dim_Address.District_ID=Dim_District.District_ID
10 INNER JOIN Dim_City ON Dim_District.City_ID=Dim_City.City_ID
11 WHERE City_Name='İstanbul' and Year BETWEEN 2010 AND 2020
12 ORDER BY Total_Score DESC

```

## Query 3

```

1 SELECT Facility_Name AS 'Facility Name', District_Population AS
2 'Population' FROM Dim_City
3 INNER JOIN Dim_District ON Dim_District.City_ID=Dim_City.City_ID
4 INNER JOIN Dim_Address ON
5 Dim_Address.District_ID=Dim_District.District_ID
6 INNER JOIN Dim_Facility ON
7 Dim_Facility.Address_ID=Dim_Address.Address_ID
8 INNER JOIN Fact_Risk_Assessment ON
9 Fact_Risk_Assessment.Facility_ID=Dim_Facility.Facility_ID
10 WHERE Total_Score=(SELECT MAX(Total_Score) FROM
11 Fact_Risk_Assessment)

```

## Query 4

```

1 SELECT Facility_Name AS 'Facility Name', City_Name,
2 SUM(Negative_Parameter_Score) AS 'Score' FROM
3 Fact_Facility_Negative_Parameter_Evaluation
4 INNER JOIN Dim_Facility ON
5 Dim_Facility.Facility_ID=Fact_Facility_Negative_Parameter_Evaluation.F
6 acility_ID
7 INNER JOIN Dim_Address ON
8 Dim_Address.Address_ID=Dim_Facility.Address_ID
9 INNER JOIN Dim_District ON
10 Dim_District.District_ID=Dim_Address.District_ID
11 INNER JOIN Dim_City ON Dim_City.City_ID=Dim_District.City_ID
12 WHERE Seismic_Zone=1 and Building_Ownership='Rent' and
13 Negative_Parameter_Value_ID BETWEEN 7 AND 10
14 GROUP BY(Facility_Name) ORDER BY SUM(Negative_Parameter_Score) ASC

```

## Query 5

1	SELECT Number_of_Floors AS 'Floor', AVG(Total_Score) 'Risk Assessment Score'
2	FROM Dim_Facility
3	INNER JOIN Fact_Risk_Assessment ON
4	Fact_Risk_Assessment.Facility_ID=Dim_Facility.Facility_ID
5	GROUP BY Number_of_Floors ORDER BY AVG(Total_Score) DESC

#### 4. Conclusion

Earthquake is the fact of life. However, it is known that most of Turkey's cities are considered at moderate or high risk of a major earthquake. With the country becoming ever more urbanized, risk reduction for earthquake is getting a significant long-term problem especially for public authorities who are responsible for taking precautions for the risks of earthquakes. In case of a major earthquake, it is more important for healthcare facilities to survive among public buildings. Previous studies highlight that prioritization should be paid to healthcare facilities. For this reason, there is an urgent need for a comprehensive and systematic assessment of healthcare facilities. In this regard, this study proposes a data warehouse model for assessing risk levels of healthcare buildings in terms of earthquake.

The data warehouse was proposed by taking legislation for risky buildings published by Turkish Environment and Urban Ministry into account. The fact constellation schema model and a multi-dimensional data modeling approach are utilized in the proposed data warehouse. A set of queries were used to illustrate the simplicity of extracting information from the proposed data warehouse. The proposed data warehouse involves an informational database whose data can be extracted from existing operational databases. As far as we know, this study is a first attempt to design a data warehouse for earthquake risk assessment of buildings. It serves a valuable reference for future research in the development of the fact constellation schema for similar purposes. The proposed data warehouse also allows decision makers and policy makers to have more control on the buildings' structural status and risk level in advance.

This paper can be extended in several directions in future research. First, the proposed data warehouse was not implemented, and its performance was not evaluated. In this manner, future research work can focus on the implementation of the proposed design with real data from healthcare facilities. Thus, it can be put into practice and it is possible to compare its performance with a conventional database. Further, it may be useful to adapt the proposed solution to buildings in other domains. Finally, further research might extend the proposed data warehouse design to make it fully compatible with the legislations from different countries.

#### References

- [1] AFAD, "DDA Catalog," 2020. <https://depem.afad.gov.tr/ddakatalogu?lang=en> (accessed Nov. 30, 2020).
- [2] M. M. Önal and S. Çellek, "A Study To Determine The Quality Of Building Stock And Earthquake Risk In Kirsehir," *Int. J. Sci. Technol. Res.*, vol. 6, no. 4, pp. 105–111, 2017.
- [3] A. Ketsap, C. Hansapinyo, N. Kronprasert, and S. Limkatanyu, "Uncertainty and Fuzzy Decisions in Earthquake Risk Evaluation of Buildings," *Eng. J.*, vol. 23, no. 5, pp. 89–105, 2019, doi: 10.4186/ej.2019.23.5.89.
- [4] "OECD Health Statistics," OECD, 2020. [https://stats.oecd.org/viewhtml.aspx?datasetcode=HEALTH\\_REAC&lang=en#](https://stats.oecd.org/viewhtml.aspx?datasetcode=HEALTH_REAC&lang=en#) (accessed Dec. 11, 2020).
- [5] "Health Statistics Yearbook," Ankara, 2018. [Online]. Available: <https://dosyamerkez.saglik.gov.tr/Eklenti/39024,haber-bulteni-2019pdf.pdf?0>.
- [6] G. Garani, A. V. Chernov, I. K. Savvas, and M. A. Butakova, "A Data Warehouse Approach for Business Intelligence," in *28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises*, 2019, pp. 70–75, doi: 10.1109/WETICE.2019.00022.
- [7] S. H. A. El-Sappagh, A. M. A. Hendawi, and A. H. El Bastawissy, "A proposed model for data

- warehouse ETL processes,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 23, no. 2, pp. 91–104, 2011, doi: 10.1016/j.jksuci.2011.05.005.
- [8] A. G. P. Kujur and A. Oraon, “A Data Warehouse Design and Usage,” vol. 3, no. 11, pp. 335–337, 2016, Accessed: Nov. 29, 2020. [Online]. Available: [https://www.academia.edu/34467981/A\\_Data\\_Warehouse\\_Design\\_and\\_Usage](https://www.academia.edu/34467981/A_Data_Warehouse_Design_and_Usage).
- [9] M. Y. Santos, B. Martinho, and C. Costa, “Modelling and implementing big data warehouses for decision support,” *J. Manag. Anal.*, vol. 4, no. 2, pp. 111–129, 2017, doi: 10.1080/23270012.2017.1304292.
- [10] B. A. Devlin and P. T. Murphy, “An architecture for a Information, business and System,” *IBM Syst. J.*, vol. 27, no. 1, pp. 60–80, 1988, doi: 10.1007/978-3-319-77703-0\_114.
- [11] J. M. Abraham Navamani, A. Kannammal, and P. R. Jeba Thangaiyah, “Building multi-dimensional cube to analyze public examination results: A business intelligence approach,” *Appl. Mech. Mater.*, vol. 622, pp. 11–22, 2014, doi: 10.4028/www.scientific.net/AMM.622.11.
- [12] I. Güratan, “The Design and Development of a Data Warehouse Using Sales Database and Requirements of a Retail Group,” *Dokuz Eylul University*, 2005.
- [13] D. Moody and M. A. . Kortink, “From Enterprise Models to Dimensional Models: A Methodology for Data Warehouse and Data Mart Design,” in *Proceedings of the International Workshop on Design and Management of Data Warehouses (DMDW’2000)*, 2000, pp. 1–12.
- [14] D. Moody and M. A. . Kortink, “From Enterprise Models to Dimensional Models: A Methodology for Data Warehouse and Data Mart Design,” in *International Workshop on Design and Management of Data Warehouses (DMDW’2000)*, 2000, vol. 28, pp. 1–12.
- [15] E. Demir and C. Eren Atay, “Akciğer Kanseri Verilerinin Karar Destek Sistemleri için Veri Ambarında Saklanması,” *DEU Muhendis. Fak. Fen ve Muhendis.*, vol. 19, no. 57, pp. 987–997, 2017, doi: 10.21205/deufmd.2017195785.
- [16] G. N. A. Dharma, K. O. Saputra, and N. P. Sastra, “Data Warehouse Design for Hospital Executive information using the Facts Constellation Method (Case Study: Bali Mandara Eye Hospital),” vol. 5, no. 2, pp. 6–10, 2020.
- [17] S. Asghar, S. Fong, and T. Hussain, “Business intelligence modeling: A case study of disaster management organization in Pakistan,” in *ICCIT 2009 - 4th International Conference on Computer Sciences and Convergence Information Technology*, 2009, pp. 673–678, doi: 10.1109/ICCIT.2009.318.
- [18] T. Panrungsri and E. Sangiamkul, “Business intelligence model for disaster management: A case study in Phuket, Thailand,” 2017.
- [19] K. A. B. Permana, D. A. P. Wulandari, and P. A. Mertasana, “Online Analytical Processing ( OLAP ) for Disaster Report,” *Int. J. Eng. Emerg. Technol.*, vol. 3, no. 1, pp. 26–29, 2018.
- [20] E. Işık, M. F. Işık, and M. A. Bülbül, “BetonarmeBinaların Web Tabanlı HizDeğerlendirilmesi,” *Uludağ Univ. J. Fac. Eng.*, vol. 23, no. 1, pp. 225–234, 2018, doi: 10.17482/uumfd.417978.
- [21] S. Sarı and T. Türk, “Depremde Meydana Gelebilecek Bina Hasarlarının Coğrafi Bilgi Sistemleri ile İncelenmesi,” *Türkiye Coğrafi Bilgi Sist. Derg.*, vol. 2, no. 1, pp. 17–25, 2020, Accessed: Nov. 28, 2020. [Online]. Available: <https://dergipark.org.tr/tr/pub/tucbis>.
- [22] *Riskli Yapıların Tespit Edilmesine İlişkin Esaslar*. Turkey, 2019, p. 55.
- [23] Y. Bassil, “A Data Warehouse Design for a Typical University Information System,” *J. Comput. Sci. Res.*, vol. 1, no. 6, pp. 12–17, 2012.