Machines and Algorithms
http://www.knovell.org/mna

Editorial

Volume 3 Issue 1

Editor: Dr. Khadija Kanwal
Institute of Computer Science and Information Technology, The Women University, Multan, 60000, Pakistan

From the Editor

Cutting-edge research that combines the domains of artificial intelligence (AI), predictive analytics, cancer detection, and the Internet of Things (IoT) is the focus of current issue. The journal showcases advancements in tailored treatment strategies, early cancer detection, and the incorporation of IoT technology in healthcare by investigating creative uses of AI and predictive modeling. In order to enhance patient outcomes and promote the future of healthcare, we want to encourage cooperation and knowledge exchange between researchers, physicians, and technologists.

The paper titled, "A Framework for Analysis of SNPs in TAGAP Gene" provides a framework for analyzing single nucleotide polymorphisms (SNPs) in the TAGAP gene, with a focus on their structural and functional impacts. TAGAP, a protein involved in T cell regulation, is linked to autoimmune and infectious diseases. The research employs computational tools like SIFT, PolyPhen-2, PROVEAN, and I-Mutant 3.0 to study non-synonymous SNPs (nsSNPs) that potentially affect protein stability and function. Key findings include identifying nine deleterious nsSNPs associated with structural instability and potential disease pathways, including diabetes and multiple sclerosis. The study discusses TAGAP gene interactions, conservation analysis, and the role of harmful nsSNPs in protein dysfunction. However, it notes limitations such as computational bias, lack of experimental validation, and insufficient diversity in genetic datasets. The research concludes that deeper experimental analysis is required to improve the understanding of SNP-induced changes in TAGAP functionality and their broader implications in human health.

"Personalized Education Enhanced by AI and Predictive Analytics", discusses the use of artificial intelligence (AI) and predictive analytics to enhance personalized education through e-learning systems. It proposes a framework for recommending learning sequences to new learners based on historical data from previous learners. The study uses time-series analysis with two models, Vector Auto-Regression (VAR) and Auto-Regressive Integrated Moving Average (ARIMA), to analyze patterns in learning behaviors. Key findings indicate that the ARIMA model provides higher accuracy and a better model fit, while the VAR model captures more directional changes and explains more variance in the data. The research demonstrates how predictive analytics can personalize e-learning experiences, improve learning content organization, and offer tailored recommendations. Future extensions include customizing learning content, tests, and assignments, further enhancing education personalization.

The paper "Software-defined Network based Fog Computing for IoT Networks" presents a framework for secure fog computing in IoT networks using software-defined networking (SDN). It introduces a novel architecture combining SDN with fog computing to address challenges like scalability, latency, and security. Key features include Fog Management Nodes (FMNs) which manage access control and monitor fog nodes for trustworthiness. Newly connected fog nodes are assigned non-sensitive tasks and undergo trust evaluation based on their behavior. Weighted trust management and simulation add to this proposed framework and was tested using iFogSim, demonstrating effective detection and elimination of malicious fog nodes while maintaining secure interactions between nodes. The results highlight improved network security and reliability by combining SDN and fog computing, making the approach suitable for real-time IoT applications.

In the work titled, “Performance Evaluation of Machine Learning Models for Breast Cancer Prediction” evaluates the performance of six machine learning models—Logistic Regression, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Gaussian Naïve Bayes, Decision Tree, and Random Forest—for predicting breast cancer using the Wisconsin Diagnostic Breast Cancer dataset. The study aims to identify the most effective model for classifying breast cancer as malignant or benign by evaluating their performance across key metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. Among the models, SVM demonstrated the best overall performance with a testing accuracy of 98.25%, a precision of 100%, and an F1-score of 97.83%. Logistic Regression, Random Forest, and KNN also performed well, though they were slightly outperformed by SVM. Decision Tree exhibited perfect training accuracy but struggled with generalization, indicating overfitting. The study concludes that SVM is the most reliable model for breast cancer prediction, particularly effective in minimizing false negatives, which is critical in healthcare. It underscores the importance of evaluating models across multiple metrics and ensuring generalizability rather than focusing solely on accuracy.

In “Optical Character Recognition for Nastaleeq Printed Urdu Text using Histogram of Oriented Gradient Features “the researchers explore a segmentation-free method for Optical Character Recognition (OCR) of Nastaleeq-printed Urdu text, leveraging Histogram of Oriented Gradients (HOG) and statistical features for ligature-based classification. Using the UPTI dataset, ligature images were segmented via connected component labeling and processed with an SVM classifier with an RBF kernel for recognition. The system achieved a 97.3%-character recognition rate, outperforming many previous methods. Key findings include the effectiveness of HOG features for classification, resilience to font size variations, and challenges such as over-segmentation and inability to process handwritten text. The study emphasizes automation in Urdu text digitization and suggests future work on handling text overlap, diacritics, and multi-font recognition.

To conclude, the articles featured in this issue underscore the transformative potential of artificial intelligence, predictive analytics, and IoT in addressing diverse challenges across healthcare, education, and network security. By leveraging AI-driven frameworks for cancer detection, SNP analysis, and personalized learning, alongside innovative architectures like SDN-based fog computing, these studies demonstrate how interdisciplinary approaches can drive impactful advancements. However, the need for experimental validation, diverse datasets, and real-world implementations remains critical to further refining these solutions. Through continued collaboration among researchers, technologists, and practitioners, these cutting-edge innovations hold promise for shaping the future of healthcare, education, and IoT-enabled networks.

Machines and Algorithms
http://www.knovell.org/mna

Research Article

Software-Defined Network based Fog Computing for IoT Networks

Muhammad Tehseen Irshad[1, *]

[1]Department of Computer Science, Bahauddin Zakariya University, Multan, 60000, Pakistan
[*]Corresponding Author: Muhammad Tahseen Irshad. Email: tehseenirshad7370@gmail.com
Received: 25 October 2023; Revised: 1 November, 2023; Accepted: 28 December 2023; Published: 14 March 2024
AID: 003-01-000031

Abstract: The Internet of Things (IoT) connects smart gadgets. The IoT solution streamlines data collecting and processing. High-quality end-user services are making IoT systems appealing. Users can't get high-quality cloud services quickly. Fog computing computes quickly and provides excellent services. Fog computing is a novel processing layer between the cloud and consumer layers in the standard cloud computing concept. The fog layer uses distributed computing with tiny smart devices and access points. Use in diverse applications raises numerous major challenges. Challenges include network security, capacity, scalability, and latency. Security is a major concern for IoT applications. This paper introduces a novel Internet of Things architecture that blends software-defined networking with fog computing. We suggested access control management and trust evaluation algorithms. Our methodology allows fog computing systems to dynamically add fog nodes. Newly connected nodes get non-sensitive jobs. IoT devices and fog nodes may interact, exchange services, and report fog node activity to the Fog Manager node. The FMN measures fog node weight confidence based on behavior. This assessment checks for harmful devices that might compromise system or protection quality. Due to such diagnostics, the fog system filters away unreliable nodes and overweights confident nodes. We tested our technique in iFogSim using Java. The simulation results show that our system can recognize and eliminate harmful attacks/interactions among fog nodes in the fog environment.

Keywords: IoT; Fog Computing; SDN; Cloud; Access control; Weighted Trust Management Security; Dynamic behaviors;

1. Introduction

Smart devices in a networked environment form the Internet of Things (IoT) architecture. By 2025, more than 75 billion IoT devices will be operational, according to CISCO [1], [2]. Our application is more flexible with cloud computing since you can add and remove processing nodes during runtime. Cloud computing has been successful in many contexts. This is unsuitable for receiving reliable, low-latency inputs in smart grids, industrial automation, and ITS [3]. The Internet of Things has become an essential element of our daily lives and garnered attention in recent years. IoT is a network of smart systems, infrastructure, objects, and sensors [4]. Global networking of everyday products (refrigerators, fans) and smart city apps drive this growth. Wireless communications and electronics are also driving smart web-connected object adoption [5]. Creating scalable programs that serve many users at once is driving rapid innovation and cloud computing. Cloud computing allows service providers to run projects with little infrastructure without big proprietary data warehouses [3]. Not all IoT systems are perfect.

Edge computing—real-time device processing—is one of the hardest tasks. Tasks that need extra processing or storage are often moved to the cloud. Service delivery may be delayed. The device-cloud connection is difficult since there is no framework for data sharing and verification [2]. Fog computing is ideal for IoT networks and applications. Cisco launched fog computing as an extension of cellular edge computing. Several research examine IoT Fog Computing [6]. Fog computing is safer than cloud computing since data is temporarily stored and analyzed in local fog nodes near the data source. Instead of connecting to a central cloud, fog computing installs several cloud computing resources at the network's edge. End-users gain from fog computing layer latency reduction, QoS improvement, and entertainment [7]. End-user closeness to the cloud, geographic reach, and mobility support distinguish fog computing from cloud computing. By adding a layer of geographically dispersed fog nodes in the middle, fog improves cloud services [1].

Scalability, real-time data transport, and mobility are not supported by traditional network architectures. Most researchers use fog computing-based SDNs for real-time data transmission. Each network device's traditional communication method combines the control plane and data plane, whereas the SDN separates them [1]. SDN and network function virtualization have transformed network management. A logically central network control layer in SDN enables complex resource optimization algorithms [5]. These attempts rely on a central SDN-managed plane to handle fog computing infrastructure islands. This could severely damage dependability and performance due to increased traffic and failure. Create touchpoints. Dispersed control planes are closer to fog islands, making them more vulnerable to network events. Several researchers employ a distributed SDN control plane [8].

IoT networks struggle with reliability, access control, authentication, bandwidth, and latency. Additionally, IoT systems use device-to-device communication. Users benefit from this great functionality, but communicating data with other IoT devices puts their security and privacy at risk. Additionally, malevolent IoT devices may provide false data or use it for their own gain. These issues render IoT systems ineffective and potentially harmful [2], [7], [9]. The nearest IoT gateway connects several competing nodes. Portals connect to different countries via the Internet. Due of the need to disconnect and rejoin with the gateway, IoT nodes must be flexible [10]. Fog nodes are evaluated based on their network activity when they collaborate and exchange data to perform tasks. The malevolent fog nodes are removed from the network, and researchers struggle to keep them from rejoining.

Most researchers examined direct and indirect trust. Direct trust fog nodes evaluate fog node trust based on their own experience, while indirect trust is based on prior behavior. None of the models characterize the trust evaluator's honesty. Our model describes the trust evaluator's honesty and provides a novel framework for access restriction and dynamic weighted trust management. A centralized architecture for access control and trust management is proposed. The Fog Manager Node (FMN) controls access to newly connected fog nodes. The FMN gives new devices non-essential tasks. You can assign tasks to new shared devices using the FMN. Suggestions include calculating the reliability of newly connected devices. SDN controllers govern network infrastructure. It alerts all FMN about the malicious fog node so they can't allow it into their network.

The remainder of the study is laid out as follows. The literature review is described in Section II. The proposed framework is presented in Section III. Section IV contains the findings and discussion of the implemented model, and Section V contains the concluding remarks

2. Literature Review

Dynamically integrating malfunctioning fog nodes into trusted model and role-based access control computer systems was described in [3]. In appearance, the new dynamic nodes perform mathematical functions. Static, processing, and dynamic nodes in the system are faulty. Fog Nodes Manager handled all extra nodes. FNM looks for additional issues and assigns tasks based on trust. Several moving nodes will enter and leave the fog system. The fog manager recently sorted jobs into linked nodes and calculated confidence using confidence equations. Character classification using computer parameters determined accuracy. These include availability, data dependability, and inverse efficiency. Once any variable node

gains trust, the Fog Network extends its duty to the insured Fog nodes. System C builds the Fog computer platform. They test using Model A SystemC, which represents each processing component and may communicate with other operating systems over the SystemC channel. Each processor element operates at 500MHz–4GHz. The system registers applications based on random responsibility distribution over time. A suitable technique for Internet of Things networks is presented in [2]. Fog estimates underpin the proposed trust and dependability paradigm. The authors examine the framework and its integration into Fog Computing IoT. They use [9], Designed for the new frame. Trust and reputation addresses the previous framework's weaknesses. Each IoT device utilizes error codes to evaluate all IoT devices' dependability and connects to one that exceeds a certain level. This check ensures that no harmful apps are present that might harm the system or dissatisfy customers. The notion is defended against unfavorable exposure, on-off, and self-promotion. Several IoT devices are simulated in the testbed. The collection is on Epinions.com. The 500-testbed concept was modified to utilize IoT apps for various dangers. A Bad-Mouthing assault was repeated for hundreds of rounds from round 20 to 100. Bad-Mouthing attackers start at 10% and rise to 60%. the On-Off assault experiment with 30% and 60% attackers. When hostile gadgets attack at round 20 and persist until round 40, confidence drops. The data shows 60% even with various attacker counts. This dangerous gadget seeks to recover trust between 15 and 35. It began devious behavior in round 40 but was quickly defeated. Rebuilt its trust score between 85 and 100, and removed the device; joined at round 175.

Users can evaluate service providers and choose who to contact in mobile agent systems using an adopted trust and reputation model [9]. The review considers investigators' and witnesses' backgrounds. The credibility of witnesses was also checked to avoid dishonest reporting. Multiple tests are combined using novel, adjustable, and changeable weights depending on contact frequency, size, and witness accuracy. Discounts and punishments encourage witnesses to give accurate information. Another unique feature of the suggested security system is supplementary options for disrespectful witnesses to improve their reputations. The framework might manage detached agent behavior using this way. A testbed simulation assessed the trust and reputation paradigm. The matter is being examined by six auditors and 25 service providers. The user compared five service providers in this scenario. They presume the system user must connect with witnesses with a 70% trust rating (threshold conf = 0:7). If a service provider's Trust value exceeds 60% (threshold trust= 1:2), they are trustworthy. Threshold = 0:4 for average simulated application interaction. The exam is repeated 50 times.

According to [10], modern computing and the scale and variety of IoT devices should include distributed trust management. Trust between IoT entities was established via a multi-layered architecture. Identity, gateway, IoT, node, and server frameworks existed in the cloud. Their architecture considers all of these to provide reliable end-to-end IoT data flow. Identity creation is the first of four communication flows between endpoints, gateways, and identity providers. (2) Identifying the endpoint with the identity server to start the procedure. (3) Gates determine endpoints. Second gateway starts terminal device authentication.

Using a Hidden Markov Model (HMM), the authors developed a statistical, safe, and scalable rogue fog node detection method [11]. A trained HMM might spot dangerous fog nodes. Calculations are fast and accurate. Three steps comprise HMM-based detection. 1) instruction, 2) observation, 3) detection. A rogue Fog node in the network might compromise user data. Thus, connecting to a rogue Fog node was hazardous because attackers may steal sensitive user data. The problem may be pervasive if a larger network collaborated. To avoid malicious fog nodes, they recommend addressing security and privacy at every level of fog computing system architecture. Fog network systems must be secured and malicious fog nodes detected. A device that acts abnormally after being hacked by hostile users or hackers is called. MATLAB R2016a and Eclipse IDE were tried in Java to finish these operations. Badmouthing, self-promotion, on-off attack, and ballot stuffing assault results were recorded in a data file and loaded into MATLAB R2016a for Markov model predictions. Two devices were created to test the system model with different demands. The first fog device averages 0.75 attack chances, whereas the second averages 0.67. Individual device requests determine the aggregate attack probabilities, which are represented as Legitimate nodes (Chance 25%, 50%). 75% wanted 250, 500, and 750). An assault on nodes (37.5%, 25%, and 12.5% requests on 375, 250, and 125). incorrect node (37.5%, 25%, and 12.5% requests on 375, 250, and 125).

The authors examined the functional and non-functional aspects of an Intelligent Transportation Systems (ITS) model and its authorization issues in [12]. Privacy, interoperability, context awareness, resource restrictions, network coverage, selection delay, monitoring, and responsibility in fog computation are addressed. Their ITS architecture splits the ICT environment into four parts: cloud infrastructure (CI), roadside, cars and people, sensors, and enablers. Transportation infrastructure relies on mission-critical security and fog computing. Expect IoT-enabled devices on SA roads temporarily or permanently. Fixed nodes communicated with carriers using short-range radio communication [13]. Sensors like roadside actuators and smart traffic lights are mounted to the automobile. Cloud services were often used in CI to provide alternative routes during traffic bottlenecks. The affected location should have sensors deployed in traffic data systems to monitor traffic conditions. When using the service, the automobile must additionally provide its location and destination. Data will be examined and compared to other vehicle data via the cloud service. Fog computing might boost cloud capabilities and overcome these limits in RS and VH. Cloud services with fog computing might employ RS and VH field features to minimize latency, localize and decentralize apps, and link applications and users (vehicles). It was important to remember that automobiles and people may react unexpectedly. They outline the main traits of a good access control system. 1) Attribute-Based Access Control. In further research, they examined successful methods for establishing reference monitors, arguing for regulations, and supporting offline operations.

In [14], the authors advise assessing the trust value and rating of each fog IoT and fog node/device based on their interactions to ensure routing and handoff. A trust manager between the fog and IoT layers tracks all fog nodes in its lookup table and identifies malicious fog and IoT nodes. Fog nodes also define the IoT layer's capacity and provide services via the most dependable pathways. They examined one-way links to understand the paradigm. In the first SITO approach, nodes generate trust levels between 0 and 1 and assign them to neighbors based on recent interactions. All FN ratings and TV/TF data will be added to the TM lookup table. The Tidal Trust method starts with a randomly picked FN and estimates its nearest nodes' TVs at various stages of the dialog. The tidal confidence technique calculates step confidence scores. In general, the tidal Trust technique calculates trust and ratings depending on each BS level in two steps. First layer fog node was used to compute i+1 layer. Fog nodes determine end-user dependability and then pick the most reliable intermediary nodes to transmit services in the second phase. The testbed held this experiment. Microsoft Azure Cloud DS2 provisioned three virtual machines. Each fog environment started with 2,000 nodes and added 50 every minute to evaluate the architecture's scalability. The three NS2 configurations use the suggested method to increase fog node reliability. The recommended approach allows fog and IoT node combinations and changes.

The authors propose in [15] that fog node authentication verifies data owners and requesters without third parties. Create a fog node-smart contract system to validate user access to IoT devices using smart contract tokens. The procedure was also used to study backups. The model works properly when the smart contract is deployed, boosting security and search result trust. This scenario managed IoT devices, fog nodes, and end users using Ethereum smart contracts. Five primary pieces of the system model access Ethereum smart contracts online. Each participant utilizes cloud and fog nodes to engage with smart contracts using the Ethereum client and has an Ethereum address (EA). Decentralized cloud storage solutions should use this authentication method. The suggested approach comprises five phases: device registration, mapping, authentication, token production, and data sharing. Admin, Smart contract, End-user, Fog nodes, and IoT devices comprise the decentralized storage system. The organization handles donor accounting and exchange. A proposed authentication technique was supplied for Python testing. How expanding the group tail from 50 to 200 bits affects shader node authentication. The tale is valid for 4,009,083 minutes, ranging from 76,915 seconds to 76 times, but the validation approach is superior.

For fog computing, the author introduces access control Ciphertext-policy attribute-based encryption (CP-ABE) with outsourcing and attribute modification [16]. Determining the encrypted data's owner and decryption user is independent of the key's access structure and other factors. Updates to attributes are cheap since they just update the variable value's ciphertext. The proposed system has five agencies. Cloud service providers, fog nodes, data owners, end-users, and key authorities are examples. A effective fuzzy computing

access approach involved outsourcing and theme adjustments in five stages. to test java, run the system, produce keys, encrypt and decrypt data, and alter attributes The CPAB Toolkit and Java Pairing Cryptographic Library implemented these laws. Java on Android phones with quad-core CPUs and 3GB RAM encrypts and decrypts data owners and users. Their key generation computational cost was half that of other models. Consumers found it handy that the data owner generated and sent the ciphertext to the fog node in 0.615 seconds and the user decrypted it in 0.459 seconds. Smart low-resource devices.

The persistent memory leakage model (CML) was one of the most powerful models for allowing continuing loss of user and master secret keys [17]. Their fog computing technology protects against third-party dangers, allows privacy, and controls access. The two settings have distinct threats, thus using the same technology might generate issues. Fog nodes are close to users and widely utilized in public spaces, making physical attacks (side-channel attacks) conceivable. With a smart gateway, the attacker may see device power usage and operating time. Therefore, typical functional encryption may not provide enough security in a foggy computing environment. Basically, fog computing access control is best solved using functional encryption. Due to new dangers, the previous definition may not be enough. Their technique converts LR-FE modeling and design to conditional coding. It has two designs: double encryption with paired encryption leak prevention. From a sealed pair encoder, make a sealed FE encoder. They offer equations and assertions to substantiate their stance.

A security architecture based on IoT and fog collaboration is suggested in [18]. Secure cooperation across resources and operational components is achieved by combining efficient access control with monitoring. Their security approach includes a thorough resource scheduling and allocation method to maximize system performance. Fog computing systems are divided into cells with various fog nodes in their study. The FNM FMN leads the cluster and manages each unit. Different service classes are found in fog nodes[19]. As long as the service was available, IoT users received it. Its major function was to handle new IoT users. Their second FNM assignment classifies resources. Provide an algorithm that boosts credibility and rating for high- or medium-access devices. Devices with lower service class will have mediocre access. They developed his network architecture (iFogSim) and a Java application to test his concept. The fog node service and the proposed trust access control and fog computation algorithm (TACRM) have the most resource management mechanisms and the greatest response, whereas the cloud server service takes longer and has a range of at least 90ms. Group Task Scheduler (GTS) 12.46 ms, 5.0328ms interval.

SDN-based public infrastructure (PKI) trust management solutions are sluggish to adopt in the cloud. TRUFL, a distributed trust mechanism, was devised to build and validate SDN trust [20]. Distributed trust management is consistent, resulting in faster transfers than centralized trust management. Unlike previous studies, the TRUFL system scales effectively as OpenFlow rules increase. Their OpenStack control node had an SDN controller. The Open Day Light Controller, a management vessel, has various uses, including a flow rule conflict checker and one or more Certificate Authorities (CAs) to verify node computer science dependability. A network-based intrusion detection system (NIDS) mirrors data plane traffic between numerous data plane virtual machines via port mirroring. NIDS uses a Neutron API (OpenStack Network Manager). TRUFL transfer verification time has grown from 7 to 28 milliseconds, while OpenFlow rules have increased from 10,000 to 50,000. SDPA had 130-145ms latency for 50k rules, while Net-Syn had 65ms. Their testing findings made SDPA delay data difficult to determine. Hassle delays 50k regulations by 6 seconds. The DPDK's distributed trust management reduces packet processing and authentication to milliseconds, making TRUFL's authentication quicker.

In [21], the authors present a model for assessing node trustworthiness that accounts for detrimental node behaviour. The suggested trust and reputation model addresses the security issues with delegation mechanisms in distributed systems. A distributed architecture with N nodes was investigated. These N home gateways (HG) can connect to the central server via one or more hops. Nodes were rated dependable or unreliable based on their past activity. Use selfish and accusing nodes to test his trust modeling. Trust-based trust management systems struggled with trust measurements and dependability data. The WSN agent-based trust and credit management (ATRM) strategy is explained. Our reputation is the backbone of

their trust model. In this example, nodes support each other make honest evaluations. Two layers make up this model. A trust model protects nodes against selfishness on the first layer. This implies that selfish nodes will be identified, penalized, and maybe banned from participating. To interrupt network activity, hostile nodes falsely accuse other nodes of being untrustworthy. The Trust Modeling layer defends against this. Four nodes (3, 9, 5, and 8) wish to send packets. Checks confidence matrix before transmitting. This 8 knots selfishly. They also employ 4 load nodes (2, 6, 12, and 14). The research shows that nodes 2, 6, 12, and 14 have low confidence levels and the number 2 is wrong.

The LoRaWAN method for connecting end devices to the network server is analyzed for serious security flaws in [22]. They uncovered protocol flaws, including the use of random integers in join packets to thwart replay attacks. They transmit two communications between the end device and the web server to join and accept the application. Terminals send network server connection requests. The web server will permit the receiver if the end device is authorized to access the network. No answer will be delivered to the last device if the membership request is denied. They then create different quantities of concealed visible data, which the attacker must retrieve using Entropy, the average voltage. This technique solves several security difficulties. Start by creating a single DevNonce for the device's random generators, which assess this event's likelihood and the network server's response. Second, DevNonce struggles to provide real randomness, especially for low-cost devices. SX1272 transceivers add the least significant bit of the received signal strength (RSSI) signal to form the random bit pattern. They study key generator quality factors and specific attacks that might reduce number unpredictability for long-term number production. This WiMOD SK-iM880A test shows how the DevNonce generator works without attack and with jammers. They discovered $\rho = 320.6$ without interference, identical to the comparative article, proving attacker-free random number generators' applicability. The first experiment, in which the jammer transmitted random signals, yielded findings identical to the 1m and no jammers. The second experiment's minimal entropy, which is lower than the others, is most essential. The minimal entropy of 12.66 means the maximum value is created every 6451.6 operations.

In [23], the authors provide a cloud resource provider's credentials and capabilities-based trust model. They demonstrate how to construct SLAs that integrate customer experience with cloud service provider capabilities. They expand the trust model and provide a cloud resource trust value equation based on QoS needs including dependability, availability, processing time, and data integrity. They also explain how to pick reliable and useful cloud resources. Architecture revolves around the system manager. It cooperates with the system. SLA managers negotiate and compromise on user QoS demands. The provisioning service component links system administrators with middleware agents. Provisioning solutions virtualize cloud customers' work environments and separate IaaS, PaaS, and SaaS systems from the cloud, affecting SLAs. The three main services were monitoring, metering, and billing. Control subsystems distributed and used resources. The middleware agent manages VM creation, modification, sharing, administration, and network deployment. Trustworthy repositories may be trusted. Adds a trust manager that stores and processes trust values from the trust store. A trust store contains cloud resource trust values. They also control trust management algorithms. They simulated their infrastructure with CloudSim. Revenue efficiency, dependability, availability, and data integrity were examined separately for each QOS configuration. They compared their model to two others using 5000 tasks per feature. QoS trust outperformed others.

They provide a detailed overview of fog computing data access control [24]. They evaluated fog computing access control needs latency, efficiency, generality, aggregation, privacy, resource limitation, and policy administration. They also discussed access control models. The DAC model allows the data owner to decide and determine access to others based on the identity of the group members. The MAC Paradigm resource is designed with users in mind for matching. Thus, it suits distributed systems better than DAC. Role-based access control (RBAC) prioritizes article responsibility above authorship. RBAC-assigned roles allow users to access system elements. Attribute-based access control (ABAC) and attribute-based encryption (ABE) are ideal cloud computing access control solutions because they preserve data privacy and let data owners manage access. Usage-control-based access control (UCON), RMAC, and

proxy re-encryption. They concluded by discussing exporting expensive computation, regulating access policies, and fog computing access control research.

The authors of [25] compare Fog security approaches using IoT security criteria. They evaluate fog-based strong authentication for IoT devices and how it meets security goals. No standard fuzzy computing architecture to handle trust, privacy, and other obstacles can lead to IoT security difficulties, as shown in this article. IoT devices and fog visibility are also goals of the new approach. Traditional fog and IoT hardware manufacturers still report to many cloud service providers. Many issues require decentralized design. Authentication methods with the lowest ratings on architectural elements including security, usability, and productivity were investigated. They then addressed how they want to enlarge this pool and establish a qualitative and quantitative diagnostic framework for IoT authentication systems. In [26], the authors establish a model to assess reputation and trust management and set the scene for integrating trust and reputation into a security architecture to ensure WSN data security, dependability, integrity, and trustworthiness. The authors examined fog computing topologies and authentication methods. They also reveal IoT security innovations. IoT device security and privacy risks are growing, he says. Thus, they must build security mechanisms against a range of threats to give the system a convincing foundation for detecting an intruder, even if the threat changes. Fog computing has several risks, thus uniform standards may help solve some of them. Since IoT network security depends on fog computing architectures, they must be developed with security in mind. Communication is wireless between N sensor nodes. Multi-hop routing sends regulated data from sensor nodes to other networks or the Internet via a group leader or gateway. Every member node has an environment-wide reputation value. A packet that gets such a signal keeps just neighboring data and ignores other node data. Through experimentation and research, their algorithm found a solution. This optimisation approach improved communication dependability and cost-efficiency.

Researchers [7] describes a fog computing trust management (COMITMENT) strategy that uses quality of service and quality of protection experience metrics from earlier direct and indirect fog node encounters to assess and manage node trust. They built a model approach to assist foggers make the optimal judgments during degassing by working with them. The authors study a distributed fog topology with nodes effectively spread over many sites and linked by a communication mechanism that assigns each node an IP address. Fog nodes may interact without a central console (mesh network), making resource distribution and function transparent. Since there is no central authority to designate trusted nodes in the network, Fog nodes regularly generate trust scores and a local list of trusted nodes for their neighbors. They evaluated the suggested technique in MATLAB (2018b) and found that it outperformed random wax discharge (RWO) and proximity fog discharge (NFO) using competing conventional methods. Every node without 75% fog increases the network's fog to 5%. The average number of successful and failed contacts during the test showed that network abuse raised the proportion of unsuccessful interactions and lowered the percentage of successful interactions.

The Ciphertext Policy Attribute-Based Encryption (CP-ABE) algorithm and blockchain technology are used in [27] to remove rogue fog nodes and minimize network latency between the cloud service provider (CSP) and fog nodes. The blockchain stores on-chain tracking tables, and the smart contract verifies fog nodes' identities before they can access the CSP's encrypted data. FNs that purposefully change the on-chain tracking table are tagged as rogue fog nodes due to blockchain immutability. Transferring data from cloud storage to end-user devices is expensive and latency-prone. To address these challenges, Fog Computing was invented. However, BSs are vulnerable to malicious attacks that compromise user data. The blockchain uses and integrates the contemporary cryptographic primer CP-ABE algorithm to secure communication between the BS and the CSP with secrecy, integrity, and access to end-user data. Blockchain may allow CSP FN to spread access control authority.

A cloud fog control middleware that manages service requests with limits is proposed in [28] to merge cloud and fog computing. Cloud fog management middleware may be efficiently maintained without additional design components if one wants to join a new fog node, group, or node goes down. Their fog node manager (FNM) only works in fog. A fog group maintains all fog nodes or related fog nodes. They

think their model will save energy use and service times. Cloud-IoT-Fog interaction generates massive data that may need change and integration. Attacks might happen in sequence. Ineffective resource rules and user activity monitoring might lead to such assaults. Basically, poor security implementation might cause security vulnerabilities. Fog systems must be protected from resource exploitation, malware, and other threats [3]. They must provide anti-fogging and resource management to speed up turnaround. Fog systems analyze vast amounts of data well.

3. Proposed Methodology

This study introduces safe fog computing using software-defined fog computing. The fog computing platform protects end-user and fog node processing. Fog management nodes monitor fog nodes. The fog management node (FMN) validates IDs and assigns non-sensitive jobs to fog nodes entering the network in our idea. FMN watches the new fog node after access and estimates trust based on activity. Trust between fog-2-fog nodes depends on recent activity. FMNs control fog nodes. A harmful fog node is removed from the network and designated malicious. All FMNs get malicious node status and SDN controller alert. This section covers FMN. Three subsections comprise this section. Part 1 introduces our SDN-enabled fog computing architecture. The second and third components involve access control and weighted trust evaluation.

3.1. Software-Defined network-based fog computing

This section describes our SDN-enabled fog computing architecture's layers. The IoT devices, Fog, SDN controller, and cloud layers have different methods. Figure 1 shows these layers.

3.1.1. IoT devices layer

IoT devices employ sensors, actuators, microcontrollers, RFID tags, and transceivers to process final commands and deliver applications. Internet of Things isn't one technology. Instead, it's a combination of technologies. People interact with their surroundings via sensors and actuators. Users must store and interpret sensor data wisely to gain insights. We define "sensor" generally; a microwave oven or cell phone can be a sensor provided it delivers current state information. An actuator, like an air conditioner temperature controller, affects the surroundings.

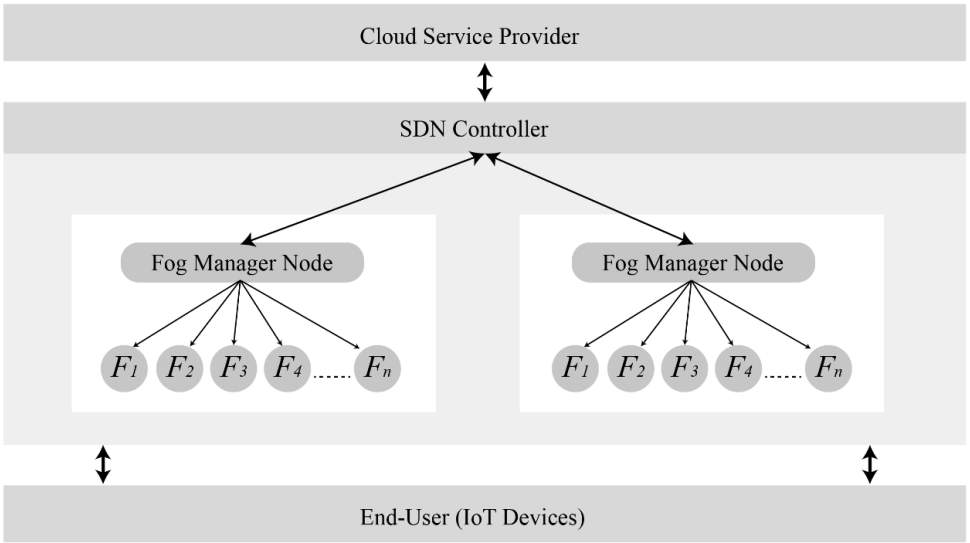


Figure 1: Proposed SDN-Based Fog Computing Framework

3.1.2. Fog computing layer

Cisco said edge computing penetrates deeper than fog computing, which includes smart doors and sensors. This model implies motors, pumps, and lights can process data intelligently. Network edge devices should preprocess as much data as feasible.

Our fog computing paradigm contains parent and child nodes. Fog management nodes, or fog heads, are parents, whereas fog nodes are children.

Fog nodes: Software on IoT devices is called cloud nodes. These nodes connect with IoT devices using CoAP and SNMP. Less devices than computer resources are required. A router, access point, switch, gateway, firewall, and dedicated server can be cloud nodes. Cloud nodes can be linked directly to an SDN device like a switch or router or configured with one. The following operational modules make up each cloud node. This server accepts the Cloud Manager node's request. Monitors IT services implementation includes this module. Database - This module saves the received request in the database, updates the current state of the cloud node system, and available resources, and monitors the readiness of the data.

Fog manager node: The central console of the fog network is the node of the fog manager. The node must be connected to the FMN every time it joins the network. If the connected node is an edge node, the FMN will send the addresses of the active fog nodes based on their proximity. The edge node will later establish a connection with the nearest fog node and start transmitting data. If a fog node goes down, it will send the address of the next available nodes to connect to.

When a smart device tries to join the fog network, the FMN sets the lowest level of trust connecting the node. If a node needs to be removed from the network (due to an outage or permanent disconnection), FMN will revoke that node's access rights and update the list of active fuzzy nodes for the edge sensor, SDN, and cloud layer. SDN will notify other fog management nodes of unauthorized nodes.

Fog Manager Node performs the following operations.

1. Access control management.
2. Monitoring of fog nodes.

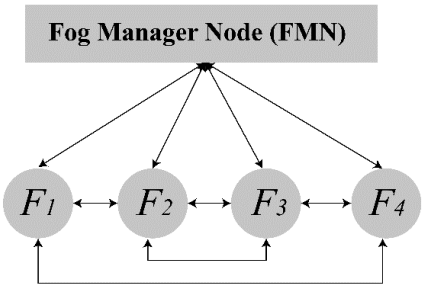


Figure 2: Fog Manager Node and Fog Node

3.1.3. Software-defined network (SDN) Controller

Software-Defined Networking (SDN) is a network development in which the data layer is separated from the console level and all actions, management, and control are concentrated on a single console. Because of its administrative features, it finds applications in other countries, such as cloud computing and cloud computing, to manage asymmetric communication between nodes, thereby increasing security performance.

SDN manages the network in our framework by updating the fog manager nodes. It notifies other hostile nodes of all activities so that they might be spared from the harmful nodes. If a fog node engages in

malicious behavior, a fog manager node estimates its trust and notifies the software-defined network controller, which subsequently notifies the offending node's ID.

3.1.4. Cloud Service Provider (CSP)

Because it won't be calculated, unnecessary data gets transferred to the cloud. Fog computing cannot compete with cloud computing for data computation. Fog computing enhances the computation workflow by reducing latency. When a fog node gains trust, it may connect with the cloud and receive and send data without a fog manager node.

3.2. Access Control Management

The first function here is the Fog Manager node to control access to the new IoT user. When user requests are added, the trust level is calculated at the time. The accuracy of each user is determined by their actions. The authorization assigned to each user is done on a proxy-defined Fog node (FMN) level basis. Delegation is used by resource allocation method. Once the data in this cell is authenticated, the user does not need to access the cloud node method.

New fog nodes (Fns) or user devices must obtain permission from the fog manager node or network head to join the network. This enables the fog node to switch between services. The fog manager node (FMN) will match the new fog node's ID to the malicious fog list. We need the ID verification phase to prevent malevolent nodes from joining another fog network after removal. If FMN decides the new fog is malicious, he will deny the fog node's request to join the network. If the fog node lacks an ID, the fog head will verify and issue one. All network nodes will get the new fog node's ID from the fog management node (FMN). All neighbor fog nodes (NFN) will exchange services with the newly joined node. After that, the fog manager node will assign a non-sensitive task to avoid affecting the surrounding fog node. One algorithm describes the whole procedure.

Algorithm 1 discusses sending network access requests to fog management nodes. Consider this scenario. fog F_n seeks network membership. F_n queries the fog management node in line 1. If the ID and maliciousness of F_n match those on the malicious fog list in lines 2-4, the fog management node will reject their request. Lines 5-7 show the fog management node (FMN) assigning the ID and job to fog node F_n after verification. The wireless sensor network assigns roles based on reputation to facilitate role-based access control.

- **Algorithm 1:** Fog node joining into the network and assigning a task.

1 ***Input:*** *FogManagerNode (FMN); NewFogNode (F_n); Maliciousfognode(ML_f)*
2 ***Parameters:*** *FogList (F_L);*
3 ***Result:*** *Give Access control and assign a task to NewFogNode (F_n)*
4 *NewFogNode (F_n)* will request *FogManagerNode (FMN)* for joining the system.
5 **If** $F_n \epsilon\, ML_F$ **then** ▷ *FMN* will check the ID in MF_L for verification F_n
6 **Declined** ▷ *FMN* will remove the untrusted node
7 **else**
8 $F_n \leftarrow ID$ ▷ *FMN* will assign Id and task to the new fog node
9 $F_L \leftarrow ID(F_n)$ ▷ *FMN* will update the list & send the *Id* of F_n in the network
10 **End**
11 **End**
12 return;
13 **End**

3.3. Weighted Trust Management

Monitoring continues after user authorization. Additionally, reporting unusual behaviors activates a deactivation feature. The Fog Manager Node (FMN) monitors the system to track user activities. The FMN tracks all cloud-fog computing traffic. The FMN notifies multiple fog nodes within a small cell when connected users engage in harmful behavior. Weighted trust management has two parts: FMN evaluates fog node trust, and FMN updates fog node honesty by informing about harmful and genuine nodes. The following subsections summarize the entire scenario.

3.3.1. Trust Evaluation

This procedure involves fog nodes sharing services based on quality of protection. If one fog node (F_1) requests service from another (F_2). If F_2 behaved maliciously or positively with F1, the fog management node (FMN) created a list of F_2's neighbors (NFn) for verification and trust evaluation. FMN selected dynamically weighted witness neighbors. The fog manager node (FMN) tests F_2's trust using eq1.

Let's say the neighbor's fog F_1 gives F_2 a favorable assessment, F_3 negative, F_4 positive, and so on. F_1, F_3, F_4, and F_n have dynamic weights w_1, w_3, w_4, and w_n, respectively, therefore service provider fog node trust is computed in eq1.

Let

$$\{ \propto_1 = (F_2, +),\ \propto_3 = (F_2, -),\ \propto_4 = (F_2, +), \cdots,\ \propto_n = (F_2, n)\}$$

Then trust evaluation is calculated as

$$EV'(F_2) = \sum_{n\in\{1,\cdots,N\}\setminus 2}^{N} w_n * \propto_n \tag{1}$$

Where N is the total number of fog nodes, n is the neighbor nodes interacting with F_2, w is the dynamic weight, and $\propto$ is F_2's positive or negative behavior with n. We add the freshly assessed F_2 trust to the existing trust. Eq2 calculates F_2's trust.

$$Trust\ EV(F_2) = \ EV\ (F_2) + EV'\ (F_2) \tag{2}$$

The trust evaluation for fog node F_2 is EV(F_2). The table now displays the updated final trust evaluation for fog node F_2. Two instances will follow. In the first scenario, fog node F_2 is considered valid if its trust evaluation exceeds the threshold and updates its ultimate trust score. In example 2, the fog node F_2 is malevolent if its trust evaluation is below the threshold. FMN removes harmful nodes from the table/list and adds them to the malicious fog list. FMN notifies both the network fog node and the SDN about the presence of the malicious node. The parent fog manager node will get the malicious node's ID via SDN. Therefore, the second Fog management node will not permit the presence of malicious nodes in the future. We calculate all fog nodes' trusts, enabling FMN to update them and remove malicious nodes.

3.3.2. Honesty updating

In this subsection, the honesty of informer is updated based on their honesty. We see in previous scenario that the fog node *F_1* inform the FMN about the behavior of fog node *F_2*. So based on their behavior with neighbor FMN node evaluate their trust. But in this scenario the honesty of *F_1* is increase or decrease based on their honesty. Let's suppose the *F_1* inform the behavior of the *F_2* to FMN. Then there will be two scenarios.

First scenario: fog node F_1 sends F_2 negative input, FMN recalls neighbor as witness. The witnesses also critique. Positive fog node F_1 feeds F_2. Even witnesses spoke well. Therefore, F_1's claim about F_2 is true. Equation 4 rewards F_1's honesty.

$$H(\ F_1)' = H\ (F_1) + \ H(F_1) * \ \beta \tag{3}$$

In this case, β represents the reward threshold. Multiply by H(F_1) and add to current honesty. A node's existing honesty from past honest feedback. Fog node F_1's updated honesty is H(F_1)'.

Second scenario: fog node F_1 sends F_2 negative feedback, but the nearby witness offers positive or opposite input. Or fog node F_1 feeds F_2 positively. And witnesses criticize. Therefore, F_1 is erroneous about his

assertion with F_2. F_1's honesty will suffer as a punishment. F_1 will be removed from the network and tagged as malicious in eq5 if its honesty is below threshold.

$$H(F_1)' = H(F_1) - H(F_1) * \beta \quad (4)$$

In this case, β represents the penalty threshold. This value multiplies and subtracts existing honesty.

- **Algorithm 2:** Weighted Trust Management
  Input: *NeighborFogNode (F_n), TotleNumberOfFogNode (N); FogManagerNode (FMN); SoftwareDefinedNetwork (SDN); Maliciousfognode(ML_f)*

  Parameters: *HonestyEvaluation; WeightedTrustEvaluation; FogList (F_L);*

  Initialization: *Honesty = 1; Trust = {Ø};* $\propto = \pm n; \beta = n;$ *w = 1; Thr = n;* $F_L = \{f_1, \cdots, f_n\}$
  Result: Weighted neighbor trust and honesty evaluations *' F_1 toward F_2*

1. **Procedure 1:** Weighted Trust Calculation;
2. $F_1 \rightarrow F_2$ ▷ *F_1* will interact with *F_2*
3. $FMN \leftarrow F_1(F_2, Comment)$ ▷ *F_2* will report the behavior of *F_2* to *FMN*
4. $FMN \rightarrow NF_2 = [F_3, \cdots, F_n]$ ▷ *FMN* will create witness neighbor fog node list
5. $$EV'(F_2) = \sum_{n \in \{1,\cdots,N\} \setminus 2}^{N} w_n * \propto_n$$ ▷ Evaluate the trust of *F_2*
6. $Trust\ EV(F_2) = EV(F_2) + EV'(F_2)$ ▷ Calculate the overall trust of *F_2*
7. **if** $Trust\ EV(F_2) < Thr$ **then**
8. $FMN = F_L - F_2$ ▷ Remove untrusted node and marked as malicious
9. $SDN \leftarrow FMN(F_2, update)$ ▷ *FMN* will Update the status of *F_2* to *SDN*
10. $FMN' \leftarrow SDN(F_2, informID)$ ▷*SDN* will inform the ID of *F_2* to all parent *FMN'*
11. **Else**
12. $F_L = update(F_2)$ ▷ update list
13. **End**
14. **End**
15. return;
16. **End**

Procedure 2: Update the Honesty of *F_1* by;

17. **If** $(Trust\ EV(F_2) < 0\ \&\ \propto_1 < 0)\ OR$ $(Trust\ EV(F_2) > 0\ \&\ \propto_1 > 0)$ **then** ▷ In both cases F_1 was true about F_2

18. $H(F_1)' = H(F_1) + H(F_1) * \beta$

19. $F_L = updated(F_1)$ ▷ update honesty of F_1 in list

20. **else if** $(Trust\ EV(F_2) > 0\ \&\ \propto_1 < 0)\ OR\ (Trust\ EV(F_2) < 0\ \&\ \propto_1 > 0)$ **then** ▷ In both cases F_1 was false about F_2

21. $H(F_1)' = H(F_1) - H(F_1) * \beta$

22. $F_L = updated(F_1)$ ▷ update honesty of F_1 in list

23. **End**

24. return;

25. **End**

Algorithm 2 explains how to get a suggestion from nearby fog nodes and calculate weighted trust. If fog F_1 interacts with fog F_2, F_1 follows. Procedures 1 and 2. In the procedure 1, F_1 interacts with fog F_2 and gives feedback based on its behavior to the fog manager node from lines 1-3. The fog management node obtains the neighbor list and assesses the trustworthiness of fog F_2 in lines 4 through 6. Based on the final trust score, the fog management node erases or updates the fog F_2 trust and informs the SDN, preventing any network contact with the malicious node in lines 7-13. In the procedure 2, FMN assesses Fog F_1 honesty and updates. F_1's honesty increases in lines 17-20, while their dishonesty decreases in lines 21-23.

3.3.3. Experimental Setup

Table 1: System Setup and Simulation Settings

Parameter	Value
Operating system	Win 10
Processor	Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz
RAM	8.00 GB
System Type	64-bit operating system, x64-based processor
Simulation environment	iFogSim, Java jdk-8u241 and Eclipse-IDE
Number of fog nodes	10
Number of IoT devices	10
$\boldsymbol{Thr_{trust}}$	1.5
$\boldsymbol{Thr_{honesty}}$	1.5
$\propto$	0.5
$\boldsymbol{\beta}$	0.5

In this part, we evaluate the safe Fog-2-Fog collaboration paradigm to enable secure fog service requests. We used Java to sketch our network architecture, testing the suggested access control and trust evaluation technique. We simulate using iFogSim [29]. iFogSim is used to manage IoT services within a fog infrastructure. Figure 3 shows how iFogSim visualizes the proposed design.

3.4. Simulation Settings

Table 2 shows the system functions that were used in the simulation. We provide simulation parameters in terms of network topology, propagation and transmission delay, uplink and downlink capabilities, and Fog interaction.

3.4.1. Network Topology

Fogs are represented as a mesh network when the network topology is characterized as an indirect graph. The simulation comprises ten fog nodes (i.e., Fn = 10) that are connected via an internal communication network. In our simulation, we used iFogSim default settings, which include a 100 ms delay for cloud and proxy connections and a 2 ms latency for SDN controller and fog-to-end devices.

3.4.2. Network Bandwidth

The link bandwidth depends on the type of service's request; hence, heavy-request will require more bandwidth than light request. Table 2 shows the default entity settings in iFogSim. It shows the fog node and IoT device default specifications in terms of MIPS and RAM in gigabytes.

Table 2: Default entity configurations in iFogSim

Attribute	**F1, F2, F3, F4 F5, F6, F7, F8, F9, F10**	**FMN1, FMN2**	**SDN**	**IoT devices**
MIPS	2000	10000	10000	1500
RAM (GB)	10	4	10	2
Uplink Bw	10000	100	200	10000
Downlink Bw	270	10000	20000	10000

3.4.3. Fogs interactions

We allow fog nodes to interact and request services. To calculate fog node trust, we used a mesh network to allow fogs to communicate and report their service providers to the fog management node. Binary confidence score values representing excellent or poor interactions are stored locally as 1 and 0. Using all prior node interactions/collaborations, a weighted trust is generated to assess the partner fog node's trustworthiness.

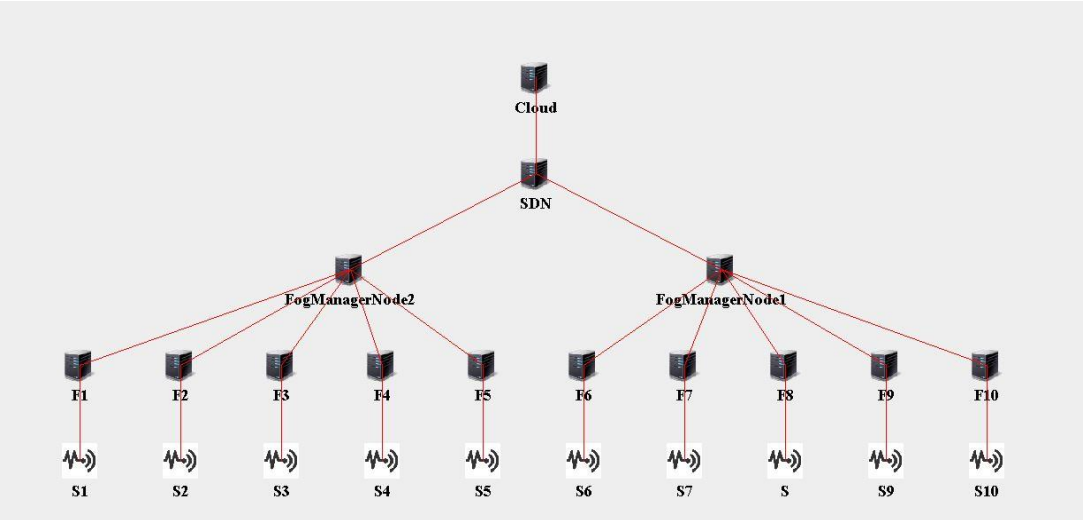


Figure 3 Overall Simulated Architecture in iFogSim

As you can see that we planted to FMN(FMN) and each FMN have five child nodes. These nodes can communicate with each other and also with their parent's head node. Fog manager nodes are connected with SDN controller. When any node leaves the network due to their negative behavior are informed to SDN so that he could inform other fog manger node so they will be aware of malicious node and did not give permission to these nodes to enter in the network.

4. Results and discussion

This section shows the numerical results of the experimentations on the proposed model to validate the accuracy of our secure Access control based on the weighted trust model. We first evaluate the performance of the Proposed weighted trust algorithm and then the performance of the access control algorithm. We compare our proposed trust algorithm on different parameter and did two experiments to gets the results

4.1. Experiment I

At the first experiment, we set the α and β respectively 0.5 and 0.5. Initial trust value of all fog nodes *Trust* = 1 and *Honesty value* = 1. and we keep a 1.5 threshold. The results of simulation analysis demonstrate a favorable impact on network usage and trust in our proposed algorithm, where Figs. 6.1, 6.2, and 6.3 show that the proposed algorithm detects malicious nodes very precisely.

4.1.1. Trust Evaluation

Fig 6.1 shows how the F_1 and F_5 maintain their trust value and achieve the highest Trust and how the F_2, F_3, and F_4 did the malicious activity and these are removed from the network.

When the trust value of the fog node decreases from this threshold then that fog node is removed from the network. Trust and honesty increase with the passage of time and their behavior. Trust value increase and decrease by 0.5 after each transaction.

In fig 6.1 we can see that when F_1 and F_5 did positive behavior then its value increased from 1 to 1.5 and likewise when F_2, F_3, and F_4 misbehave then their values decrease 0.5. and at the end the value of these fog nodes becomes 0 and these nodes are removed from the network.

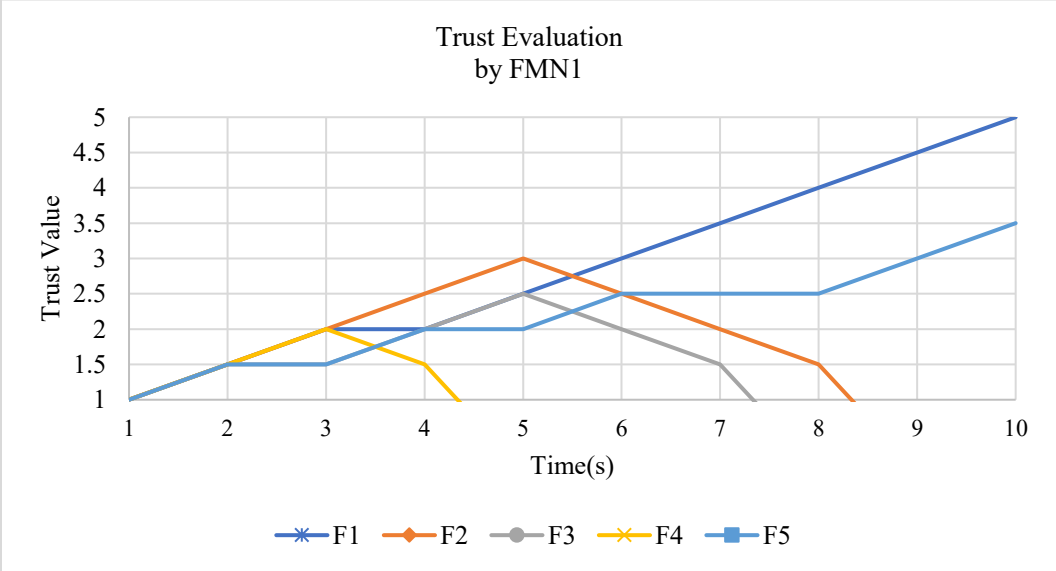


Figure 4: Trust Evaluation of Fog Nodes by Fog Manager Node (FMN1)

Fig 6.2 shows how the F_6, F_8, and F_9 maintain their trust value and achieve the highest Trust and how the F_7 and F_9 do the malicious activity and were removed from the network.

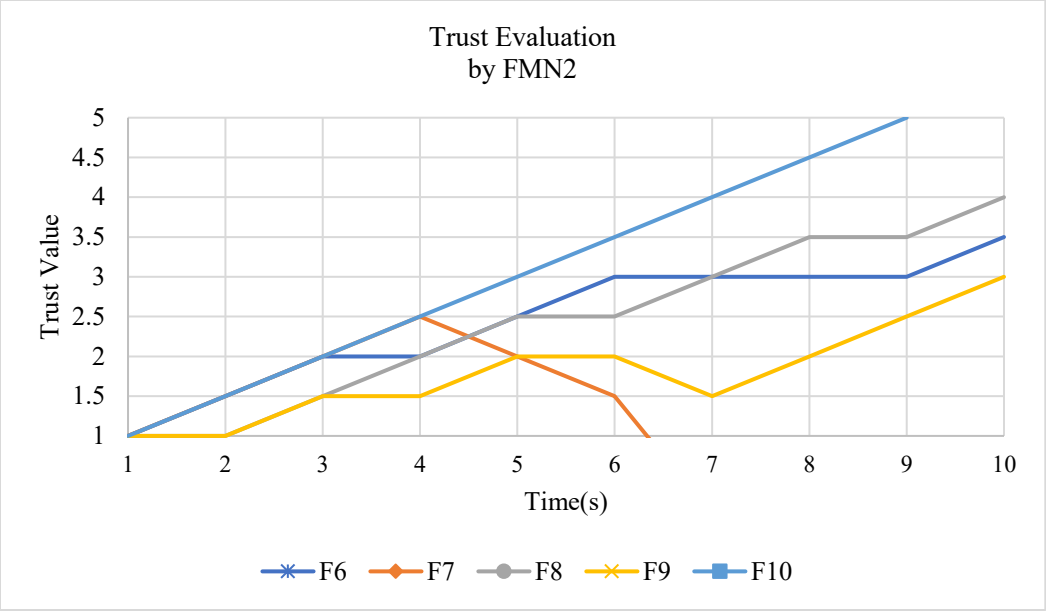


Figure 5: Trust Evaluation of Fog nodes by Fog Manager Node (FMN2)

In above fig 6.2 we can see that when the F_6, F_8, and F_9 did positive behavior then their values increased from 1 to 1.5 and finally reached they're to the highest value. and likewise, when F_2, F_3, and F_4 misbehave than their values decrease by 0.5. and at the end, the value of these fog nodes becomes 0 and finally these nodes are removed from the network.

In fig. 6.3 below we can see the overall simulation result of all fog nodes in the network. F_2, F_3, F_7, F_2, and F_9 are removed from the network due their malicious activity.

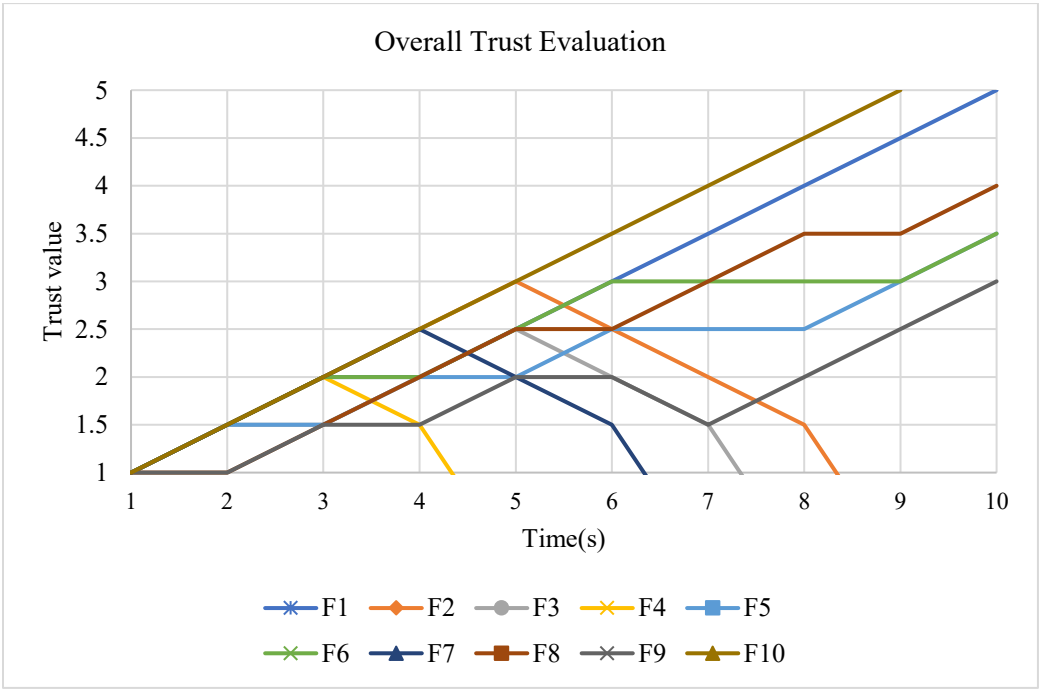


Figure 6: Trust Evaluation of all Fog nodes by their Fog Manager nodes (FMN)

4.1.2. Honesty updating

After the trust evaluation of service provider fog nodes, the FMN updates the honesty of the informer. Informer is the fog node that interacts with the service provider. It gives feedback about the service provider. The honesty of the informer is updated based on their honest review of the service provider.

Figure 7, 8, and 9 show how the honesty of the fog node is updated. Honesty is the reward and penalty of honest or dishonest interactor. In below fig F_1, F_3, and F_5 fog node become dishonest by providing wrong information about the service provider.

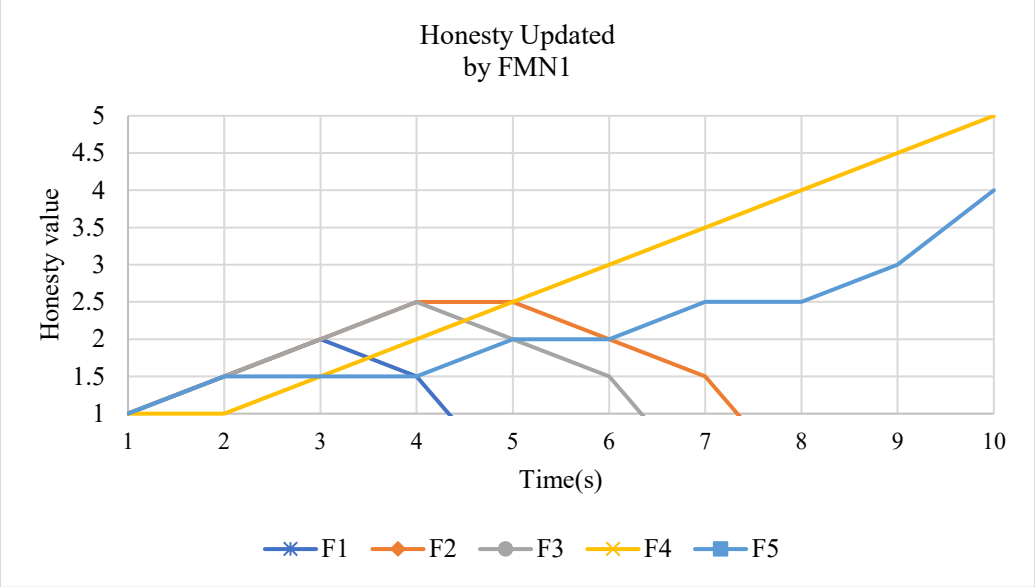


Figure 7: Honesty Updating of Fog Nodes by Fog Manager node (FMN1)

In Fig 7 we can see that F_1, F_3, and F_5 interact with service provider fog nodes and give honest feedback about them. So, the Fog manager node increases their honesty by 0.5. and that's how by the passage of time they get the highest honesty on every honest feedback about their service provider in the network. On another hand, F_2, F_3, and F_4 misbehave and give dishonest feedback about their service provider and that's why they lose their honesty and are removed from the network.

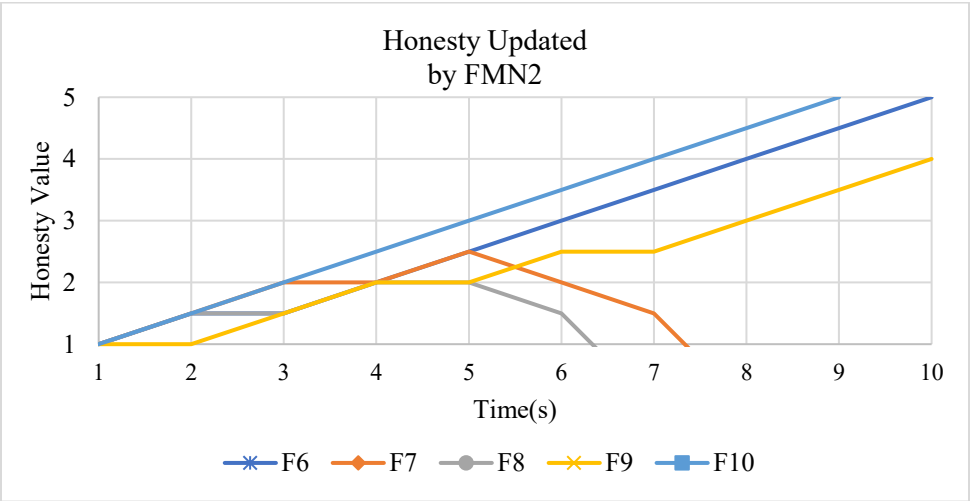


Figure 8: Honesty updating of Fog Nodes by Fog Manager Node (FMN2)

In figure 8 fog node F_6, F_8, and F_9 get highest trust value by giving honest reports to the Fog manager node of the service provider. And F_7 and F_{10} send the wrong report to the fog manager node and loss their honesty.

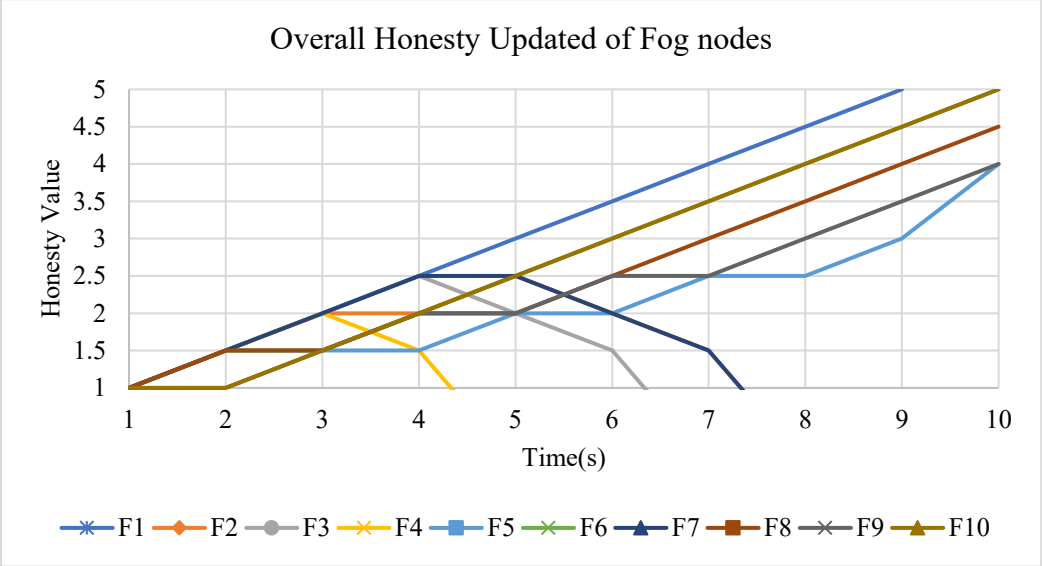


Figure 9: Honesty Updating of all Fog Nodes by their Fog Manager Nodes (FMN)

In figure 9 we can see the overall honesty updated of fog nodes in their networks. Fog manager node remove the dishonest reporter and give an update to the SDN controller so that he could report all fog manager nodes in the network.

The final result we can see in table 3 and 4.

Table 3: Fog Nodes Present in Network with their ID, Honesty, and Trust level

Node ID	***Node Trust***	***Node Honesty***
F_1	5.0	5.0
F_2	-	-
F_3	-	-
F_4	-	-
F_5	4.0	5.0
F_6	3.5	5.0
F_7	-	-
F_8	4.0	4.5
F_9	3.0	4.0
F_{10}	5.0	5.0

Malicious fog node ID are kept in Malicious fog node list so when malicious node tries to re-enter in the network and request the fog manager node then FMN match their ID in fog list and declined their request.

Table 4: Malicious Fog Node List (ML_F)

Malicious Node ID	Value
F_2	0.0
F_3	0.0
F_4	0.0
F_7	0.0
F_{10}	0.0

4.2. Experiment II

At the first experiment, we set the α and β respectively 0.3 and 0.3. Initial trust value of all fog nodes *Trust* = 1 and *Honesty value* = 1. and we keep a 1.3 threshold. we can see the result in Fig 6.7, 6.8, and 6.9.

4.2.1. Trust Evaluation

In figure 10 F_1, and F_3 try to misbehave and are removed from the network. on other hand F_2 F_4 and F_5 become most weighted and trusted fog node of network.

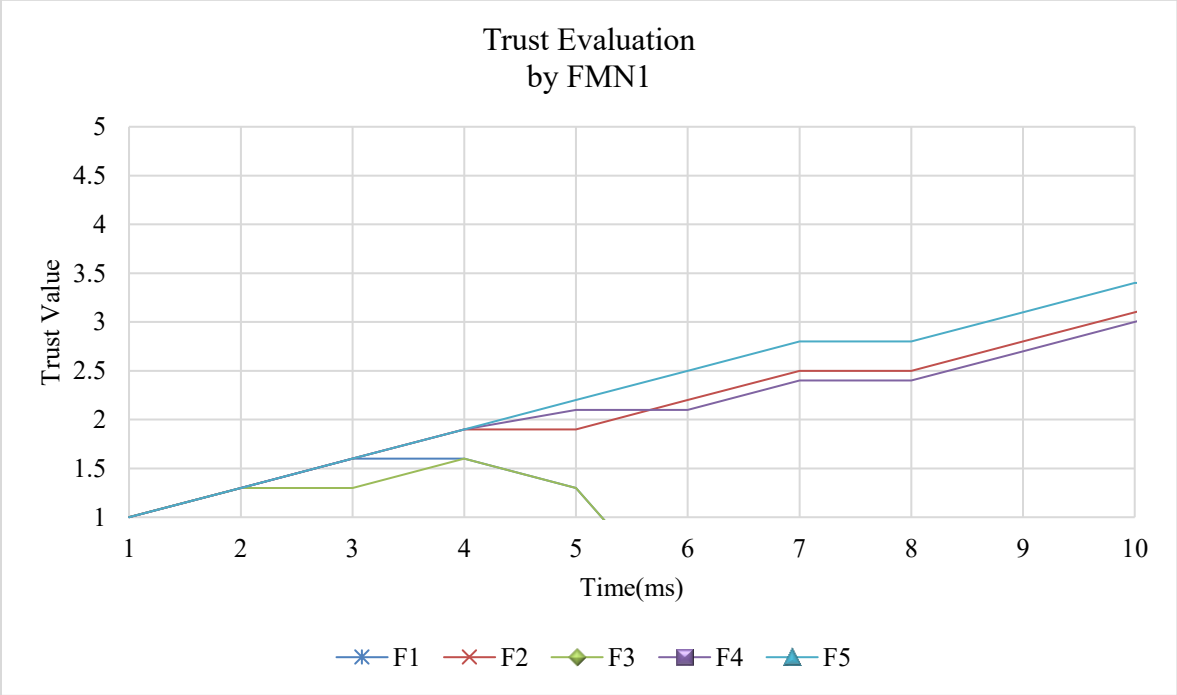


Figure 10: Trust Evaluation by FMN1

In figure 11 we can see that F_9 and F_7 did malicious activity and removed from the network. F_9 loss their trust at 3ms after two transactions. and F_7 after six transactions. F_6, F_8, and F_{10} reached at highest trust level.

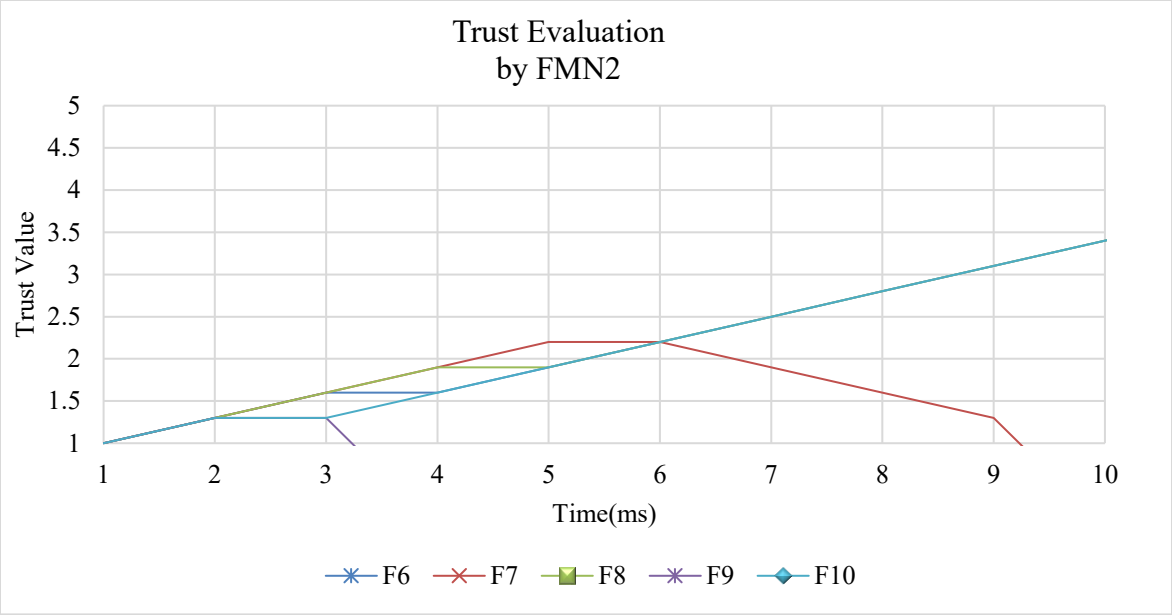


Figure 11: Trust Evaluation by FMN2

In figure 12 we can see overall trust evaluated for all fog nodes.

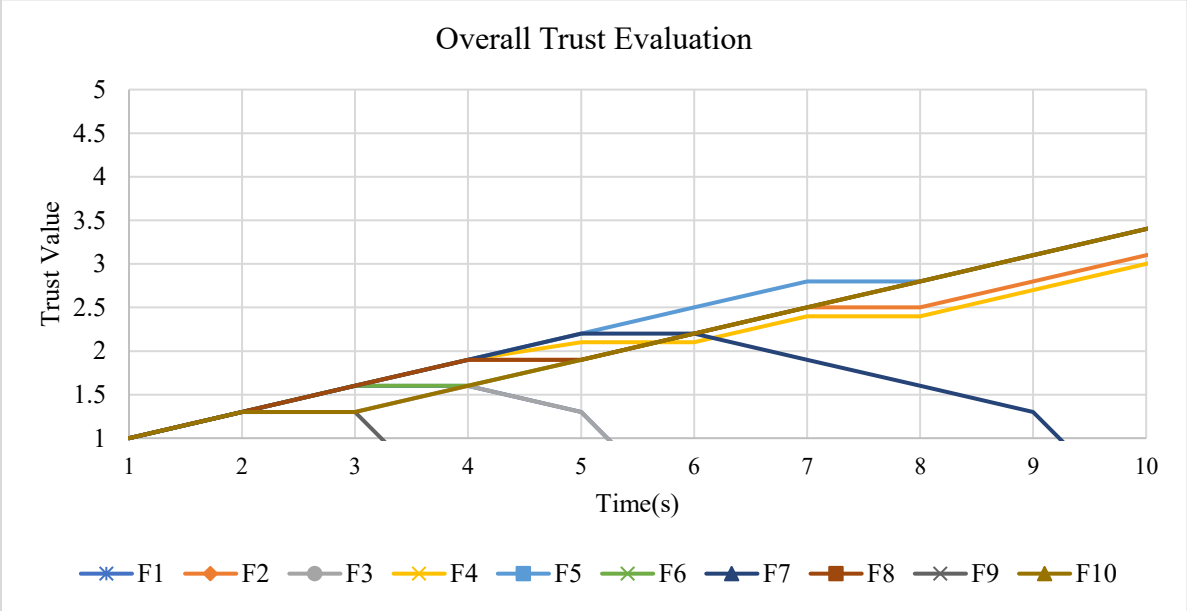


Figure 12: Overall Trust Evaluation

4.2.2. Honesty updating

In figure 13 F_3 and F_1 interact with service provider and report negative to the fog manager node. While the service provider is not malicious node. So, F_3 and F_1 is malicious and loss their honesty. F_2, F_4, and F_5 behave positive and increase their honesty.

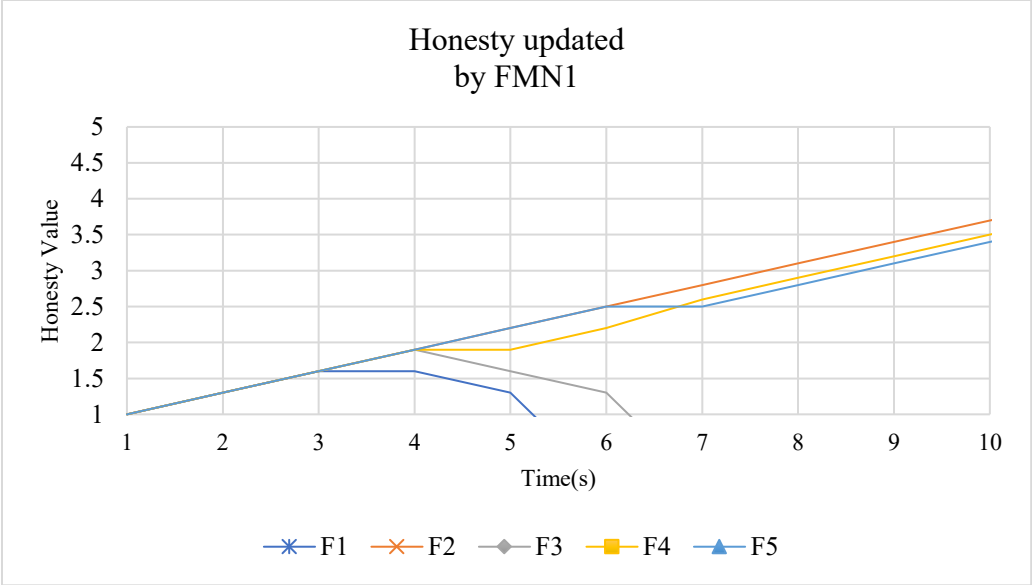


Figure 13: Honesty Updated by FMN1

In figure 14 we can see that F_6, F_8, and F_{10} are increases their honesty by providing honest report to fog manager about service provider. While on the other hand F_7 and F_9 provide dishonest report and loss their honesty.

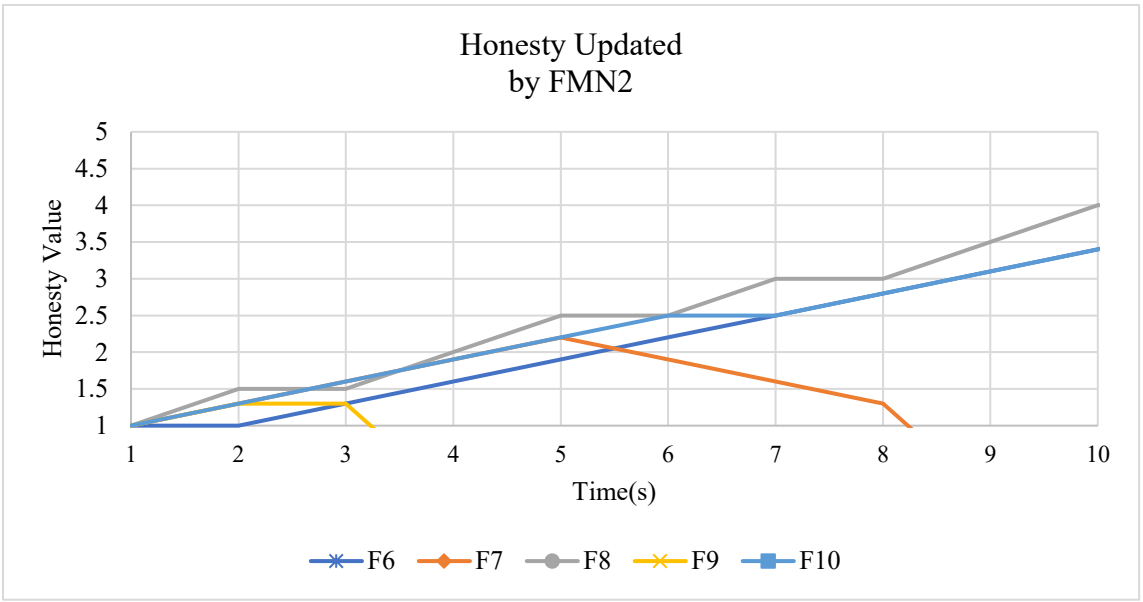


Figure 14: Honesty Updated by FMN2

In figure 15 we can see overall activity of fog nodes in network.

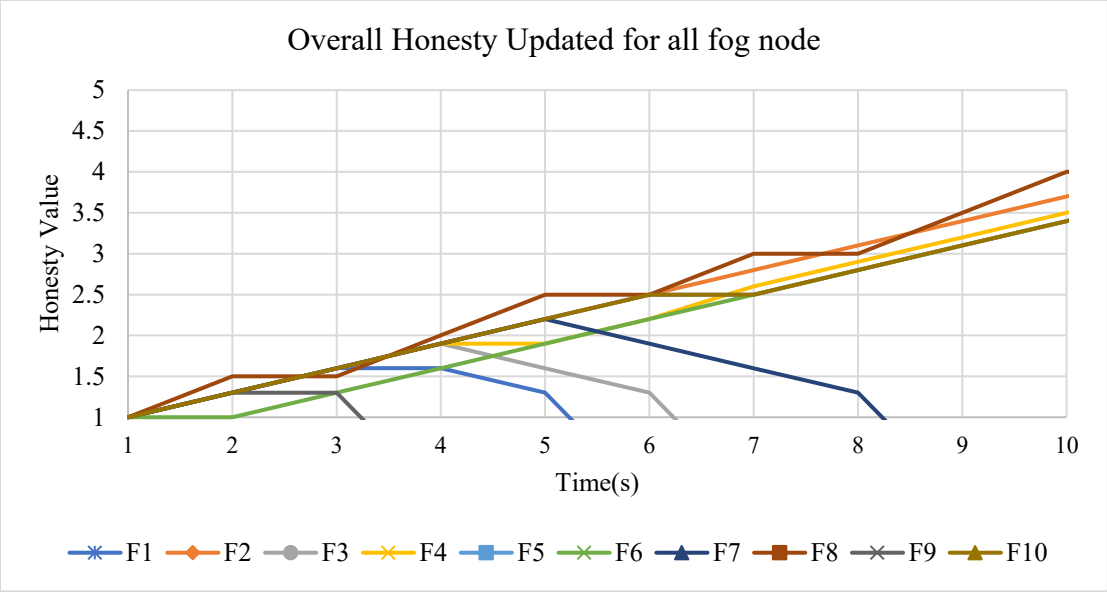


Figure 15: Overall Honesty Updated for all Fog Nodes

We can see the Final output in table 5 is obtained from our proposed weighted trust evaluation algorithm. Untrusted fog nodes are removed from the list and added in malicious fog list.

Table 5: Fog nodes present in network with their ID, Honesty, and Trust level

Node ID	*Node Trust*	*Node Honesty*
F_1	-	-
F_2	3.1	3.7
F_3	-	-
F_4	3.0	3.5
F_5	3.4	3.4
F_6	3.4	3.4
F_7	-	-
F_8	3.4	4.0
F_9	-	-
F_{10}	3.4	3.4

Table 6: Malicious Fog Node List (MLF)

Malicious Node ID	*Value*
F_1	0.0
F_3	0.0
F_7	0.0
F_9	0.0

4.3. Access Control Management

Table 7 shows iFogSim data for fog nodes joining networks. Performance measurements include access control method implementation time in milliseconds. Some malicious fog nodes are rejected by the fog management. Fog nodes cannot connect to the network due to access control. The maximum fog node

access duration has raised from 2.3 to 3.4 milliseconds. When malicious fog attempts to enter the network, FMN matches their ID with the malicious fog node list and denies their request.

Table 7: Example of the Various Metrics Reported by iFogSim

Fog ID	STATUS	Fog Manager Node	Time
F_1	FAILURE	-	-
F_2	ACCEPT	1	2.5
F_3	FAILURE	-	-
F_4	ACCEPT	1	3.5
F_5	ACCEPT	1	3
F_6	ACCEPT	2	3.4
F_7	FAILURE	-	-
F_8	ACCEPT	2	2.3
F_9	FAILURE	-	-
F_{10}	ACCEPT	2	3
F_{11}	ACCEPT	1	3.6
F_{12}	FAILURE	-	-
F_{13}	FAILURE	-	-
F_{14}	ACCEPT	2	2.6
F_{15}	ACCEPT	1	3.4

5. Conclusion

Many modern IoT devices need communication to other IoT devices, gateways, or network components due to their processing power, mobility, and increased discovery. Thus, an untrusted fog node may get illegal access to resources, compromising IoT network functionality. Malicious nodes can also alter data. Malware can also hinder system performance. To enable global growth and end-user migration, fog computing brings cloud computing and services to the network edge. SDN-based fog computing infrastructure security architecture was described in this work. To boost fog layer performance, we recommended using adjacent smart devices like smartphones and tablets' idle resources. This approach raises security challenges. We offer access control and trust evaluation model. We showed how our design can solve fog network security issues. Our approach calculates trust levels from node behavior. We provided algorithms for applying our hypothesis. Our implementation showed that the fog system could distinguish trusted and untrusted dynamic nodes. In iFogSim, we showed the fog network access control and trust evaluation algorithms' results. Simulation results showed the importance of integrating our technique within the Fog paradigm to resolve the trust and access control issue of newly joining nodes and IoT devices. We tested our algorithms on different parameters and thresholds and found and eliminated malicious fog nodes using FMN in fog computing.

References

[1] Tomovic, Slavica, Kenji Yoshigoe, Ivo Maljevic, and Igor Radusinovic. "Software-defined fog network architecture for IoT." *Wireless Personal Communications* 92 (2017): 181-196.

[2] Shehada, Dina, Amjad Gawanmeh, Chan Yeob Yeun, and M. Jamal Zemerly. "Fog-based distributed trust and reputation management system for internet of things." *Journal of King Saud University-Computer and Information Sciences* 34, no. 10 (2022): 8637-8646.

[3] Hosseinpour, Farhoud, Ali Shuja Siddiqui, Juha Plosila, and Hannu Tenhunen. "A security framework for fog networks based on role-based access control and trust models." In *Research and Practical*

Issues of Enterprise Information Systems: 11th IFIP WG 8.9 Working Conference, CONFENIS 2017, Shanghai, China, October 18-20, 2017, Revised Selected Papers 11, pp. 168-180. Springer International Publishing, 2018.

[4] Kumar, Sachin, Prayag Tiwari, and Mikhail Zymbler. "Internet of Things is a revolutionary approach for future technology enhancement: a review." *Journal of Big data* 6, no. 1 (2019): 1-21.

[5] Diro, Abebe Abeshu, Haftu Tasew Reda, and Naveen Chilamkurti. "Differential flow space allocation scheme in SDN based fog computing for IoT applications." *Journal of Ambient Intelligence and Humanized Computing* (2024): 1-11.

[6] Khakimov, Abdukodir, Abdelhamied A. Ateya, Ammar Muthanna, Irina Gudkova, Ekaterina Markova, and Andrey Koucheryavy. "IoT-fog based system structure with SDN enabled." In *Proceedings of the 2nd international conference on future networks and distributed systems*, pp. 1-6. 2018.

[7] Al-Khafajiy, Mohammed, Thar Baker, Muhammad Asim, Zehua Guo, Rajiv Ranjan, Antonella Longo, Deepak Puthal, and Mark Taylor. "COMITMENT: A fog computing trust management approach." *Journal of Parallel and Distributed Computing* 137 (2020): 1-16.

[8] Hakiri, Akram, Bassem Sellami, Prithviraj Patil, Pascal Berthou, and Aniruddha Gokhale. "Managing wireless fog networks using software-defined networking." In *2017 ieee/acs 14th international conference on computer systems and applications (aiccsa)*, pp. 1149-1156. IEEE, 2017.

[9] Shehada, Dina, Chan Yeob Yeun, M. Jamal Zemerly, Mahmoud Al-Qutayri, Yousof Al-Hammadi, and Jiankun Hu. "A new adaptive trust and reputation model for mobile agent systems." *Journal of Network and Computer Applications* 124 (2018): 33-43.

[10] Sadique, Kazi Masum, Rahim Rahmani, and Paul Johannesson. "Trust in Internet of Things: An architecture for the future IoT network." In *2018 International Conference on Innovation in Engineering and Technology (ICIET)*, pp. 1-5. IEEE, 2018.

[11] Patwary, Abdullah Al-Noman, Nishat Hossain, and Mohammad Aslam Sami. "A detection approach for finding rogue fog node in fog computing environments." *Am. J. Eng. Res* 8 (2019): 2320-0847.

[12] Salonikias, Stavros, Ioannis Mavridis, and Dimitris Gritzalis. "Access control issues in utilizing fog computing for transport infrastructure." In *Critical Information Infrastructures Security: 10th International Conference, CRITIS 2015, Berlin, Germany, October 5-7, 2015, Revised Selected Papers 10*, pp. 15-26. Springer International Publishing, 2016.

[13] Panja, Biswajit, Sanjay Kumar Madria, and Bharat Bhargava. "A role-based access in a hierarchical sensor network architecture to provide multilevel security." *Computer Communications* 31, no. 4 (2008): 793-806.

[14] Rathee, Geetanjali, Rajinder Sandhu, Hemraj Saini, M. Sivaram, and Vigneswaran Dhasarathan. "A trust computed framework for IoT devices and fog computing environment." *Wireless Networks* 26 (2020): 2339-2351.

[15] Pallavi, K. N., and V. Ravi Kumar. "Authentication-based access control and data exchanging mechanism of IoT devices in fog computing environment." *Wireless Personal Communications* 116 (2021): 3039-3060.

[16] Zhang, Peng, Zehong Chen, Joseph K. Liu, Kaitai Liang, and Hongwei Liu. "An efficient access control scheme with outsourcing capability and attribute update for fog computing." *Future Generation Computer Systems* 78 (2018): 753-762.

[17] Yu, Zuoxia, Man Ho Au, Qiuliang Xu, Rupeng Yang, and Jinguang Han. "Towards leakage-resilient fine-grained access control in fog computing." *Future Generation Computer Systems* 78 (2018): 763-777.

[18] Daoud, Wided Ben, Mohammad S. Obaidat, Amel Meddeb-Makhlouf, Faouzi Zarai, and Kuei-Fang Hsiao. "TACRM: trust access control and resource management mechanism in fog computing." *Human-centric Computing and Information Sciences* 9 (2019): 1-18.

[19] Misra, Sudip, and Ankur Vaish. "Reputation-based role assignment for role-based access control in wireless sensor networks." *Computer Communications* 34, no. 3 (2011): 281-294.

[20] Chowdhary, Ankur, Dijiang Huang, Adel Alshamrani, Myong Kang, Anya Kim, and Alexander Velazquez. "TRUFL: distributed trust management framework in SDN." In *ICC 2019-2019 IEEE*

International Conference on Communications (ICC), pp. 1-6. IEEE, 2019.

[21] Ukil, Arijit. "Secure trust management in distributed computing systems." In *2011 Sixth IEEE International Symposium on Electronic Design, Test and Application*, pp. 116-121. IEEE, 2011.

[22] Tomasin, Stefano, Simone Zulian, and Lorenzo Vangelista. "Security analysis of lorawan join procedure for internet of things networks." In *2017 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pp. 1-6. IEEE, 2017.

[23] Manuel, Paul. "A trust model of cloud computing based on Quality of Service." *Annals of Operations Research* 233 (2015): 281-292.

[24] Zhang, Peng, Joseph K. Liu, F. Richard Yu, Mehdi Sookhak, Man Ho Au, and Xiapu Luo. "A survey on access control in fog computing." *IEEE Communications Magazine* 56, no. 2 (2018): 144-149.

[25] Al Harbi, Saud, Talal Halabi, and Martine Bellaiche. "Fog computing security assessment for device authentication in the internet of things." In *2020 IEEE 22nd International Conference on High Performance Computing and Communications; IEEE 18th International Conference on Smart City; IEEE 6th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pp. 1219-1224. IEEE, 2020.

[26] Ukil, Arijit. "Trust and reputation based collaborating computing in wireless sensor networks." In *2010 Second International Conference on Computational Intelligence, Modelling and Simulation*, pp. 464-469. IEEE, 2010.

[27] Alshehri, Mohammed, and Brajendra Panda. "A blockchain-encryption-based approach to protect fog federations from rogue nodes." In *2019 3rd Cyber Security in Networking Conference (CSNet)*, pp. 6-13. IEEE, 2019.

[28] Apat, Hemant Kumar, Bibhudatta Sahoo, and Prasenjit Maiti. "Service placement in fog computing environment." In *2018 International Conference on Information Technology (ICIT)*, pp. 272-277. IEEE, 2018.

[29] Awaisi, Kamran Sattar, Assad Abbas, Samee U. Khan, Redowan Mahmud, and Rajkumar Buyya. "Simulating fog computing applications using iFogSim toolkit." *Mobile edge computing* (2021): 565-590.

Machines and Algorithms
http://www.knovell.org/mna

Research Article

Optical Character Recognition for Nastaleeq Printed Urdu Text using Histogram of Oriented Gradient Features

Muhammad Awais[1, *], Fatima Yousaf [2] and Tanzeela kousar[3]

[1]Department of Computer Science, Bahauddin Zakariya University, Multan, 60000, Pakistan
[2]Department of Computer Science and Information Technology, University of Chakwal, Chakwal, 48800, Pakistan
[3]Institute of Computer Science and Information Technology, The Women University Multan, 60000, Pakistan
[*]Corresponding Author: Muhammad Awais. Email: awaisahmadd555@gmail.com
Received: 12 October 2023; Revised: 1 January, 2024; Accepted: 5 February 2024; Published: 14 March 2024
AID: 003-01-000032

Abstract: The focus of research on optical character recognition (OCR) has been to digitize text in images. Urdu OCR is a challenging task because of its complexity, where a character can have multiple inflections depending on its position in the word, making it more difficult than English and similar languages. The proposed research aims to detect offline Urdu printed text using a segmentation-free approach, which means a holistic approach is taken. Horizontal histogram projection is used to extract text lines from an image, while connected components labelling is used for ligature segmentation in the extracted image to text line. To train the proposed model, a set of 14 statistical features along with HOG features are extracted for each sub-word/ligature. An open-source dataset UPTI is used to train and test the proposed algorithm, and SVM with RBF kernel function is used for the classification of ligatures. The proposed algorithm achieves a 97.3%-character recognition rate on the given dataset.

Keywords: Urdu language; Optical Character Recognition; HOG features; Connected Components; Support Vector Machine;

1. Introduction

Pattern recognition is a crucial aspect of both data science and computer vision, and its primary aim is to identify specific patterns within data and understand their connections. Optical character recognition (OCR) is a well-known example of this type of problem. Researchers have devoted significant effort to OCR over the last 30 years, but despite many advancements, there is still a requirement for more effective techniques to be developed [1].

Extracting text from printed documents is a difficult task for the Urdu language. With advancements in machine learning, there has been an increased expectation for text extraction from images, leading to the development of various approaches for Urdu optical character recognition (OCR) [2]. The field of OCR has seen significant improvement over the last 20 years, with widespread use in industries such as pharmaceuticals, accounting, and medical. OCR has numerous everyday applications, especially in the banking and financing sectors, such as reading and digitizing handwritten banker checks, verifying signatures, and sorting checks by zip code [3]. This technology has significantly reduced turnaround time, resulting in significant economic benefits. OCR is also being used for data processing by government and non-governmental organizations that require processing of thousands of survey forms [4]. The process of

computerizing large volumes of paper documents and books typically requires significant human effort and time. However, automation of this process through OCR can efficiently save both time and human resources [5].

Urdu belongs to the group of cursive scripts, which is characterized by separate or linked characters that form partial words called Ligatures. The commonly used fonts for printing Urdu are Naskh and Nastaleeq. The proposed methodology focuses on recognizing printed Urdu text using the Nastaleeq font. To achieve this, Histogram of Oriented Gradients (HOG) is used as a feature descriptor, which is known for its effectiveness in image segmentation and object detection using machine learning classification models. In OCR, these descriptors can also be applied to represent sub-word or character images.

The focus of this article is on recognizing printed Urdu text in Nastaleeq font offline and the obstacles involved. To improve the recognition accuracy, gradient features have been utilized to classify individual sub-words and characters. Additionally, the approach taken in this study is holistic, treating each ligature as a recognition unit. The paper also emphasizes the procedure for gathering training data, which includes ligature images and corresponding class IDs (labels) obtained from text line images in the UPTI [6] dataset, stored in a separate file.

1.1. Contributions to the Proposed work

The following are the main achievements of our suggested investigation.:

- A new method has been presented in this research study for automatic extraction of training and validation data from the UPTI dataset. By utilizing this technique, ligature images for training can be produced from the UPTI dataset, while also obtaining their corresponding class ID.
- A collection of characteristics that can enhance the accuracy of recognition has also been proposed in this study. This set of features is composed of HOG-based and statistical features.

The rest of the paper is structured in the following manner: Section 2 contains an extensive discussion of the key contributions made towards creating a recognition system for printed Urdu text in an image. Several techniques for recognizing printed Urdu text are explored in this section. The implementation methodology and pertinent information about our proposed system are outlined in Section 3. Finally, Section 4 presents the output results of our experiments and a discussion of their implications.

2. Literature Review

The Urdu language shares a script similar to that of Arabic, which means that OCR techniques developed for Arabic can also be utilized for Urdu. Therefore, this section discusses the OCR work done on both languages. Research conducted by Saeeda Naz et al. [7] recognized Urdu script through an implicit segmentation method when combined with a Multidimensional-LSTM Recurrent Network operating on UPTI dataset information. The developed system displayed a Nastaleeq Urdu font recognition precision rate of 98%. A printed Urdu Nastaleeq font text recognition methodology proposed by Israr Ud Din et al. [8] uses a sliding window approach to extract nine statistical features totaling 116 dimensions for each sub-word image. An accuracy rate of 92% was achieved by applying these features to the UPTI dataset when using a Hidden Markov model (HMM) for training.

An all-encompassing approach served as the basis for Urdu text recognition work conducted by Toflk et al. [9]. The text lines in the document get separated by using horizontal histogram projection before any text recognition process begins. A connected component algorithm segments each sub-word through its procedure. Feature descriptors of SIFT and SURF are measured on each separated sub-word output with segmentation. The system generates 1600 categories of sub-words which include multiple diacritic marks. The system matches features from input document sub-words against the 1600 category sub-words for identification. When feature matching leads to the highest score a designated sub-word receives its corresponding ID value. The ID becomes comparable to the sub-word file for identifying the matching sub-word. A training of 23204 sub-words/ligatures within the system achieved a 95% accuracy rate.

Israr Uddin et al. [10] have introduced a comprehensive strategy for recognizing Urdu language, specifically focusing on the Nastaleeq Urdu Script. To accomplish this task, the authors utilized the Discrete Wavelet Transformation technique to extract features from sub-words, which were then utilized to train a Hidden Markov Model as a classifier. The authors evaluated the system's performance using 2000 distinct and commonly used Urdu ligatures from the center for language engineering (CLE) dataset [11]. The findings indicate that the system achieved a recognition accuracy rate of 88.87% on 10,000 Urdu sub-words.

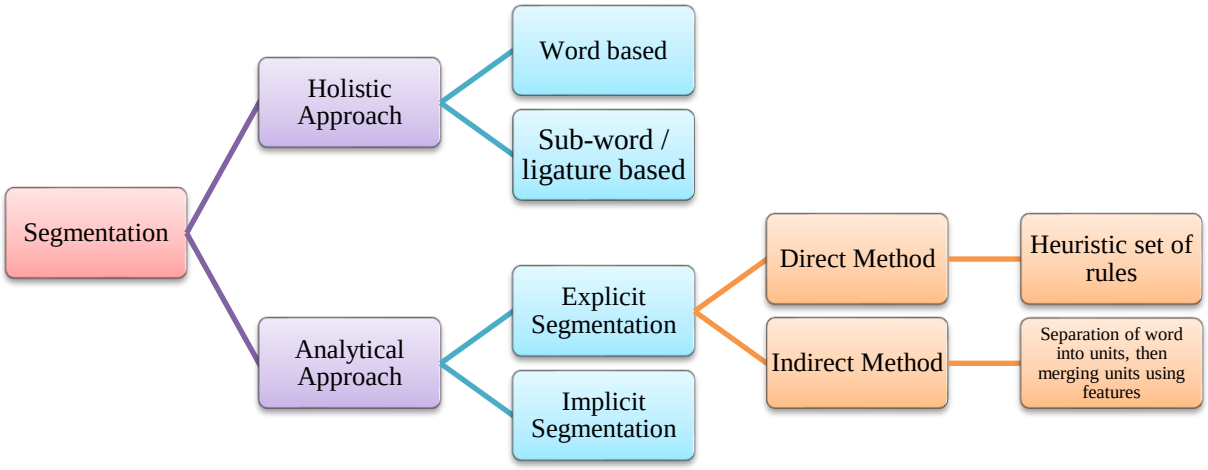


Figure 1: Categorization of OCR Techniques from the Segmentation Aspect

Pal & Sarkar [12] developed a technique to recognize isolated Urdu characters. The system segments the document image into text lines and then into individual characters. Character images are represented using water reservoir, topological, and contour features. The features include reservoir number, direction, flow level, position, and height, and topological components, loops, position relative to the character boundary, and loop-to-height ratio. The character contours are represented by projection profile features. A decision tree classification technique is used for character recognition.

Hussain et al. [13] developed an analytical approach for Urdu text recognition. They extracted primary and secondary ligatures from scanned document images and preprocessed them for noise removal. Characters were grouped into four classes, and character endpoints were computed using a local window sliding over every primary ligature's thinned image. They segmented 5,249 primary ligatures into 79,093 graphemes with 250 unique shapes. Low-frequency DCT features were computed using right-to-left sliding windows for all graphemes, and separate HMMs classifiers were trained for each grapheme class. For recognition, a query sub-word/ligature was split into primary and secondary ligatures, and the primary ligature was segmented into individual graphemes, which were recognized using trained HMM classifiers. The ligature was then generated by combining the recognized graphemes. The system achieved an 87.44% accuracy rate on 18,409 query ligatures.

In another work, Hassan et al. [14] employed BLSTM with CTC output layer to recognize Urdu text lines. Text line height is normalized to 30 pixels, and each column of a text-line image is fed to train the classification network. Results show accuracy rates of 94.85% and 86.43% for recognizing printed Urdu text lines with and without considering variations in character shapes, respectively. Ahmed et al. [26] used the same technique for recognizing cursive and isolated scripts. Hassan et al. achieved an accuracy of 96% on the UPTI dataset.

Table 1: Summary of Different Recognition Techniques

Study	Dataset	Recognition Techniques	Language Script	Results (accuracy)
Farhan M. A. Nashwan et al. [15]	Custom	DCT and center of gravity with Euclidean Distance score comparison	Arabic	84.8%
Ouled Jaafri Yamina et al. [16]	Custom dataset (30500 samples)	Set of 14 statistical features with SVC classifier	Arabic	95.03%
Hussein Osman et al. Error! Reference source not found.**]**	Watan-2004 APTI	ANN	Arabic	97.94%
Saad M. Darwish, Khaled O. Elzoghaly [18]	PATS-A01, APTI	14 statistical features from grey level Co-occurrence matrix with fuzzy KNN	Arabic	98.69%
Israr Uddin et al. [10]	CLE	DWT with HMM	Urdu	88.87%
Nazly Sabbour, Faisal Shafait [6]	UPTI	Shape context with KNN	Urdu, Arabic	86% (Arabic), 91% (Urdu)
Israr Ud Din et al. [19]	UPTI	116-dimensional Statistical features with HMM	Urdu	92%
Tofik et al. [9]	Custom	SIFT and Surf with Brute force Feature Matching	Urdu	95%
Mujtaba Husnain et al. Error! Reference source not found.	Custom	Statistical features and Raw pixels with CNN	Urdu	96.5%
Saeeda Naz et al. [7]	UPTI	MDLSTM (Analytical Approach)	Urdu	98%

Javed et al. [21] proposed an optical character recognition system using HMM classifier with 1282 high-frequency ligatures (HFLs). DCT-based features were used to represent each ligature image using sliding windows, and a separate class of models was trained for each ligature. A set of rules was used to associate recognized primary and secondary ligatures based on diacritic and dot position information. The system achieved a recognition rate of 92% on a dataset of 3655 ligatures, with errors mainly due to the system's inability to distinguish between ligatures that share the same primary main ligature body but differ only in diacritic and dot positions.

3. Proposed Methodology

The Following section describes the proposed methodology.

3.1. Dataset Description

The UPTI (Urdu Printed Text Images) dataset [22] is a freely available dataset that has been extensively employed for assessing various printed Urdu character recognition systems. Sabbour and Shafait [22]

created this dataset in 2013, and it comprises 10,063 lines of printed Urdu text written in the Nastaleeq font, all of which were sourced from the Daily Jang newspaper [23]. The dataset is segmented into three sub-groups: images of printed lines, images of printed ligatures, and images with noise. Figure 3 displays some sample images from the dataset.

(a) قائم ہے اور اسے صوبوں کو منتقل کیا جانا چاہئے۔

(b) قائم ہے اور اسے صوبوں کو منتقل کیا جانا چاہئے

(c) قائم ہے اور اسے صوبوں کو منتقل کیا جانا چاہئے

Figure 2: UPTI dataset printed text line Samples (a) Line text image – non-degraded (b) ligature-based image non-degraded (c) degraded/noisy ligature image.

If the dash that indicates the end of a sentence is taken away, the text image in a line is transformed into an image that is based on ligatures. To train the proposed system, distinct ligature training images are extracted from UPTI text line images by using the connected component labelling algorithm. The system's performance is then tested by using validation ligature images from the UPTI dataset that were not utilized in the training of the recognition model.

3.2. Methodology

This section provides a detailed explanation of the proposed recognition method. Each sub-word along with ligature serves as the core unit for recognition purposes within the developed approach. Holistic segmentation was selected over complex segmentation tasks because these approaches require extensive computation and pose significant identification challenges. Text lines have to be divided into sub-words/ligatures before an SVC classifier applies the recognition process for digitization of words through its output. The classification training process uses annotated ligature images drawn from text line images for extraction purposes. The training of the classifier depends on annotated ligature images. The trained classifier identifies predicted IDs for individual segmented ligatures contained in text-line images throughout the recognition phase. The predicted IDs are then matched with corresponding ligature text from a separate file containing all ligatures' texts and their IDs. The training process is explained in detail in the following section.

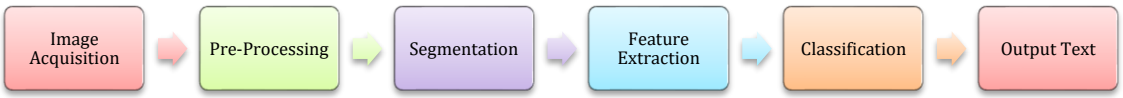


Figure 3: Ligature Recognition Process of Proposed OCR System

3.2.1. Preprocessing

The text lines provided in the dataset are in grayscale, but to make them more convenient for the designed classifier, they are converted to a binary form where each pixel is either 0 or 1. Figure 4 illustrates the process of converting the lines to binary.

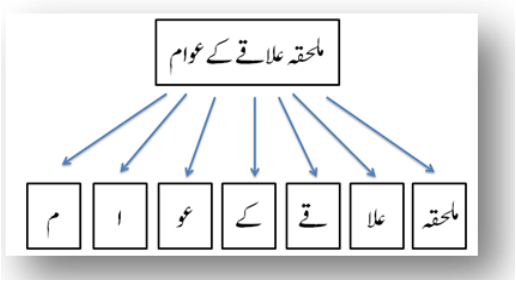
a) b)

Figure 4: a) Grayscale Image & b) Binarized Image

3.2.2. Ligature Segmentation

This section of the methodology focuses on the method of separating each ligature from a line of text. The connected component labelling algorithm is utilized to divide ligatures from images of text lines. This method of segmentation not only splits complete ligatures (combinations of primary and secondary ligatures) from the text line image but also separates primary and secondary ligatures from each other.

The process of extracting ligatures comes after binarization. This involves separating each image from a text line image and assigning a specific label to it, which is then replaced by a corresponding number during classification. The labels and their corresponding numbers are recorded in a CSV file. An illustration of this process can be seen in the figure below, which shows how a text line image is broken down into ligatures.



Unique Urdu Ligature	Ligature ID
ا	1
ملحقہ	5
علا	6
قے	7
کے	8
عو	9
م	10

Figure 5: Ligature IDs of Unique Urdu Ligatures

Each Urdu script word requires one to several linked characters in order to form itself. The Urdu script characters follow predetermined rules while locking together into multiple ligature combinations. We can classify these ligatures into three types: complete, primary, and secondary. One complete ligature includes Urdu words with their diacritical marks and unwanted dots. With its diacritics and dots removed, the letter becomes a simple ligature structure. Secondary ligatures develop through dots alongside diacritics found in ligatures. The figure 6 under this section shows how these three ligature categories appear.

a) b) c)

Figure 6: a) Complete Ligature, b) Primary Ligature and c) Secondary Ligature

The process of connected component labelling generates a primary and secondary ligature list from images of text lines with unique labels. The primary ligature list (PL_list) and secondary ligature list (SL_list) are produced. To create the complete ligature, the next step is to match the secondary ligatures with their corresponding primary ligatures. All complete ligatures are then saved in the complete ligature list (CL_list). The secondary ligatures are identified and separated from the combined list of primary and secondary ligatures extracted from each text line image during the segmentation process. The height of each ligature is calculated using its contour boundary, and stored in a separate list. After conducting experiments, it was determined that ligatures with a height less than 30% of the tallest ligature in the list are considered secondary and are stored in the SL_list. Starting indexes of diacritic marks in the SL_list are compared to the starting and end indexes of other primary ligatures in the combined list. If the starting index of a diacritic mark or secondary ligature falls between the starting and end indexes of a primary ligature in the PL_list,

it is considered a part of that primary ligature and its label is replaced with the label of the associated primary ligature. All labels are stored in the ligature label list (LL_list).

Figure 7: Segmentation of text line using connected component labelling

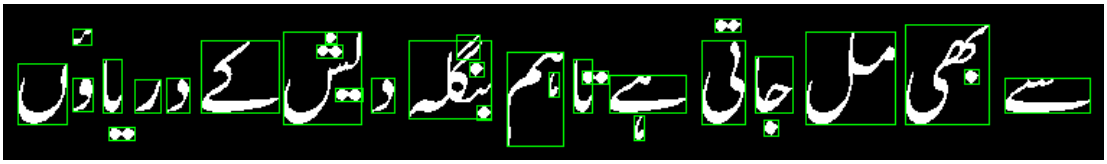


Figure 8: Identifying secondary ligatures from the ligatures list

Figure 9 illustrates how primary and secondary ligatures are linked. The significant role of excluding secondary ligatures from the complete list of all ligatures is emphasized in the association of ligatures. If the exclusion of secondary ligatures is not performed accurately, the segmentation process will be unable to separate complete ligatures from the text line images.

- **Algorithm 1: Algorithm for associating primary and secondary Ligatures**

Input: (*PL_list*, *SL_list*, *LL_list*)

Output: Updated connected components list having correct labels values for associated primary and secondary ligatures

DEFINE FUNCTION **Ligature_Association**(*PL_list*, *SL_list*, LL_*list*):

FOR (i ← 1 to **length**(*PL_list*))
  SET *PL_st_idx* ← *PL_list* [i][0]
  SET *PL_en_idx* ← *PL_list*[i][1]
  FOR (j ← 1 to **length**(*SL_list*)):
    // Starting index of a diacritic mark in the diacritic list
    SET *SL_st_idx* ← *SL_list*[j][0]
    // Connected components label of a diacritic mark in the diacritic list
    SET *SL_label* ← *SL_list*[j][4]
    IF (*SL_st_idx* >= *PL_st_idx* **AND** *SL_st_idx* <= *PL_en_idx*):
      // Assigning primary ligature label to its associated secondary ligature
      // Connected components label of primary ligature
      SET *PL_label* ← *PL_list*[i][4]
      // updating labels
      SET *LL_list*[LL_*list* == *SL_label*] ← *PL_label*
    END IF
  END FOR
END FOR
RETURN *LL_list*

Figure 9: Association of Primary and Secondary Ligatures

Algorithm 1 generates images of ligatures, which are subsequently labeled with their respective ground truth files for each text line derived from the UPTI dataset. The resulting output of the algorithm is presented below.:

Figure 10: UPTI text line segmentation using Connected component labelling

The ground truth for extracted ligatures is constructed using the following algorithm.

- **Algorithm 2: Text split algorithm**

1. Joiners ← ["ض","ص","ش","س","خ","ح","ج","ث","ت","ب",
"ن","م","ل","ك","ق","ف","غ","ع","ظ","ط","ى","ئ","ي","ه"]

2. non_joiners ← ["إ","أ","ة","ء","ؤ","و","ز","ر","ذ","د","ا"]

3. **DEFINE FUNCTION** split(***Text_line***):

4. **FOR** word **IN** Text_line **do**

5. **SET** Text_ligature_list ← [ ] // initialize with empty list

6. **SET** Temp_Characters_list ← [ ]

7. **SET** complete_Characters_list ← [ ]

8. **FOR** char_id **and** char **IN enumerate**(word) **do** // taking each character from
// single Arabic word to check
// whether it is joiner or nonjoiner.

9. **FOR** joiner_character **IN** joiners **do** // first checking **char** using joiner list

10. **IF** char **EQUALS** joiner_character **do** **//if** a character is a joiner

11. **ADD** char **IN** Temp_Characters_list[ ] //keep adding characters
//in list .

12. **END IF**

13. **END FOR**

14. **FOR** nonjoiner_character **IN** non_joiners **do** //checking **char** using nonjoiners

15. **IF** char **EQUALS** nonjoiner_character **do**

16. **ADD** char **IN** Temp_Characters_list [ ]

17. **SET** complete_Characters_list ← Temp_Characters_list

18. **SET** Temp_Characters_list ← [ ] //clearing the list to reuse
// for the next word in the next iteration

19. **SET** complete_ligature ← NULL //variable to combine all
// Characters in characters' list
// to generate single
// Ligature or sub-word

20. **FOR** each_character **IN** complete_Characters_list **do**

21. **SET** complete_ligature **TO** each_character **+** complete_ligature
22. **END FOR**
23. **ADD** complete_ligature **IN** Text_ligature_list [ ]
24. **END IF**
25. **END FOR**
// In the case we don't find any non-joiner, then all joiner characters
// stored in temp_characters_list will be output as a complete sub-
// word/ligature at the end of the word string length
26. **IF** char_id **EQUALS** length of (**word**) **do** //if it is the last index of Arabic
//Word string
27. **SET** complete_Characters_list ← Temp_Characters_list
28. **SET** Temp_Characters_list ← [ ]
29. **SET** complete_ligature ← NULL
30. **FOR** each_character **IN** complete_Characters_list **do**
31. **SET** complete_ligature **TO** each_character **+** complete_ligature
32. **END FOR**
33. **ADD** complete_ligature **IN** Text_ligature_list [ ]
34. **END IF**
35. **END FOR**
36. **END FOR**
37. **RETURN** Text_ligature_list [ ]

The annotated ligatures are produced through algorithm-2. Table 2 shows the distribution of ligatures used in our experiments.

Table 2: Detail of Training and Validation Set

Sets	No. of ligatures
Training set	3,005
Validation set	92,315

The training set includes the standard and unique ligatures whereas the validation set is a rough set that contains duplicate ligatures with varying sizes which include noise of different levels.

3.2.3. Feature Extraction

A feature descriptor is an algorithm that generates a set of feature vectors from an image, which represent the most prominent features in the image. We utilized a feature set, which included the Histogram of Oriented Gradients (HoG) with 9 orientations and statistical features, to classify ligatures. The HoG descriptor captures the image's shape and structure by extracting gradients (changes in x and y direction) and orientations (magnitude and direction) of the features. The image is partitioned into smaller regions, and for each region, the gradients and orientation are computed. To create the Histogram of Oriented Gradients (HoG), a histogram is generated for each smaller region based on the gradients and orientations of the pixel values. The SVM is capable of handling a large number of classes by transforming the multi-class problem into multiple binary classification problems. To compute the HoG feature of a single sub-

word/ligature image, the image is partitioned into several sub-portions, and gradients are computed for each block of 16x16.

The Gradient along the y-axis $\boldsymbol{G_y}$ of an Image I(x, y) is defined as the difference between the south pixels and north pixels of an image I(x, y).

$$\boldsymbol{G_y} = I(x, y+1) - I(x, y-1) \tag{1}$$

Similarly, a Gradient along the x axis $\boldsymbol{G_x}$of an Image I(x, y) is defined as the difference between the east pixels and west pixels of an image I(x, y).

$$\boldsymbol{G_x} = I(x+1, y) - I(x-1, y) \tag{2}$$

After computing the gradients of a ligature image along the x and y axis, its magnitude and direction are computed using equation 3.

$$G = \sqrt{(G_x)^2 + \left(G_y\right)^2} \tag{3}$$

$$\theta = arctan\left(\frac{G_y}{G_x}\right) \tag{4}$$

Computing Histogram of Gradients in 16×16 cells is done in the following steps.

- Each image of a ligature is divided into 16×16 cell blocks
- Along each 16×16 cell block HoG is calculated
- This gradient histogram is basically a 1D vector of 9 buckets (numbers) corresponding to angles ranging from 0 to 180 degrees (gap increments of 20 degrees).
- Values of these 256 cells (16X16) are binned and added into the 9 buckets of gradient histogram cumulatively.
- This process essentially reduces 256 values into 9 values for each cell block.

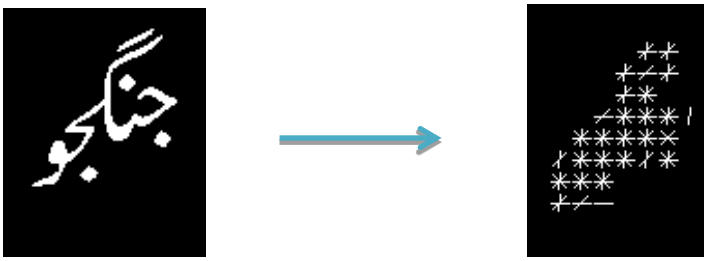


Figure 11: HOG features of the UPTI Urdu Ligature

The following statistical features are used in the system:

- Horizontal transition
- Vertical transition
- The ratio of black pixels over white pixels

To put it differently, the horizontal transition process entails examining of a sub-word image in the horizontal direction and tallying the number of instances where the pixel value changes from 1 to 0 or vice versa. Likewise, the vertical transition count method involves scanning the sub-word image from top to bottom and keeping track of the number of pixel value transitions.

Each sub-word image is divided into four parts. Top left portion, Top right portion, Bottom left portion, and Bottom right portion. The next 10 features are mentioned below.

- The Ratio of Black pixels over white pixels in the top left area of an image
- The Ratio of Black pixels over white pixels in the top right area of an image
- The Ratio of Black pixels over white pixels in the Bottom left area of an image

- The Ratio of Black pixels over white pixels in the Bottom right area of an image
- Number of Black pixels in Top left / Number of Black pixels Top Right
- Number of Black pixels in Bottom left / Number of Black pixels in Bottom Right
- Number of Black pixels in Top left / Number of Black pixels in Bottom left
- Number of Black pixels in Top right / Number of Black pixels in Bottom right
- Number of Black pixels in Top left / Number of Black pixels in Bottom right
- Number of Black pixels in Top right / Number of Black pixels in Bottom left

Holes: The number of holes presents within the sub-word image.

Our proposed system utilizes 15 features to classify sub-words or ligatures. These features are derived from the sub-word or ligature and passed on to the classifier to generate a class ID. Once the class ID is predicted, the corresponding sub-word/ligature text is selected from a separate file and added to a list to form a paragraph. Out of the 15 features, 14 are represented by a single integer or decimal value, while the HOG feature descriptor generates a feature map comprising 900 feature values distributed along 9 different orientations (100 features per orientation).

Table 3: Summary of features computed during feature extraction phase

Features	Features Name	Feature size
Gradient Features		
F1	HOG (9 orientations)	900
Gradient Features		
F2	Holes/loops	1
Statistical Features		
F3	Horizontal Transition	1
F4	Vertical Transition	1
F5	Black to White Ratio	1
F6	Black to White Ratio in Top left area.	1
F7	Black to White Ratio in Top right area.	1
F8	Black to White Ratio in the bottom left area.	1
F9	Black to White Ratio in the bottom right area.	1
F10	Black pixels in the Top left area / Black pixels in the Top right area	1
F11	Black pixels in the bottom left area / Black pixels in the bottom right area	1
F12	Black pixels in the top left area / Black pixels in the bottom left area	1
	Total Features	914

3.2.4. Training and Validation of Classifier

- Training of SVC classifier

The proposed system utilizes the Support Vector Classifier (SVC) as its machine learning classifier to predict the class ID for each segmented sub-word/ligature image based on a given set of features. The system is initially trained on annotated training data from the UPTI dataset, and during the recognition phase, the SVC classifier predicts the class ID for each segmented query image in a sequence similar to the sequence of words/sub-words in a paragraph. The predicted class ID for each sub-word is then stored in a list, which is used to select the corresponding sub-word/ligature text from a separate file.

The SVC classifier uses a hyperplane to separate data samples based on their feature points in n-dimensional feature space. To select the hyperplane that has the maximum distance from the nearest data points in either category, the concept of a kernel function K is introduced to transform non-linearly separable data in its input space into linearly separable data in a higher dimensional feature space. While selecting a hyperplane for linearly separable data is not challenging, most real-world problems deal with non-linearly separable data, making it more difficult to choose a hyperplane.

- Top of Form

$$K(x_i, x_j) = \emptyset(x_i) \,.\, \emptyset(x_j) \tag{5}$$

The right side of the equation $\emptyset(x_i) \,.\, \emptyset(x_j)$ represents the non-linear SVM function.

There are 4 kernel functions mostly used in SVM classification. In this work, we have used the Radial Basis function.

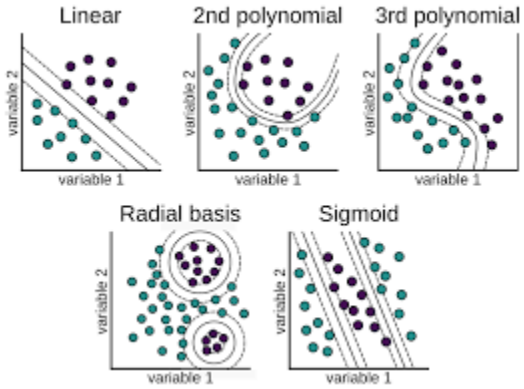


Figure 12: Different Kernel functions of the SVC Classification model.

Radial-based Function (RBF):

$$K(x_i, x_j) = exp\left(-\gamma \ \left|\left|x_i - x_j\right|\right|\right) + C \tag{6}$$

- Classifier Validation

The recognition task involves segmenting the printed Urdu text line images into individual ligatures images using the text line segmentation technique that was used during the training phase. This involves binarizing each text line image using the OTSU thresholding method and then extracting the ligatures from the image. Once the ligatures have been segmented, their features are used to recognize them and match them to their corresponding text. The training section provides more information about these features.

The dataset used for the validation purpose is more complex than the one used in training. Here are a few examples of the validation dataset.

Figure 13: Examples of Validation Set Extracted from UPTI

These features of each ligature image are passed to the SVC classifier that predicts their corresponding ligature ID. Using that predicted ligature ID the ligature text is selected from a separate ligature ID file that holds ligature text corresponding to their IDs.

4. Results and evaluation

This section presents the evaluation results of our proposed system on validation ligature images taken from the UPTI [6] dataset. The approaches mentioned in the proposed methodology section are evaluated on the validation set extracted from the UPTI dataset.

The proposed system is composed of two distinct classifiers. The first one is trained on 3005 unique Urdu ligature extracted from the UPTI dataset where each sub-word image has its own size (height and width) based on the ligature's dimensions. All statistical features are computed without resizing the ligature image. For the training process of calculating HoG features, each ligature image is resized to 100x100, and the Hog features are extracted along 9 orientations.

Figure 14: Ligature Images Samples Extracted from the UPTI Dataset.

4.1. Result Evaluation Metric

To evaluate the proposed system ligature recognition rate metric is used which is given below:

$$LRR = \frac{No \text{ of } ligatures\ correctly\ classified}{Total\ Number\ of\ ligatures}$$

Along with each query image, we have placed the ground truth. Using the proposed text split algorithm explained in preprocessing section, the ground truth paragraph text is converted into individual sub-word texts and stored in a list. Then each predicted sub-word is matched to the sub-word/ligature present in the ground truth sub-words list. Whenever the segmented sub-word/ligature is not classified correctly, the matching score is not added to the score list. This score list is equal to the number of correctly classified sub-words and it is divided by the total number of sub-words to evaluate the accuracy of our recognition system.

The proposed system is evaluated on a validation set extracted from the UPTI dataset that comprises 92,000 images of printed Urdu ligatures belongs to 3,005 unique ligatures classes.

Table 4: Comparative Analysis of Various Studies.

Study	No. of Unique ligatures	LRR	Recognition of complete ligatures
Israr Uddin et al. [19]	2028	97.93%	No
Javed and Hussain [24]	1692	92.73%	No
Akram et al. [25]	1475	97.87%	No
Javed et al. [21]	1282	92.00%	No

Akram et al. ERROR! REFERENCE SOURCE NOT FOUND.	1475	87.15%	Yes
Israr Uddin et al. [10]	2017	88.87%	Yes
Proposed	3005	97.39%	Yes

5. Conclusion and Discussion

Our research work has presented a method for recognizing printed Arabic and Urdu Script without using segmentation. Our approach includes incorporating HOG feature descriptors, which have produced promising recognition results. We obtained our training data from the UPTI dataset, using 3005 unique ligature images of printed Urdu Script, and annotated the training and validation ligatures using the ground truth text files of the UPTI dataset. This research work has the following key findings, including HOG feature-based classification that outperforms other methods. The font size of the text in the image doesn't impact the recognition performance of the system when the system is trained using these HOG features. We also discovered that over-segmentation or under-segmentation can negatively impact recognition, and that appropriate preprocessing, such as thinning and noise removal, is crucial to avoid these issues. However, our proposed technique is limited to printed text and cannot handle handwritten text due to the complexity of ligature overlapping and variations in shape. Finally, future research directions may include exploring recognition in multi-font text, addressing text overlap during segmentation, and incorporating diacritics.

Declaration of Competing Interests: The Authors declare that they have no competing interest that could have been appeared to influence the work reported in this paper.

Acknowledgements: The research in this work has been supported by Center for Artificial Intelligence Research (CAIR), Department of Computer Science, Bahauddin Zakariya University, Multan, Pakistan. We also acknowledge the publishers of UPTI dataset which is used in this work.

References

[1] Singh, Amarjot, Ketan Bacchuwar, and Akshay Bhasin. "A survey of OCR applications." *International Journal of Machine Learning and Computing* 2, no. 3 (2012): 314.

[2] Khan, Naila Habib, and Awais Adnan. "Urdu optical character recognition systems: Present contributions and future directions." *IEEE Access* 6 (2018): 46019-46046.

[3] Agrawal, Prateek, Deepak Chaudhary, Vishu Madaan, Anatoliy Zabrovskiy, Radu Prodan, Dragi Kimovski, and Christian Timmerer. "Automated bank cheque verification using image processing and deep learning methods." *Multimedia Tools and Applications* 80 (2021): 5319-5350.

[4] Peng, Xujun, Huaigu Cao, Srirangaraj Setlur, Venu Govindaraju, and Prem Natarajan. "Multilingual OCR research and applications: an overview." In *Proceedings of the 4th International Workshop on Multilingual OCR*, pp. 1-8. 2013.

[5] Doucet, Antoine, Gabriella Kazai, Sebastian Colutto, and Günter Mühlberger. "Icdar 2013 competition on book structure extraction." In *2013 12th International Conference on Document Analysis and Recognition*, pp. 1438-1443. IEEE, 2013.

[6] Sabbour, Nazly, and Faisal Shafait. "A segmentation-free approach to Arabic and Urdu OCR." In *Document recognition and retrieval XX*, vol. 8658, pp. 215-226. SPIE, 2013.

[7] Naz, Saeeda, Arif Iqbal Umar, Riaz Ahmed, Muhammad Imran Razzak, Sheikh Faisal Rashid, and Faisal Shafait. "Urdu Nasta'liq text recognition using implicit segmentation based on multi-dimensional long short term memory neural networks." *SpringerPlus* 5 (2016): 1-16.

[8] Ud Din, Israr, Imran Siddiqi, Shehzad Khalid, and Tahir Azam. "Segmentation-free optical character recognition for printed Urdu text." *EURASIP Journal on Image and Video Processing* 2017 (2017): 1-18.

[9] Ali, Toflk, Tauseef Ahmad, and Mohd Imran. "UOCR: A ligature based approach for an Urdu OCR system." In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 388-394. IEEE, 2016.

[10] Uddin, Israr, Imran Siddiqi, and Shehzad Khalid. "A holistic approach for recognition of complete Urdu ligatures using hidden Markov models." In *2017 International Conference on Frontiers of Information Technology (FIT)*, pp. 155-160. IEEE, 2017.

[11] CLE (2015) Urdu digest POS tagged corpus. (*http://www.cle.org.pk/software/localization.html)*

[12] Pal, U., and Anirban Sarkar. "Recognition of printed Urdu script." In *Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings.*, vol. 3, pp. 1183-1183. IEEE Computer Society, 2003.

[13] Hussain, Sarmad, Salman Ali, and Qurat ul Ain Akram. "Nastalique segmentation-based approach for Urdu OCR." *International Journal on Document Analysis and Recognition (IJDAR)* 18, no. 4 (2015): 357-374.

[14] Ul-Hasan, Adnan, Saad Bin Ahmed, Faisal Rashid, Faisal Shafait, and Thomas M. Breuel. "Offline printed Urdu Nastaleeq script recognition with bidirectional LSTM networks." In *2013 12th international conference on document analysis and recognition*, pp. 1061-1065. IEEE, 2013.

[15] Nashwan, Farhan MA, Mohsen AA Rashwan, Hassanin M. Al-Barhamtoshy, Sherif M. Abdou, and Abdullah M. Moussa. "A holistic technique for an Arabic OCR system." *Journal of Imaging* 4, no. 1 (2017): 6.

[16] Yamina, Ouled Jaafri, Mamouni El Mamoun, and Sadouni Kaddour. "Printed Arabic optical character recognition using support vector machine." In *2017 International Conference on Mathematics and Information Technology (ICMIT)*, pp. 134-140. IEEE, 2017.

[17] Osman, Hussein, Karim Zaghw, Mostafa Hazem, and Seifeldin Elsehely. "An efficient language-independent multi-font OCR for Arabic script." *arXiv preprint arXiv:2009.09115* (2020).

[18] Darwish, Saad Mohamed, and Khaled Osama Elzoghaly. "An enhanced offline printed Arabic OCR model based on bio-inspired fuzzy classifier." *IEEE Access* 8 (2020): 117770-117781.

[19] Khattak, Israr Uddin, Imran Siddiqi, Shehzad Khalid, and Chawki Djeddi. "Recognition of Urdu ligatures-a holistic approach." In *2015 13th International conference on document analysis and recognition (ICDAR)*, pp. 71-75. IEEE, 2015

[20] Husnain, Mujtaba, Malik Muhammad Saad Missen, Shahzad Mumtaz, Muhammad Zeeshan Jhanidr, Mickaël Coustaty, Muhammad Muzzamil Luqman, Jean-Marc Ogier, and Gyu Sang Choi. "Recognition of Urdu handwritten characters using convolutional neural network." *Applied Sciences* 9, no. 13 (2019): 2758.

[21] Javed, Sobia T., Sarmad Hussain, Ameera Maqbool, Samia Asloob, Sehrish Jamil, and Huma Moin. "Segmentation free nastalique urdu ocr." *World Academy of Science, Engineering and Technology* 46 (2010): 456-461.

[22] Sagheer, Malik Waqas, Chun Lei He, Nicola Nobile, and Ching Y. Suen. "Holistic Urdu handwritten word recognition using support vector machine." In *2010 20th international conference on pattern recognition*, pp. 1900-1903. IEEE, 2010.

[23] Jang News Paper *https://jang.com.pk/*

[24] Javed, Sobia Tariq, and Sarmad Hussain. "Segmentation based urdu nastalique ocr." In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 18th Iberoamerican Congress, CIARP 2013, Havana, Cuba, November 20-23, 2013, Proceedings, Part II 18*, pp. 41-49. Springer Berlin Heidelberg, 2013.

[25] Hussain, Sarmad, Aneeta Niazi, Umair Anjum, and Faheem Irfan. "Adapting Tesseract for complex scripts: an example for Urdu Nastalique." In *2014 11th IAPR International Workshop on Document Analysis Systems*, pp. 191-195. IEEE, 2014.

[26] Qurat-ul-Ain Akram, Sarmad Hussain, Farah Adeeba, Shafiq ur Rehman, and Mehreen Seed. " Framework for urdu nastalique optical character recognition", 2014.

Machines and Algorithms
http://www.knovell.org/mna

Research Article

A Framework for Analysis of SNPs in TAGAP Gene

Haider Usman[1], Qaisar Rasool[1,*], Saad Rasool[1] and Hifssa Aslam[1]

[1]Department of Computer Science, Bahauddin Zakariya University, Multan, 60000, Pakistan

[*]Corresponding Author: Qaisar Rasool Email: qrasool@bzu.edu.pk

Received: 22 November 2023; Revised: 6 December, 2023; Accepted: 20 December 2023; Published: 14 March 2024

AID: 003-01-000033

Abstract: The role of TAGAP in the T cells plays a vital role and the analysis of how the structure and functional impact SNPs have on the TAGAP is the main focus that has been analyzed within the provided studies. Analysis of the genetic maps that are developed within the human body has been analyzed with testing the impact of the mutation on the TAPAG in T cells. The examination of non-synonymous Single Nucleotide Polymorphisms (nsSNPs) on the TAGAP gene functionality and structural characteristics. Four tests have been performed are SIFT, Polyphen-2, PROVEAN, and the stability of the protein was tested through the I-Mutants 3.0. Diabetes Mellitus and Auditory Agnosia are the two diseases that have been analyzed to depict the importance of the TAGAP in the T cells.

Keywords: TAGAP; T cells; Single Nucleotide; Polymorphisms (SNPs); Non-synonymous SNPs (nsSNPs); Protein structure; Protein functionality;

1. Introduction

The majority of autoimmune and infectious diseases are linked to the locus syntenic to the TAGAP genes. Some of these conditions include multiple sclerosis and candidemia, which are related to infections and the immune system. Unlike most studies, very little is known about the role of TAGAP beyond its function in T cells. SNP (Single Nucleotide Polymorphism) refers to the sequence or combination of DNA that arises from the substitution of a single nucleotide. After the change, the sequence of combined DNA is referred to as SNP. This data consists of sequences of four nucleotide bases: Adenine (A), Cytosine (C), Guanine (G), and Thymine (T). Simply put, SNP is the most common method used for gene transmutation. According to research, most humans share this mutation, and having a single SNP is common within the human population [1]. Once a study is completed and the association between genetic variations and their phenotypes is understood, the resulting data can help us understand how various diseases are connected to these mutations. These mutations can occur in the coding or non-coding regions of genes. SNPs located in the coding region of genes do not alter the amino acid sequence of newly formed proteins. There are two kinds of SNPs in the coding region. One type, known as synonymous SNPs, has no impact on the shape or sequence of amino acids. The other type, nonsynonymous SNPs, affects the amino acids and can potentially alter the protein sequence. Nonsynonymous SNPs are further divided into nonsense and missense SNPs. These types of SNPs modify the amino acids and can disrupt the function of proteins, causing potential harm [2].

Numerous SNPs occur naturally within the human body, leading to mutations and the development of different genetic maps. This results in various diseases that respond differently to drugs. It is clear that diseases are directly related to these mutations. In other cases, most SNPs are neutral and do not affect

protein function. Therefore, understanding the role of SNPs is essential for unraveling the complex traits and diseases present in humans [3].

This study aims to achieve the following objectives:

- To perform a computational analysis of SNPs associated with TAGAP, focusing on their occurrence and role in the TAGAP locus.
- To investigate how interactions between SNPs and TAGAP predict patient behavior and aid in diagnosing autoimmune, infectious, or mutation-related diseases.
- To evaluate the structural and functional impact of deleterious SNPs on TAGAP proteins, focusing on their role in disease mechanisms.
- To identify functional SNPs with significant relevance that can be utilized for predicting, diagnosing, or treating fungal infections, diabetes, or other mutation-related conditions.

The research hypotheses are as follows:

- SNPs associated with TAGAP significantly influence susceptibility to autoimmune and infectious diseases, such as multiple sclerosis and candidemia.
- Deleterious SNPs in TAGAP alter protein structure and function, contributing to disease pathogenesis.
- Functional SNPs in TAGAP can be computationally identified and utilized to develop predictive models for patient behavior and disease diagnosis.
- Structural changes in TAGAP proteins due to nonsynonymous SNPs are correlated with their impact on immune system functionality and the development of genetic disorders.

In this section, we discuss a brief introduction to the paper, including the study's aim and hypothesis. The second section is a follow-up with the literature review. The methods used to compile the results are detailed in Section 3. The study's findings and results are presented in the fourth part. Section five contains the discussion, while all sections end the paper with conclusions.

2. Literature Review:

A peer-reviewed work by Anne Ndungu et. all [4] that presents multi-SNP models for gene expression analysis. Article describes gene-level investigation of SNP prediction models using TWAS. The research report analyses 43 human tissues using GTEx (Genotype-Tissue Expression Project) multi-SNP gene models and an iterative modeling scheme. Compared to multi-SNP analysis, cis-eQLT variance is lower. About 90% of the 826 gene metabolite pairings analyzed in the paper were dominant of the single eQTL. The framework described in the research was important since more than 90% of colonization signals were created within QTLs in the multi-SNP model. The research found that 8% of TWAS were linked to the causative genes. The results show considerable gaps in the research, and more legitimate SNP analysis employing multi genes is needed.

GWAS (huge genome-wide association studies) have enhanced RA genetic risk factor knowledge. The genetic susceptibility of groups with high Amerindian admixture is unknown. The research aimed to assess prior RA locus reports' generalizability with admixed ancestry in Latin America. In linkage equilibrium, researchers selected about 128 SNPs (single nucleotide polymorphisms) with high RA association in non-Amerindian groups. To genotype roughly 118 SNPs in 313RA participants/487 healthy controls, mid-density polymerase chain reaction was used. The second cohort (250 cases/290 controls) confirmed the connection. The SNP rs2451258 was shown to be linked to RA and 18 additional markers were suggestive. It was upstream of the TAGAP. Haplotype testing showed a strong correlation between neighbouring SNPs and RA near the single transcription activator and transducer4. The study found no replication of earlier genetic link reports to RA. These data suggest that LA population admixture mapping and GWAS can identify novel loci associated with RA. This study can help clinicians and researchers comprehend this disease's pathogen omics. Research on trans-population disparities in RA is also possible [5].

Pehliva et. all [6] studied paediatric TAGAP polymorphism gene patients. There are a number of hypotheses about how the rs1738074 T/C SNP affects TAGAP. No study had focused on Turkish paediatric

patients. The researchers wanted to examine the connection between celiac disease and diabetes mellitus type 1 in Turkish paediatric patients with TAGAP gene SNPs. Researchers used IBM SPSS. This study included 127 paediatric patients and 100 healthy youngsters. We found the polymorphism using an allele-specific polymerase chain reaction. The researchers used IBM SPSS, Arlequin 3.5.2, and Statistics 25.0 for statistical analysis. Researchers may not be biased. The data also showed that 72% of 154 CD patients had C alleles. In addition, 28%, or 60 CD patients, carried the T allele. We also found the C and T alleles in 43.5% and 57.5% of patients with celiac disease and type 1 diabetes, respectively. The control group comprised 67% C alleles and 33% T alleles. The results also showed no significant difference in allele frequencies and genotype between the control group and the patient. The investigation found no significant link between the polymorphism and disease risk.

Rheumatoid arthritis, the most widespread, persistent, and progressive inflammatory illness, damages joints and increases mortality. The C-C motif ligand 21 is a cytokine that is involved in immunological regulation and inflammation. As a result, SNP analysis is the most important in the CCL21 gene because it helps evaluate their functional and structural relevance in finding potential treatment targets for immune-related illnesses like RA. This work identified the most harmful non-synonymous SNPs that affect CCL21 protein function and structure using in silico methods. The main roles in this research may include SNPs&GO, PROVEAN, PolyPhen2, and SIFT. We validated the functional and structural effects, stability, and conservation profile of the other tools using MutPred, I-Mutant, and ConSurf. The results revealed a post-translational modification site. The research is also important for identifying and analyzing human functional SNPs and TAGAP proteins. Chimera v1.11 proposes 3-D protein modeling with I-TASSER. The findings also suggest that non-synonymous SNPs may cause CCL2 protein dysfunction and autoimmune disorders like RA. According to the research, these non-synonymous SNPs may be the most important in examining the CCL21 gene's link to autoimmune disorders like Crohn's Disease (CD) and RA. To determine their suitability for genome editing and pharmacogenomics, these SNPs must be tested in animal studies and tissue samples from diseases [7].

SNPs in the CCR6 gene may cause Lupus nephritis, systemic sclerosis, rheumatoid arthritis, psoriasis, and other autoimmune illnesses. Functional and structural identifications are important polymorphisms for therapeutic and dysfunctional target research. Bioinformatics methods have identified damaged nsSNPs that affect CCR6 function and structure. They utilized PolyPhen2, SNP&GO, SIFT, and PROVEAN to model proteins in 3D in a computer, and Gene MANIA and STRING to guess how genes would interact with each other. The three nsSNPs rs751102128, rs1185426631, and rs1376162684 do the most damage to the CCR6 gene. On the other hand, the seven missense rs139697820, rs1282264186, rs768420505, rs1263402382, rs139697820, rs769360638, and rs1438637216 go back to stop codons. Because of its probable phosphorylation location, rs1376162684's highlighted post-transcriptional alteration is feasible. Gene-gene interactions have demonstrated CCR6's role in several co-expressions and pathways. After that, we can use these ten nsSNPs to study disorders related to CCR6 [8].

SNPs can help DNA develop and repair, according to another study. This study demonstrates that mitochondrial dysfunction repairs DNA and helps it fight the body. These dysfunctions strengthen DNA with SNPs and reactive oxygen and nitrogen species. DNA is composed of SNPs, reactive oxygen, and nitrogen. All of these substances help DNA resist genetic mutations. When its ability increases, DNA will attempt to repair itself and stop mutations. When DNA does not change its amino acid pattern when it becomes too powerful to battle mutations. This increases immunity, DNA strength, and the body's ability to fight cancer. Thus, SNPs and mitochondrial dysfunction strengthen the body, immune system, and DNA repair [9].

3. Methodology

3.1. Collection and compilation of the Database

We retrieved the T cell protein sequence for TAGAP from the protein database website (www.uniprot.org/UniProt/Q8N103). Obtain nsSNP retrieval information for experimental data analysis

from the database.We utilized the National Center for Biotechnology Information (NCBI) at ncbi.nlm.nih.gov/SNP. The NCBI and Entrez retrieval systems have utilized it. The experiment may utilize a variety of computing algorithms. Ensure that dbSNP errors won't affect study outcomes [10]. This classifies nsSNPs as tolerant and neutral using the computing algorithm. Protein stability is crucial for experimental findings, and its stability may be determined via computer data analysis.

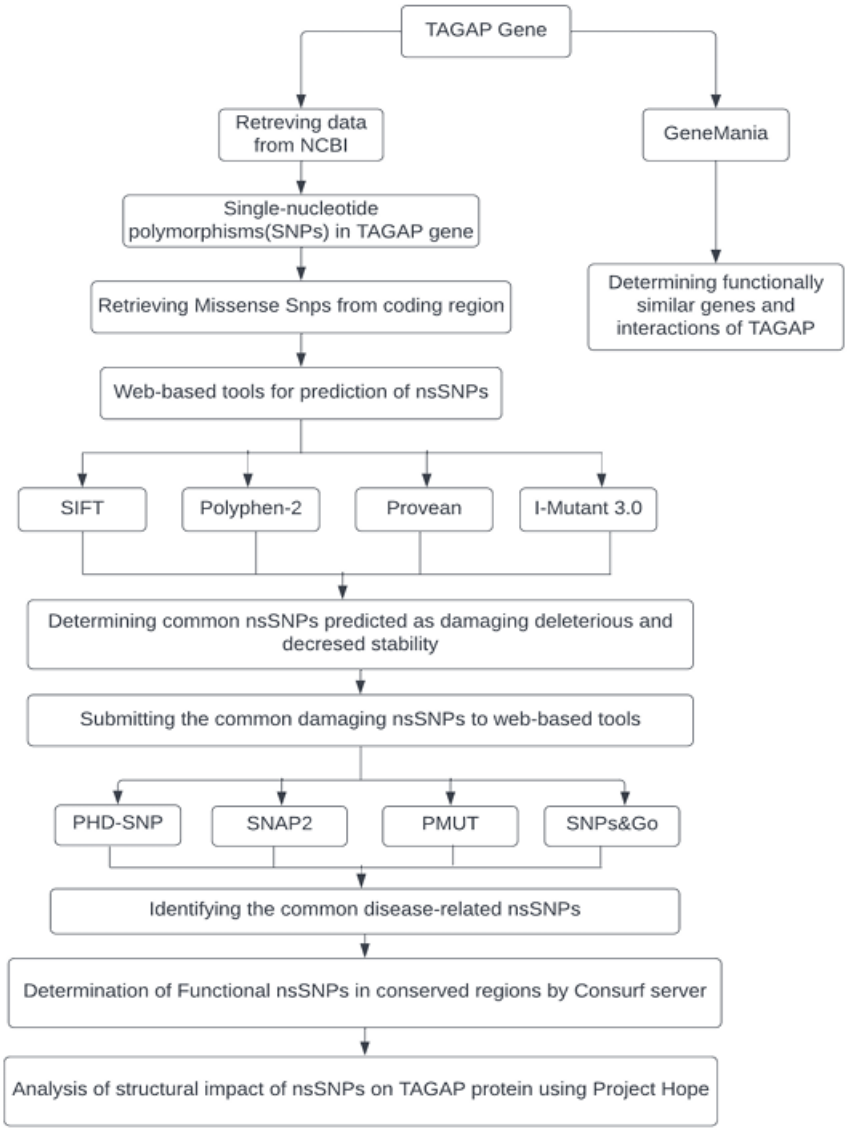


Figure 1: Methodology of the TAGAP Gene Analysis

3.2. Determining the Functional nsSNPs

There were eight distinct types of prediction tools tested to demonstrate the computational usefulness of nsSNPs. First, four prediction techniques were used to show functional nsSNPs, including the following.

- I Mutant (http://gpcr2.biocomp.unibo.it/cgi/predictors/I-Mutant3.0/)
- SIFT(https://sift.bii.a-star.edu.sg/)
- Polyphen-2 (http://genetics.bwh.harvard.edu/pph2/)
- ROVEAN (http://provean.jcvi.org/index.php)

After analysing nsSNPs that may harm the protein, PhD-SNP (https://snps.biofold.org/phd-snp/phd-snp.html) was used. The last one is SNPs&GO (https://snps.biofold.org/snpsand-go/snps-and-go.html). These simple tools will assist analyze disease and neutral SNPs. Further details on all computational tools are provided above in figure 1.

3.2.1. SIFT

Using Sorting Tolerant from Tolerant (SIFT) server, study reveals harmful coding non-synonyms in SNP. SIFT is mostly used to identify the link between phenotypic variation and mutations [11]. This involves assessing the influence of amino acids on protein and testing their capabilities. The SIFT analyses protein families based on their ability to preserve amino acids, a pre-determined criterion. Damage can be assessed by observing changes in well-conserved places.

The prediction software-defined will get a query first. Co-ordinates will be created from nsSNPs in prediction software. SIFT works by analysing given queries and applying alignment information in numerous ways. The program scores 3.0 for depicting protein sequences based on the MSCS (Median Sequence Conservation Score). It is easier to determine if nsSNPs are harmful or tolerated. The SIFT performance involves several processes, including:

- First, select a comparable sequence.
- Next, choose a similar sequence based on functionality.
- These sequences are analysed to find several alignments.
- Each location's probabilities are compared to probable substitutions.

Next, select a cut-off value to determine if a replacement is harmful or accepted at a certain spot. If the probability cut-off value exceeds the feasible possibilities, the program declares it detrimental for further analysis. A cut-off value of TI > 0.05 is defined for specific algorithms.

3.2.2. PROVEAN

Protein Variation Impact The PROVEAN analyzer tool predicts and determines the influence of amino acid substitutions on protein functioning inside the system software [12]. The initial stage was retrieving nsSNP information from the NCBI database. Clustering and supporting sets will be used to eliminate redundancy. The PROVEAN requires average delta storage computation as the second step. Effective output was achieved by clustering 75% of globally recognized BLAST proteins using captured sequences. Next, a succession of supporting sets will be created from the top clusters in the experimental study. Research indicates that variations in protein similarity can harm the studied protein. Changes in the delta score are evaluated to see if the protein's functioning has changed, aiding in experimental data collecting. A low delta score for a protein modification might have a greater detrimental influence on its functioning. A score of less than 2.5 indicates higher damage to protein functioning, whereas scores above 2.5 indicate neutrality. The purpose of PROVEAN is to collect nsSNPs with scores below 2.5 for future experimentation.

3.2.3. Polyphen-2

The Polyphen-2 test is included to demonstrate how substituting an amino acid affects the protein's functioning and structure. This assay analyses protein sequence alignments from several sources and describes a three-dimensional structure [13]. The Bayesian classifier was used to examine the impact of mutations. Two query types are used in the Polyphen-2 test: gene ID and protein sequences. The Uniport is used to get information, including amino acid replacement. The protein sequence was used in FASTA format. The substitute score will be categorized as 30 through server prediction. The position-specific independence count score (PSIC) is a projected score classification with a value between 0 and 1. Increased PSIC scores indicate more impact on protein analysis through amino acid replacement, and vice versa. Scores vary from 0 to 1 and are divided into three types: Potentially dangerous, perhaps destructive, and benign are categorized based on the score.

3.2.4. PhD-SNP

A similar technique has been used to illustrate how amino acid substitutions might change protein functioning and potentially cause illnesses. Various forms of support vector machines are utilized for PhD SNP testing [14]. The protein sequence was obtained by testing to illustrate the study report results. Retrieving Swiss-PROT codes from the NCBI database and obtaining the protein sequence in FASTA format was crucial. Mutation levels were recorded in the analyzed output table of protein sequences. The output table displays both novel and Wild-type amino acid sets. Two types of nsSNPs exist: neutral and disease-related. A crucial step is to use PhD-SNP to distinguish disease-related nsSNPs from benign ones.

3.2.5. I-Mutant 3.0

I-Mutant 3.0 analyzes protein alterations that may affect stability. Gibbs' free energy change process helps analyze protein heat capacity variations, including temperature transitions [15]. Alterations to protein stability can be identified using I-Mutant 3.0. The format sequence will be retrieved from Uniport to achieve the appropriate format (FASTA). The data input includes the new residue of correlated values and the mutation site to determine the change in free energy, known as Delta Delta G (a measure for forecasting the impact of a single point mutation on protein stability). There are three prediction categories based on DDG value: DDG values below -0.5 kcal/mol indicate severe instability, while values above 0.5 kcal/mol indicate severe stability. The neutral value is shown as 0.5kcal/mol DDG.

3.2.6. SNAP2

A useful tool for analysis is SNAP2, which assigns amino acid substitutions based on their impact and neutrality categories [16]. The impact of SNPs on protein function will be studied utilizing biophysical properties, including structural and evolutionary information. The sequences in the study report will be shown in the same FASTA format as when downloaded from Uniport. SNAP2 employs a neutral network method. The TAPAG gene sequence should be entered into SNAP2 to classify amino acid substitutions and analyze their influence on protein features like neutrality or effect. The Reliability Index RI shows the value from +100 to -100. An analysis will be based on the score, with a strong neutral value of -100 and a strong influence on protein characteristics and functionality at +100.

3.2.7. SNPs&GO

This tool determines the disease-nsSNP connection. Additionally, the method helps determine disease or neutral SNP effects on protein [17]. SVM is the tool's main substitution classification technique. The three-dimensional protein structure and protein sequence profile from the tool that explains the protein's functions are used to retrieve the information. A stringently specified trained and tested approach is used to examine cross validation utilizing a prediction tool. Provide input in FASTA format to determine and inspect output. If the likelihood 32 score exceeds 0.5 in this test, the system evaluates the disease-related impact.

3.2.8. PMut

This free tool aids analysis. The PMut mechanism operates on two levels, obtaining information from local databases, mostly as mutational hotspots. Protein SNPs with specified identities are evaluated in the second step of PMut analysis. The study report requires extensive mutational effort to portray the possible mutations. This tool shows the association between protein illness and amino acid substitution type, as well as the impact of amino acid replacement on protein neutrality. Mendelian mutational analysis is performed using the technique shown in the paper.

3.3. Determination of Functional SNPs that are based on Conserved Region

The test relies on conserved amino acids for experiment evolution. After collecting SNPs from the computational tool, the conserved evolutionary amino acids in the TAPAG gene will be examined to provide the research findings. Next, the TAPAG discussion score will be analysed and shown as study results. The ConSurf server will analyse harmful SNPs for the study report.

3.3.1. ConSurf Server

Certain amino acids include a protein, and particular nucleic acids in RNA or DNA must balance the inherent inclination of the protein to mutate [18]. Macromolecules must also keep their structural and informational identities. The ConSurf service will analyze evolutionarily present amino acids in RNA, DNA, and proteins. The ConSurf Server is used to extract phylogenetic relationships between homologous substance sequences. A unique phylogenetic tree was created using the ConSurf Server, which also analyzes evolutionary conservation. When the score ranges from 1 to 4, factors are named, intermediate scores are depicted, and conserved scores are recorded (7 to 9). The Bayesian calculation approach is used in research report analysis. The ConSurf Server defines protein sequences in FASTA format for prediction as per analysis. A tabular structure helps define the overall conservation score, which is displayed through a color scheme. Protein features linked to functioning and structural complexity are predicted and assessed. To analyze the experimental technique and acquire data, highly conserved amino acids will be chosen.

3.4. Examination of the impact of nsSNPs on protein structural properties

The most important step to follow is an analysis of the impact and effect of mutations of protein structural properties that have been depicted through the deleterious nsSNPs.

3.4.1. Project Hope Utilization

The aim is to compare the wild-type amino acids used in the study report to the mutant amino acids in real time. The program gathers several sources of information to create a three-dimensional picture of protein [19]. This data includes sequence annotations from the study paper. This software program was used to calculate 3D coordinate prediction and services like DAS in the research project. The research uses a specific technique to analyze and illustrate residual traits, whether they are native or new. nsSNPs identified in earlier phases and filtered to meet study objectives are evaluated using this tool. Variants based on various protein sequences have been submitted to Hope software [20]. After submitting the response, the difference between wild-type amino acids and novel residue amino acids was examined. Two approaches have been devised to compare the size and hydrophobicity of new and ancient residues. The difference analysis helps determine the impact of the mutation on protein functioning and structural features.

3.5. Analysis of the TAGAP Gene Interactions

GeneMANIA's database helps display the organization's gene base characteristics. The research project used functional association data for experimentation. Genetic relationship including protein co-localization and co-expression with a comparable pathway. The research study analyzes gene expression by co-expression [21].

Table 1: TAGAP and its functionally similar genes

Gene	Description	Rank
TAGAP	T cell activation RhoGTPase activating protein	N/A
RHOH	ras homolog family member H	1
RHOV	ras homolog family member V	2
RHOT1	ras homolog family member T1	3

RHOF	ras homolog family member F	4
RHOU	ras homolog family member U	5
RHOBTB1	Rho related BTB domain containing 1	6
RHOBTB2	Rho related BTB domain containing 2	7
RHOT2	as homolog family member T2	8
CD69	CD69 molecule	9
RHOD	as homolog family member D	10
RHOJ	as homolog family member J	11
RAC2	Rac family small GTPase 2	12
RHOB	as homolog family member B	13
RHOC	as homolog family member C	14
RHOG	as homolog family member G	15
RHOQ	as homolog family member Q	16
RAC3	Rac family small GTPase 3	17
ARHGEF3	Rho guanine nucleotide exchange factor 3	18
PLEK	pleckstrin	19
DENND1C	DENN domain containing 1C	20

Their expression must be considered to perform the analysis with implementing and subjecting to different situations. A genetic expression is a tool that helps in the prediction of one gene and its functionality that has been associated with another type of gene that has been analyzed in the research report. The perturbation effect of one gene to another gene depict about the functionality that has been associated with one another according to the structure and functionality of the gene that has been analyzed.

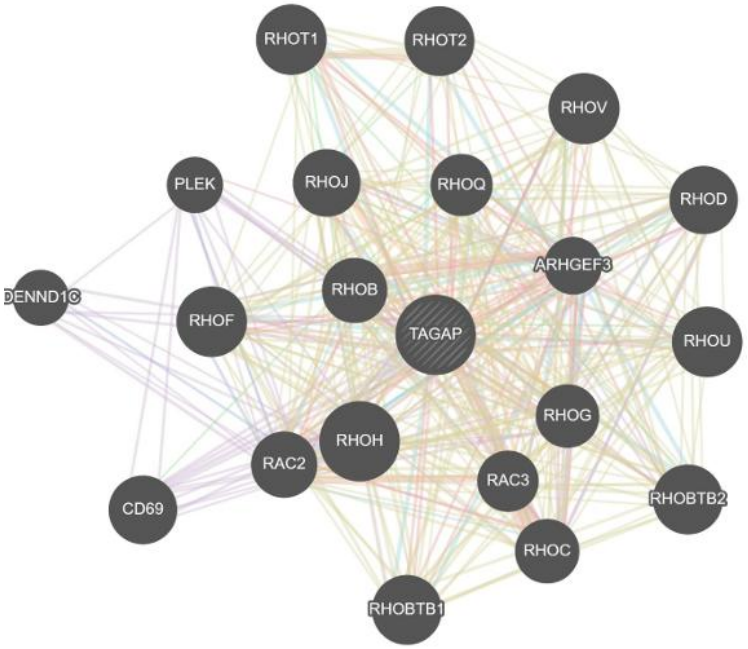


Figure 2: Gene-MANIA Interaction network of TAGAP gene

Studying gene products that are linked may suggest the existence of a related gene. The analytical approach will follow the following procedure to describe the research investigation.

Initially, data will be retrieved from the NCBI website to illustrate experimental data. SNP analysis in the TAGAP gene and missense Snaps from the coding area are required. Four useful online tools for analysis include SIFT, Polyphen-2 PROVEAN, and I-Mutant 3.0. Submit Common harmful nsSNPs to PhD-SNP, PMut, and SNPs&GO tools. The examination of nsSNPs in prevalent diseases led to the creation of the ConSurf Server and Hope Project analysis.

4. Results

4.1. SNPs dataset

Research indicates that database SNP effectively captures SNP interest due to its huge dataset. Research indicates over 100 million SNPs exist globally, with unique or common variances across individuals. These DNA alterations are usually found across genes.

As biological markers, they can help researchers find disease-linked genes. SNPs in a genotype or gene's promoter regions can directly affect sickness by altering gene activity. According to the chart below, the TAGAP gene has 17% nsSNPs, 9% 3'UTR SNPs, 8% 5'UTR SNPs, and 66% other SNPs [22].

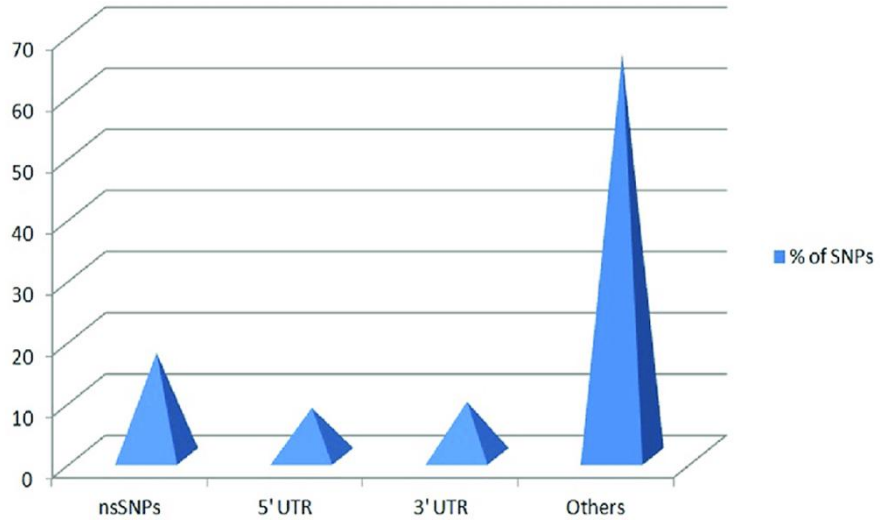


Figure 3: The pyramid in the form of the cluster is showing the SNPs percentage in TAGAP Gene

4.2. Sift, Polyphen-2, PROVEAN, I-Mutant 3.0 results

This study examined the functional effects of nsSNPs using SIFT, PROVEAN, and PolyPhen2. The I-Mutant was used to study the impact of nsSNPs on protein stability. The SIFT results indicate that nsSNPs have an intolerant scoring tolerance index. In Proven, mutations are often considered harmful if the final score is below -2.5 or if nsSNPs may be harmed. Polyphen-2 identifies potentially harmful, likely detrimental, and perhaps benign non-synonymous SNPs. Benign nsSNPs and destructive possibilities are the most accurate predictions compared to the other two. Most estimates are based on the independent count score and position-specific difference, with score 1 being the most detrimental. The research highlights nsSNPs that are prevalent in up to four algorithmic techniques and have a substantial score of zero in SIFT and PolyPhen-2. This is done to emphasize only very harmful SNPs.

TAGAP protein is significant as it is its sole functional domain. Therefore, these nsSNPs may be considered as potential causes of TAGAP dysfunction illnesses, facilitating treatment discovery and development [23].

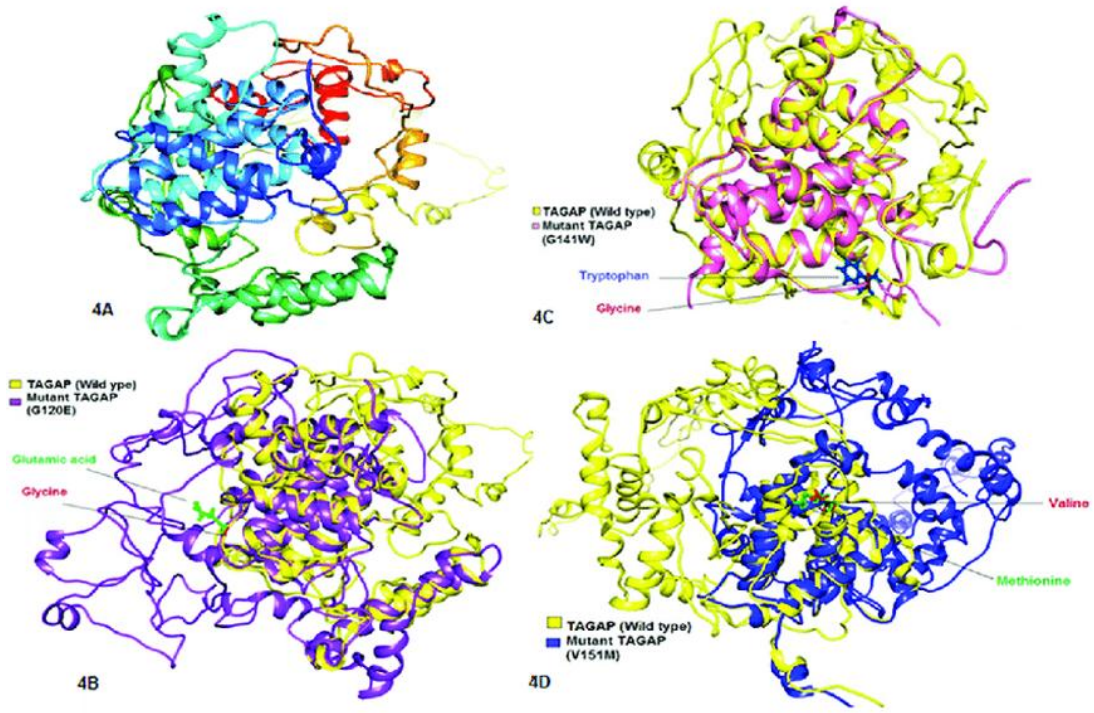


Figure 4: TAGAP structure of a protein with its mutant form [23]

4.3. PhD-SNP, SNAP2, PMut, SNPs&Go results

This study examined the functional effects of nsSNPs using PMut, SNAP2, PhD-SNP, and SNPs&GO. The study utilized several computational methods to identify non-synonymous SNPs that are susceptible to TAGAP protein structure and function, potentially leading to harmful disorders. The computational analysis was conducted using several tools, including Provean, SIFT, PolyPhen-2, PMut, SNPS&GO, and PhD-SNP. In addition, SNAP2, a trained neural network-based tool, uses many criteria to differentiate between sickness and benign alterations. Appreciable accuracy. We utilized the protein sequence as input and received two predictions: effect (high score) or neutral (low score) (negative score). Alternatively, SNPs&GO predicts TAGAP gene mutations using a protein sequence method. A likelihood score above 0.5 implies a disordered impact of the mutation on host protein interactions. PhD-SNP, a tool in SNPs&GO, evaluates data for harmful or neutral mutations. Additionally, PMut studies reveal that neural network intelligence yields 80% correct discoveries of each SNP's compulsive features. It's feasible to predict neutral or disorderly output [24].

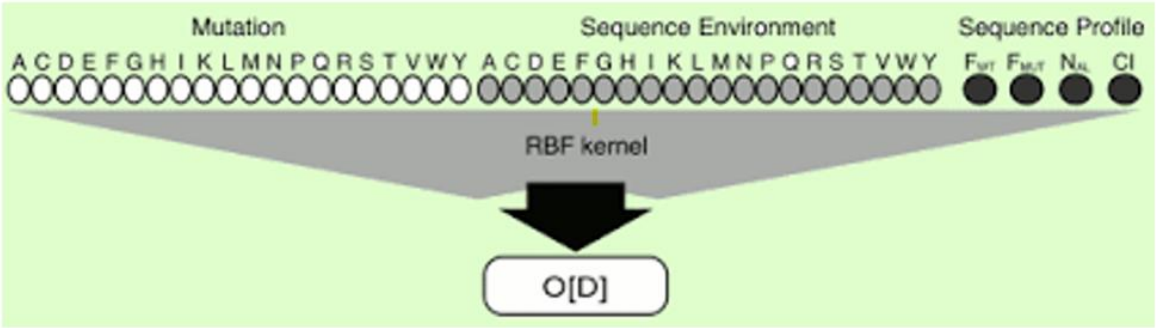


Figure 5: PhD-SNP [24]

4.4. ConSurf results

ConSurf analyzes past acylated acid rates and applies them to the search macromolecule's structure and sequencing. The ConSurf investigation can identify crucial places inside the search macromolecule, as the query interface's slowly shifting sections are crucial for operation. When the search macromolecule's infrastructure is in place, it can distinguish between slowly developing core sites, crucial for structural stabilization, and clusters of slowly changing surface locations, crucial for function [25].

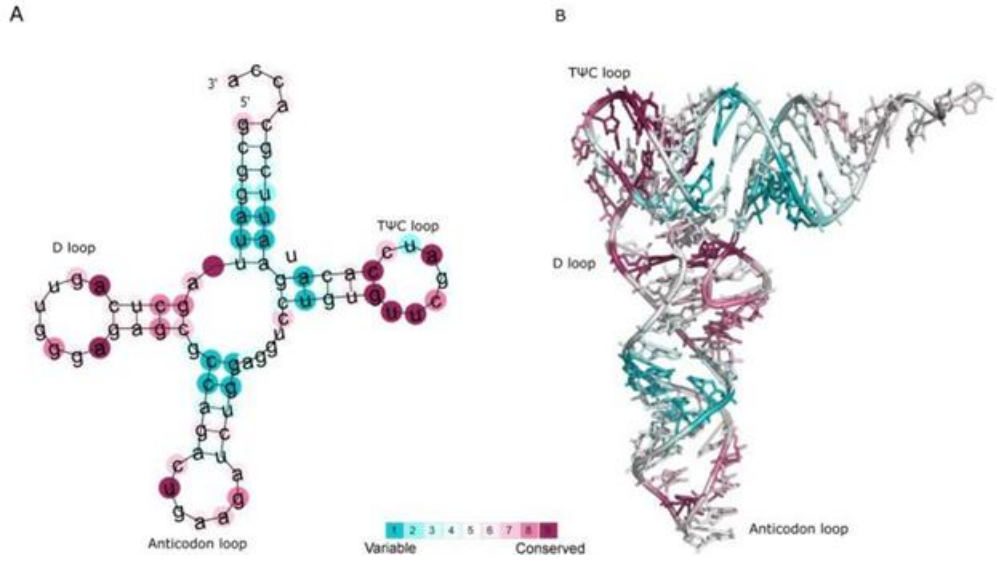


Figure 6: ConSurf Analysis

The ConSurf test in the research report showed the mutation's overall impact. Using the ConSurf service, amino acids evolutionary conversation was examined against 9 of the most damaging nsSNPs in TAGAP protein residues. The ConSurf server findings are shown using the TAGAP protein structure. Using solvent accessibility data, the ConSurf service has produced predictions about functional and structural residues. Based on conserved residue position, two alternatives are shown.

It may be on the protein's inner core or surface as defined by ConSurf. The ConSurf service analyzes the amino acid functionality of proteins to show that they are more conserved than other proteins. The more often nsSNPs are on the conserved reign, the more they damage protein functioning, according to ConSurf server data. Different findings from the ConSurf server show the TAGAP conservation level and structural and functional properties to be studied. The ConSurf service found TAGAP-exposed functional residues G141, E136, T118, N205, V151, and G141 to be highly conserved. The ConSurf server provides a p-value of 0.618 for the T118M mutation. Mutations L100F and F122L had p values of 0.573 and 0.846, respectively. Mutation G120E is 0.902 and N205S is 0.896. G141W and V151W had p-values of 0.663 and 0.676, respectively. The A126T p-value was 0.804 and E136K 0.498.

The TAGAP domain's nsSNP ID assessment findings. Rs748659041 contains amino acid change 100, L to F, and protein stability has deteriorated with RI value 9 and DDG score -0.62, including nsSNP TM Score 1. The given nsSNPs have RM SD 0. For rs368265576, the amino change is 118 T to M, and the results show lower stability as the above nsSNPs. The RI is 6 and the DDG score for that ID is -0.29. TM score is -0.29 and RMSD 0.83. nsSNP study for the ID rs764717611 showed a shift of amino acid 120 G to E, which decreased protein stability with a RI value of just 2 (ConSurf server data). The DDG score for the nsSNP ID is -0.84, with a TM score of 0.78894 and RMSD 1.98.

The stability of rs763380333 must be lowered using a ConSurf web server RI value of 3 to calculate the amino acid change from 122 F to L. The DDG value is -0.6, with a TM score of 0.98778 and RMSD of 0.83. For nsSNPs ID rs780953936, the amino acid change is 126 A to T, indicating a complete loss in stability and RI of 7 due to the ConSurf server. The DDG value is -1.04, with a TM score of 0.7912 and an

RMSD of 1.87. The stability of nsSNP ID rs866898464 is lowered, and the amino acid change is 136 E to K and score of 7 in RI. The ConSurf web server findings for rs866898464 show a DDG score of -1.12 and a Tm score of 0.98778. In ConSurf server data, an amino acid altered from 141 G to W has an ID of rs765146154, indicating low stability and a drop from its original value. The DDG is -0.58 and the TM score of the supplied nsSNPs ID is 0.78894. RMDB scored 1.98. The second final ID assessed by the ConSurf web server, rs777042268, showed decreased instability with 151 V to M. Due to the loss in stability, the total value score is RI 9 and DDG with a value score of -1.53. The RMSD score for the nsSNP ID is 1.99. The last ID evaluated with the ConSurf server is rs778438807, which shows decreased stability and a value of change in the amino acid of 205 N to S. Its RI is 2 with a DDG score of -2.45, TM Score of 0.78851, and RMSD score of 1.99.

Figure 7: Pictorial Representation of ConSurf Result of TAGAP (a)

ConSurf anticipated the amino acid-based TAGAP conversation profile. The highly conserved nsSNP ID rs748659041 with a residual and position of L100 has a conversation score of 8. The conversation score for nsSNP ID rs368265576 is 9, indicating good exposure and conservation. For rs764717611, G120 has a conservation value of 9, while for nsSNP ID analysis, F122 has a conservation score of 9. Position rs780953963 of A126 is highly exposed and has a conservation score of 9.

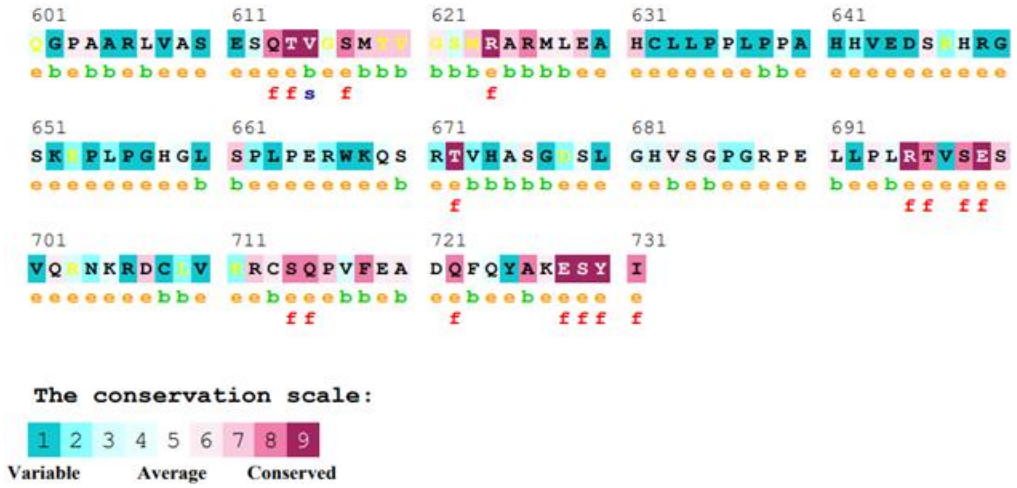


Figure 8: Pictorial Representation of ConSurf Result of TAGAP (b)

Analysis of nsSNP ID rs866898464 E136 has a conservation score of 9 indicating high exposure. The highly exposed protein G141, with nsSNP ID rs777042268 and conservation 8, was identified by ConSurf web analysis. The rs778438807 for V151 is well conserved, with a conservation score of 9 revealed. The N205 location (rs765146154) identified as highly exposed and conserved.

4.5. Project Hope Results

The HOPE results reveal that genetic polymorphism in the human genome is primarily based on SNPs, which are primarily composed of single base pair alternations in alleles. These alternations are the most common and significant type of variation in DNA sequences. The study indicates that non-synonymous SNPs cause mutations that lead to genetic diseases. Non-synonymous SNPs in the TAGAP gene are linked to severe illnesses due to their negative impact on protein structure. Results from SNP analysis are as follows. With project hope, protein SNP mutations have been explored. Various gene and protein mutations are associated to methionine synthase. The STRING database was used to examine these predictions. Further details on these mutations.

4.5.1. rs57752780 (V744L)

Mutation of amino acid to SNP V744L. Different angles have been used to see the residue where the mutation occurs. In the picture, H bonding is used for mutation. An angle of 744 on the left is employed. Mutations will occur at this angle, with SNPs present on the local bonding. All these interactions cause genetic mutation. Mutation changes the sequence of amino acids in DNA and SNPs, affecting their function.

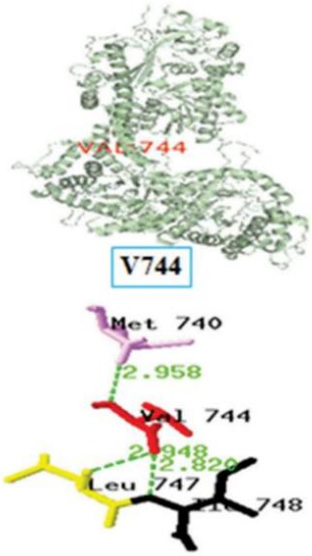


Figure 9: rs57752780 (V744L)

4.5.2. rs57752780 (L744L)

In this structure, gene mutations occur at angle 744 on the right, altering the genetic coding of DNA. Mutations can be harmful and affect the DNA sequence. These mutations also occur with DNA SNPs. This alters DNA sequence and creates a mutant gene.

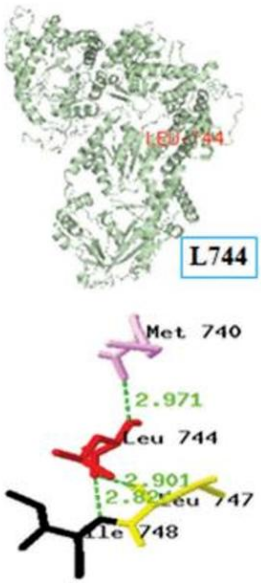


Figure 10: rs57752780 (L744L)

4.5.3. rs113914406 (G682D)

Mutations occur in amino acids rs113914406 at the angle of 682 on the left side of DNA due to hydrogen bonding. Hydrogen bonding is essential for SNP introduction and mutations. These mutations are harmful and cause genetic changes that impact human lives. Avoid this form of mutation to prevent potentially hazardous illnesses.

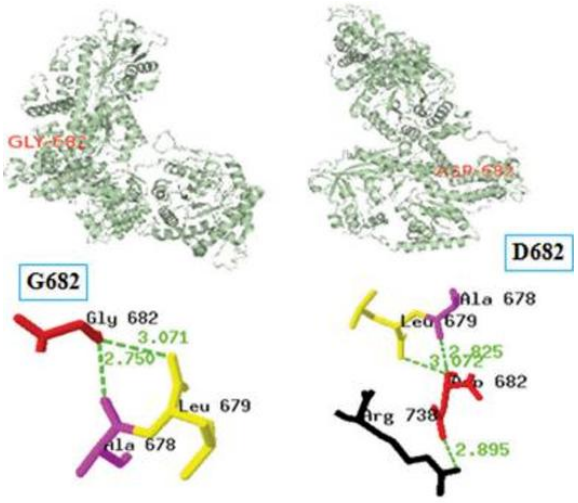


Figure 11: rs113914406 (G682D)

4.5.4. rs74710714 (V776E)

These SNP configurations enable mutation of amino acids rs74710714 on the left side of SPDBV at an angle of 766. Genetic mutation is promoted by gene interactions with GLY 828 and lle 826. Mutations can impact DNA function and the immune system. This form of mutation should be avoided. Mutations impair DNA repair. Mutations modify DNA structure, function, and ability by altering sequences via multiple mutations.

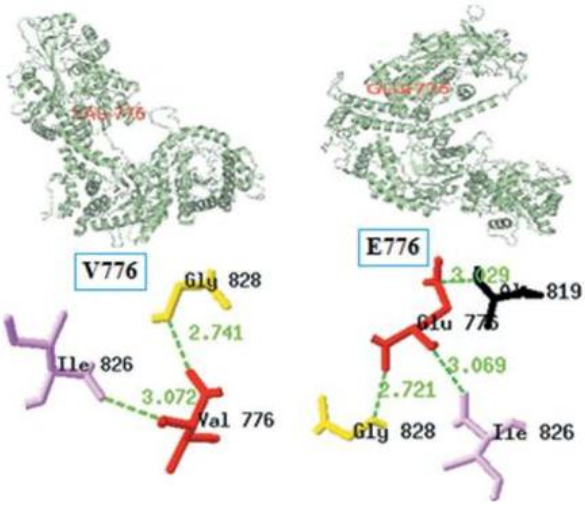


Figure 12: rs74710714 (V776E)

4.5.5. rs116836001 (R1027W)

A mutation at the right side of DNA at the angle of 1027 alters the mutation. This mutation occurs because to hydrogen bonding and SNP present. We know there are two SNP kinds. One form of SNP, synonymous, contributes to mutation. These mutations are harmful and increase the risk of cancer in humans. This sort of genetic mutation weakens the immune system, making the body more susceptible to many illnesses. Mutations can cause the death of many human cells, including white blood cells, red blood cells, and impulsive neurons in the central nervous system.

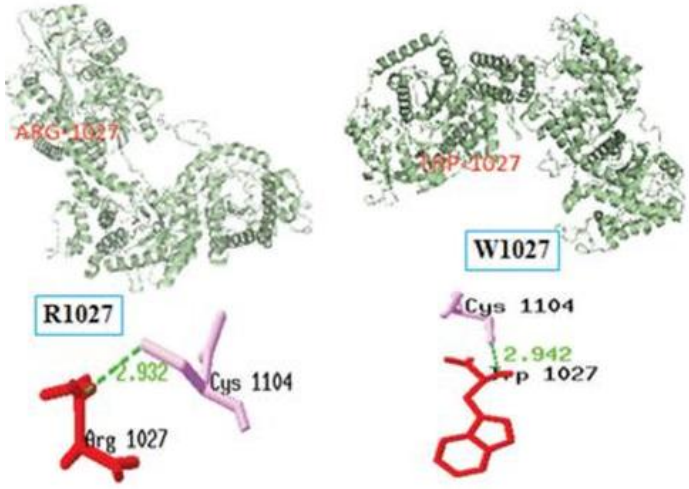


Figure 13: rs116836001 (R1027W)

5. Discussion:

The database analysis identified multiple SNPs, highlighting the challenges in determining which SNPs significantly influence protein structure and functionality. This study emphasizes the importance of distinguishing between coding and non-coding SNPs to better understand their effects on protein function, aligning with findings from Lim et al. (2021), who noted the critical role of SNP analysis in addressing genetic illnesses and improving cancer treatments. The TAGAP gene is responsible for the regulation of the immune system. Being closely related to various diseases like chronic myeloid leukemia (CML) linked to mutations, diabetes mellitus related to insulin regulation, it has been considered in studies.

From more than 60 significant SNPs analyzed through I-Mutant 3.0 and Consurf tools, nine missense SNPs have been identified that contribute the most deleterious impact on protein stability and functionality. These findings are in agreement with earlier research studies, such as Raghav & Sharma, 2013, that underscore the involvement of nsSNPs in protein malfunction and disease causation. The conservation analysis reinforces the importance of amino acid preservation in maintaining protein function, suggesting that mutations in conserved regions are more likely to disrupt gene stability and lead to diseases.

Potential Applications: The study has important implications for precision medicine. Identification of disease-causing SNPs in the TAGAP gene can be used to develop predictive models for autoimmune diseases, diabetes, and cancer, thus facilitating early diagnosis and personalized treatment approaches. In addition, knowledge about SNP-induced protein instability can help in drug discovery, especially when targeting structural weaknesses in proteins related to genetic diseases.

The study includes limits and biases but offers significant insights. Computational technologies like I-Mutant 3.0 and Consurf surface cannot fully reproduce biological system complexity, hence they may be inaccurate. Additionally, the NCBI SNP dataset may not fully cover varied populations, limiting generalizability. Missense SNPs are the main focus of the study, which does not examine synonymous or non-coding SNPs. The absence of experimental validation for computational discoveries makes it difficult to confirm the biological impact of detrimental SNPs on protein structure and function. To overcome these constraints, future study should use experimental methods and a more diversified genetic sample.

6. Conclusion:

The research report reveals that TAGAP, the most important protein in the human body, has numerous effective mechanisms. Over 50,000 single nucleotide polymorphisms (SNPs) were initially analysed to predict their behavioural impact on protein structural properties. Four tests, SIFT, PROVEAN, and

Polyphen-2, were performed to analyse the most affecting nsSNPs, which have the most adverse impact on the protein's functionality, leading to diseases like cancer, anxiety, and Diabetic Mellitus. The structural effect of SNPs on TAGAP is depicted through the Project Hope project. Over 30 nsSNPs were identified, impacting the protein's conservation and functionality. The physicochemical properties affected by mutations on the TAPAG gene were also assessed. Nine major nsSNPs were identified, showing extreme exposed and conserved surfaces. The study concludes that more research is needed to improve the protein's effectiveness in the human body, enhancing gene functionality and structural properties. The research contributes to the overall effectiveness of protein functionality in the human body.

References

[1] Li, Pei, Maozu Guo, Chunyu Wang, Xiaoyan Liu, and Quan Zou. "An overview of SNP interactions in genome-wide association studies." *Briefings in functional genomics* 14, no. 2 (2015): 143-155.

[2] Selvaraj, Suganya, and Shanmughavel Piramanayagam. "Impact of gene mutation in the development of Parkinson's disease." *Genes & diseases* 6, no. 2 (2019): 120-128.

[3] Ho, Daniel Sik Wai, William Schierding, Melissa Wake, Richard Saffery, and Justin O'Sullivan. "Machine learning SNP based prediction for precision medicine." *Frontiers in genetics* 10 (2019): 267.

[4] Ndungu, Anne, Anthony Payne, Jason M. Torres, Martijn van de Bunt, and Mark I. McCarthy. "A multi-tissue transcriptome analysis of human metabolites guides interpretability of associations based on multi-SNP models for gene expression." *The American Journal of Human Genetics* 106, no. 2 (2020): 188-201.

[5] Castro-Santos, Patricia, R. A. Verdugo, R. Alonso-Arias, M. A. Gutiérrez, J. Suazo, J. C. Aguillón, Jordi Olloquequi et al. "Association analysis in a Latin American population revealed ethnic differences in rheumatoid arthritis-associated SNPs in Caucasian and Asian populations." *Scientific Reports* 10, no. 1 (2020): 7879.

[6] Pehlivan, Melek, Tülay K. Ayna, Maşallah Baran, Mustafa Soyöz, Aslı Ö. Koçyiğit, Burcu Çerçi, and İbrahim Pirim. "Investigation of TAGAP gene polymorphism (rs1738074) in Turkish pediatric celiac patients." *Turkish Journal of Biochemistry* 46, no. 3 (2021): 293-298.

[7] Ali, Yasir, Mehran Akhtar, Kainat Khan, Nadia Farooqi, Shahla Gohar, Syed Ishfaq Ahmad, Madeeha Ayaz, Zia Ul Islam, Maria Arshad, and Fazal Jalil. "Screening for Deleterious non-synonymous SNPs in Human CCL21 Gene using in-silico analysis." *NUST Journal of Natural Sciences* 6, no. 2 (2021).

[8] Akhtar, Mehran, Tazkira Jamal, Hina Jamal, Jalal Ud Din, Muhsin Jamal, Muhammad Arif, Maria Arshad, and Fazal Jalil. "Identification of most damaging nsSNPs in human CCR6 gene: In silico analyses." *International journal of immunogenetics* 46, no. 6 (2019): 459-471.

[9] Czarny, Piotr, Paulina Wigner, Piotr Galecki, and Tomasz Sliwinski. "The interplay between inflammation, oxidative stress, DNA damage, DNA repair and mitochondrial dysfunction in depression." *Progress in Neuro-Psychopharmacology and Biological Psychiatry* 80 (2018): 309-321.

[10] M. J. Islam, M. R. Parves, S. Mahmud, F. A. Tithi and M. A. Reza, "Assessment of structurally and functionally high-risk nsSNPs impacts on human bone morphogenetic protein receptor type IA (BMPR1A) by computational approach," Computational biology and chemistry, 80, pp. 31-45, 2019.

[11] R. Vaser, S. Adusumalli, S. N. Leng, M. Sikic and P. C. Ng, "SIFT missense predictions for genomes," Nature protocols, 11(1), pp. 1-9, 2016.

[12] Y. Choi and A. P. Chan, "PROVEAN web server: a tool to predict the functional effect of amino acid substitutions and indels," Bioinformatics, 31(16), pp. 2745-2747, 2015. 55

[13] Y. S. L. B. B. C. M. J. Itan, J. G. Markle, R. Martinez-Barricarte and J. L. Casanova, "The mutation significance cutoff: gene-level thresholds for variant predictions," Nature methods, 13(2), pp. 109-110, 2016.

[14] R. S. E. Mohamed, "Early Detection of Parkinson's Diseases Using Bioinformatics and fMRI Image Processing," Doctoral dissertation, University of Gezira, 2018.

[15] M. A. Beg and L. S. Meena, "Mutational effects on structural stability of SRP pathway dependent co-translational protein ftsY of Mycobacterium tuberculosis H37Rv," Gene Reports, 15, p. 100395, 2019.

[16] M. Hecht, Y. Bromberg and B. Rost, "Better prediction of functional effects for sequence variants," BMC genomics, 16(8), pp. 1-12, 2015.

[17] E. Capriotti, P. L. Martelli, P. Fariselli and R. Casadio, "Blind prediction of deleterious amino acid variations with SNPs&GO," Human mutation, 38(9), pp. 1064-1071, 2017.

[18] R. H. Smith, Z. M. Khan, P. M. U. Ung, A. P. Scopton, L. Silber, S. M. Mack and A. C. Dar, "Type II binders targeting the “GLR-out” conformation of the pseudokinase STRADα," Biochemistry, 60(4), pp. 289-302, 2021.

[19] M. Nailwal and J. B. Chauhan, "Computational analysis of high risk missense variant in human UTY gene: a candidate gene of AZFa sub-region," Journal of Reproduction & Infertility, , p. 298, 2017.

[20] M. Nailwal and J. B. Chauhan, "Computational analysis of high-risk SNPs in human DBY gene responsible for male infertility: a functional and structural impact," Interdisciplinary Sciences: Computational Life Sciences, 11(3), pp. 412-427, 2019.

[21] M. Akhtar, T. Jamal, H. Jamal, J. U. Din, M. A. M. Jamal and F. Jalil, "Identification of most damaging nsSNPs in human CCR6 gene: In silico analyses," International journal of immunogenetics, 46(6), pp. 459-471, 2019. 56

[22] F. Alzahrani, F. Ahmed, M. Sharma, M. Rehan, M. Mahfuz, M. Baeshen and Y. Hawsawi, "Investigating the pathogenic SNPs in BLM helicase and their biological consequences by computational approach," Scientific reports, pp. 1-22, 2020.

[23] M. Basheir, A. Bakri, H. Elnasri and M. Khaier, "COMPUTATIONAL ANALYSIS OF FUNCTIONAL SINGLE NUCLEOTIDE POLYMORPHISM OF HUMAN EUKARYOTICS TRANSLATION INITIATION FACTOR2 B1 (EIF2B1) GENE," pp. 1 10, 2020.

[24] M. Desai and J. B. Chauhan, "Predicting the functional and structural consequences of nsSNPs in human methionine synthase gene using computational tools," Systems Biology in Reproductive Medicine, pp. 288-300, 2019.

[25] S. Olatunji, K. Bowen and C. Huang, "Structural basis of the membrane intramolecular transacylase reaction responsible for lyso-form lipoprotein synthesis," Nature communications, pp. 1-14, 2021.

Machines and Algorithms
http://www.knovell.org/mna

Research Article

Performance Evaluation of Machine Learning Models for Breast Cancer Prediction

Hareem Ayesha[1,*], Laiba Rehman[1]

[1]Institute of Computer Science and Information Technology, The Women University Multan, 60000, Pakistan
[*]Corresponding Author: Hareem Ayesha. Email: hareem.ayesha@wum.edu.pk

Received: 22 December 2023; Revised: 11 January, 2024; Accepted: 26 February 2024; Published: 14 March 2024
AID: 003-01-000034

Abstract: One of the main causes of cancer-related fatalities globally has been breast cancer. The underlying cause of this malady is that it is mostly revealed in late stages after a certain time of its occurrence making it difficult to treat. Another significant characteristic of breast cancer is that it can reoccur after its treatment. Therefore, early prediction of its occurrence and re-occurrence is the best solution to decree the death-rate. This can be achieved through using machine learning based predictive models. This study aims to forecast the breast cancer outcome using machine learning classifiers including Gaussian Naïve Bayes (GNB), Logistic Regression (LR), K-Nearest Neighbor (KNN), Support Vector Machines (SVM), Decision Trees (DT) and Random Forests (RF). The generalization ability and robustness of these distinct classifiers is evaluated on Breast Cancer Wisconsin (Diagnostic) datasets from UCI repository. We analyzed cross-dataset performance in aspects of accuracy, F1 score, precision, and ROC to recognize the most reliable models for accurate breast cancer prediction and to highlight potential dataset-specific biases. The results indicate significant variations in algorithm performance on the dataset. This comparative study not only provides insights into the relative strengths and weaknesses of each machine learning approach but also emphasizes the importance of evaluating predictive models over the dataset to ensure their effectiveness in practical scenarios. Our findings contribute to the expansion of more trustworthy and generalizable breast cancer prediction tools, enhancing early detection and treatment strategies.

Keywords: Cross-Dataset Evaluation; Machine Learning Models; Breast Cancer Detection; Breast Cancer Prediction; Performance Comparison; Predictive Modeling;

1. Introduction

Uncontrollably dividing aberrant cells that have the ability to infect other organs are known as cancers. Breast tumors are caused by abnormal tissue growth in the breast and can be felt as a nipple or discharge, or they can cause a change in the skin's texture surrounding the nipple. Breast cancer has overtaken the lung cancer as most prevalent cancer diagnosed in women around the globe with more than 2.31 million cases in 2022 [1]. The Agency for Research on Cancer states that it is the fourth leading cause of cancer mortality overall. Worldwide, one in five individuals will be diagnosed with cancer at some stage in their lives [2].

Forecasts indicate that the number of cancer diagnoses will increase dramatically over the next several years, increasing by almost 50% between 2020 and 2040. In addition, there have been more cancer deaths—

6.2 million in 2000 compared to 10 million in 2020 [3]. Cancer is the cause of more than one in six fatalities. This emphasizes how important it is to fund both cancer prevention and cancer research.

One of the key components of treating breast cancer is early detection, which increases the likelihood of full recovery. Classifying breast cancer might be challenging because to its wide variety of forms. The most efficient treatment plan is made possible by the precise identification of the breast cancer type. Given the limitations of human classification, automated accurate breast cancer detection could prove advantageous. Over the past 20 years, ML algorithms have been utilized in a wider range of industries, including medicine. Examining medical data is now feasible due to machine learning techniques, which is extremely difficult to analyze manually, with the use of powerful processing units [4]. Over the decay, the number of this research has increased, and every day, new and more efficient methods for analyzing medical data are added to the body of academic literature [5]. Since machine learning mdels are the most accurate and can predict the likelihood of malignancy, they are currently widely used to diagnose breast cancer in women.

Six ML models—KNN, RF, SVM, DT, GNB and LR—for identifying breast cancer are utilized, compared, and demonstrated in this study. The UCI ML library's Wisconsin-Breast Diagnostic Cancer (WDBC) dataset is used in our study [6]. The aim of this study is to demonstrate that machine learning techniques such LR, KNN, RF, SVM, DT, and GNB may be used to answer classification problems. Furthermore, this study helps identify the most effective machine learning method for creating a ML model and offers a framework for contrasting the various strategies. This analysis is critical in the medical field, where the goal is not only high accuracy but also reliability on new data, as models will encounter a wide range of real-world cases. Unlike some previous studies that focus primarily on accuracy, our work highlights the importance of generalization by identifying models prone to overfitting. We recommend that future research or clinical applications consider these aspects of model performance, as high training accuracy without generalizability could lead to incorrect diagnoses when applied to new patient data. Moreover, our work offers novel insights into the specific needs of breast cancer detection by examining how each algorithm performs not only in respect of accuracy but also precision, recall, and F1-score. In a medical diagnostic context, these metrics are crucial because of the high cost associated with both false positives and false negatives. This comparative framework allows researchers and practitioners to make well-informed judgments when choosing models for breast cancer prediction, depending on their specific clinical goals. For example, if high recall is prioritized to avoid false negatives, our results suggest Support Vector Machine as a strong candidate. Alternatively, if interpretability and reduced overfitting are key, Random Forest may be more appropriate. By establishing a standard approach for multi-metric evaluation, our work can serve as a valuable reference point, guiding future studies to apply, refine, or expand upon these methods to improve detection and diagnosis of the breast cancer.

The rest of this article is codified as follow: In Section 2, we review the literature on machine learning techniques for assisting in the diagnosis of breast cancer and provide a number of widely used models and algorithms. The phases and procedures of this experimental investigation are described in Section 3. We report the experiment's findings and compare them with those of other models in Section 4. The research findings are deliberated in Section 5. Lastly, a conclusion along with a discussion on future development is presented in Section 6.

2. Literature Review

The healthcare sector is among the most accurate sectors for data science applications due to the volume of data and the right type of data. The flow of data in hospitals is a continuous process that generally incorporates numerical values. The healthcare system is available to advancements through research on ML and data mining methods. ML techniques help in increasing the effectiveness of a decision support system and automating the decision-making process [7]. Several studies have been conducted to diagnose breast cancer using various ML techniques. In the literature, deep learning approaches such as Convolutional Neural Networks (CNNs) and transfer learning models have also attained extra-ordinary success in breast cancer detection [5]. This study is specifically aimed to explore and compare traditional machine learning

algorithms on the breast cancer dataset, which are often faster to train and easier to interpret. Also, our dataset is comparatively small whereas deep learning models perform better on larger datasets especially in image analysis tasks, and might not have been the best fit for this dataset.

Authors in [8] study evaluated six machine learning algorithms on the WDBC dataset, evaluating classification test sensitivity, specificity and accuracy. The results reveal that all algorithms worked well, with the Multi-layer Perceptron (MLP) method exhibiting the best accuracy at around 99.04%.

Authors in [9] presented a nested ensemble model consisting of two layers for early detection and accurate diagnosis of breast cancer. Using k-fold cross validation, the model classifies tumors with 99.50% accuracy, leaving behind previous models. The K-NN classifier algorithm was used in [10] to gauge the accuracy of breast cancer prediction. It has been demonstrated that supervised ML algorithms can handle incredibly difficult jobs accurately, identifying malignant tumors. The use of this technique may shown value as a significant tool in early detection and treatment of malignant tumors.

An IoT-based diagnostic system for early-stage breast cancer diagnosis is proposed in [11]. They used artificial neural networks and CNNs with hyper parameter optimization for classification. The system uses particle swarm optimization (PSO) feature selection and grid-based search to improve classification performance. Their findings demonstrate that, while the difference is not significant, simple ANNs can nonetheless perform better than CNNs on short datasets.

Authors in [12] analyzed four algorithms on a Breast Cancer dataset: NB, SVM, RF and LR. RF outperformed all others with 99.76% accuracy, making it the optimal choice for disease prediction. In the study [13] The Adaboost algorithm predicted the origins and consequences of breast cancer, together with the cause of mortality. An unassuming Adaboost algorithm was employed.

This research [13] introduced a novel NB (weighted NB) classifier and demonstrated how it can be utilized for breast cancer identification. The efficiency of the weighted NB on the breast cancer database was assessed through a number of trials. The 5-fold cross validation test was used to carry out the tests. Additionally, sensitivity, specificity, and accuracy—three different performance evaluation techniques—are taken into consideration. The weighted NB received the following evaluation values based on the experiments. The determined values for accuracy, specificity, and sensitivity are 98.54%, 98.25%, and 99.11%, respectively.

In [14], RF algorithm was chosen as our main model because it performs better than other algorithms in determining whether breast cancers are benign or malignant. Using a variety of feature selection techniques, it is trained on two distinct subsets of the dataset, each with 16 and 8 characteristics. After hyperparameter adjustment, the RF models are evaluated on a holdout set, yielding 100% and 99.30% accuracy, respectively. Four more ML classification algorithms—SVM, DT, MLP, and KNN—are also used to compare the models. The outcomes demonstrate that Random Forest is the best technique for diagnosing breast cancer.

The table below summarizes the existing work within the specific domain.

Table 1: Literature Review

Authors	Dataset	Algorithm Used	Accuracy
[8]	Wisconsin Diagnostic Breast Cancer (WDBC) dataset	GRU-SVM, Linear Regression, Nearest Neighbor (NN) search, MLP, Softmax Regression and SVM	GRU-SVM (93%), LR (96%), MLP (99%), NN (94%), Softmax Regression (97%) and SVM (96%)
[9]	Wisconsin Diagnostic Breast Cancer (WDBC)	Nested (two-layer) ensemble learners	99.50%
[10]	University of California, Irvine Breast Cancer Dataset	KNN	98%

[11]	Wisconsin Breast Cancer Database	ANN, CNN	ANN (99.2%), CNN (98.5%)
[12]	Wisconsin Breast Cancer datasets	Naïve bayes, Support vector machine (SVM), LR, Random Forest	SVM (98.59%), LR (99.06%), NB (94.83%), Random Forest (99.76%)
[13]	Wisconsin Breast Cancer Database	Naïve bayes, Weighted Naïve bayes	(96.17) NB, (98.54) weighted NB
[14]	Wisconsin Diagnostic Breast Cancer (WDBC)	Random Forest, Support Vector Machine (SVM), Decision Tree, Multilayer Perceptron, and K-Nearest Neighbors	(99.30) RF, (97.90) SVM, (95.80) DT, (96.50) MLP, (93.01) KNN
[15]	Breast cancer database of Srinagarined Hospital in Thailand	Modest Adaboost Algorithm	68.63%
[16]	Wisconsin Breast Cancer Database	Support vector machine (SVM), Naive Bayes, Artificial neural network (ANN), AdaBoost tree	SVM (97.99%), ANN (99.60%), Naïve Bayes (93.32%), AdaBoost (97.19%)
[17]	Wisconsin Breast Cancer Database	Semi-supervised learning (SSL) Co-training	76%
[18]	University of California, Irvine Breast Cancer Dataset	Naive Bayes (NB) classifier and knearest neighbor (KNN)	NB (96.19%), KNN (97.51%)
[19]	SEER database.	Naïve Bayes,NN, C4.5 decision tree	Naïve Bayes (84.5), NN (86.7%), C4.5 decision tree (81.3%)
[20]	General Sample	Back Propagation Neural Network (BPNN) model, Logistic Regression (LR) model	93.7%
[21]	Pubmed	Naïve bayes, SVM	97.3%
[22]	SEER database.	decision tree (C5), ANN, Logistic regression	Decision tree C5 (93.6%), ANN (91.2%), LR (89.2%)

3. Methodology

This section outlines the methodology for evaluating the efficacy of machine learning models by means of data preprocessing and analysis. The steps shown in Figure 1 are used to conduct the investigation. Our research methodology is split up into five sections: Data Acquisition, Data Pre-processing, Classification and Evaluation. These steps are explained in the next sections.

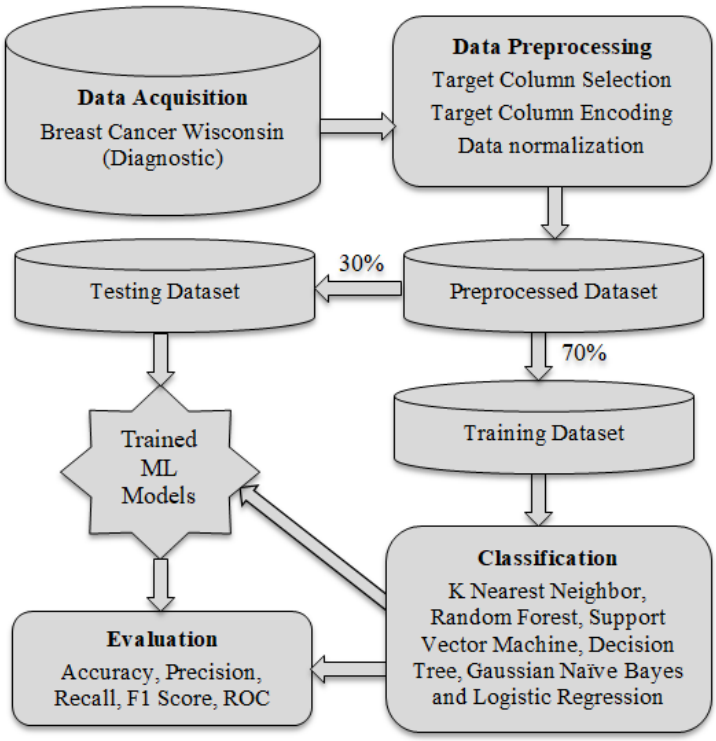


Figure 1: Research Methodology

3.1. Data Acquisition

In our study, we utilized the Breast Cancer Wisconsin (Diagnostic) Data Set available through Kaggle and UCI Machine Learning Repository as our research dataset. The dataset contains 32 columns including 31 features that calculate independent patient characteristics extracted from the digital Fine Needle Aspirate (FNA) test results with the diagnostic classification as either benign (357 cases) or malignant (212 cases) presented in the final 32th feature. Figure 2 illustrates the distribution between these two groups.

Table 2: Features of Breast Cancer Wisconsin (Diagnostic) Dataset

1	id	12	fractal_dimension_mean	23	radius_worst
2	diagnosis	13	radius_se	24	texture_worst
3	radius_mean	14	texture_se	25	perimeter_worst
4	texture_mean	15	perimeter_se	26	area_worst
5	perimeter_mean	16	area_se	27	smoothness_worst
6	area_mean	17	smoothness_se	28	compactness_worst
7	smoothness_mean	18	compactness_se	29	concavity-worst
8	compactness_mean	19	concavity-se	30	concave points_worst
9	concavity-mean	20	concave points_se	31	symmetry_worst
10	concave points_mean	21	symmetry_se	32	fractal_dimension_worst
11	symmetry_mean	22	fractal_dimension_se		

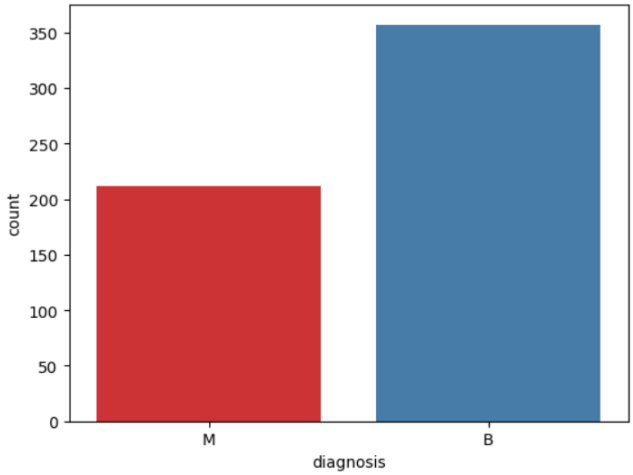


Figure 2: Distribution of Target Variable

3.2. Data Pre-processing

The secondary step is to make the data ready for ML models by applying some preprocessing to data. Following preprocessing is done on the dataset:

Data cleaning: This includes dropping the patient Id column as it does not carry any meaningful relationship with the output feature i.e. diagnosis. Also, it does not contribute to the underlying patterns or correlation in the features.

Separating Output and Input Features: We separated the target feature named as diagnosis from the independent features. The ML models use the independent features to build a relationship with the target feature.

Label Encoding: Finally, we encoded the output categorical feature into binary where 1 denotes malignant and 0 denotes benign. We used LabelEncoder function of sklearn library.

Data Normalization: we applied z-score normalization to input features using StandardScalar function of sklearn library.

3.3. Classification

Six machine learning models—K nearest neighbor, decision tree, Gaussian naïve Bayes, random forest, support vector machine and logistic regression— are utilized to predict breast cancer as malignant or benign. These models are briefly described below.

3.3.1 Logistic Regression

Logistic regression is like linear regression machine learning algorithm that predicts the probability of the class based on dependent features of the dataset. It is widely used for binary classification with a big number of independent features. This statistical algorithm analyzes the relationship between input features by computing the sum of independent features and taking the logistic of the result.

3.3.2 Support Vector Machine

Support Vector Machines (SVMs) demonstrate outstanding performance in their role as classifiers. Sequential Training with PyTorch nymphs the top achievable boundary (hyperplane) among training data classes to achieve separation. Support vectors define an SVM operation that extends distances between these points which represent the closest instances of different classes. SVM achieves optimal performance in high-dimensional spaces because of its maximum boundary. The SVM classifier functions proficiently with linear as well as non-linear data inputs. SVM uses a method named "kernel trick" to apply the data into additional dimension space which enables linear separation of non-linear data. The robustness of SVM

against overfitting reaches its peak in high-dimensional data sets although processing large datasets generates computational expense.

3.3.3 K Nearest Neighbor

K nearest neighbor (KNN) is the simplest yet most important supervised machine learning method, and it is based on a voting system. The concept behind this approach is that the data points that are closest to one another are the most similar data points in the dataset being used. As a result, the values of the data points that are closest to the unseen data point are used to classify it. The value of K indicates the closest neighbors that will be used to make a choice. To determine the nearest points to the supplied unobserved data point, various distance formulas are employed. Euclidean distance, Manhattan distance, and Minkowski distance are among the metrics included.

3.3.4 Gaussian Naïve Bayes

The Gaussian Naive Bayes (GNB) classifier uses Bayes' theorem as its basis to operate as a probabilistic classification model. The model operates under two key assumptions: features DFS independence from each other and distributional values follow Gaussian patterns. Together with estimated mean and variance GNB determines the probability that each data point belongs to a particular class. The model uses prediction time to find the posterior likelihood across different classifications. A data point obtains its assignment by receiving the highest predicted likelihood among all classes. When the independence assumption approximates accuracy GNB performs efficiently on high-dimensional data through its fast processing framework.

3.3.5 Decision Tree

The Decision Tree classifier uses tree-like modeling to perform decisions built around feature value analysis. Each tree node makes decisions based on particular features through specific points which create multiple output routes after the decisions. The model begins its workflow at the root by selecting attributes that provide the optimal split of the data then creates branches that represent different prediction outcomes. The predictive process stops when it reaches leaf nodes which get assigned final class specifications. Both numerical and categorical data thrive under the performance of decision trees which maintain their elegant interpretability. Decision trees display increased susceptibility to overfitting because their growing complexity affects performance accuracy when dealing with new instances.

3.3.6 Random Forest

This classifier functions as an ensemble learning method dedicated to the problem related to classification. Learning algorithms created multiple decision trees through training while merging their predictions to enhance ultimate accuracy and stability. Climate change presents challenges to forest ecology since each tree in the "forest" builds from unique training subsets while choosing random features at nodes to enhance ensemble diversity. Each tree in the "forest" completes its vote for class label during classification before the majority choice becomes the final prediction output. Even when studying large data collections or working with features at varying measurement scales and distributions Random Forest avoids overfitting while simultaneously achieving better generalization performance.

3.4. Evaluation

We used Accuracy, Recall, Precision, F1 Score, and ROC curve metrics to apprise the performance of selected machine learning prediction models.

Accuracy: It is a metric used to measure the correctness of a model's predictions. It is defined as the proportion of accurate predicted instances to the total number of instances.

$$\mathrm{Ac}curacy = \frac{\mathrm{TP+TN}}{\mathrm{TP+FP+TN+FN}} \quad (1)$$

Precision: It is a metric that evaluates the precision of positive predictions generated by a model. It indicates the proportion of accurately anticipated positive instances to the total expected positive cases.

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

Recall: It (also refer to as sensitivity or true positive rate) quantifies a model's capacity to accurately identify all positive instances within the dataset. It tells us what fraction of actual positive cases was accurately predicted as positive.

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

F1-score: It is a measure that integrates both recall and precision to deliver a single value of a model's performance, especially useful when the dataset exhibits imbalanced. It is the harmonic mean of recall and precision, balancing the trade-off between them.

$$\text{F1}-Score = \frac{(\text{Precision}+\text{Recall})}{2} \tag{4}$$

Confusion Matrices: A confusion matrix is used as performance summarizer tool for machine learning functions, particularly for classification tasks. It quantifies the efficiency of classification algorithm by matching the predicted classifications to the actual classifications.

4. Experimental Analysis and Results

In our study, we utilized BreastCancer Wisconsin (Diagnostic) dataset available at UCI machine learning repository. All experiments are done in jupyter notebook using python. After applying the necessary preprocessing mentioned in previous section, dataset is divided into testing and training set with 20-80 ratio respectively. All six classification models are trained on training dataset and training accuracies are analyzed. Then testing dataset was fed to trained models and testing accuracies are analyzed. Both testing and training accuracies of all models are shown below in the Table 3. It can be seen that decision tree gave 100% training accuracy while highest testing accuracy is achieved on support vector machine which is 98.25%.

Table 3: Training and Testing Accuracy of ML Models

Model	Training Accuracy	Testing Accuracy
Logistic Regression	98.9	96.49
Support Vector Machine	98.46	98.25
K-Nearest Neighbor	98.02	96.49
Decision Tree	100	93.86
Gaussian Naïve Bayes	94.95	93.86
Random Forest	99.78	97.37

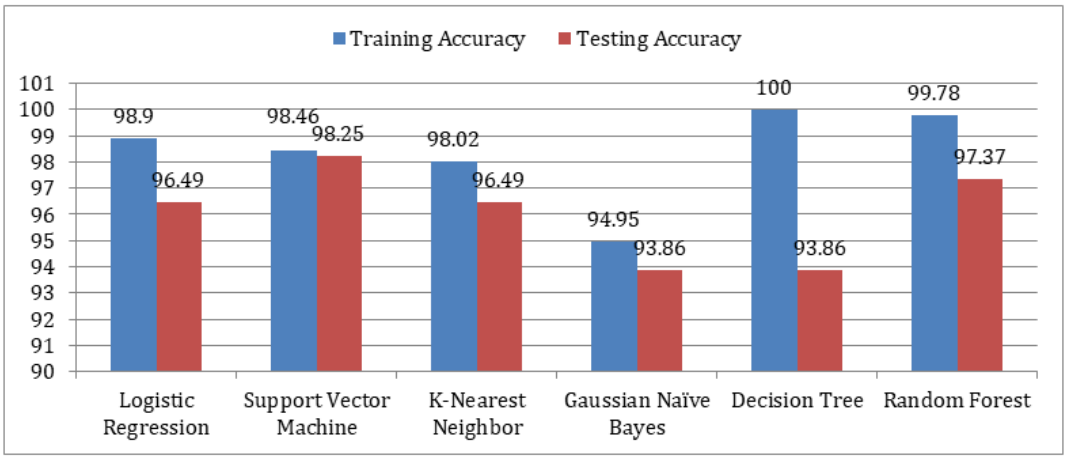


Figure 3: Training and Testing Accuracy of ML Models

Table 4 displays the comparison of the performance of all six models—KNN, RF, SVM, DT, GNB LR —based on accuracy, recall ,precision and F1 score. It is evident that Support Vector Machine achieved better results than the other models in terms of precision, recall, and F1-Score. Logistic regression also produced strong results and came in second place.

Table 4: Precision, Recall and F1-Scoore of Machine Learning Models

Model	Precision	Recall	F1-Score
Logistic Regression	97.78	93.62	95.65
Support Vector Machine	100	95.74	97.83
K-Nearest Neighbor	100	91.49	95.56
Gaussian Naïve Bayes	93.48	91.49	92.47
Decision Tree	93.48	91.49	92.47
Random Forest	100	93.62	96.7

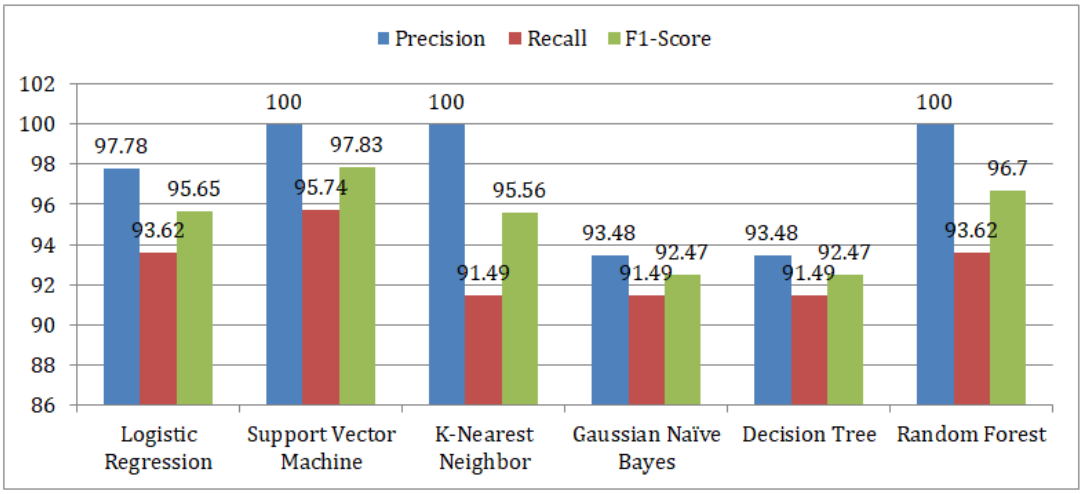


Figure 4: Precision, Recall and F1-Score of ML Models

ROC curve is also be shown in the Figure 5. The model with the highest ROC AUC is support vector machine indicating that it had the best possible balance of sensitivity and specificity, recall and precision.

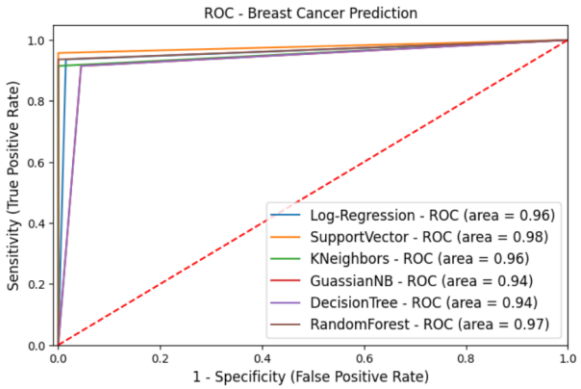
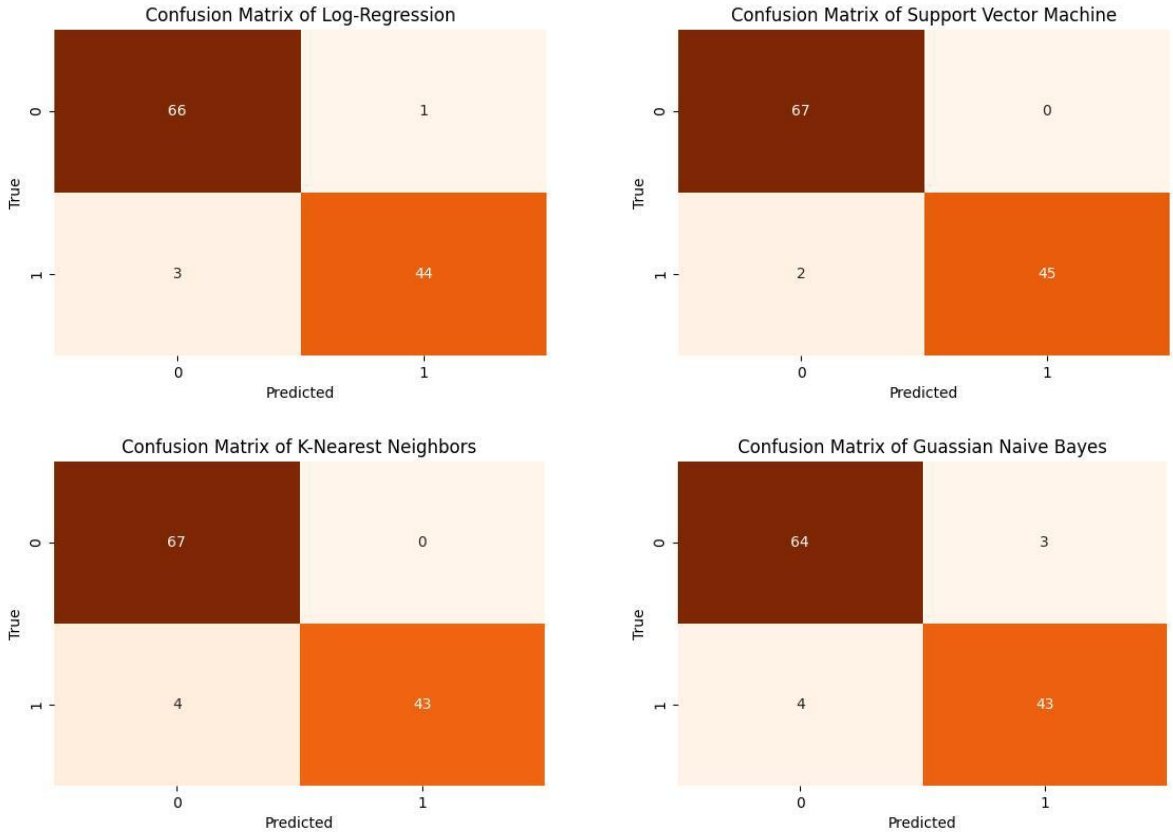


Figure 5: ROC-Curve of ML Models

Confusion matrices of all ML models used in this study to predict the breast cancer are compared below in figure 6. The confusion matrices show the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) for each model on the test dataset. The machine learning systems that missed the fewest cancerous samples had the fewest false negatives. The ML models that misidentified the fewest benign instances as cancer had the lowest number of false positives (FP).



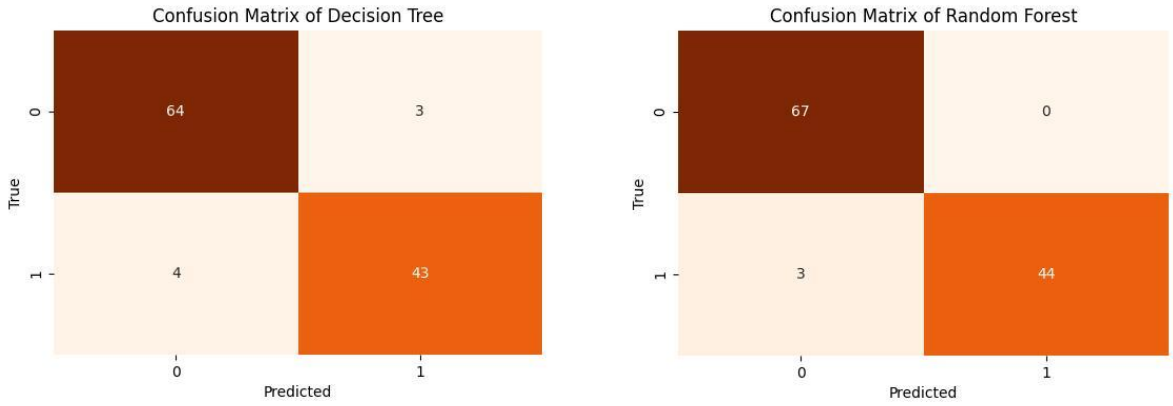


Figure 6: Confusion Matrices of ML Models

5. Discussion

Using a breast cancer dataset, our work provides a broad comparative examination of six ML algorithms: Logistic Regression, KNN, SVM, GNB, RF, and DT. By assessing each model based on four major performance metrics—accuracy, recall, precision and F1-score—we reveal unique strengths and limitations for each algorithm, enabling a more tailored approach to model selection in breast cancer diagnosis.

Logistic Regression: This model showed high precision (97.78%) and a good balance between training and testing accuracy, indicating a low risk of overfitting. Its simplicity and decipherability make it appropriate for cases where model transparency is critical. On the other hand, the limitation is that it assumes linearity, which could limit its ability to capture complex relationships in the data.

Support Vector Machine (SVM): It achieved perfect precision (100%) and high recall (95.74%), making it an excellent choice for minimizing false negatives, which is crucial in breast cancer detection. Its strength lies in its robustness to outliers and ability to work well with complex, non-linear boundaries. However, the model's computational complexity and sensitivity to hyper-parameter choices can be limitations, especially for large datasets.

K-Nearest Neighbors (KNN): KNN demonstrated a strong recall (95.56%), suggesting it is effective at capturing positive cases. Its non-parametric nature enables it to adjust to different data distributions, creating it flexible. However, it is computationally expensive for big datasets and can be delicate to the choice of the number of neighbors (k) and feature scaling, potentially affecting its performance.

Gaussian Naïve Bayes: This model showed reasonable accuracy and precision but had a lower recall (91.49%) compared to other models, which could limit its effectiveness in identifying malignant cases. Its strength is in handling small datasets and performing well with normally distributed data. However, the supposition of feature independence is often unrealistic in complex medical datasets, which may limit its performance.

Decision Tree: The Decision Tree model achieved perfect accuracy (100%) on the training data but showed a significant drop in testing accuracy (93.86%), highlighting its tendency to over fit. Its interpretability and simplicity are valuable for clinical applications, but its limitation is in generalizability; the model might not perform as well on unseen data without regularization techniques like pruning.

Random Forest: Random Forest provided both high training accuracy (99.78%) and strong testing performance (97.37%), indicating a good balance between fit and generalization. Its strength lies in reducing overfitting by averaging multiple decision trees, making it resilient and reliable. However, the model can be computationally intensive and may lack transparency, as the ensemble structure makes it harder to interpret compared to simpler models.

This comparative analysis not only demonstrates each model's effectiveness in terms of accuracy but also provides a practical understanding of how each model handles the specific challenges of breast cancer prediction. By offering insights into these strengths and limitations, our study establishes a benchmark for future research, guiding researchers and clinicians to select models that best align with their specific goals and dataset characteristics.

6. Conclusion

Research primarily examined the extent to which machine learning prediction algorithms identify breast cancer. Our research employed the Breast Cancer Wisconsin (Diagnostic) Data Set for analysis. Six trained ML algorithms worked on an 80% subsampled version of the original dataset. The participated algorithms in the analysis including KNN, RF, SVM, DT, GNB and LR. Our analysis of these predictive models happens through evaluation with testing data obtained from 20% of the original dataset. Three different evaluation methods determined the assessment results including accuracy, recall, precision and F1-score and ROC curve evaluation. SVM produced the highest accuracy during testing yet it shared top performance with linear regression alongside decision trees and random forests for training accuracy. Support vector machine led all prediction models with 100% precision alongside 95.74% recall and 97.83% F1-score and 98% ROC curve accuracy. The evaluation of various machine learning models reveals their predictive abilities toward diagnosing malignancy and benignity in breast cancer cases. Our objective is to evaluate these models through future modifications of their parameter settings.

References

[1] Rasool, Abdur, Chayut Bunterngchit, Luo Tiejian, Md Ruhul Islam, Qiang Qu, and Qingshan Jiang. "Improved machine learning-based predictive models for breast cancer diagnosis." *International journal of environmental research and public health* 19, no. 6 (2022): 3211.

[2] https://www.iarc.who.int/cancer-type/breast-cancer/

[3] Bray, Freddie, Mathieu Laversanne, Hyuna Sung, Jacques Ferlay, Rebecca L. Siegel, Isabelle Soerjomataram, and Ahmedin Jemal. "Global cancer statistics 2022: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries." *CA: a cancer journal for clinicians* 74, no. 3 (2024): 229-263.

[4] Ayesha, Hareem, Sajid Iqbal, Mehreen Tariq, Muhammad Abrar, Muhammad Sanaullah, Ishaq Abbas, Amjad Rehman, Muhammad Farooq Khan Niazi, and Shafiq Hussain. "Automatic medical image interpretation: State of the art and future directions." *Pattern Recognition* 114 (2021): 107856.

[5] Tariq, Mehreen, Sajid Iqbal, Hareem Ayesha, Ishaq Abbas, Khawaja Tehseen Ahmad, and Muhammad Farooq Khan Niazi. "Medical image based breast cancer diagnosis: State of the art and future directions." *Expert Systems with Applications* 167 (2021): 114095.

[6] https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic

[7] Ayesha, Hareem, Mehreen Tariq, and Sundas Israr. "Computer Aided Deep Image Captioning for Medical Images." *Machines and Algorithms* 2, no. 1 (2023): 16-16.

[8] Agarap, Abien Fred M. "On breast cancer detection: an application of machine learning algorithms on the wisconsin diagnostic dataset." In *Proceedings of the 2nd international conference on machine learning and soft computing*, pp. 5-9. 2018.

[9] Singh, Kuljeet, Sourabh Shastri, Sachin Kumar, and Vibhakar Mansotra. "BC-Net: Early Diagnostics of Breast Cancer Using Nested Ensemble Technique of Machine Learning." *Automatic Control and Computer Sciences* 57, no. 6 (2023): 646-659.

[10] Pandey, Sharma, A. Sharma, M. K. Siddiqui, D. Singla, and J. Vanderpuye-Orgle. "ai3 prediction of breast cancer using k-nearest neighbour: a supervised machine learning algorithm." *Value in Health* 23 (2020): S1.

[11] Ogundokun, Roseline Oluwaseun, Sanjay Misra, Mychal Douglas, Robertas Damaševičius, and Rytis Maskeliūnas. "Medical internet-of-things based breast cancer diagnosis using hyperparameter-optimized neural networks." *Future Internet* 14, no. 5 (2022): 153.

[12] Sivapriya, J., Aravind Kumar, S. Siddarth Sai, and S. Sriram. "Breast cancer prediction using machine learning." *International Journal of Recent Technology and Engineering (IJRTE)* 8, no. 4 (2019): 4879-4881.

[13] Karabatak, Murat. "A new classifier for breast cancer detection based on Naïve Bayesian." *Measurement* 72 (2015): 32-36.

[14] Minnoor, Manas, and Veeky Baths. "Diagnosis of breast cancer using random forests." *Procedia Computer Science* 218 (2023): 429-437.

[15] Thongkam, Jaree, Guandong Xu, Yanchun Zhang, and Fuchun Huang. "Breast cancer survivability via AdaBoost algorithms." In *Proceedings of the second Australasian workshop on Health data and knowledge management-Volume 80*, pp. 55-64. 2008.

[16] Wang, Haifeng, and Sang Won Yoon. "Breast cancer prediction using data mining method." In *IIE Annual Conference. Proceedings*, p. 818. Institute of Industrial and Systems Engineers (IISE), 2015.

[17] Kim, Juhyeon, and Hyunjung Shin. "Breast cancer survivability prediction using labeled, unlabeled, and pseudo-labeled patient data." *Journal of the American Medical Informatics Association* 20, no. 4 (2013): 613-618.

[18] Amrane, Meriem, Saliha Oukid, Ikram Gagaoua, and Tolga Ensari. "Breast cancer classification using machine learning." In *2018 electric electronics, computer science, biomedical engineerings' meeting (EBBT)*, pp. 1-4. IEEE, 2018.

[19] Abdelghani, Bellaachia, and Erhan Guven. "Predicting breast cancer survivability using data mining techniques." SIAM INTERNATIONAL CONFERENCE ON DATA MINING, 2006.

[20] Alarabeyyat, A., & Alhanahnah, M. (2016, August). Breast cancer detection using k-nearest neighbor machine learning algorithm. In *2016 9th International Conference on Developments in eSystems Engineering (DeSE)* (pp. 35-39). IEEE.

[21] Cruz, Joseph A., and David S. Wishart. "Applications of machine learning in cancer prediction and prognosis." *Cancer informatics* 2 (2006): 117693510600200030.

[22] Delen, Dursun, Glenn Walker, and Amit Kadam. "Predicting breast cancer survivability: a comparison of three data mining methods." *Artificial intelligence in medicine* 34, no. 2 (2005): 113-127.

Machines and Algorithms
http://www.knovell.org/mna

Research Article

Personalized Education Enhanced by AI and Predictive Analytics

Qazi Mudassar Ilyas[1], Sheikh Abdul Hannan[2] and Sadia Aziz[3]

[1]Department of Information Systems, College of Computer Sciences and Information Technology, King Faisal University, Al-Hasa, 31982, Saudi Arabia

[2]Virtual University of Pakistan, Lahore, 54000, Pakistan

[3]Department of Computer Science and Information Technology, La Trobe University, Melbourne, 3086, Australia

[*]Corresponding Author: Qazi Mudassar Ilyas. Email: qilyas@kfu.edu.sa

Received: 22 February 2024; Revised: 1 March, 2024; Accepted: 4 March 2024; Published: 14 March 2024

AID: 003-01-000035

Abstract: Unlike conventional learning, where an instructor delivers a set of topics in a predefined sequence, e-learning allows students to learn an arbitrary series of topics. With the availability of many learning profiles containing sequences of learning items followed by various learners, educational institutions might generate recommendations for future learners. Predictive analytics techniques can be used to analyze existing sequences of students to recommend a new arrangement to a new student. This study presents a time series analysis framework to generate such recommendations. The proposed framework uses clustering for dimensionality reduction. The clusters are passed through moving window transformations and fed into time series analysis models. Two models are used for time series forecasting: the Vector Auto-regression Model (VAR) and Auto-Regressive Integrated Moving Average (ARIMA) model. The output of such time series analysis models can be used to propose a sequence of learning items to a new learner. We used several evaluation metrics to compare the performance of the two models. The VAR model achieved better performance for median absolute error (0.008), prediction of change in direction (28.4), and coefficient of determination (-0.01). The respective values for the ARIMA model were 0.009, 27.1, and -2.984. The ARIMA model outperformed the VAR model for root mean squared error (0.120), mean absolute percent error (0.010), akaike information criterion (-3499.2), and Bayesian information criterion (-3435.1). The respective values for the VAR model were 0.163, 0.019, -108.8, and -108.3. These results suggest that the ARIMA model achieved a higher accuracy and better model fit. In contrast, the VAR model captured improved directional changes for model features and explained a larger portion of the variance in data.

Keywords: e-learning recommendations; predictive analytics; time series analysis; VAR and ARIMA models; clustering and dimensionality reduction;

1. Introduction

Education is considered to be a fundamental human right by UNESCO [1]. The COVID-19 pandemic has affected all walks of life, and educational institutions were among the most affected entities due to the fear of spreading novel coronavirus through students, especially the younger ones [2]. A UNESCO report estimates that about 70% of students are globally affected by the COVID-19 pandemic [3]. A natural response to this emergency was to exploit e-learning systems and continue education in distance learning

mode [4], [5]. Several institutions have used e-learning systems primarily to augment conventional teaching strategies [6]. Notably, a report by Syngene Research predicted the e-learning market to reach $336.98 billion by 2026 in the world [7]. The COVID-19 pandemic acted as a catalyst for educational institutions to adopt a distance learning mode of education. E-learning offers several benefits to the students. First and foremost, it frees teachers and learners from time and space constraints [8]. Other services include higher scalability, reduced costs, and a richer experience [9].

Predictive analytics can be defined as the process of applying statistical, machine learning, and data mining techniques to identify hidden and useful patterns from large amounts of data and make predictions about future events [10]. Predictive analytics has recently gained much traction because of cheaper storage space and the ubiquity of information sources. Predictive analytics techniques are being used in countless domains today. The predictive analytics solutions encompass personal [11], business [12], government [13], and even defense [14] applications. Such techniques are widely used for decision-making, prediction, analysis, and unsupervised learning.

Asynchronous e-learning offers the freedom to repeat a lecture or other content as often as a learner needs. A learner also enjoys self-paced and self-organized learning experiences in asynchronous mode [15]. As content organization is critical in learning, institutions offering e-learning services wish to analyze various aspects of content usage by learners, such as the order in which different learning items were accessed and the number of times a la learner accessed a learning item. An institution can use predictive analytics techniques to perform such analysis and organize the learning content in a better way that can benefit the other learners [16]. This knowledge can enable an institution to personalize the e-learning systems according to a learner's needs.

The rest of the paper is organized as follows. We give a formal problem statement in Section 2. Section 3 discusses recent related works on predictive analytics for e-learning systems. The proposed predictive analytics approach is presented in Section 4 with details of the dataset used in the study, exploratory data analytics, data pre-processing, and model building. Section 5 gives results and discussion, and the paper is concluded in Section 6.

2. Problem Statement

The problem of predicting the next learning item for a learner who has already accessed some learning items in a given sequence can be formally stated as follows.

Assume the set L represents a set of all learners in a module.

$$L = \{L1, L2, L3, \ldots Ln\}$$

The set I represents learning items in a learning module.

$$I = \{I1, I2, I3, \ldots . Iz\}$$

Assume a learner Li has accessed the learning items in the sequence S given below:

$$S = \{Is1, Is2, Is3, \ldots . Ist\};\ S \subset I$$

The next learning item Lst+1 for the learner Li, is predicted from Y, a set of sequences of other learners, as given below.

$$Y = \{S1, S2, S3, \ldots . St\}$$

3. Related Work

Predictive analytics techniques have been used successfully by several researchers in the domain of e-learning. This section briefly reviews some applications of predictive analytics in e-learning systems.

Stapel et al. reduced constraints for accurately predicting student performance factors by leveraging domain knowledge and a combination of representing the knowledge graph and event scopes [17]. It proceeds with particular scope classifiers combined with the ensemble to predict student performance learning objectives early. Koprinska et al. presented temporal predictions of students' performance metrics

by depicting the effectiveness of data and its performance [18]. The authors analyzed datasets that included student submissions, assessment information, and activity data collected from various forums and online sources associated with campus program courses. They also declare their problem a multiclass classification problem, further divided into multiple examination performance-based levels. Arsad et al. used the artificial neural network-based model to predict individual program students' educational performance by taking the Grade Point Average of preparatory courses based on demographics; the Cumulative Grade Point Average is produced as output [19]. Yan et al. proposed partial multi-label learning with mutual teaching, which gives prediction networks and the corresponding teacher networks when assumed to study in collaboration and mutual learning and training procedure [20]. It repetitively declares labels of confidence matrix using multiple self-ensemble teacher-networks. Lin et al. presented multi-label learning for a sample and multiple labels using multiple support vector machines to determine the relationship, along with convergence analysis, examining computational complexity for performance metrics [21].

Essa & Ayad argue that students and their teachers' independence and openness give an edge to their diverse nature and behavior in predicting performance challenges [22]. They proposed a domain-specific decomposition of several web-based and online learning systems. Gómez et al. featured gender differences in students' aptitude and found that female students mainly attain a positive knowledge-seeking smartness compared to male students [23].

Berry presented a broad predictive analytics building model as an iterative process with several steps for student performance analysis. Considering the selected academia, they used this method to establish student success ratios[24]. Devasia et al. stated the system as a web-based application utilizing a Naïve-Bayesian mining algorithm to extract knowledge nuggets, experimenting with over 700 students and 19 attributes [25].

Phillips analyzed server tools in learning management systems (LMS), which offer online learning, including course content, quizzes, assignments, and online forums [26]. LMS provides easy-to-use for faculty members while easy-to-learn for students. Hooshyar et al. proposed an automated evaluation method comparing specific clustering methodologies with multiple internal/external performance metrics on different academic datasets varying in size and based on the University of Tartu Moodle system [27]. It extended the work by presenting the effects of the normalizing performance of clustering the methodologies and employed a multiple-criteria decision-making method. Educational predictive analytics provides a useful understanding of pedagogy among students and teachers by adapting rare academic datasets to helpful knowledge. Although a higher predicting accuracy model could be obtained by supervised learning, they are frequently inapplicable compared to educational data without class labels. [28], [29]. Tomasevic et al. presented a comparative analysis of supervised machine learning approaches to solve the task of student examination and predict their performance [30].

Shapiro et al. considered three categories of supervised machine-learning techniques: similarity-based, model-based, and probabilistic approaches [31]. The similarity-based method was used to predict exam performance, which is leveraged by discovering students with similar past performances. A second approach is a model-based approach driven by estimating implicit correlation among input learning data comprising the underlying model. The supervised probabilistic method was used to fit probability distribution features and their representation methods to find students at high risk of dropping out of courses. They were also evaluated for examination performance classification and regression activities.

4. Proposed Predictive Analytics Approach

As stated earlier, asynchronous e-learning offers self-paced and self-organized learning in which learners can define their sequence of learning items. The institution can analyze the usage of learning objects to improve their predefined sequence [15]. This learning can also be coupled with learners' analytics to provide a personalized learning experience for each learner [32]. A new learner's learning profile may be matched with past learners' learning profiles, and their learning sequence can be used to provide an enhanced learning experience for the new learner [9]. The technique used for this kind of analysis is called time series analysis. As the name implies, time series analysis learns from a sequence of temporal events

to predict such sequential outcomes in the future. A time-series analysis requires a sequence of panel data to learn the sequence. Several models have been developed for performing time series analysis. These models fall into three main categories: autoregressive, moving average, and integrated models. These three classes of models have also been combined to propose hybrid models like Autoregressive Moving Average (ARMA), Autoregressive Integrated Moving Average (ARIMA), and Autoregressive Fractionally Integrated Moving Average (ARFIMA) models. An interested user may refer to several resources related to the topic [33], [34], [35], [36].

4.1. Dataset and Exploratory Data Analytics

We have used the Open University Learning Analytics dataset for this case study, a public dataset available for download from *https://analyse.kmi.open.ac.uk/open_dataset*. It consists of academic records and the students' personal information. The following data tables are available in this dataset:

1. Student Info
2. Courses
3. Student Registration
4. VLE
5. StudentVLE
6. Assessments
7. Student Assessments

We have used student info, VLE, and Student VLE tables for this case study, which are briefly described below.

The "Student Info" table contains the students' personal information. There are 12 attributes in this data table namely code_module, code_presentation, id_student, gender, region, highest_education, imd_band, age_band, num_of_prev_attempts, studied_credits, disability, and final_result. The attributes "code_module" and "code_presentation" represent a course in a module. Code_presentation consists of the year when the course is presented while appending B or J for course offerings in February and October, respectively. The rest of the attributes are obvious by their names.

The VLE table contains information about items in the virtual learning environment. The attributes in this table are id_site, code_module, code_presentation, activity_type, week_from, and week_to. While the other characteristics are apparent, activity_type needs a little more elaboration. It is used to categorize course material into one of 20 activities such as homepage, subpage, content, resource, forum, HTML activity, or external quiz.

The StudentVLE table is our main table, with over ten million records. It stores information about student interaction with the items in the virtual learning environment. The table contains code_module, code_presentation, id_student, id_site, date and sum_click attributes. The id_site attribute is the unique ID for every VLE item, while sum_click represents how many times a student accessed a given item.

4.2. Data pre-processing

The following pre-processing prepares the dataset for the time series analysis task.

First of all, the three tables are merged into one table. StudentVLE is considered to be the master table. The information about students and VLE is extracted from respective tables and added to this master table. Every student's sequence in which they interacted with the items is preserved using the data attribute in the StudenVLE table.

Data can be prepared for time series analysis either in long or wide format. Long format holds one item accessed by a student in one record, while wide format appends all items accessed by a student one after the other in a single record. We have used a long format as several student interactions are not uniform for all students.

Close observation of the data reveals that the range of values for different attributes is not uniform. This may have the undesirable consequence of a variable with large values dictating a learning algorithm's output. All variables have been normalized to overcome this issue.

Finally, the number of records was reduced due to a limitation of the tool for calculating the Silhouette coefficient in clustering.

4.3. Model Building

Figure 1 below shows the model used to perform a time-series analysis on the sequence of learning items in the virtual learning environment. The workflow of the model is described below:

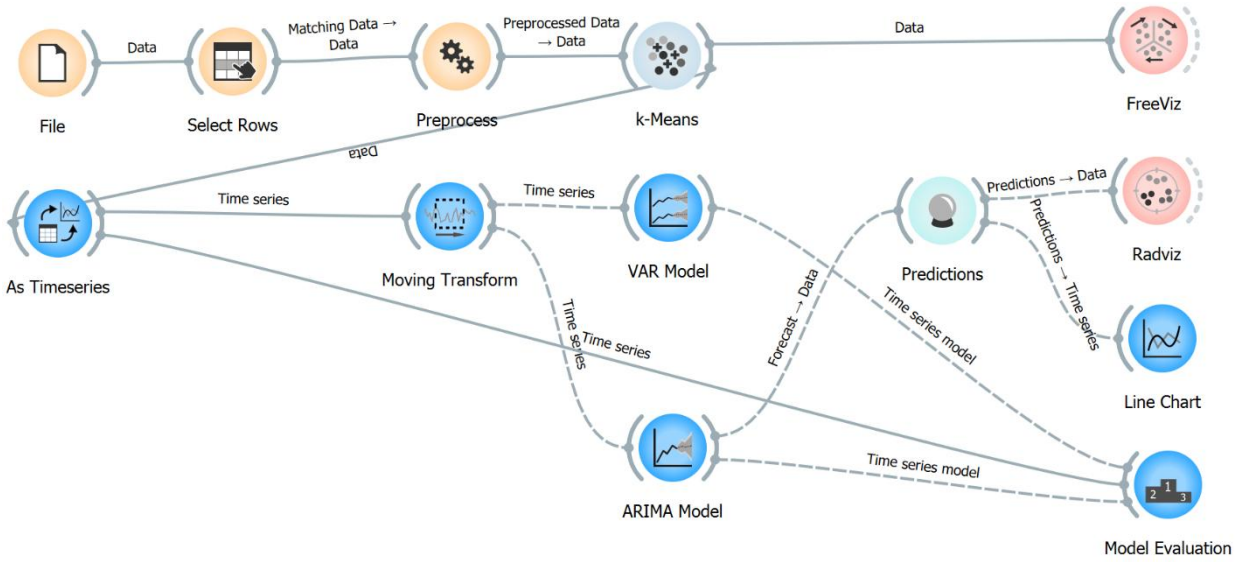


Figure 1: Proposed framework

1. First, data is imported, and the number of records is reduced, as described earlier.
2. The pre-processing step is used to normalize the attributes as described above.
3. Clustering is used as a dimensionality reduction technique, and the clusters are used to improve the performance of the time series process. The algorithm used for clustering is the k-mean clustering algorithm.
4. The output of clustering is visualized using the FreeViz chart.
5. Data, now in the form of clusters, is passed on to the time series process.
6. "Moving transform" is used to perform aggregation operations by applying rolling window functions.
7. The transformed data is ready for time series analysis. This data is fed into two time-series models: the Vector Autoregression Model (VAR) and the Auto-Regressive Integrated Moving Average (ARIMA) model.
8. The predictions of both models are visualized using RadViz and line charts.
9. Model performances are compared, and the results are exported in the final step.

5. Results and discussion

As stated above, clustering is used as a dimensionality reduction process. The k-means clustering algorithm is used to form data clusters. Figure 2 presents the output of the clustering process.

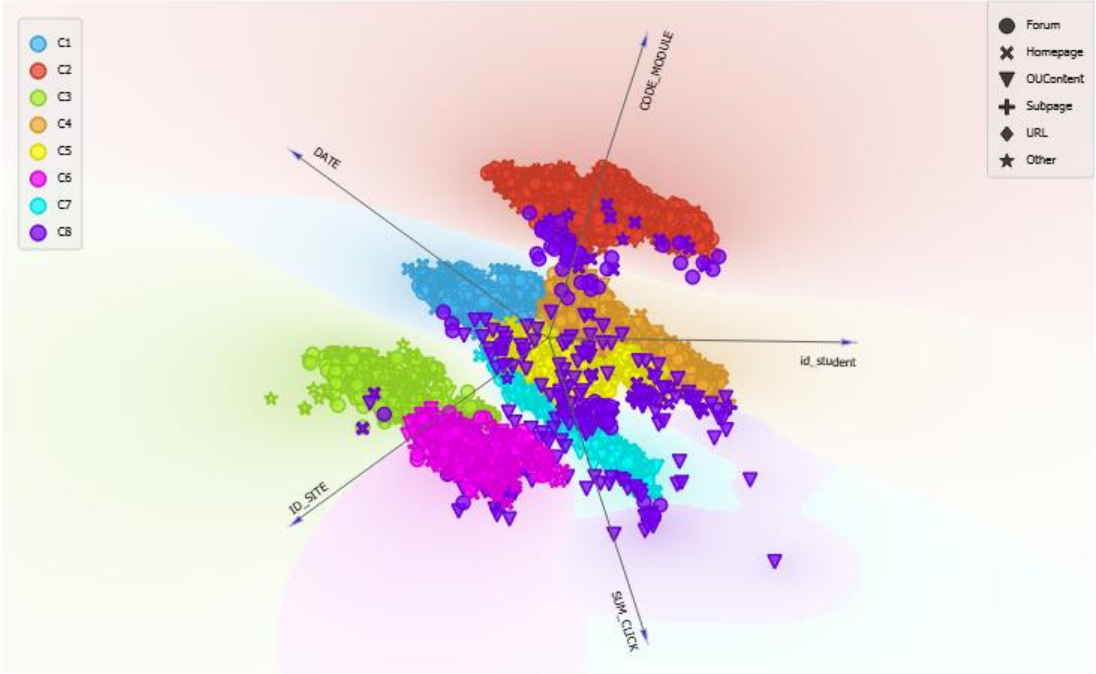


Figure 2: Data set divided into eight clusters

A total of eight clusters are produced. Further cluster analysis reveals a substantial similarity between URL and OUContent activity types, which suggests that these activity types share similar characteristics. A justification for this high similarity is the use of descriptive URL identifiers for content. Similarly, there is a high overlap between Forum and OUContent because of the discussion of topics on the forum.

The sequence predictions produced by the VAR model can be visualized in Figures 3 and 4. As "id_site" has been used as the sequential attribute, and this attribute's values are very close to each other, the sequential output also looks like a cluster. An exploded version of the chart may provide better visualization. It can also be noted in Figure 4 that the predictions converge after some time.

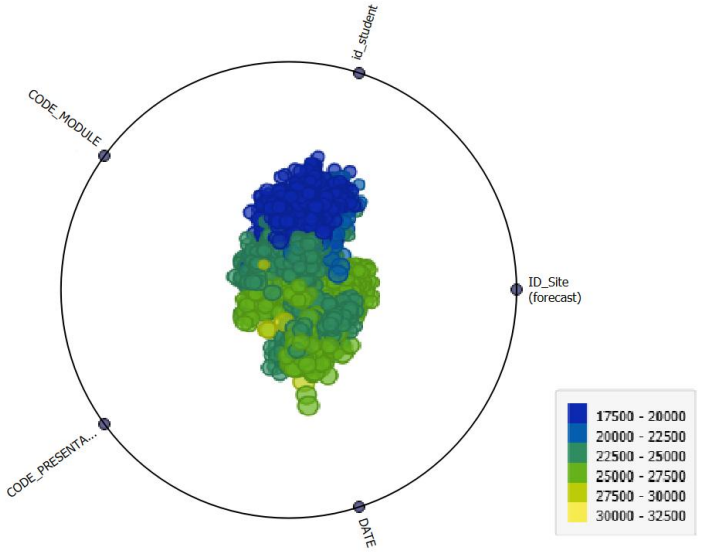


Figure 3: A visualization of predictions by the VAR model

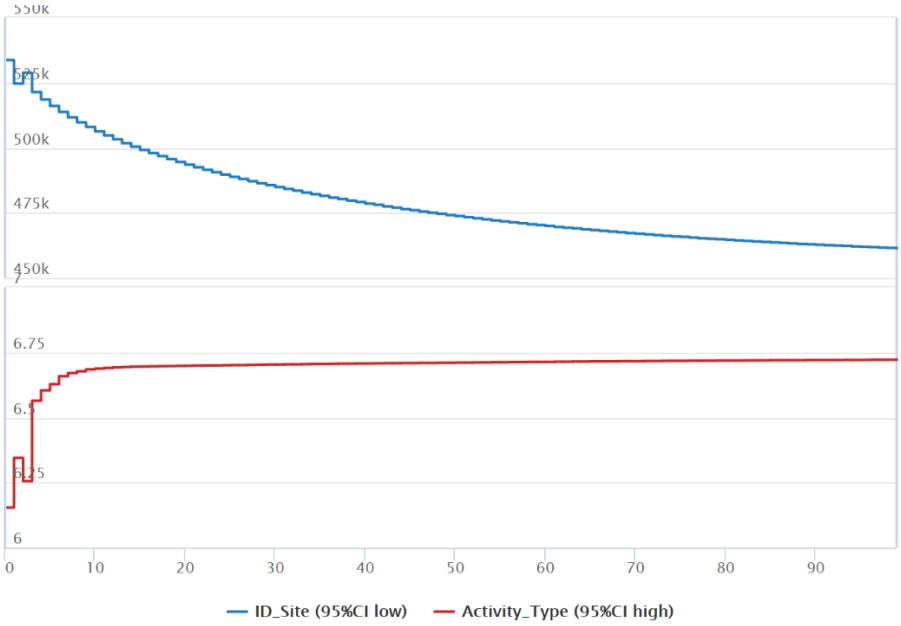


Figure 4: Step line charts showing predictions for ID_Site and Activity type with 95% confidence interval

A comparison of the VAR and ARIMA models is presented in Table 1. The evaluation measure used for comparison includes Root Mean Squared Error (RMSE), Median Absolute Error (MAE), Mean Absolute Percent Error (MAPE), Prediction of Change in Direction (POCID), Coefficient of Determination (R^2), Akaike Information Criterion (AIC), and Bayesian Information Criterion (BIC). As shown in the results, the performance of VAR and ARIMA is comparable for some measures, while each model achieved better results for some metrics and performed poorly for others. The ARIMA model outperformed the VAR model in terms of RMSE (0.120 vs 0.163) and MAPE (0.010 vs 0.019). VAR achieved slightly better MAE with 0.008 compared to 0.009 for ARIMA. The VAR model also achieved slightly better performance regarding POCID (28.4 vs 27.1), indicating slightly better performance in capturing directional changes for model features. VAR also achieved significantly better performance for R^2 (-0.012 vs. -2.984), which shows that the VAR model explains a larger portion of the variance in data compared to the ARIMA model. The ARIMA model achieved a very low score for AIC (-3499.2) compared to the VAR model (-108.8), showing a better-fit model that balances goodness of fit and complexity. The ARIMA model also outperformed the VAR for BIC (-3435.1 vs. -108.3), indicating a better-fit model.

Table 1: Results of sequence prediction by VAR and ARIMA models

Model	RMSE	MAE	MAPE	POCID	R^2	AIC	BIC
VAR	0.163	0.008	0.019	28.4	-0.012	-108.8	-108.3
ARIMA	0.120	0.009	0.010	27.1	-2.984	-3499.2	-3435.1

6. Conclusion

In today's age of personalized e-services, it is natural to offer e-learning services to learners according to their specific needs. One possible way of personalizing e-learning systems is to recommend a sequence of learning items. This study presents a case study to perform a time-series analysis of previous learners' learning object sequences to recommend a personalized arrangement to a new learner. Institutions can use it to provide a better quality of service, an improved learning experience, and a higher satisfaction rate among students. One may think of further enhancing the proposed framework by customizing the learning

content. Other possible extensions include personalized tests, personalized assignments, and course recommendations.

References

[1] UNESCO, "United Nations Decade of Education for Sustainable Development (2005-2014): International Implementation Scheme," *Sustainable Development*, 2005.

[2] Andersen, Kristian G., Andrew Rambaut, W. Ian Lipkin, Edward C. Holmes, and Robert F. Garry. "The proximal origin of SARS-CoV-2." *Nature medicine* 26, no. 4 (2020): 450-452.

[3] UNESCO, "COVID-19 Educational Disruption and Response," *Unesco.Org*, 2020.

[4] C. for S. N. CoSN, "COVID-19 Response : Preparing to Take School Online," *(Consortium for School Networking)*., no. March, 2020.

[5] G. Tam and D. El-Azar, "3 Ways the Coronavirus Pandemic Could Reshape Education," *World Economic Forum*, 2020.

[6] Halkiopoulos, Constantinos, and Evgenia Gkintoni. "Leveraging AI in e-learning: Personalized learning and adaptive assessment through cognitive neuropsychology—A systematic analysis." *Electronics* 13, no. 18 (2024): 3762.

[7] Syngene Research LLP, "Global E-Learning Market Analysis 2019," 2019.

[8] Gligorea, Ilie, Marius Cioca, Romana Oancea, Andra-Teodora Gorski, Hortensia Gorski, and Paul Tudorache. "Adaptive learning using artificial intelligence in e-learning: a literature review." *Education Sciences* 13, no. 12 (2023): 1216.

[9] Ozyurt, Ozcan, Hacer Ozyurt, and Deepti Mishra. "Uncovering the educational data mining landscape and future perspective: A comprehensive analysis." *Ieee Access* 11 (2023): 120192-120208.

[10] Dada, Michael Ayorinde, Johnson Sunday Oliha, Michael Tega Majemite, Alexander Obaigbena, and Preye Winston Biu. "A review of predictive analytics in the exploration and management of us geological resources." *Engineering Science & Technology Journal* 5, no. 2 (2024): 313-337.

[11] Garett, Renee, and Sean D. Young. "The role of artificial intelligence and predictive analytics in social audio and broader behavioral research." *Decision Analytics Journal* 6 (2023): 100187.

[12] Dehankar, Pooja, A. Amudha, S. Jayasudha, R. Pallavi, Devendra Kumar Doda, and Nelson Mandela. "Predictive Analytics Powered by Artificial Intelligence." In *2023 2nd International Conference on Futuristic Technologies (INCOFT)*, pp. 1-5. IEEE, 2023.

[13] Qadadeh, Wafa, and Sherief Abdallah. "Governmental data analytics: an agile framework development and a real world data analytics case study." *International Journal of Agile Systems and Management* 16, no. 3 (2023): 289-316.

[14] Khan, Fahad Ali, Gang Li, Anam Nawaz Khan, Qazi Waqas Khan, Myriam Hadjouni, and Hela Elmannai. "AI-Driven Counter-Terrorism: Enhancing Global Security Through Advanced Predictive Analytics." *IEEE Access* 11 (2023): 135864-135879.

[15] Bayly-Castaneda, Karla, María Soledad Ramirez-Montoya, and Adelina Morita-Alexander. "Crafting personalized learning paths with AI for lifelong learning: a systematic literature review." In *Frontiers in Education*, vol. 9, p. 1424386. Frontiers Media SA, 2024.

[16] Ayeni, Oyebola Olusola, Nancy Mohd Al Hamad, Onyebuchi Nneamaka Chisom, Blessing Osawaru, and Ololade Elizabeth Adewusi. "AI in education: A review of personalized learning and educational technology." *GSC Advanced Research and Reviews* 18, no. 2 (2024): 261-271.

[17] Stapel, Martin, Zhilin Zheng, and Niels Pinkwart. "An Ensemble Method to Predict Student Performance in an Online Math Learning Environment." *International Educational Data Mining Society* (2016).

[18] Koprinska, Irena, Joshua Stretton, and Kalina Yacef. "Predicting student performance from multiple data sources." In *Artificial Intelligence in Education: 17th International Conference, AIED 2015, Madrid, Spain, June 22-26, 2015. Proceedings 17*, pp. 678-681. Springer International Publishing, 2015.

[19] Arsad, Pauziah Mohd, and Norlida Buniyamin. "A neural network students' performance prediction model (NNSPPM)." In *2013 IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)*, pp. 1-5. IEEE, 2013.

[20] Yan, Yan, Shining Li, and Lei Feng. "Partial multi-label learning with mutual teaching." *Knowledge-Based Systems* 212 (2021): 106624.

[21] Lin, Luyue, Bo Liu, Xin Zheng, Yanshan Xiao, Zhijing Liu, and Hao Cai. "An efficient multi-label learning method with label projection." *Knowledge-Based Systems* 207 (2020): 106298.

[22] Essa, Alfred, and Hanan Ayad. "Student success system: risk analytics and data visualization using ensembles of predictive models." In *Proceedings of the 2nd international conference on learning analytics and knowledge*, pp. 158-161. 2012.

[23] González-Gómez, Francisco, Jorge Guardiola, Óscar Martín Rodríguez, and Miguel Ángel Montero Alonso. "Gender differences in e-learning satisfaction." *Computers & Education* 58, no. 1 (2012): 283-290.

[24] Berry, Michael A., and Gordon S. Linoff. "Mastering data mining: The art and science of customer relationship management." *Industrial Management & Data Systems* 100, no. 5 (2000): 245-246.

[25] Devasia, Tismy, T. P. Vinushree, and Vinayak Hegde. "Prediction of students performance using Educational Data Mining." In *2016 international conference on data mining and advanced computing (SAPIENCE)*, pp. 91-95. IEEE, 2016.

[26] Phillips, Rob. "Tools used in Learning Management Systems: analysis of WebCT usage logs." In *Proceedings of the 23rd Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education. Sydney University Press*, pp. 663-673. 2006.

[27] Hooshyar, Danial, Yeongwook Yang, Margus Pedaste, and Yueh-Min Huang. "Clustering algorithms in an educational context: An automatic comparative approach." *IEEE Access* 8 (2020): 146994-147014.

[28] Dutt, Ashish, Maizatul Akmar Ismail, and Tutut Herawan. "A systematic review on educational data mining." *Ieee Access* 5 (2017): 15991-16005.

[29] C. Anuradha, T. Velmurugan, R. Anandavally, and A. Professor, "Clustering Algorithms in Educational Data Mining: A Review…C.Anuradha et al., CLUSTERING ALGORITHMS IN EDUCATIONAL DATA MINING: A REVIEW," *International Journal of Power Control and Computation(IJPCSC)*, 2015.

[30] Tomasevic, Nikola, Nikola Gvozdenovic, and Sanja Vranes. "An overview and comparison of supervised data mining techniques for student exam performance prediction." *Computers & education* 143 (2020): 103676.

[31] Shapiro, Heather B., Clara H. Lee, Noelle E. Wyman Roth, Kun Li, Mine Çetinkaya-Rundel, and Dorian A. Canelas. "Understanding the massive open online course (MOOC) student experience: An examination of attitudes, motivations, and barriers." *Computers & Education* 110 (2017): 35-50.

[32] Sahu, Sourav, Neelamadhab Padhy, Satyam Mohapatra, Amrutansu Patra, Anurag Kumar, and Rajiv Kumar Choudhary. "Educational Data Mining for Personalized Learning: A Sentiment Analysis and Process Control Perspective." In *Proceedings*, vol. 105, no. 1, p. 77. MDPI, 2024.

[33] Kirchgässner, Gebhard, Jürgen Wolters, and Uwe Hassler. *Introduction to modern time series analysis*. Springer Science & Business Media, 2012.

[34] Wong, Chun Shan, and Wai Keung Li. "On a mixture autoregressive model." *Journal of the Royal Statistical Society Series B: Statistical Methodology* 62, no. 1 (2000): 95-115.

[35] Weiß, Christian H. *An introduction to discrete-valued time series*. John Wiley & Sons, 2018.

[36] Koosha, Mohaddeseh, Ghazaleh Khodabandelou, and Mohammad Mehdi Ebadzadeh. "A hierarchical estimation of multi-modal distribution programming for regression problems." *Knowledge-Based Systems* 260 (2023): 110129.