

# Triplet-Loss–Driven Contrastive Autoencoding for Anomaly-Based Handover Prediction Using Drive-Test Data in Dynamic Cellular Environment

M M Sadman Shafi\*, K M Istiaque, Tasnia Siddiqua Ahona, Shafayet Sadik Sowad, Mohammad T. Kawser

Affiliations: Department of Electrical and Electronic Engineering, Islamic University of Technology, Board Bazar, Gazipur 1704, Bangladesh.

Corresponding Author's Email Address: [sadmanshafi@iut-dhaka.edu](mailto:sadmanshafi@iut-dhaka.edu)

## Abstract

This paper presents a Triplet-Loss–driven Contrastive Autoencoder (CAE) framework for anomaly-based handover prediction in dynamic LTE networks using real-world drive-test data. High-resolution measurements were collected along a 13 km urban route in Dhaka, Bangladesh, capturing serving and neighbor cell parameters such as RSRP, RSRQ, and CINR under diverse mobility and interference conditions. These measurements formed the empirical foundation for training the CAE, which employs triplet loss to structure the latent space through anchor–positive–negative relationships. This organization enforces discriminative separability between stable link states and handover-triggering anomalies, enabling robust generalization across dynamic radio environments. Reconstruction deviation from the learned latent manifold serves as an intrinsic indicator of handover likelihood, refined through statistical and hybrid thresholding strategies. The proposed CAE consistently surpasses state-of-the-art supervised classifiers—including Logistic Regression, Random Forest, Gradient Boosting, CatBoost, Artificial Neural Networks and conventional Autoencoders. Unlike these models, which rely on balanced data and manually tuned thresholds, the CAE inherently adapts to class skewness and network variability through representation learning. Beyond numerical gains, the framework introduces a novel and interpretable paradigm for handover prediction—robust to label scarcity, delayed signaling and parameter drift—while offering low inference latency and practical deployability for proactive mobility management in 4G and emerging 5G systems.

**Keywords:** Handover Prediction; Contrastive Autoencoder; Triplet Loss; LTE Drive Test Data; Anomaly Detection; Representation Learning; Proactive Mobility Management; Quality of Service (QoS) Optimization

## 1. Introduction

Long Term Evolution (LTE) represents the global standard for fourth-generation (4G) cellular systems, designed to support mobile broadband applications with significantly higher data rates and lower latency compared to previous generations. It enables advanced services such as Voice over LTE (VoLTE) and enhanced Multimedia Broadcast Multicast Services (eMBMS), which extend the capabilities of traditional cellular networks [1]. The LTE architecture follows an all-IP design and is primarily composed of two components: the Evolved Universal Terrestrial Radio Access Network (E-UTRAN) and the Evolved Packet Core (EPC) [2]. Within the E-UTRAN, evolved NodeBs (eNBs) manage radio communications with User Equipments (UEs), while the EPC is responsible for mobility management, authentication, and Quality of Service (QoS) provisioning [3]. This integrated structure allows LTE networks to deliver ubiquitous coverage, high reliability, and efficient spectrum utilization, making them suitable for large-scale deployments with diverse user demands. By ensuring seamless coordination between the E-UTRAN and EPC, LTE achieves superior performance in terms of throughput, latency, and service continuity, thereby meeting the increasing requirements of data-intensive applications and real-time communication services.

Handover is a fundamental mechanism in LTE that enables a UE to maintain uninterrupted connectivity while moving across the coverage areas of different eNBs. It is essential for seamless mobility and stable communication, especially in dense network deployments designed to meet the growing demand for high-capacity and reliable services [4]. However, network densification introduces new challenges for traditional handover methods. Shorter sojourn times in small cells increase the frequency of handovers, leading to higher signaling overhead, inefficient resource usage, and potential service disruptions. A particularly problematic scenario is the ping-pong effect, where a UE oscillates repeatedly between neighboring cells within a short time, often triggered by rapid fluctuations in signal strength in overlapping areas. Ping-pong handovers not only waste network resources but also severely degrade QoS by causing packet loss, increased latency, reduced throughput, and even handover failures (HOFs) or radio link failures (RLFs).

The handover process can be conceptually divided into three phases: information gathering, decision making, and execution [5]. In the information gathering phase, relevant network and UE data are collected to evaluate the need for handover. The decision-making phase determines whether a handover should occur and identifies the most suitable target cell, taking into account factors such as signal quality, user mobility, and service requirements. Finally, the execution phase carries out the handover according to the decisions made. Optimizing the decision-making phase is particularly critical, as premature or inappropriate handover choices can trigger unnecessary transitions and ping-pong events, directly impacting QoS. Therefore, effective handover management that balances mobility, resource efficiency, and service stability is essential for maintaining high-quality user experiences in modern LTE networks [6, 7].

The growing complexity of LTE networks and the increasing demand for seamless connectivity have made predictive modeling an important tool for proactive QoS management. Unlike conventional reactive approaches, predictive modeling can anticipate user mobility patterns and network dynamics by leveraging historical and real-time data. This enables more informed handover decisions, reducing the occurrence of unnecessary and ping-pong handovers that degrade performance. By minimizing disruptions, optimizing resource utilization, and ensuring more stable connections, predictive models can significantly enhance QoS in terms of throughput, latency, and reliability, thereby improving the overall user experience.

The key contributions of this work are:

- Our work is the first to integrate a Contrastive Autoencoder with triplet loss for handover prediction, enabling the model to learn highly discriminative and semantically structured latent representations, thereby improving the identification of handover events.
- We address the limitations of conventional predictive approaches, which often suffer from noisy or delayed labels, severe class imbalance, limited generalization, and reliance on hand-crafted features. By leveraging contrastive learning with triplet-based representation, the proposed framework enhances robustness, adaptability, and scalability, effectively capturing anomalies under diverse network conditions.
- To validate the framework, extensive drive tests were conducted in a densely deployed urban LTE environment, capturing high-resolution real-world measurements, including mobility patterns, interference, and environmental impairments.
- Finally, we show that the proposed framework outperforms state-of-the-art models in handover prediction, achieving higher accuracy, F1 score, and reliability, offering a practical solution for proactive mobility management and QoS assurance.

The rest of the paper is organized as follows: Section 2 provides an overview of prior work in handover prediction and related methods. Section 3 discusses the limitations of the conventional methodologies and the motivation behind this study. Section 4 describes the drive test setup and data collection methodology. Section 5 details the proposed Contrastive Autoencoder framework and its implementation. Section 6 presents the evaluation results and comparative analysis of the framework. Finally, Section 7 concludes the paper and outlines avenues for future investigation.

## **2. Literature Review**

To address the challenges of maintaining seamless connectivity and ensuring robust system performance, researchers have explored a variety of predictive and analytical approaches. These efforts include methods for anticipating handover events, optimizing mobility management, and mitigating the negative effects of rapid transitions, as well as techniques for extracting meaningful patterns from complex data to improve decision-making. Studies have leveraged field measurements, simulations, and advanced feature representation strategies to better understand network dynamics and system behavior. Such approaches provide a foundation for designing models that enhance both operational efficiency and overall service quality across diverse network and data-driven contexts.

### **2.1 Handover prediction and management**

An ensemble learning-based handover prediction framework was developed in [8] for underwater wireless sensor networks (UWSNs), using simulated mobility data derived from real marine current measurements. Multiple machine learning models were evaluated on the collected handover events, and the ensemble learning approach was found to outperform traditional classifiers such as Gaussian Naïve Bayes and K-Nearest Neighbors. In terrestrial cellular networks, drive tests were conducted to evaluate RF parameters and assess handover performance using field-collected log data [9]. Dahouda et al. in [10] proposed a machine learning-based framework for non-terrestrial networks to reduce signaling storms during satellite handovers. An LTE handover mechanism combining direction prediction and adaptive time-to-trigger (TTT) reduced unnecessary handovers and handover trigger failures in [11]. Supervised learning approaches were applied in next-generation and O-RAN networks to predict handover events using real network measurements, enabling anticipatory service adaptations and operational cost reduction [12, 13]. The authors in [14] developed a Q-learning-based handover optimization framework that integrates subtractive clustering and real-world LTE drive test data to reduce unnecessary and ping-pong handovers. In [15], adaptive handover optimization in LTE was evaluated using fuzzy logic and Deep Q-Network (DQN) models. Furthermore, Shafi

et al. in [16] introduced a contextual bandit reinforcement learning framework for LTE handover optimization using real-world data to improve link stability.

## **2.2 Artificial Neural Networks**

In various communication and networking contexts, Artificial Neural Networks (ANNs) have been explored for tasks such as signal equalization, mobility prediction, and handover decision-making. For instance, ANN has been applied in underwater wireless optical communication to mitigate nonlinear impairments and enhance data rates over long distances in [17]. In mobile wireless sensor networks, researchers have used ANNs to predict node mobility and manage adaptive handovers, improving network stability and efficiency [18]. Additionally, ANNs and hybrid neural models have been employed to forecast handover success rates in GSM networks and to optimize handover decisions in hybrid LiFi–WiFi systems, demonstrating their potential to improve throughput, reduce handover frequency, and maintain connectivity under varying network conditions [19, 20].

## **2.3 Autoencoders**

Deep autoencoder-based methods, including hybrid and domain-adaptation approaches, were applied for anomaly detection and medical image classification, leveraging reconstructed feature representations for improved pattern recognition in [21–23]. Authors in [24, 25] proposed multiple-input and graph-based variational autoencoders for unsupervised anomaly and outlier detection in heterogeneous and graph-structured data, improving the identification of deviations from learned normal distributions. Lu et al. in [26] developed an unsupervised locality-preserved autoencoder using complexity-invariant distance for machinery fault diagnosis, enabling discriminative feature learning from complex signals. Deep autoencoder-based methods, including deep and convolutional architectures, were proposed in [27, 28] for error mitigation and channel estimation in 5G and intelligent wireless communication systems. An autoencoder-driven semantic communication framework and a capacity-oriented coding scheme were proposed in [29, 30], where robustness against channel impairments and improved information transfer in machine-to-machine and general communication systems were demonstrated. Authors in [31, 32] proposed hybrid quantum-classical autoencoder frameworks for end-to-end communication, demonstrating feasibility, robustness against fading and noise, and reduced trainable parameters compared to classical baselines.

## **2.4 Contrastive autoencoder**

A contrastive predictive autoencoder was designed in [33] to learn self-supervised spatiotemporal representations from dynamic point cloud sequences by jointly optimizing contrastive prediction and reconstruction tasks. In [34], a contrastive autoencoder with multi-resolution segment-consistency discrimination was proposed for multivariate time series anomaly detection, enabling robust contextual representation learning through segment-wise contrastive objectives. In [35], Luo et al. combined contrastive learning with an autoencoder in an unsupervised framework for out-of-distribution detection, where contrastive learning enhanced feature separability and the autoencoder refined data representations. The authors proposed a contrastive variational autoencoder integrated with deep metric learning in [36] for attack detection in blockchain-cloud-IoT systems, effectively distinguishing normal and malicious traffic through shared and private feature generation. A contrastive loss-enhanced autoencoder framework was developed for unsupervised outlier detection in IoT networks by Nikougoftar [37], integrating reconstruction and contrastive objectives to distinguish normal and anomalous data. The authors of [38] proposed a contrastive autoencoder for phoneme recognition that leverages class labels to learn task-specific invariant features, achieving superior performance on the TIMIT dataset compared to conventional and deep autoencoder-based approaches. An unsupervised cross-modal contrastive hashing framework was developed in [39], which combines autoencoder-based semantic similarity and a multi-objective contrastive loss to enhance retrieval accuracy between remote sensing images and text. Zhang *et al.* [40] introduced C-MELT, a contrastive-enhanced

masked autoencoder framework for ECG-language pretraining, which effectively integrates physiological and textual modalities to achieve state-of-the-art cross-modal representation and diagnostic performance. A contrastive autoencoder-based framework was proposed in [41] for weather-aware RLF prediction, combining multi-modal feature fusion and survival analysis.

Although autoencoders and contrastive autoencoders have been applied for anomaly detection and representation learning, their application to handover prediction remains unexplored. In this work, we address this gap by developing a novel contrastive autoencoder framework with triplet loss that learns structured latent representations to accurately identify event-triggering conditions in dynamic LTE networks. By uncovering hidden patterns indicative of inefficient handover events, the framework enables early detection of unnecessary and ping-pong handovers, supporting proactive handover management and ultimately improving QoS by reducing failures, minimizing delays and lowering signaling overhead.

### **3. Limitations of Conventional Handover Prediction Approaches and Motivation for the Proposed Framework**

Conventional supervised learning exhibits significant limitations for handover (HO) prediction in dynamic cellular networks. These models assume the availability of accurate and timely labels, an assumption frequently violated due to noisy Radio Resource Control (RRC) signaling, logging delays, and temporal misalignments. Such label drift can cause models to learn incorrect or delayed associations. Moreover, the severe class imbalance, where HO events constitute only ~5% of samples, leads to overfitting and a strong bias toward the majority “no-HO” class, resulting in missed critical events such as Unnecessary Handovers (UHOs) or Radio Link Failures (RLFs). Supervised classifiers also tend to memorize HO-triggering patterns observed during training, limiting generalization to unseen network conditions, new cell deployments, or evolving mobility profiles. Any changes in network configurations—including Mobility Robustness Optimization (MRO) parameters, cell layouts, scheduling strategies, or fading environments—often require costly retraining. In addition, collecting large-scale, high-quality labeled datasets is expensive and error-prone, further constraining practical applicability [35].

Traditional deep learning architectures, including Artificial Neural Networks (ANNs) and Long Short-Term Memory (LSTM) networks, are similarly constrained. ANNs, as feedforward architectures, fail to capture temporal dependencies in mobility-driven KPI fluctuations (e.g., RSRP, RSRQ, CINR), producing monotonous feature representations with poor generalization across mobility and fading conditions. LSTMs incorporate recurrent gating to model temporal context, but remain highly data-driven, prone to overfitting under limited labeled data and class imbalance [22]. They are also vulnerable to catastrophic forgetting and gradient degradation over long sequences [33], and their sequential computation incurs high complexity for large-scale KPI datasets. Both ANNs and LSTMs operate as black-box models, offering limited interpretability of mobility degradation causes and require frequent retraining to adapt to changing network topologies, propagation conditions, and MRO configurations.

Conventional anomaly detection methods, such as threshold-based or statistical rule-based techniques, also face significant challenges [21]. In HO prediction, an anomaly corresponds to an event-triggering condition—a deviation from normal signal behavior that precedes HOs, UHOs, or RLFs. These approaches rely on predefined thresholds or simplistic statistical assumptions, which fail to capture the nonlinear, high-dimensional, and temporally correlated relationships among KPIs [37]. Threshold calibration is context-dependent and rarely generalizes across cells or mobility patterns, often resulting in false alarms, missed detections and limited adaptability to evolving network conditions.

To address these challenges, we propose a contrastive autoencoder-based framework with triplet loss. This approach learns discriminative and semantically structured latent representations of mobility behavior, allowing deviations from reconstructed normal states to serve as intrinsic indicators of event-triggering conditions. By eliminating reliance on fixed thresholds and focusing on representation learning, the framework enhances robustness, scalability and adaptability, effectively identifying anomalies under diverse propagation environments and network configurations.

A key aspect of this study is grounding the framework in real-world network measurements. Extensive drive tests were conducted to capture irregular mobility patterns, interference, and weather-induced propagation impairments, ensuring that the learned representations and reconstruction-based anomaly detection are validated under realistic and challenging conditions. Drive tests play a crucial role in accurately characterizing the dynamic behavior of cellular networks, providing empirical evidence of signal fluctuations, handover events, and link degradation that cannot be fully captured through simulation-based approaches [42, 43]. This confirms the practical applicability of our framework for proactive mobility management and QoS assurance.

#### **4. Empirical Data Acquisition through Drive Test Measurements**

To evaluate the performance of the proposed contrastive autoencoder for handover prediction in LTE networks, a comprehensive dataset was collected through structured drive tests in a densely deployed urban cellular environment in Dhaka, Bangladesh. This section details the experimental setup, including the equipment used, route selection, utilization of *OpenCellID* for site identification, time period of data collection, software configuration, data extraction procedures, and a description of the resulting dataset. The methodology emphasizes high-resolution capture of real-world radio conditions to support machine learning-driven mobility studies, ensuring the data's suitability for modeling handover dynamics under varying traffic and velocity scenarios.

##### **4.1 Equipment**

The drive test employed professional-grade tools to capture granular LTE measurements. The core setup included XCAL-M software (developed by Accuver) for logging radio frequency (RF) parameters and Layer 3 signaling events. This was interfaced with a Samsung Galaxy S10 smartphone equipped with an Exynos System-on-Chip (SoC) chipset, enabling diagnostic mode access to chipset-level key performance indicators (KPIs). A laptop hosted the XCAL-M application and facilitated real-time data acquisition.

To augment location and mobility tracking, an Arduino Uno microcontroller was integrated with a Neo-7M GPS module. This auxiliary system provided precise geolocation (latitude and longitude) and user equipment (UE) velocity data, synchronized with the RF measurements. The entire apparatus was mounted in a private vehicle to simulate realistic urban mobility. Compared to consumer-grade Android applications like G-NetTrack Pro, which are limited to one-second sampling intervals due to operating system constraints [44], XCAL-M offers millisecond-level granularity, making it ideal for capturing transient handover events and radio fluctuations.

The schematic diagram of the drive test setup is illustrated in Fig. 1, highlighting the interconnection between hardware components for seamless data collection.

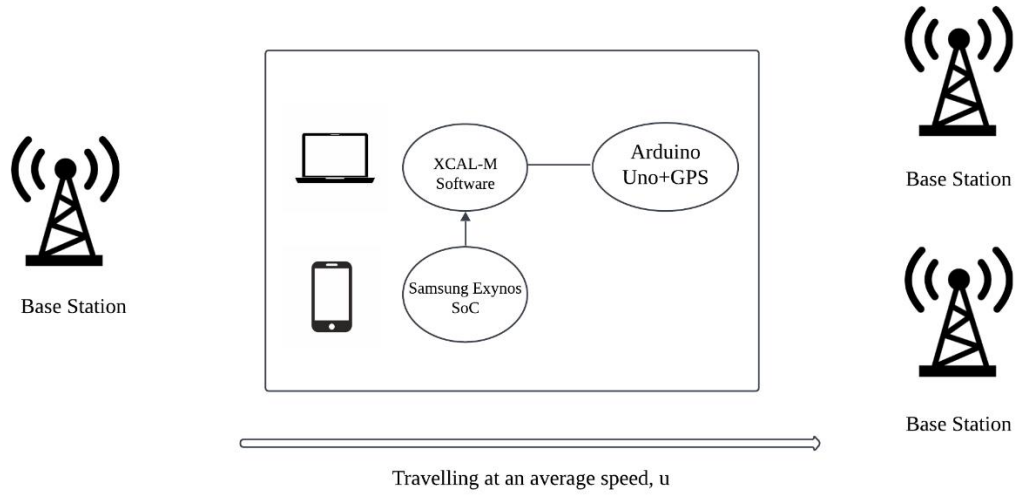


Fig.1. Schematic Diagram of Drive Test.

#### 4.2 Route

The drive test route was carefully selected to encompass a diverse urban environment characterized by high cell density, mixed traffic conditions, and frequent inter-cell handovers. Spanning approximately 13 km, the route commenced at Le Meridien Dhaka (Uttara) and terminated at the BRAC University campus, traversing busy corridors in Dhaka city. This path included segments with stationary periods, pedestrian-like low speeds, and vehicular movements under congestion, reflecting real-world mobility patterns in a densely populated area.

Measurements were conducted on the Grameenphone LTE network, one of Bangladesh's largest mobile network operators (MNOs). The route's design ensured repeated encounters with the same LTE cell IDs under varying conditions, enhancing temporal diversity for machine learning applications. However, due to the focus on a single MNO and specific urban/suburban zones, the results may not generalize to other operators or broader network topologies.

#### 4.3 OpenCellID Utilization

Route selection was informed by the *OpenCellID* database, a publicly available repository of cellular base station geolocations across multiple radio access technologies (RATs), including GSM, UMTS, and LTE. For this LTE-focused study, data were filtered to target eNodeBs, identifying regions with high LTE base station density to maximize handover events. Dhaka's urban landscape, with its clustered deployments, was prioritized to capture edge-of-cell conditions and mobility-induced signal variations critical for handover prediction models.

This data-driven approach ensured the route's alignment with areas prone to frequent handovers, facilitating the collection of a mobility-aware dataset. The *OpenCellID* mapping confirmed the presence of multiple overlapping cells, promoting inter-cell interactions essential for evaluating contrastive learning in autoencoder-based prediction. The cell ID database is visualized in Fig. 2.

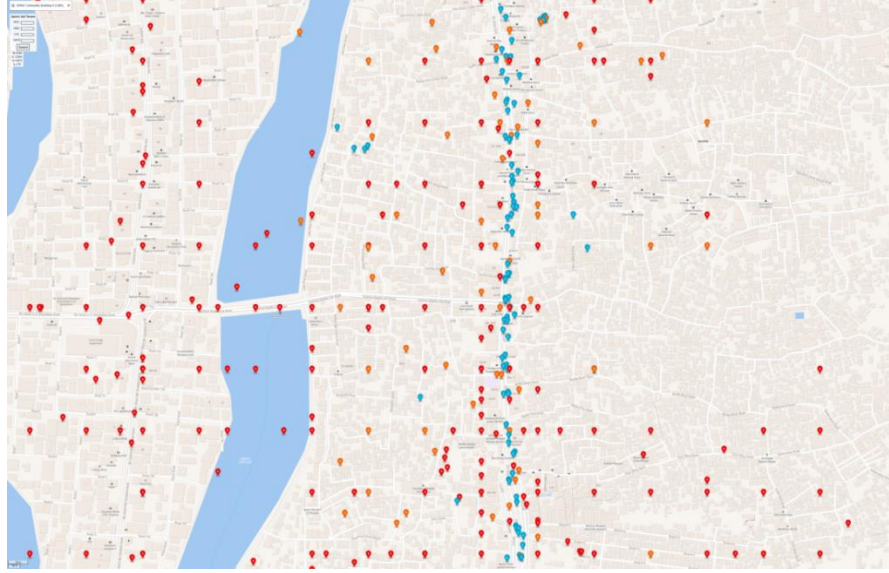


Fig. 2. Cell ID Database from *OpenCellID*.

#### 4.4 Time Period

Data collection occurred over three separate days: October 15, November 6, and November 16, during evening peak hours from 16:00 to 22:00 local time. On each day, the route was traversed in two complete rounds (outbound and return), with a 30-minute interval between trips. This timing coincided with high user density and network load, amplifying handover dynamics due to resource contention and signal interference.

Traffic congestion along the route introduced non-uniform mobility, such as stop-and-go patterns, which influenced radio measurements like reference signal received power (RSRP) fluctuations and signal-to-interference-plus-noise ratio (SINR) degradation at cell edges. No significant changes in network infrastructure (e.g., eNodeB commissioning/decommissioning) were observed during the period, attributing variations in KPIs primarily to mobility and traffic factors. Ambient conditions averaged 29°C temperature and 72% relative humidity, typical for Dhaka's post-monsoon season.

#### 4.5 Software Setup

XCAL-M was configured on the laptop and connected to the Samsung Galaxy S10 via USB in diagnostic mode, enabling the capture of Layer 1 to Layer 3 signaling, RF metrics, and protocol data. The software's high sampling rate supported detailed analysis of handover performance in dense urban LTE deployments. The Arduino-GPS module ran concurrently, logging UE position and speed for correlation with network events.

Unlike coarser Android-based tools, XCAL-M's millisecond precision captured subtle radio condition changes, essential for time-to-trigger (TTT) analysis in handover modeling. The vehicle-mounted setup ensured real-time synchronization of all data streams as the route was traversed.

#### 4.6 Data Extraction

Following the completion of the drive tests, network performance data were extracted using the post-processing capabilities integrated within XCAL-M software. The process began with accessing handover-related key performance indicators through the "Layer3 KPI" module, specifically by selecting "LTE" and navigating to the "LTE Handover Event Statistics" window, as illustrated in Fig. 3. This step provided detailed metrics, including the number of handover

LTE Handover Event Statistics									
Re1		Total							
Clear		<input type="checkbox"/> Show Chipset Time							
Time	Event	Freq	Base	Result	Attempt	Success	Fail	Count	
21:05:32.769	Intra LTE-HO	(-/-) to (1600/325)	X2	Attempt	1	0	0		
21:05:32.769	Intra LTE-HO	(-/-) to (1600/325)	X2	Success	0	1	0		
21:05:42.751	Idle Mode Load Balancing	EUTRA(275/7) EUTRA(40688/6		Attempt	1	0	0		
21:08:08.143	Idle Mode Load Balancing	EUTRA(275/7) EUTRA(40292/6		Attempt	1	0	0		
21:09:10.132	Idle Mode Load Balancing	EUTRA(275/7) EUTRA(40292/6		Attempt	1	0	0		
21:10:28.298	Idle Mode Load Balancing	EUTRA(275/7) EUTRA(40292/6		Attempt	1	0	0		
21:11:54.217	Intra LTE-HO	(-/-) to (275/33)	X2	Attempt	1	0	0		
21:11:54.217	Intra LTE-HO	(-/-) to (275/33)	X2	Success	0	1	0		
21:12:04.415	Idle Mode Load Balancing	EUTRA(1600/7) EUTRA(40292/		Attempt	1	0	0		
21:14:18.986	Intra LTE-HO	(-/-) to (1600/34)	eNB	Attempt	1	0	0		
21:14:18.987	Intra LTE-HO	(-/-) to (1600/34)	eNB	Success	0	1	0		
21:14:20.174	Intra LTE-HO	(-/-) to (1600/33)	eNB	Attempt	1	0	0		
21:14:20.276	Intra LTE-HO	(-/-) to (1600/33)	eNB	Success	0	1	0		
21:14:30.323	Idle Mode Load Balancing	EUTRA(1600/7) EUTRA(40292/		Attempt	1	0	0		
21:16:07.588	Idle Mode Load Balancing	EUTRA(1600/7) EUTRA(40292/		Attempt	1	0	0		
21:16:55.885	Idle Mode Load Balancing	EUTRA(1600/7) EUTRA(40292/		Attempt	1	0	0		
Time	Event	Freq	Base	Result	Duration(ms)				
21:11:53.617	LTE Measurement Report		A5						
21:11:53.916	LTE Measurement Report		A5						
21:11:53.916	LTE Measurement Report		A5						
21:11:53.916	LTE Measurement Report		A5						
21:11:54.116	LTE Measurement Report		A3						
21:11:54.116	LTE Measurement Report		A5						
21:11:54.116	LTE Measurement Report		A5						
21:11:54.116	LTE Measurement Report		A5	Trigger					
21:11:54.316	LTE Measurement Report		Unknown						
21:11:54.217	Intra LTE-HO	(-/-) to (275/33)	X2	Attempt					
21:11:54.217	Intra LTE-HO	(-/-) to (275/33)	X2	Success	6				
21:11:54.917	LTE Measurement Report		A1						
21:11:54.917	LTE Measurement Report		A1						
21:11:55.017	LTE Measurement Report		A4						

Next, serving cell parameters were retrieved by utilizing the "Call Statistics" option, selecting the "LTE Summary Table," and filtering to the "Serving Cell" view. This yielded essential identifiers such as the physical cell ID and carrier-to-interference-plus-noise ratio.

Signaling Message
Re1

Message Filter: None
 Filtering
Filtering 2
Pause
Export

Hex
Vertical
Clear
Fed

String Color Setting
 Detail
Show Capture Time

Show SP

Time	UE-MET	Channel	ID	Message
20:52:37.892	DL	PCCH	LTE	LTE paging
20:52:38.493	DL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:38.494	UL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:38.693	DL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:38.693	UL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:38.893	DL	PCCH	LTE	LTE paging
20:52:39.193	UL	DCCH	LTE	LTE measurementReport
20:52:39.294	DL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:39.294	UL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:39.294	DL	PCCH	BCH	LTE MasterInformationBlock
20:52:39.295	DL	DL	DL	LTE systemInformationBlockType1
20:52:39.295	DL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:39.295	UL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:39.295	DL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:39.296	UL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:39.393	DL	DCCH	LTE	LTE measurementReport
20:52:39.393	UL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:39.593	UL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:39.792	DL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:39.793	UL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:40.093	UL	DCCH	LTE	LTE measurementReport
20:52:40.093	DL	DCCH	LTE	LTE measurementReport
20:52:40.894	UL	DCCH	LTE	LTE measurementReport
20:52:41.894	DL	PCCH	LTE	LTE paging
20:52:42.393	DL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:42.393	UL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:42.393	DL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:42.393	UL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:42.394	DL	DCCH	LTE	LTE rrcConnectReconfigurationComplete
20:52:42.394	UL	DCCH	LTE	LTE rrcConnectReconfigurationComplete

PC Timestamp [2024 Nov 16 20:52:40.894]

Msg sequence: 46817

UAS Command: 0x02DM (LTE Data Out Message)

Command Type: 0x52

Time: 323575716

DLDataLength: 12

value UL-DCCH-Message =>

message c1: measurementReport

criticalExtensions c1: measurementReport-v8:

measResults

measId 2

measResultPCelId

rsrpResult 48

rsrqResult 17

measResultHighCells measResultListEUTRA:

physCellId 235

measResult

rsrpResult 42

rsrqResult 9

3 Data in Hex

00 10 80 40 43 AD A4 24

Fig. 4. Signaling Messages from XCAL Software.

## 4.7 Data Description

The dataset comprises three processed subsets corresponding to drive tests conducted on three distinct days, ensuring temporal diversity in urban mobility patterns. These subsets, denoted as Dataset\_1, Dataset\_2, and Dataset\_3, capture high-resolution LTE measurements under varying traffic conditions, with each dataset reflecting a complete round-trip traversal of the 13 km route. The processed versions incorporate cleaning for hardware anomalies, filtering of non-LTE events, timestamp-based interpolation for missing values, and time-to-trigger (TTT) handling at 320 ms to align with handover execution logic, rendering them suitable for machine learning applications in handover prediction. The complete dataset has been made publicly available to facilitate further research [45].

Table 1 summarizes the features across the datasets, including spatiotemporal attributes (timestamp, longitude, latitude, speed in km/h), serving cell metrics (cell ID, RSRP, RSRQ, CINR variants), and neighbor cell parameters (cell ID, RSRP, RSRQ). The target variable, handover trigger, is binary-labeled based on detected serving cell ID transitions, facilitating supervised modeling of mobility events.

**Table 1**

Dataset Columns

Features	Target Variable
Timestamp	Handover Trigger
Longitude	
Latitude	
Speed (km/h)	
Serving Cell ID	
Serving Cell RSRP	
Serving Cell RSRQ	
Serving Cell CINR-1	
Serving Cell CINR-2	
Serving Cell CINR	
Neighbour Cell ID	
Neighbour Cell RSRP	

The structural overview of the datasets is provided in Table 2, highlighting variations in column count and data points post-preprocessing: Dataset\_1 encompasses 11 columns and 922 points; Dataset\_2 includes 14 columns and 592 points; and Dataset\_3 features 14 columns and 641 points. These dimensions reflect the progressive refinement across sessions, with Dataset\_1 derived from the initial test lacking certain neighbor metrics.

**Table 2**

Structural overview of the datasets

Dataset Names	Number of Columns	Data Points
Dataset_1	11	922
Dataset_2	14	592
Dataset_3	14	641

Handover instances relative to traveled distance are detailed in Table 3, quantifying mobility dynamics: Day 1 recorded 129 handovers over 26.21 km (203 m per handover); Day 2 captured 95 handovers across 25.54 km (268 m per handover); and Day 3 documented 107 handovers in 27.28 km (254 m per handover). This granularity underscores the dataset's utility for velocity-aware analysis.

**Table 3**

Handover Instances with respect to travelled distance

Time Period	Handover Count	Total Distance Travelled (km)	Distance Travelled per Handover in meter
Day 1	129	26.21	203
Day 2	95	25.54	268
Day 3	107	27.28	254

Figure 5 illustrates the distribution of handover and non-handover events across the three datasets. As observed, Dataset\_1, Dataset\_2, and Dataset\_3 contain only 14%, 16%, and 16.7% HO events, respectively, highlighting their highly imbalanced nature. This pronounced class imbalance poses significant challenges for conventional predictive models, which may become biased toward the majority non-handover class. Consequently, specialized approaches are required to accurately identify the minority HO events and ensure robust prediction performance across all datasets.

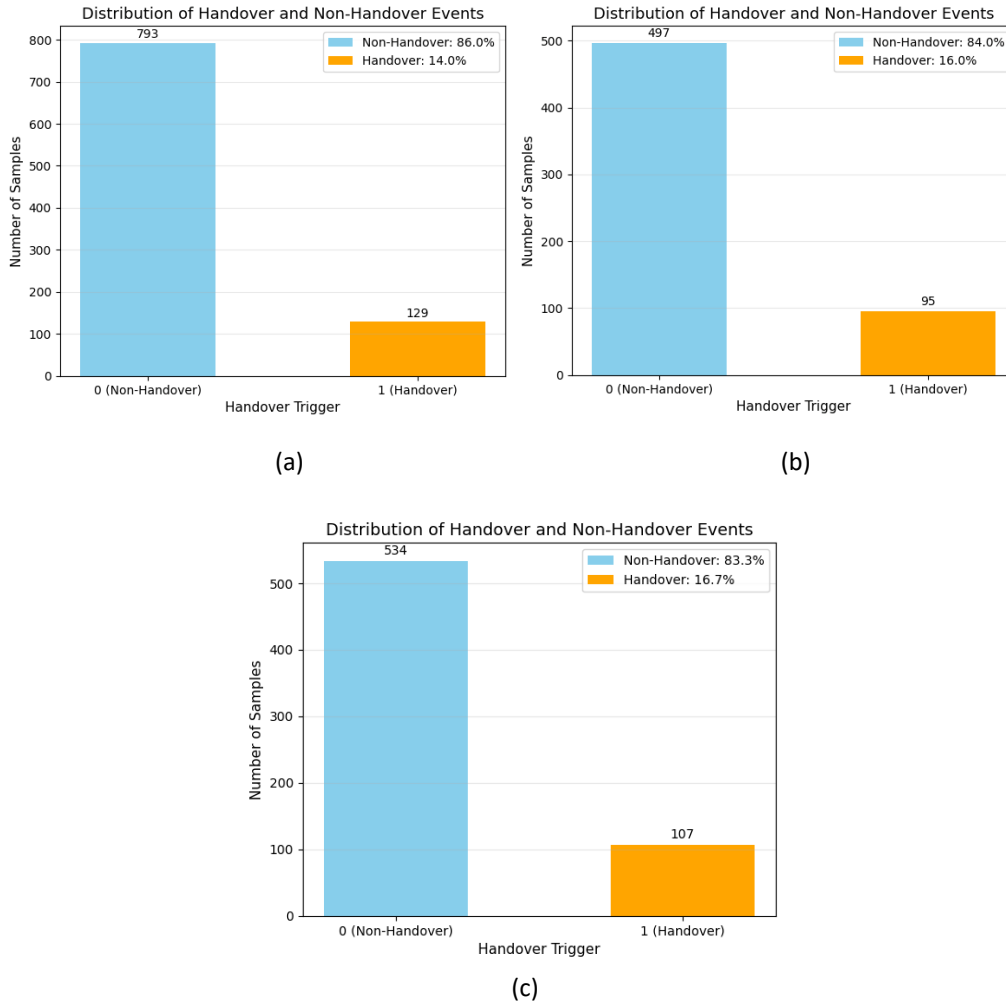


Fig. 5. Distribution of Handover and Non-Handover Events of (a) Dataset\_1, (b) Dataset\_2, (c) Dataset\_3

Parameter distributions for Dataset\_2, illustrating the Gaussian-like profiles of key metrics such as serving cell RSRP, RSRQ, and speed, are depicted in the bell curves of Fig. 6, revealing mean-centered signal strengths typical of dense urban LTE deployments and informing feature normalization strategies for the contrastive autoencoder.

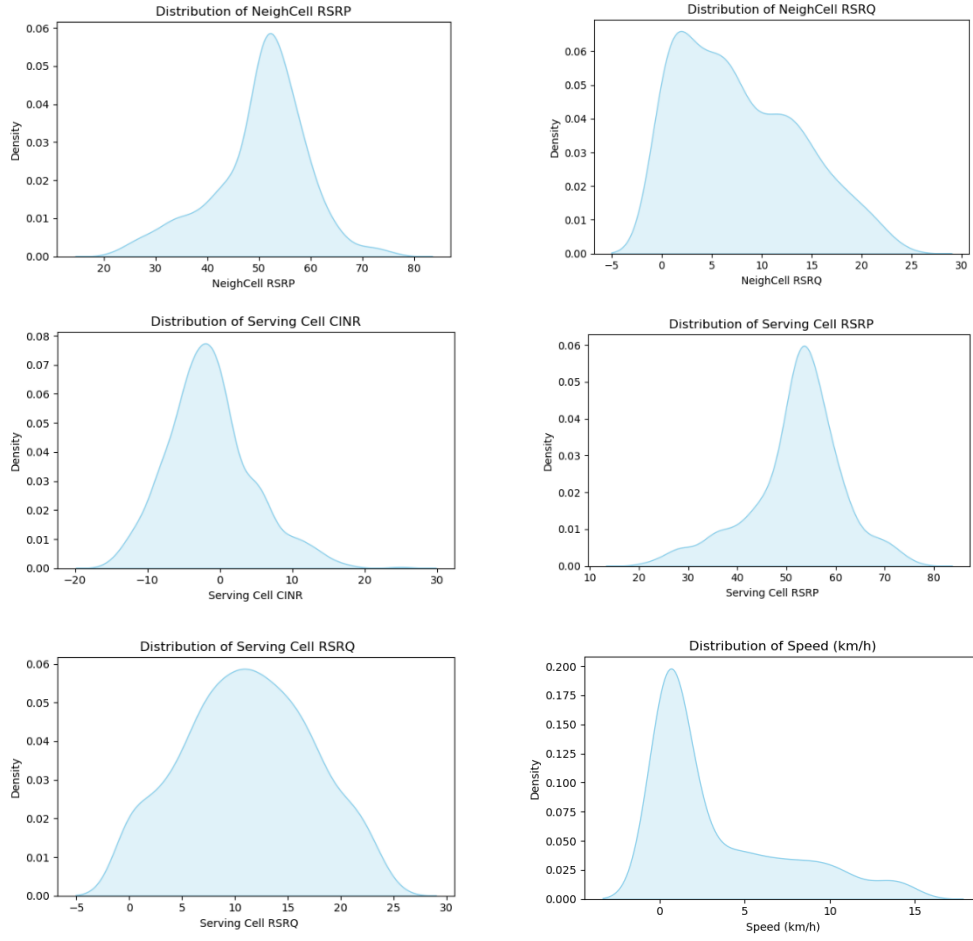


Fig. 6. Distribution of network parameters of Dataset\_2

## 5. Methodology and Implementation

This section details the methodology used to develop and evaluate the proposed Contrastive Autoencoder (CAE) framework for handover prediction. It begins with an overview of the conventional Autoencoder, describing its encoder–decoder architecture, training objective, and anomaly scoring mechanism. The Autoencoder formulation for handover prediction is then presented, followed by the proposed CAE design, which incorporates contrastive representation learning with triplet loss to enhance feature discrimination. Finally, the training and testing procedures, including the encoder–decoder structure and latent space representation, are discussed.

### 5.1 Autoencoder Framework

Autoencoders (AEs) are a class of neural network architectures designed to learn compressed representations of input data through unsupervised learning. The model aims to capture the intrinsic structure of the data distribution by encoding the input into a latent feature space and subsequently reconstructing it with minimal loss of information. Owing to their reconstruction capability, AEs are widely employed in representation learning, noise reduction, and anomaly detection tasks, where deviations in reconstruction error serve as indicators of abnormal patterns.

An AE consists of two parametric components: an Encoder and a Decoder.

Let the input observation be denoted as  $x^{(i)} \in \mathbb{R}^d$  where  $d$  represents the input dimensionality and  $i=1, 2, \dots, N$  indexes the total  $N$  training samples.

#### Encoder

The Encoder performs a non-linear transformation from the input space to a compact latent representation  $z^{(i)} \in \mathbb{R}^{d_z}$ , where  $d_z < d$ . This mapping is defined as:

$$z^{(i)} = f(x^{(i)}; \phi) = \sigma(W_e x^{(i)} + b_e) \quad (1)$$

where:

- $f(\cdot; \phi)$  is the encoding function parameterized by  $\phi = \{W_e, b_e\}$ ,
- $W_e \in \mathbb{R}^{d_z \times d}$  is the weight matrix of the Encoder,
- $b_e \in \mathbb{R}^{d_z}$  is the Encoder bias vector,
- $\sigma(\cdot)$  is a non-linear activation function such as ReLU or tanh,
- $z^{(i)}$  represents the compressed latent embedding of  $x^{(i)}$ .

#### Decoder

The Decoder reconstructs the original data from its latent representation through a parametric function  $g(\cdot; \theta)$ , expressed as:

$$\hat{x}^{(i)} = g(z^{(i)}; \theta) = \sigma(W_d z^{(i)} + b_d) \quad (2)$$

where:

- $g(\cdot; \theta)$  is the decoding function parameterized by  $\theta = \{W_d, b_d\}$ ,
- $W_d \in \mathbb{R}^{d \times d_z}$  denotes the weight matrix of the Decoder,
- $b_d \in \mathbb{R}^d$  is the bias vector of the Decoder,
- $\hat{x}^{(i)}$  is the reconstructed approximation of the original input  $x^{(i)}$ .

#### Training Objective

The AE is trained to minimize the discrepancy between the input  $x^{(i)}$  and its reconstruction  $\hat{x}^{(i)}$ . The reconstruction loss is typically computed using the mean squared error (MSE) across all samples:

$$\mathcal{L}_{AE}(X; \phi, \theta) = \frac{1}{N} \sum_{i=1}^N \|x^{(i)} - \hat{x}^{(i)}\|_2^2 \quad (3)$$

where  $\|\cdot\|_2$  denotes the Euclidean norm, and  $X = \{x^{(1)}, x^{(2)}, \dots, x^{(N)}\}$  represents the training dataset. The minimization of  $\mathcal{L}_{AE}$  enforces the Encoder–Decoder pair to learn compact yet informative latent features that effectively capture the distribution of normal data.

#### Anomaly Scoring

Once trained, the AE evaluates a new observation  $x^{(i)}$  by computing its reconstruction error, which serves as the anomaly score:

$$s_{AE}(x^{(i)}) = \frac{1}{d} \sum_{t=1}^d (x_t^{(i)} - \hat{x}_t^{(i)})^2 \quad (4)$$

Here,  $x_t^{(i)}$  and  $\hat{x}_t^{(i)}$  denote the  $t^{th}$  feature of the input and reconstructed sample, respectively. Higher values of  $s_{AE}(x^{(i)})$  indicate a greater deviation from the learned data manifold, implying a higher likelihood of anomaly.

In general, AE-based anomaly detection assumes that the majority of data samples are normal ( $n_a \ll n_b$ ), enabling the model to predominantly learn the characteristics of nominal behavior. Post-training, the top  $n_a$  samples with the largest anomaly scores  $s_{AE}(x^{(i)})$  are designated as anomalous instances. Fig. 7 shows the general structure of an autoencoder.

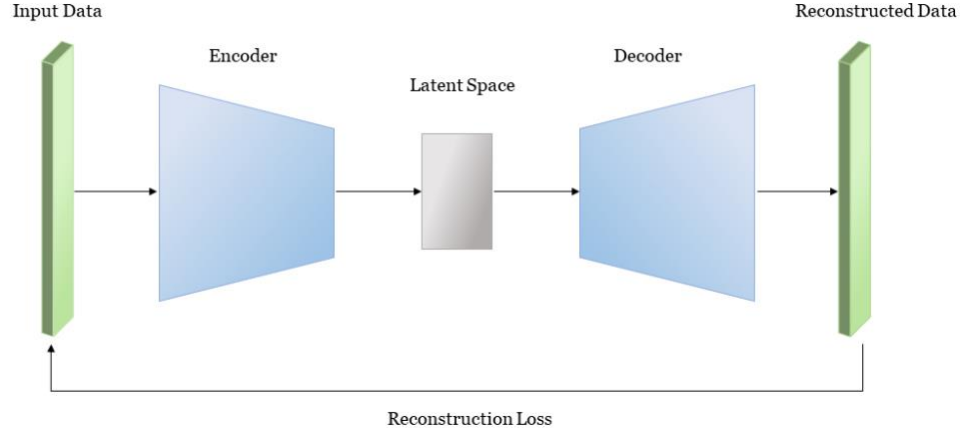


Fig. 7. General structure of an autoencoder.

## 5.2 Autoencoder Formulation for Handover Prediction

In the context of LTE networks, the handover (HO) process is initiated when the signal quality of the serving cell degrades below a predefined threshold while a neighboring cell exhibits comparatively higher quality. Such conditions are governed by event-triggering mechanisms, as standardized in 3GPP specifications. Typical events include A1–A5, where for instance, *Event A3* is triggered when the received signal strength of a neighboring cell exceeds that of the serving cell by a hysteresis margin for a specific duration. These event-triggering instances indicate potential transitions between cells and represent dynamic changes in the wireless environment.

Let the time-series data of radio measurements at a given UE location be denoted by

$$x^{(i)} = [RSRP_s, RSRQ_s, RSRP_n, RSRQ_n, CINR, \dots] \in \mathbb{R}^d \quad (5)$$

where the subscript  $s$  and  $n$  refer to serving and neighboring cells, respectively, and  $d$  represents the number of measurement parameters collected at each instant. Each sample  $x^{(i)}$  corresponds to the network state at a particular time instance.

Under normal operating conditions, the measurements exhibit stable patterns reflecting nominal connectivity with the serving cell. These are denoted as normal data points, forming the distribution  $\mathcal{X}_n = \{x_n^{(1)}, x_n^{(2)}, \dots, x_n^{(N)}\}$ . Conversely, when event-triggering conditions (e.g., A3, A5) are met, the feature vector deviates significantly due to rapid variations in received power and quality metrics. Such samples constitute anomalous data points, represented by  $\mathcal{X}_a = \{x_a^{(1)}, x_a^{(2)}, \dots, x_a^{(N)}\}$ , where  $N_a \ll N_n$ . These anomalies correspond to potential handover events, reflecting the moment when the network must decide whether a cell transition is necessary.

An AE can be formulated to learn the underlying manifold of the normal radio conditions and identify deviations that indicate imminent handover triggers. The Encoder function  $f(\cdot; \phi)$  compresses each measurement vector into a latent representation:

$$z^{(i)} = f(x^{(i)}; \phi), \text{ for } x^{(i)} \in \mathcal{X} \quad (6)$$

For nominal conditions  $x^{(i)} \in \mathcal{X}_n$ , the encoded representation  $z^{(i)}$  lies within a compact latent subspace that captures typical channel behavior. For anomalous states  $x^{(i)} \in \mathcal{X}_a$ , the latent representation tends to diverge from this manifold due to the non-stationary signal variations associated with the HO event.

The Decoder  $g(\cdot; \theta)$  reconstructs the input measurement from its latent embedding:

$$\hat{x}^{(i)} = g(z^{(i)}; \theta) \quad (7)$$

where the reconstruction aims to approximate the nominal feature distribution. The corresponding reconstruction error,

$$s_{HO}(x^{(i)}) = \frac{1}{d} \sum_{t=1}^d (x_t^{(i)} - \hat{x}_t^{(i)})^2 \quad (8)$$

acts as an HO anomaly score that quantifies the deviation of current radio conditions from the learned normal baseline. When  $s_{HO}(x^{(i)})$  exceeds a predefined threshold  $\tau_{HO}$ , i.e.;

$$s_{HO}(x^{(i)}) > \tau_{HO} \quad (9)$$

the instance  $x^{(i)}$  is identified as an anomalous state, indicating an impending or ongoing event-triggered handover. The threshold  $\tau_{HO}$  can be empirically chosen or derived through percentile-based or statistical estimation techniques.

This formulation establishes an equivalence between autoencoder-based reconstruction error and handover event detection, wherein:

- Normal samples ( $s_{HO}(x^{(i)}) \leq \tau_{HO}$ ) correspond to stable serving-cell attachment, and
- Anomalous samples ( $s_{HO}(x^{(i)}) > \tau_{HO}$ ) correspond to HO-triggering events (A1-A5 conditions).

Hence, the Autoencoder serves as a data-driven model that captures normal radio behavior and predicts the onset of handover events based on deviations in reconstruction error. This provides the foundational principle upon which the Contrastive Autoencoder (CAE) is proposed to enhance discriminative representation and improve HO prediction robustness under rapidly varying network conditions.

### 5.3 Proposed Contrastive Autoencoder Architecture

The proposed Contrastive Autoencoder (CAE) is designed with a quasi-symmetric, fully connected architecture that compresses high-dimensional handover (HO) feature vectors into a compact latent representation and subsequently reconstructs them back to the original dimension. The encoder progressively reduces the feature space, allowing the network to learn low-dimensional embeddings that capture the intrinsic characteristics of the input data. Conversely, the decoder expands this representation to approximate the original feature distribution. Fig. 8 shows the structure of our proposed CAE architecture.

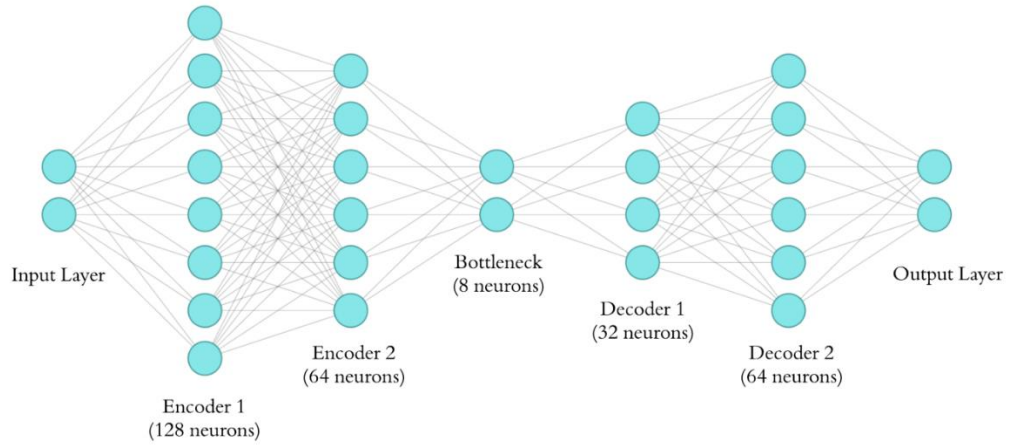


Fig. 8. Proposed CAE architecture.

The architecture is lightweight yet sufficiently expressive to capture nonlinear feature dependencies commonly present in radio signal measurements such as RSRP, RSRQ, and CINR. The latent dimension is deliberately kept small (8 neurons) to enforce strong compression, which facilitates discriminative representation learning when integrated with the contrastive objective in later stages. While the general structure of the proposed CAE follows a symmetric fully connected design, the number of hidden layers and neurons was not fixed throughout all experiments. During ablation studies, these configurations were adaptively tuned to identify the most effective architecture under varying radio conditions. Therefore, the table presents a representative configuration used in the main experiments, while acknowledging that the hidden layer dimensions were flexibly adjusted in specific cases for performance optimization.

**Table 4**  
Representative Structural Configuration of the Proposed CAE

Path	Layer	No. of Neurons	Activation Function	Functional Role
Encoder	Input Layer	Input dimension	-	Accepts normalized handover feature vectors as input
	Dense-1	128	ReLU	Performs initial nonlinear projection to capture broad feature interactions
	Dense-2	64	ReLU	Further compresses and abstracts key radio signal features
	Dense-3	8	ReLU	Generates the latent embedding representing the compact HO state space
Decoder	Dense-4	32	ReLU	Begins reconstruction by expanding latent features toward the input space.
	Dense-5	64	ReLU	Refines intermediate reconstruction for structural recovery.
	Output Layer	Input dimension	Linear	Produces the final reconstructed feature

This architectural configuration ensures a smooth information bottleneck between encoder and decoder, enabling the CAE to efficiently compress, represent and later distinguish between normal and event-triggering handover states once contrastive training is applied.

#### 5.4 Training and Testing Methodology of the Proposed CAE

The proposed Contrastive Autoencoder (CAE) is composed of three principal modules: the encoder, the latent representation space, and the decoder. The encoder and decoder operate jointly during both training and testing phases, while the latent space acts as the structural bridge that governs the discriminative mapping between normal and event-triggering (handover) conditions.

##### *Latent Space Representation and Triplet Loss Optimization*

The latent space of the proposed Contrastive Autoencoder (CAE) plays a pivotal role in transforming the input feature space into a structured, discriminative manifold that distinctly separates *normal* and *handover-triggering* instances. Unlike traditional autoencoders that rely solely on minimizing reconstruction error, the CAE explicitly constrains the latent space using a triplet-based contrastive learning objective, ensuring that the encoded representations carry semantic separation aligned with network behavior.

##### *(a) Role of Loss Function in CAE*

The loss function defines the optimization target guiding the encoder–decoder training process. In conventional autoencoders, the reconstruction loss

$$\mathcal{L}_{rec} = \frac{1}{N} \sum_{i=1}^N \|x^{(i)} - \hat{x}^{(i)}\|_2^2 \quad (10)$$

encourages the network to reproduce its inputs with minimal distortion, but it does not enforce any structural discrimination between samples of different operational states. Consequently, the embeddings of normal and handover-triggering samples may overlap in latent space, leading to ambiguous reconstruction behavior.

To address this limitation, the CAE replaces the conventional reconstruction objective with a triplet loss formulation, which introduces metric learning within the latent domain. This formulation explicitly shapes the geometry of the embedding space such that samples belonging to the same class (normal conditions) are drawn closer together, while samples corresponding to distinct classes (handover-triggering events) are repelled beyond a specified margin. The result is a well-separated latent topology where anomalies manifest as embeddings distant from nominal clusters.

##### *(b) Triplet Loss Formulation*

Formally, each triplet  $T = (x_A, x_P, x_N)$  consists of three input samples:

- Anchor ( $x_A$ ): a reference sample from the normal (non-handover) condition,
- Positive ( $x_P$ ): another sample from the same class as the anchor (normal condition),
- Negative ( $x_N$ ): a sample from the opposite class, representing a handover-triggering instance.

These samples are projected into the latent domain by the encoder  $f_\phi(\cdot)$ :

$$z_A = f_\phi(x_A), \quad z_P = f_\phi(x_P), \quad z_N = f_\phi(x_N) \quad (11)$$

where  $z \in \mathbb{R}^{d_z}$  denotes the latent representation and  $\phi = \{W_e, b_e\}$  represents encoder parameters.

The triplet loss is defined as:

$$\mathcal{L}_{triplet} = \frac{1}{M} \sum_{i=1}^M \max(\|z_A^{(i)} - z_P^{(i)}\|_2^2 - \|z_A^{(i)} - z_N^{(i)}\|_2^2 + \alpha, 0) \quad (12)$$

where:

- $\|\cdot\|_2$  denotes the Euclidean distance,
- $M$  is the number of triplets per batch,
- $\alpha > 0$  is the margin parameter, enforcing a minimum separation between positive and negative pairs.

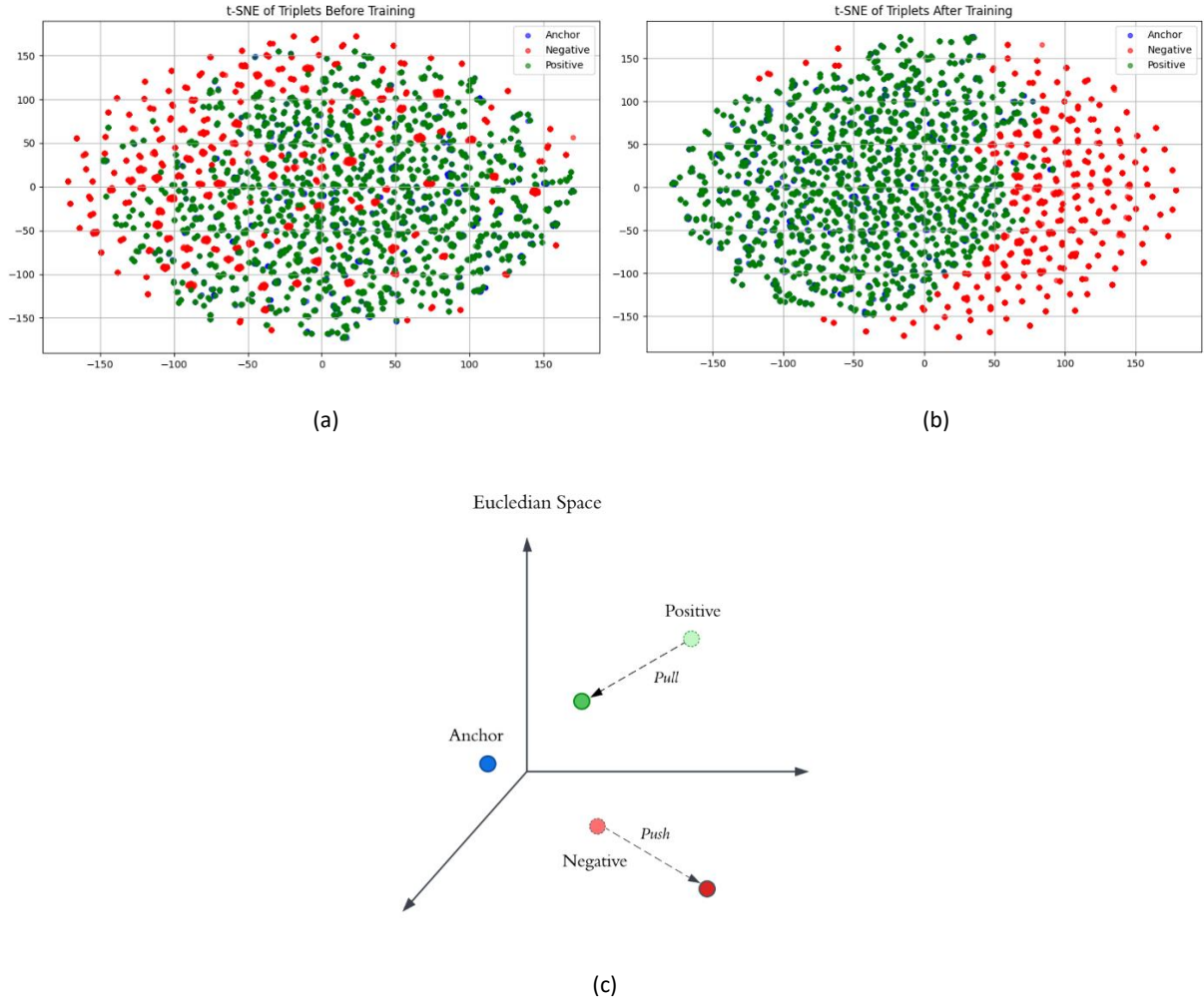


Fig. 9. Visualization of the effect of triplet loss in the proposed CAE framework: (a) t-SNE representation before applying triplet loss, (b) t-SNE representation after applying triplet loss showing improved class separability and (c) schematic illustration of the triplet loss mechanism.

#### (c) Function of the Margin Parameter

The margin  $\alpha$  defines the minimal inter-class separation required between the anchor–positive and anchor–negative pairs. It ensures that:

$$\|z_A - z_N\|_2^2 \geq \|z_A - z_P\|_2^2 + \alpha \quad (13)$$

thereby preventing trivial convergence (e.g., collapsing all embeddings into a single point). A properly tuned margin enforces robust class separation in the latent manifold, making the embedding space resilient to small perturbations or overlapping radio measurements common in LTE environments.

*(d) Effect on Latent Space Geometry for HO Prediction*

During training, the encoder receives supervision through the triplet loss and gradually molds the latent space such that:

- Non-handover (stable) samples form compact, high-density clusters.
- Handover-triggering samples are projected to distinct, sparsely distributed regions separated by at least margin  $\alpha$ .

This separation provides the decoder with a structured latent domain: embeddings within dense normal regions yield low reconstruction error, whereas embeddings lying in sparse or boundary regions (near the HO cluster) produce higher reconstruction error due to insufficient decoder generalization in those areas.

*Encoder Module*

The encoder module performs a nonlinear projection that compresses high-dimensional radio measurements (e.g., RSRP, RSRQ, CINR) into a compact, information-rich latent representation. During training, the encoder is jointly optimized with the triplet loss objective, which enforces relative similarity constraints among triplets of samples — an anchor, a positive (same class), and a negative (opposite class). This supervision instructs the encoder to learn a topological mapping in which embeddings of samples from similar radio states (non-handover) are drawn close together, while those representing dissimilar conditions (handover-triggering events) are pushed apart by at least a fixed margin.

Through this optimization process, the encoder gradually discovers an intrinsic manifold structure that reflects the underlying dynamics of network stability and transition behavior. Normal samples form dense, smooth manifolds characterized by minimal intra-cluster variability, while handover-triggering samples occupy peripheral or sparsely populated regions of the latent domain. This structured separation is not explicitly programmed but emerges naturally as the encoder minimizes the triplet loss across the dataset.

At inference time, the encoder functions in a label-free mode. It receives an unseen input sample and directly projects it into the pre-organized latent space using the previously learned geometric mapping.

- Inputs exhibiting stable radio conditions are automatically positioned within the dense, well-trained regions corresponding to non-handover states.
- Inputs exhibiting transitional or degraded signal characteristics are projected toward the outer or boundary regions associated with handover events.

This enables the encoder to maintain class-aware latent projections without requiring explicit class labels during testing, effectively converting complex multivariate radio data into discriminative latent coordinates suitable for anomaly-based HO prediction.

*Decoder Module*

The decoder performs the inverse mapping, reconstructing input-like representations from the latent embeddings. Although no explicit reconstruction loss is used, the decoder still participates in training through the triplet loss, which is computed in the reconstruction (decoder output) space rather than the latent space.

In this setting, the triplet loss simultaneously updates both encoder and decoder parameters so that:

- Reconstructions of normal (anchor–positive) pairs become similar.
- Reconstructions of anomalous (anchor–negative) pairs diverge beyond the margin.

This implicit regularization causes the decoder to learn consistent mappings for normal samples — effectively mimicking the behavior of a reconstruction loss. By minimizing the distance between reconstructed normal samples, the decoder develops a structural bias to accurately reproduce normal patterns while distorting reconstructions of anomalous ones.

As a result, the reconstruction error still emerges as a meaningful indicator of handover likelihood:

- For samples encoded within the dense normal manifold, reconstructions remain accurate, producing low errors.
- For samples lying near the sparse or boundary HO manifold, reconstructions deviate significantly, yielding higher errors.

During testing, this encoder–decoder interaction acts as a self-organized, label-free predictor. The encoder maps new inputs into their corresponding latent regions, while the decoder’s reconstruction fidelity reflects how closely these inputs align with learned normal behavior. Consequently, even without an explicit reconstruction objective, the triplet-induced mapping ensures that reconstruction error remains a reliable and discriminative metric for identifying handover-prone conditions.

#### *Training Procedure*

The model was implemented in TensorFlow/Keras using a quasi-symmetric fully connected architecture. LTE drive-test datasets were split 80–20, then combined for training. Features such as RSRP, RSRQ, CINR and speed were standardized using z-score normalization.

Triplet samples were created where anchors and positives represented non-handover states, and negatives represented handover-triggering states. Around 10,000 triplets were used for training. The model was trained for 50 epochs using the Adam optimizer with a margin-based triplet loss, batch size of 128. The training encouraged the decoder to consistently reconstruct normal radio conditions while separating the latent representations of handover states, enabling effective handover inference.

**Table 5**

Training Configuration of the Proposed CAE

Parameter	Value
Optimizer	Adam
Batch size	128
Epochs	50
Margin ( $\alpha$ )	1
Number of triplets	10000

### **5.5 Reconstruction Error Thresholding for Handover Prediction**

In the context of handover prediction, thresholding on reconstruction error enables the CAE to distinguish between stable link conditions and those likely to trigger a handover event. After training, the CAE reconstructs non-handover (normal) patterns accurately while producing higher reconstruction errors for feature combinations indicative of handover-triggering states. Applying a suitable threshold to this error distribution converts continuous

deviations into binary decisions — identifying when radio link dynamics shift from stability to a potential handover condition. To ensure robustness and adaptability across varying scenarios, five thresholding techniques have been adopted in this study.

#### I. Percentile-Based Dynamic Thresholding

This method establishes an adaptive cutoff on the reconstruction error distribution using a percentile of the non-handover (normal) class. Given the absolute reconstruction error  $e_i = \frac{1}{d} \sum_{j=1}^d |x_{ij} - \hat{x}_{ij}|$ , where  $d$  is the feature dimension, the threshold  $\tau$  is selected as the 97th percentile of  $e_i$  among normal samples. Any test instance with  $e_i > \tau$  is identified as anomalous (handover). This non-parametric approach assumes that normal errors follow a stable distribution and higher deviations signify degraded radio conditions, enabling efficient detection without requiring probabilistic modeling or classifier training.

#### II. Isolation Forest–Based Anomaly Segmentation

The Isolation Forest (IF) algorithm treats anomaly detection as a process of isolating samples through recursive random partitioning. In this approach, reconstruction errors from non-handover samples are used to train the isolation model. The algorithm builds multiple random trees and computes an average path length  $h(x)$  for each point, where anomalous samples typically have shorter path lengths due to their isolation in fewer splits. The anomaly score is derived as  $s(x) = 2^{-\frac{E(h(x))}{c(n)}}$ , with  $c(n)$  as a normalization term. Samples yielding high anomaly scores are labeled as potential handover-triggering instances. This technique effectively models nonlinear error distributions and is robust to skewed or multimodal data.

#### III. Kernel Density Estimation (KDE)–Driven Likelihood Thresholding

Kernel Density Estimation provides a smooth, non-parametric estimate of the probability density function of reconstruction errors from normal conditions. The density  $p(e)$  is approximated as

$$p(e) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{e - e_i}{h}\right) \quad (14)$$

where  $K(\cdot)$  is a Gaussian kernel and  $h$  is the bandwidth. For each test sample, the log-likelihood  $\log(p(e))$  is computed, and anomalies are identified when the likelihood falls below the 5th percentile of the non-handover log-likelihood distribution. This approach captures the underlying probability landscape of normal reconstruction patterns and adaptively rejects samples that reside in low-density regions, indicating unusual network behavior.

#### IV. Youden’s J–Optimized ROC Thresholding

This supervised thresholding method derives an optimal cutoff from the Receiver Operating Characteristic (ROC) curve, balancing true positive rate (TPR) and false positive rate (FPR). The Youden’s J statistic is defined as  $J = TPR - FPR$ , and the threshold that maximizes  $J$  corresponds to the best trade-off between sensitivity and specificity. Using this criterion, the optimal reconstruction error boundary is data-driven and tuned according to the discriminative behavior of the CAE’s reconstruction quality between normal and handover samples. This ensures performance-optimized decision boundaries under varying network dynamics.

#### V. Latent–Reconstruction Fusion with XGBoost Classification

In this final approach, a hybrid thresholding mechanism was implemented using a supervised XGBoost classifier trained on features derived from the CAE. The dataset was partitioned through a stratified multi-stage split: 35% of the data was used for triplet-based CAE training, 45% for training the XGBoost classifier, and the remaining 20% was reserved as an unseen evaluation set, ensuring class balance and preventing information leakage. All input features

were standardized based on the CAE training subset. The CAE generated latent embeddings representing compressed radio-state information, along with reconstruction errors capturing deviations from normal conditions. These complementary representations were concatenated to form a fused feature vector, combining structural and anomaly-sensitive cues. The XGBoost classifier learned a discriminative boundary between normal and handover-triggering states, while evaluation on the held-out set assessed generalization. By integrating latent geometry with reconstruction-based deviations, this method provides an adaptive, data-driven thresholding mechanism that enhances handover prediction across heterogeneous radio conditions.

## 5.6 Vanilla Autoencoder with MSE Loss

The baseline architecture in this study employed a *vanilla autoencoder* trained to reconstruct radio condition features of non-handover (non-HO) samples using a mean squared error (MSE) objective. The AE acts as an unsupervised anomaly detector by learning the underlying distribution of normal handover-free network states. During inference, deviations from this learned manifold are quantified through reconstruction error, which serves as an implicit anomaly score for identifying potential handover triggers.

Prior to training, all features were normalized using z-score scaling to eliminate scale dependency and improve gradient stability. Only samples labeled as non-HO were used for training to ensure that the model captured the intrinsic radio environment characteristics under stable connectivity conditions. The network comprised a symmetric, fully connected encoder-decoder configuration: four encoding layers with progressively decreasing units (64-32-16-8-4) compressed the high-dimensional input into a 4-dimensional latent representation, while the decoder mirrored this structure to reconstruct the input feature space. ReLU activations were employed throughout the hidden layers to promote non-linear expressiveness, and a lightweight L1 activity regularization was introduced in the first encoding layer to enhance sparsity and mitigate overfitting. The model was trained using the Adam optimizer for 100 epochs with mini-batches of 32 samples, using 10% of the data for validation.

Once trained, the reconstruction errors were computed for all non-HO training samples, and their distribution was analyzed to establish an anomaly threshold. The 80th percentile of the training reconstruction error distribution was selected as the decision boundary, representing the upper tolerance limit of normal reconstruction behavior. During evaluation, both HO and non-HO test samples were passed through the AE; samples exhibiting reconstruction errors exceeding this threshold were classified as potential handover events.

Unlike the proposed *Triplet-based CAE*, which leverages both HO and non-HO samples to learn a discriminative latent structure, the vanilla AE relies solely on reconstruction fidelity. Consequently, it focuses on minimizing overall feature-level deviations rather than explicitly modeling inter-class separability. The absence of a contrastive learning mechanism causes the latent space to lack structural discrimination between normal and abnormal states—leading to substantial overlap between the reconstruction errors of HO and non-HO samples. While the AE effectively captures stationary radio patterns, it struggles to generalize when channel dynamics vary abruptly during imminent handover conditions.

### *Limitations of MSE-Based Reconstruction Loss*

The MSE loss function optimizes for global reconstruction accuracy without considering the relational geometry among samples in the latent space. This objective inherently encourages the network to approximate an average representation of the input distribution, resulting in *latent representation collapse* when the input features of normal and HO conditions are numerically similar. In dynamic cellular environments, where signal degradation and interference evolve gradually, the resulting reconstruction errors from MSE training often remain small even for HO-triggering samples, yielding poor anomaly separability.

In contrast, the Triplet-based loss explicitly enforces *relative distance constraints* between embeddings of normal and HO samples, guiding the encoder to cluster homogeneous (non-HO) representations tightly while pushing dissimilar (HO) representations apart. This structured latent space enables the decoder to inherently assign higher reconstruction errors to unseen or anomalous channel states, yielding superior precision and recall in HO detection. Thus, while the vanilla AE serves as a robust baseline for modeling normal connectivity conditions, its reliance on MSE reconstruction loss fundamentally limits its discriminative capability compared to the contrastive triplet learning paradigm.

5.7 Artificial Neural Network Based Handover Prediction

Artificial Neural Networks (ANNs) were employed as a supervised learning baseline to model the nonlinear dependencies between radio parameters and handover (HO) trigger events. The ANN learns a discriminative mapping from multi-dimensional radio features—including serving and neighbor cell RSRP/RSRQ, CINR, and user speed—to a binary decision space representing handover and non-handover states. Through iterative backpropagation, the network optimizes its weights to minimize the binary cross-entropy loss, effectively capturing the underlying decision boundary associated with HO-triggering conditions.

The designed ANN architecture consists of multiple fully connected (Dense) layers with rectified linear unit (ReLU) activations, interleaved with batch normalization and dropout layers to improve convergence stability and prevent overfitting. The final output layer employs a sigmoid activation to produce probabilistic handover predictions. The configuration of hidden layers and neuron counts was empirically tuned across different simulation runs to identify the most effective topology under varying channel conditions.

Table 6  
Parameter Configuration of the ANN Model

Component	Description	
Input Layer	6 features	Radio parameters (RSRP, RSRQ, CINR, speed, etc.)
Hidden Layers	[128, 64, 64, 16] neurons (tuned)	Fully connected layers for nonlinear feature extraction
Activation Function	ReLU	Enables sparse and efficient gradient propagation
Normalization	Batch Normalization after each hidden layer	Stabilizes learning and mitigates internal covariate shift
Regularization	Dropout (0.1–0.2)	Reduces overfitting by random neuron deactivation
Output Layer	1 neuron, Sigmoid activation	Generates binary probability of HO trigger
Optimizer	Adam	Adaptive learning rate for stable convergence
Loss Function	Binary Cross-Entropy	Measures classification error between predicted and actual HO labels
Epochs / Batch Size	100 / 32	Tuned for optimal trade-off between accuracy and convergence speed

The ANN operates as a supervised classifier, learning explicit decision boundaries based on labeled handover data. By mapping radio-condition features to a probabilistic output, it identifies patterns that precede HO events. This structure enables the network to generalize across varying radio conditions, offering an interpretable baseline for performance comparison against the proposed contrastive autoencoder framework.

## 5.8 Ablation and Benchmarking with Conventional Supervised Models

To provide a rigorous ablation against the proposed contrastive autoencoder, a set of conventional supervised classifiers was implemented and systematically optimized. The supervised pipeline comprised three disciplined steps: (i) training-time imbalance correction using SMOTE (Synthetic Minority Over-Sampling Technique) applied only within training folds, (ii) feature preprocessing (z-score scaling where required) encapsulated in a pipeline to avoid leakage, and (iii) hyperparameter selection by grid search with stratified k-fold cross-validation.

SMOTE addresses class imbalance by synthesizing minority-class samples in feature space, thereby reducing bias toward the majority (non-HO) class; critically, SMOTE was applied inside the cross-validation pipeline so that synthetic samples never contaminate validation folds. Grid search (scikit-learn GridSearchCV) used stratified 5-fold CV,  $n\_jobs = -1$  and deterministic seeds; the primary optimization metric was the F1 score for the positive (HO) class, with ROC-AUC retained as a secondary metric. For algorithms sensitive to feature scale (logistic regression, SVM, KNN, Naïve Bayes), a standard scaler was included in the pipeline; tree-based and boosting classifiers received raw features. Early-stopping or  $n\_estimators$  tuning was used for gradient-boosting methods to control overfitting and reduce training cost.

Below is a compact technical description and the hyperparameter search focus for each implemented classifier. The entries highlight the algorithmic principle, the principal hyperparameters tuned, and the specific rationale for its inclusion as an HO-prediction baseline.

**Table 7**

Supervised Classifier Descriptions

Classifier	Technical description	Suitability
Logistic Regression	Linear discriminative model with convex optimization; L2/L1 regularization controls weight magnitude.	Fast, interpretable baseline; tests whether HO is (approximately) linearly separable in feature space.
Random Forest	Bagged ensemble of decision trees; reduces variance via bootstrap aggregation.	Robust to heterogeneous features and noise; provides feature-importance baseline.
Support Vector Machine (SVM)	Maximum-margin classifier; kernelized (RBF / linear) enables non-linear decision surfaces.	Effective for medium-dimensional data; requires careful scaling and $C/\gamma$ tuning.
Gaussian Naïve Bayes	Generative probabilistic model assuming feature conditional independence and Gaussian likelihoods.	Lightweight probabilistic baseline; useful to assess how well simple generative assumptions perform for HO.
K-Nearest Neighbors (KNN)	Instance-based, nonparametric classifier using distance in feature space.	Simple local decision metric; provides insight into local separability of HO patterns.
AdaBoost	Stagewise additive model that reweights misclassified samples; often uses decision stumps as weak learners.	Effective for moderately nonlinear patterns; sensitivity to noisy labels tested.

GradientBoosting	Sequential tree boosting (Friedman) minimizing differentiable loss; regularized via learning rate & subsampling.	Strong baseline for tabular data; trade-off between bias and variance controlled by depth & learning rate.
XGBoost	Optimized gradient boosting with regularization and efficient tree construction.	High performance on heterogeneous, imbalanced data; scale_pos_weight useful for class imbalance.
CatBoost	Gradient boosting optimized for categorical features and ordered boosting to reduce target leakage.	Robust default settings, strong regularization and fast convergence; included as an advanced boosting baseline.

#### *Implementation notes (methodological rigor)*

- **Pipelines & leakage prevention:** For scalable reproducibility, each model was wrapped in a Pipeline that placed StandardScaler (if needed) and SMOTE (from imblearn) before the classifier. When performing grid search, SMOTE was nested inside the CV pipeline so synthetic samples were generated only from training folds. This is essential to avoid overly optimistic validation scores.
- **Cross-validation & selection criterion:** GridSearchCV employed stratified 5-fold CV; the primary scoring objective was F1 (positive class) because early detection of HO (minority class) is prioritized; ROC-AUC acted as a secondary check for overall separability. For boosting methods, early stopping was used where possible to prevent overfitting and to shorten hyperparameter searches.
- **Reproducibility and compute:** All estimators were fit with fixed random\_state values and n\_jobs=-1 where supported. For large parameter grids, randomized search or successive halving was considered to balance thoroughness with computational resources.
- **Evaluation:** Final model selection used the held-out test set (distinct from CV) to produce unbiased metrics (precision, recall, F1, ROC-AUC, confusion matrix) reported in the Results section.

#### *Rationale for design choices*

The chosen ensemble of classifiers spans linear, distance-based, probabilistic, bagging and boosting paradigms. This breadth permits a comprehensive ablation that isolates whether performance gains of the proposed CAE stem from superior representation learning (latent separation) or simply from downstream classifier capacity. SMOTE is critical because HO events are scarce; without adequate minority-class augmentation the supervised algorithms would be biased toward the majority class, yielding poor recall. Grid search provides a principled mechanism to tune model complexity and regularization so that comparisons are fair (each classifier evaluated at near-optimal hyperparameters).

## **6. Performance Evaluation**

Among the three datasets, Dataset\_2 and Dataset\_3 did not include user speed information, whereas Dataset\_1 contained additional user speed measurements obtained from the drive test equipment. Based on this distinction, the evaluation has been divided into two separate experimental cases:

- Case 1: Drive test data from Dataset\_2 and Dataset\_3, which do not include user speed as a feature.
- Case 2: Drive test data from Dataset\_1, which includes user speed as an additional input feature.

This dual-case evaluation enhances the robustness of the ablation study by testing model performance under both mobility-aware and mobility-agnostic conditions. Case 1 evaluates prediction solely from network parameters, while Case 2 incorporates user speed to capture mobility dynamics. This design verifies that the proposed framework remains effective and generalizable across varying feature availability, reflecting realistic handover prediction scenarios.

We have used five evaluation metrics for our study, namely Accuracy, Precision, Recall, F1 Score and ROC–AUC Score, to provide a comprehensive assessment of model performance in predicting handover events. These metrics are described in brief here.

#### *Accuracy*

Accuracy is a fundamental metric used for evaluating the performance of a classification model. It represents the proportion of correct predictions made by the model out of all predictions:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (15)$$

Here, TP (True Positives) and TN (True Negatives) denote the correctly predicted positive and negative instances, respectively, while FP (False Positives) and FN (False Negatives) represent the incorrectly predicted instances. Accuracy provides an overall measure of how well the model performs in correctly classifying data samples.

However, accuracy alone may not always provide a complete picture of model performance, particularly when dealing with imbalanced datasets. In such cases, a model might achieve a high accuracy by primarily predicting the majority class while failing to correctly identify the minority class. Hence, while accuracy offers a useful general overview, additional metrics such as Precision, Recall, F1 Score, and ROC–AUC are necessary to obtain a more balanced and reliable evaluation of the model’s predictive capability.

#### *Precision*

Precision evaluates the reliability of positive predictions by quantifying the proportion of correctly predicted positive instances among all instances classified as positive:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (16)$$

It indicates how accurate the model’s positive classifications are and reflects its ability to minimize false alarms. A high Precision value signifies that when the model predicts a positive event, it is likely to be correct.

#### *Recall or Sensitivity*

It measures the model’s ability to correctly identify actual positive instances. High recall ensures that the model captures most of the relevant events, which is particularly important for critical tasks such as handover prediction:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (17)$$

Recall is especially important in applications where missing a positive case can lead to significant consequences. A high Recall value indicates that the model successfully captures most of the relevant positive events, minimizing missed detections.

#### *F1 Score*

It is a performance metric that evaluates how well a classification model balances Precision and Recall, particularly useful when dealing with imbalanced datasets where one class occurs more frequently than the other. It is defined as the harmonic mean of Precision and Recall:

$$F1 \text{ Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (18)$$

The harmonic mean ensures that the F1 Score is high only when both Precision and Recall are high, making it a more balanced indicator of performance than a simple average. It penalizes extreme differences between the two metrics—if either Precision or Recall is low, the F1 Score will also decrease significantly. This makes the F1 Score especially suitable for evaluating models in imbalanced classification problems, where maintaining an appropriate trade-off between correctly identifying positive instances and minimizing false alarms is crucial.

#### ROC AUC Score

The ROC–AUC (Receiver Operating Characteristic – Area Under Curve) is a widely used performance metric for evaluating binary classification models. It measures how well a model can distinguish between positive and negative classes across different threshold levels. The ROC curve is a plot of the True Positive Rate (TPR) — also known as Recall or Sensitivity — against the False Positive Rate (FPR), defined as:

$$FPR = \frac{FP}{FP + TN} \quad (19)$$

Each point on the ROC curve represents a different decision threshold, showing the trade-off between correctly identifying positives and incorrectly classifying negatives. The AUC (Area Under the Curve) quantifies the overall ability of the model to rank a randomly chosen positive instance higher than a randomly chosen negative one:

$$ROC \text{ AUC Score} = \int_0^1 TPR(FPR) dFPR \quad (20)$$

Thus, a higher AUC value indicates better model performance. In essence, the ROC–AUC score provides a threshold-independent measure of model quality, reflecting how effectively the classifier separates the two classes.

These metrics collectively capture different aspects of classification effectiveness, ensuring a balanced evaluation of both overall correctness and the ability to identify critical events. This multifaceted evaluation is particularly important in wireless network scenarios, where both accurate and timely identification of handover events directly impact network quality and user experience.

## 6.1 Ablation Study on Case 1

**Table 8**

Comparative analysis of supervised machine learning algorithms- Baseline Model

Algorithm	Accuracy	Precision	Recall	F1 Score	ROC AUC Score
Logistic Regression	0.9271	0.7955	0.7955	0.7955	0.9711
Random Forest	0.9231	0.8378	0.7045	0.7654	0.9672
SVM	0.9352	0.8889	0.7273	0.8000	0.9709
KNN	0.9028	0.7941	0.6136	0.6923	0.9083
Naïve Bayes	0.8421	0.7273	0.1818	0.2909	0.9037
Gradient Boost	0.9352	0.8684	0.7500	0.8049	0.9694
AdaBoost	0.8988	0.7568	0.6364	0.6914	0.9425
XGBoost	0.9312	0.8462	0.7500	0.7952	0.9620
CatBoost	0.9190	0.9000	0.6136	0.7297	0.9723

**Table 9**

Comparative analysis of supervised machine learning algorithms- SMOTE and Grid Search

Algorithm	Accuracy	Precision	Recall	F1 Score	ROC AUC Score
Logistic Regression	0.9312	0.7647	0.8864	0.8211	0.9659
Random Forest	0.9393	0.8718	0.7727	0.8193	0.9680
SVM	0.9271	0.7708	0.8409	0.8043	0.9727
KNN	0.9150	0.7805	0.7273	0.7529	0.8961

Naïve Bayes	0.8704	0.7000	0.4773	0.5676	0.9240
Gradient Boost	0.9352	0.8889	0.7273	0.8000	0.9661
AdaBoost	0.9150	0.7674	0.7500	0.7586	0.9637
XGBoost	0.9352	0.8684	0.7500	0.8049	0.9689
CatBoost	0.9312	0.8140	0.7955	0.8046	0.9738

Table 8 represents the performance metrics of the baseline supervised models for Case 1. After applying SMOTE with Grid Search optimization, the results improved slightly as shown in Table 9. Among the nine classifiers, Logistic Regression showed the most balanced results with F1 score of 0.8211 and accuracy 0.9312. It also attained the best recall of 0.8864. The highest attained precision was 0.8889 and the ROC AUC Score ranged between 0.8961 and 0.9738.

**Table 10**

Comparative analysis of ANN and Vanilla AE

Algorithm	Accuracy	Precision	Recall	F1 Score	ROC AUC Score
ANN	0.9433	0.8571	0.8182	0.8372	0.9735
Vanilla AE	0.6721	0.3011	0.6364	0.4088	0.7669

The performance metrics of ANN and Vanilla autoencoder for case 1 are summarized in Table 10. ANN achieved F1 score of 0.8372 and accuracy of 0.9433. But, Vanilla autoencoder could attain F1 score of only 0.4088, accuracy of 0.6721 and a poor precision of 0.3011, indicating limited capability in distinguishing handover events without contrastive learning.

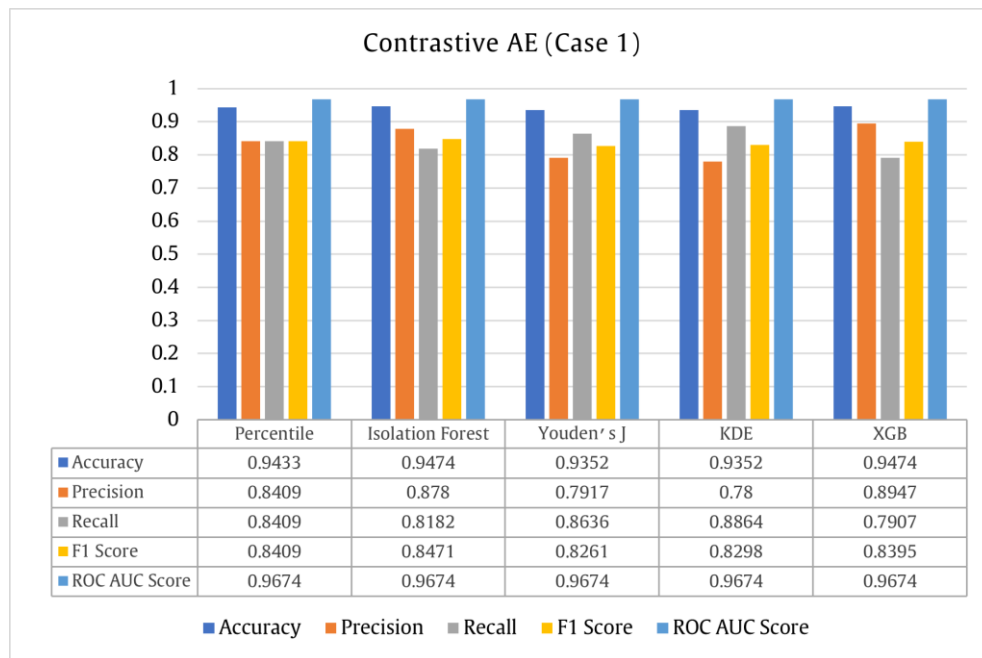
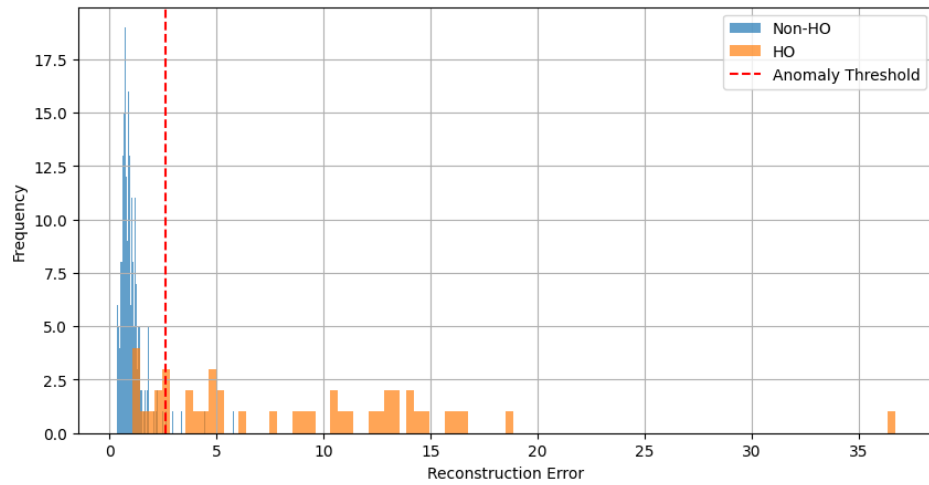


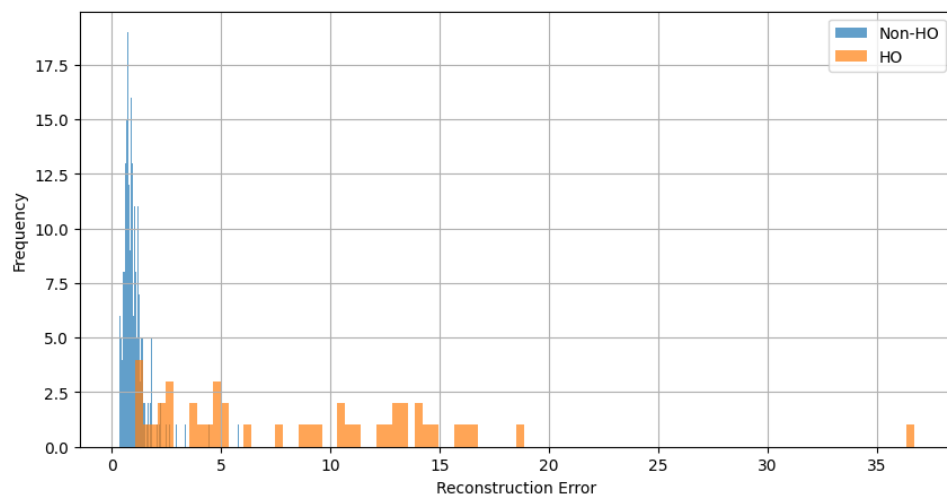
Fig. 10. Evaluation Metrics of CAE for different thresholding techniques.

For case 1, Figure 10 illustrates the F1 score, accuracy, precision, recall, and ROC–AUC score of the Contrastive Autoencoder framework with five different thresholding mechanisms – Percentile, Isolation Forest, Youden's J threshold, KDE and XGBoost. The model achieved the highest F1 score of 0.8471 with the Isolation Forest threshold, indicating a strong balance between correctly predicting handovers and minimizing unnecessary predictions. Other thresholds also performed well, with KDE attaining an F1 score of 0.8298, XGBoost achieving 0.8395, Percentile reaching 0.8409, and Youden's J obtaining 0.8261, showing the robustness of the framework across different

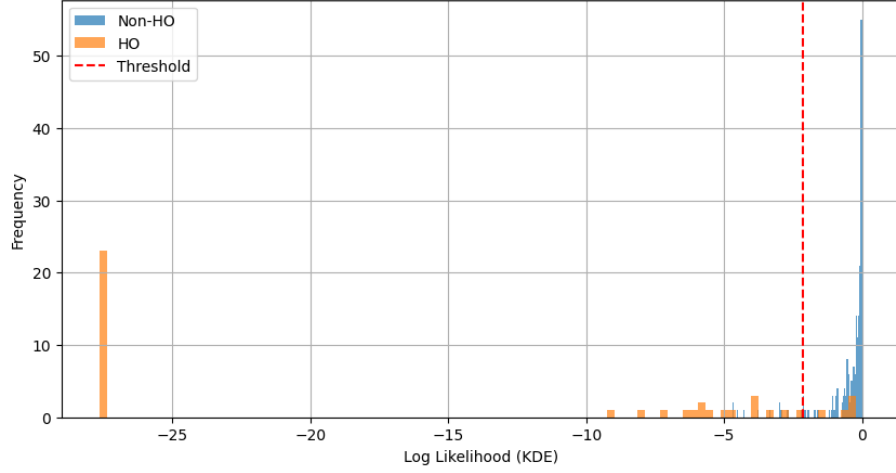
thresholding strategies. Accuracy reached 0.9474 with both Isolation Forest and XGB thresholds, reflecting that the model reliably identified both handover and non-handover instances. Precision was highest with the XGBoost threshold at 0.8947, meaning most predicted handovers were correct, minimizing unnecessary triggers. While recall ranged from 0.7907 to 0.8864, demonstrating the model's ability to capture a large proportion of true handovers, the ROC–AUC scores remained consistently high at 0.9674 for all thresholds, indicating that the learned latent space effectively separates handover and non-handover events.



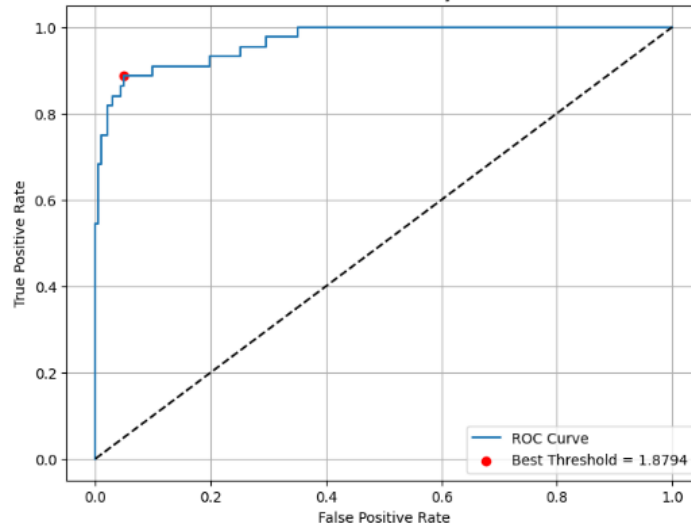
(a)



(b)



(c)



(d)

Fig. 11. Thresholding of Reconstruction Error Based on (a) Isolation Forest, (b) Percentile, (c) KDE, and (d) Youden's J Threshold.

Figure 11 illustrates the thresholding mechanisms applied to the reconstruction error distribution for distinguishing Handover (HO) and Non-HO samples. Among the four methods—Isolation Forest, Percentile, Kernel Density Estimation (KDE), and Youden's J—only the latter three differ in adaptivity and data dependency. The Isolation Forest adaptively learns anomaly boundaries through recursive feature partitioning, while KDE and Youden's J determine thresholds dynamically from the empirical distribution and ROC optimization, respectively. In contrast, the percentile-based approach relies on a fixed quantile, providing a simple yet non-adaptive baseline. Together, these thresholding techniques enable a comparative assessment between static and data-driven decision boundaries for robust reconstruction error-based HO event detection.

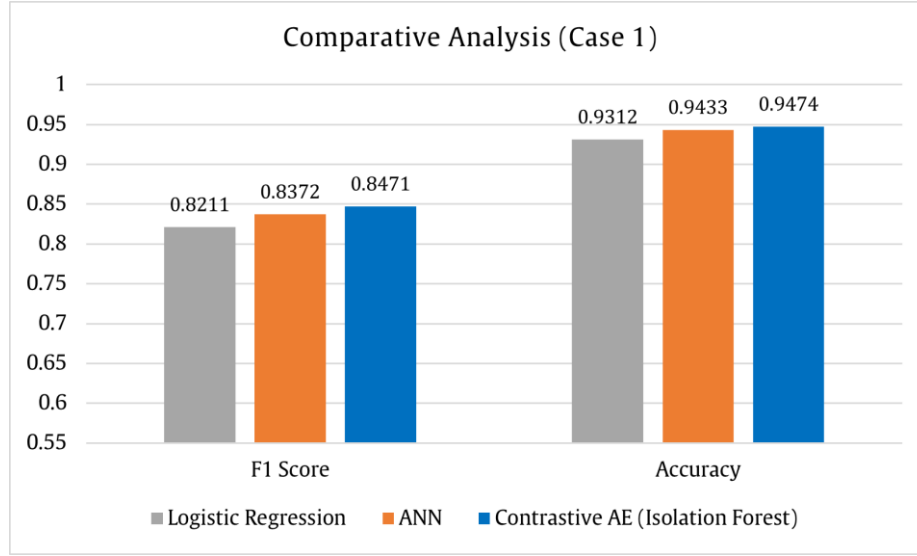


Fig. 12. Comparative Analysis of Best-Performing Models

Figure 12 presents a comparative analysis of the best-performing supervised model, ANN and the proposed CAE with Isolation Forest thresholding for Case 1. The CAE framework achieved the highest F1 score of 0.8471 and accuracy of 0.9474, outperforming both the supervised model and ANN. The consistent improvement in both metrics demonstrates that the proposed contrastive learning framework enhances the model's ability to accurately identify handover events while maintaining high reliability in overall classification. These results confirm the effectiveness of the CAE framework in capturing discriminative patterns and improving handover prediction performance compared to conventional supervised and ANN-based approaches.

## 6.2 Ablation Study on Case 2

**Table 11**

Comparative analysis of supervised machine learning algorithms- Baseline Model

Algorithm	Accuracy	Precision	Recall	F1 Score	ROC AUC Score
Logistic Regression	0.8270	0.4286	0.2000	0.2727	0.8903
Random Forest	0.9135	0.8889	0.5333	0.6667	0.9462
SVM	0.9027	0.9286	0.4333	0.5909	0.9275
KNN	0.9135	0.9375	0.5000	0.6522	0.8278
Naïve Bayes	0.8324	1.0000	0.0312	0.0606	0.7330
Gradient Boost	0.9189	0.9474	0.5625	0.7059	0.9344
AdaBoost	0.8919	0.8750	0.4375	0.5833	0.8710
XGBoost	0.9135	0.9444	0.5312	0.6800	0.8956
CatBoost	0.8973	0.8667	0.4333	0.5778	0.9290

**Table 12**

Comparative analysis of supervised machine learning algorithms- SMOTE and Grid Search

Algorithm	Accuracy	Precision	Recall	F1 Score	ROC AUC Score
Logistic Regression	0.8757	0.6400	0.5333	0.5818	0.8970
Random Forest	0.9135	0.8500	0.5667	0.6800	0.9424
SVM	0.9135	0.7917	0.6333	0.7037	0.9406
KNN	0.9081	0.7600	0.6333	0.6909	0.8361
Naïve Bayes	0.8432	0.7143	0.1562	0.2564	0.7000
Gradient Boost	0.9189	0.9048	0.5938	0.7170	0.9173
AdaBoost	0.8757	0.6552	0.5938	0.6230	0.8735

XGBoost	0.9243	0.9091	0.6250	0.7407	0.9154
CatBoost	0.9027	0.7500	0.6000	0.6667	0.9308

Table 11 presents the performance metrics of the baseline supervised models for Case 2. After applying SMOTE with Grid Search optimization, a slight improvement in performance was observed, as shown in Table 12. Among the nine classifiers, XGBoost demonstrated the most balanced performance, achieving an F1 score of 0.7407 and an accuracy of 0.9243. The highest precision and recall achieved were 0.9091 and 0.6333, respectively, while the ROC–AUC scores ranged from 0.70 to 0.9424.

**Table 13**

Comparative analysis of ANN and Vanilla AE

Case	Accuracy	Precision	Recall	F1 Score	ROC AUC Score
ANN	0.9135	0.8182	0.6000	0.6923	0.9148
Vanilla AE	0.7405	0.3393	0.6333	0.4419	0.7617

The performance metrics of ANN and Vanilla autoencoder for case 2 are summarized in Table 13. ANN achieved F1 score of 0.6923 and accuracy of 0.9135. But, Vanilla autoencoder could attain F1 score of only 0.4413, accuracy of 0.7405 and a poor precision of 0.3303, indicating a limited ability to effectively distinguish handover events in the absence of contrastive learning.

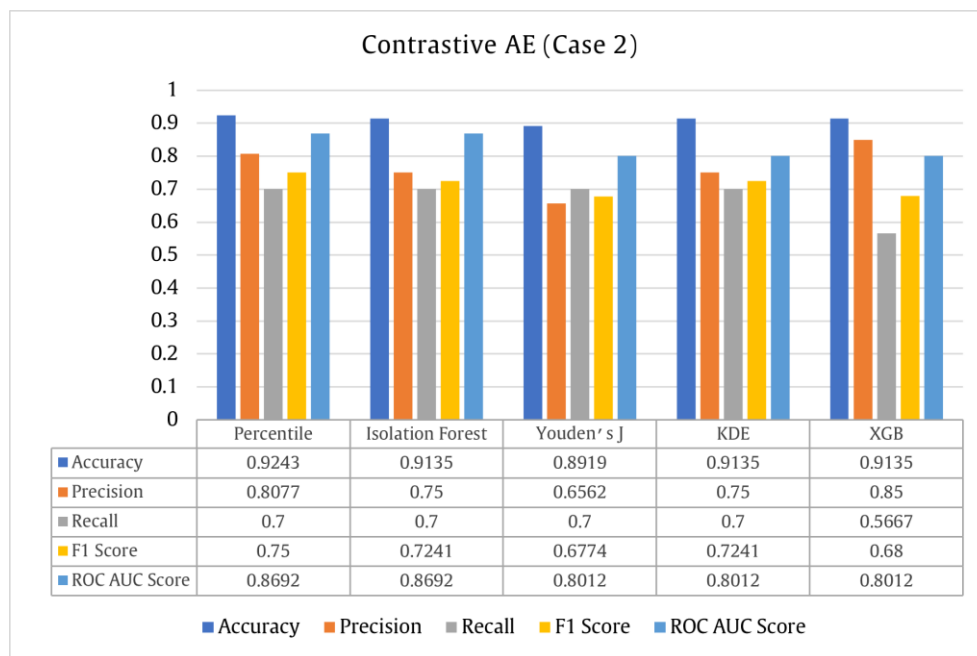
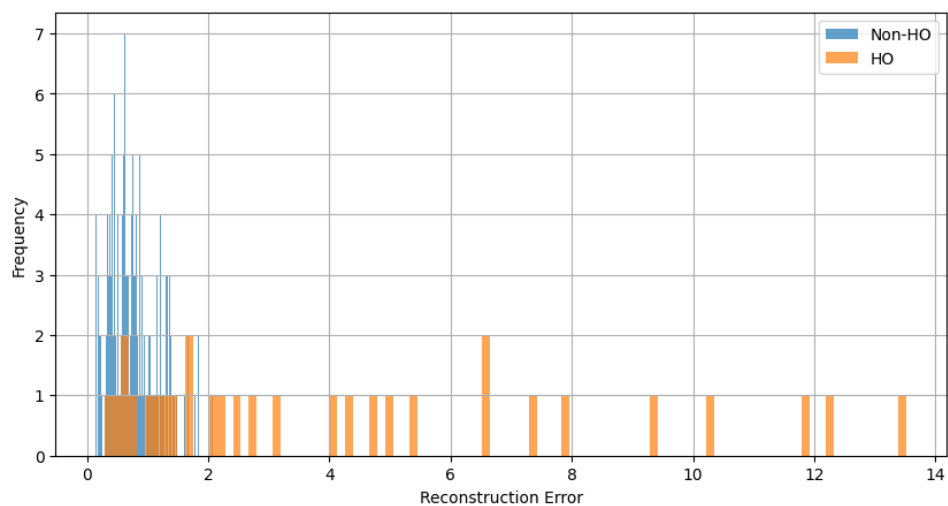


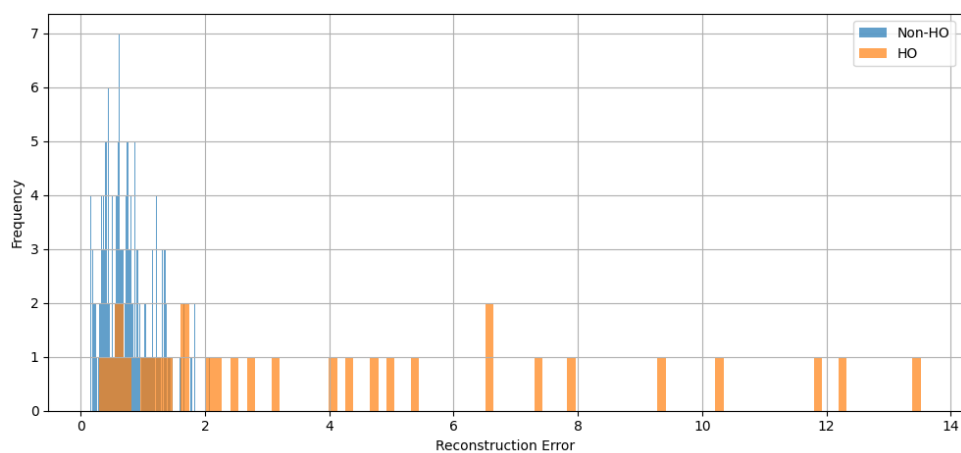
Fig. 13. Evaluation Metrics of CAE for different thresholding techniques.

For case 2, Figure 13 presents the F1 score, accuracy, precision, recall, and ROC–AUC score of the Contrastive Autoencoder framework using five thresholding mechanisms – Percentile, Isolation Forest, Youden’s J threshold, KDE, and XGBoost. The model achieved the highest F1 score of 0.75 with the Percentile threshold, demonstrating a balanced prediction performance. Isolation Forest also yielded an F1 score of 0.7241, reflecting stable performance across varying thresholding approaches. The highest accuracy of 0.9243 was also attained with the Percentile threshold, indicating reliable overall prediction capability. Precision peaked at 0.8500 with the XGB threshold, showing that most predicted handovers were accurate, while recall values went up to 0.70, confirming the framework’s ability to capture a substantial portion of actual handover events. The ROC–AUC scores, ranging

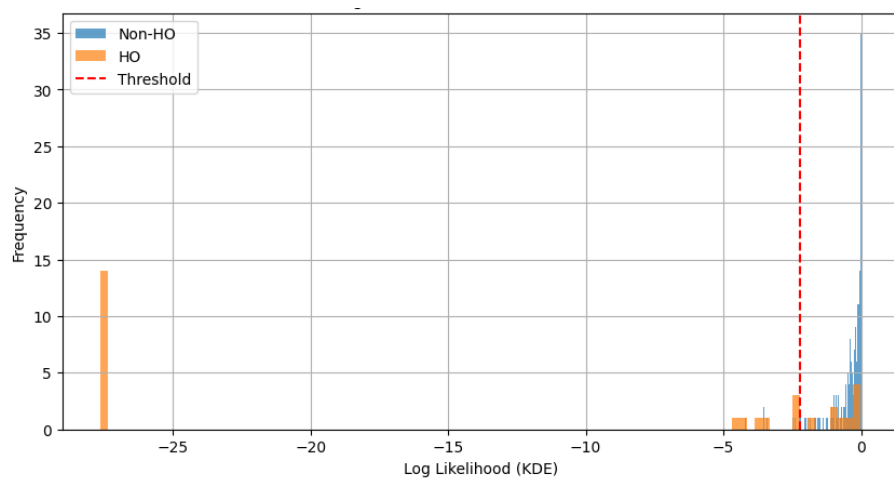
between 0.8012 and 0.8692, further demonstrate effective latent-space separation of handover and non-handover instances, even under challenging conditions.

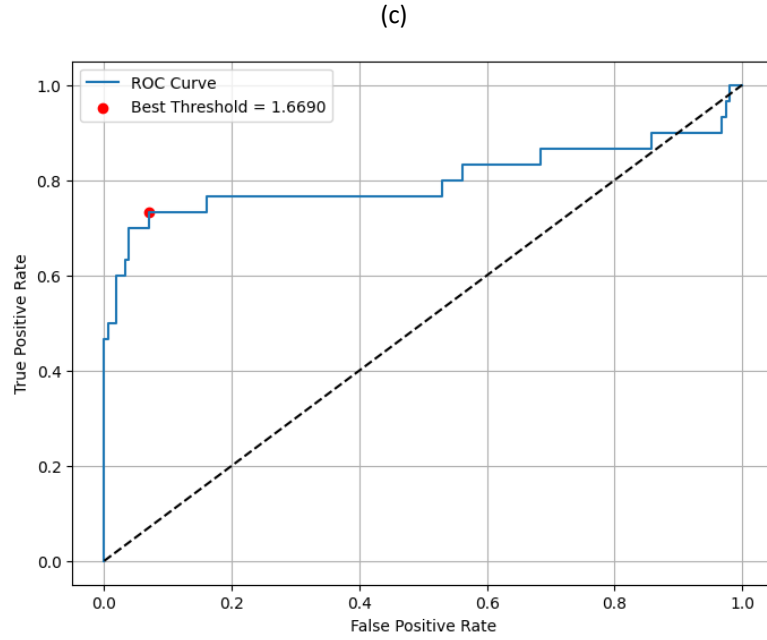


(a)



(b)





(d)

Fig. 14. Thresholding of Reconstruction Error Based on (a) Isolation Forest, (b) Percentile, (c) KDE, and (d) Youden's J Threshold.

Fig. 14. shows that, in Case 2, the inclusion of user speed as an auxiliary input significantly influences the learned latent representation, thereby altering the reconstruction error distribution. Consequently, the adaptive thresholding methods (Isolation Forest, KDE, and Youden's J) dynamically adjust their anomaly boundaries, capturing speed-induced variations in mobility behavior that were absent in Case 1.

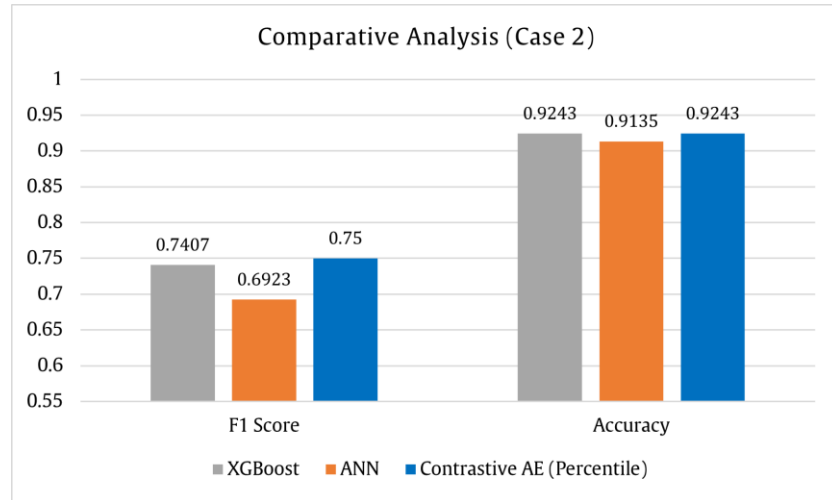


Fig. 15. Comparative Analysis of Best-Performing Models

Figure 15 presents a comparative analysis of the best-performing supervised model, ANN and the proposed CAE with Isolation Forest thresholding for Case 2. The CAE framework achieved the highest F1 score of 0.75 and accuracy of 0.9243, outperforming both the supervised model and ANN. The consistent improvement in both metrics demonstrates that the proposed contrastive learning framework reliably captures discriminative patterns and enhances handover prediction performance. The results from both Case 1 and Case 2 show that the CAE framework

performs better than the state-of-the-art methods, confirming its effectiveness and robustness across different scenarios while providing accurate, balanced, and practical handover predictions that can significantly improve network QoS and mobility performance.

## 7. Conclusion and Future Work

This study presented a CAE framework trained with triplet loss to address the inherent challenges of HO prediction in dynamic cellular networks. By manipulating the latent space through a contrastive learning objective, the CAE learns discriminative and semantically ordered representations of radio states, where normal mobility patterns and HO-triggering anomalies occupy distinct regions in the latent manifold. This latent structuring enables anomaly-based HO prediction through reconstruction error, eliminating the need for explicit binary supervision or handcrafted decision boundaries. The empirical evaluation demonstrated that the CAE consistently outperforms conventional supervised classifiers—Logistic Regression, Random Forest, Gradient Boost, CatBoost, and others—even though those models were optimized using SMOTE for class rebalancing and exhaustive grid search for hyperparameter tuning. Unlike these classifiers, the CAE inherently handles class imbalance through representation learning and reconstruction dynamics, achieving high detection fidelity without synthetic oversampling or complex preprocessing pipelines.

Beyond numerical superiority, the proposed framework mitigates several long-standing limitations of conventional HO prediction. The anomaly-based nature of CAE allows it to operate effectively even under label scarcity, delayed RRC signaling, and network parameter drift—conditions that severely degrade supervised learning performance. Moreover, as reconstruction error quantifies deviation from the learned "normal" operating state, it provides interpretable insights beyond binary prediction outcomes. In a live cellular environment, such a model can infer that *"something has changed"* in signal dynamics, offering explainable triggers for network self-optimization or proactive mobility decisions. The computational structure of the CAE, comprising feedforward encoder-decoder layers, ensures low inference latency suitable for near-real-time RAN integration, enabling faster anomaly localization and decision-making compared to retraining-dependent supervised baselines. Thus, the CAE provides a technically efficient, interpretable, and deployment-ready solution for proactive mobility management in 4G and beyond systems.

Nevertheless, the current framework does not explicitly exploit temporal dependencies among KPIs. Future research will therefore extend this work toward temporally structured and survival-aware contrastive architectures to capture the evolution and persistence of mobility anomalies, enabling predictive and prognostic capabilities within cellular environments.

Future extensions of this study may focus on:

- **Temporal-Contrastive Modeling:** Incorporate sequential encoding (e.g., LSTM/Transformer-based CAE) to capture long-term temporal correlations and mobility transition patterns.
- **Integration with Survival Analysis:** Fuse CAE latent representations with survival modeling to estimate time-to-failure or time-to-handover probabilities for proactive network adaptation.
- **Real-Time Edge Deployment:** Optimize the CAE for low-latency inference and energy efficiency on edge compute nodes or gNodeB controllers for real-world network integration.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

## Data Availability

The dataset is available on <https://data.mendeley.com/datasets/n2pvmtyn2j/1>

## Reference

- [1] S. D. Yusuf, S. I. Isa and B. J. Kwaha, "Analysis of 4G/LTE Network Performance in North-Central Nigeria: A Comprehensive Drive Test Approach," *Journal of Engineering Research and Reports*, vol. 26, no. 9, pp. 105–122, Sep. 2024, doi: 10.9734/jerr/2024/v26i91267.
- [2] S. Tripathi, V. Kulkarni and A. Kumar, "LTE E-UTRAN and its Access Side Protocols", Radisys, 2009. [Online]. Available: <https://go.radisys.com/rs/radisys/images/paper-lte-eutran.pdf>
- [3] Z. Shakir, A. Y. Mjhoor, A. Al-Thaedan, A. Al-Sabbagh and R. Alsabah, "Key performance indicators analysis for 4 G-LTE cellular networks based on real measurements," *International Journal of Information Technology (Singapore)*, vol. 15, no. 3, pp. 1347–1355, Mar. 2023, doi: 10.1007/s41870-023-01210-0.
- [4] K. Sun, Q. Han, Z. Yang, W. Huang, H. Zhang and V. C. M. Leung, "Proactive Handover Type Prediction and Parameter Optimization Based on Machine Learning," *IEEE Trans Wirel Commun*, vol. 24, no. 4, pp. 3515–3528, 2025, doi: 10.1109/TWC.2025.3531702.
- [5] R. Khoder, R. Naja, S. Ismail, N. Mouawad and S. Tohme, "Vertical Handover Decision using Machine Learning in Vehicular Platooning," *2021 3rd IEEE Middle East and North Africa Communications Conference (MENACOMM)*, Agadir, Morocco, 2021, pp. 60–64, doi: 10.1109/MENACOMM50742.2021.9678250.
- [6] E. Kim and I. Joe, "Handover Triggering Prediction with the Two-Step XGBOOST Ensemble Algorithm for Conditional Handover in Non-Terrestrial Networks," *Electronics (Switzerland)*, vol. 12, no. 16, Aug. 2023, doi: 10.3390/electronics12163435.
- [7] M. Malekzadeh and F. Rezaiee, "Impact of Inter-Enodeb Handover Parameters on Performance Optimization of LTE Networks," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 9, no. 1, pp. 212–220, Jan. 2018, doi: 10.11591/ijeecs.v9.i1.pp212-220.
- [8] E. Eldesouky, M. Bekhit, A. Fathalla, A. Salah and A. Ali, "A Robust UWSN Handover Prediction System Using Ensemble Learning," *Sensors*, vol. 21, no. 17, Sep. 2021, doi: 10.3390/s21175777.
- [9] K. M. Jaheldeen, K. H. Bilal and A. A. Mustafa, "Handover Drive Test," *International Journal of Engineering, Applied and Management Sciences Paradigms*, vol. 22, Mar. 2015. [Online]. Available: [researchgate.net/publication/317032585](https://researchgate.net/publication/317032585)
- [10] M. K. Dahouda, S. Jin and I. Joe, "Machine Learning-Based Solutions for Handover Decisions in Non-Terrestrial Networks," *Electronics (Switzerland)*, vol. 12, no. 8, Apr. 2023, doi: 10.3390/electronics12081759.
- [11] F. M. Chang, H. L. Wang, S. Y. Hu and S. J. Kao, "An efficient handover mechanism by adopting direction prediction and adaptive time-to-trigger in LTE networks," in *Computational Science and Its Applications – ICCSA 2013*, vol. 7975, pp. 270–280, 2013, doi: 10.1007/978-3-642-39640-3\_20.
- [12] C. L. Vielhaus, J. V. S. Busch, P. Geuer, A. Palaos, J. Rischke, D. F. Külzer, V. Latzko and F. H. P. Fitzek, "Handover predictions as an enabler for anticipatory service adaptations in next-generation cellular networks," in *Proc. 20th ACM Int. Symp. on Mobility Management and Wireless Access*, 2022, pp. 19–27, doi: 10.1145/3551660.3560913.

- [13] M. Dzaferagic, B. Missi Xavier, D. Collins, V. D’Onofrio, M. Martinello and M. Ruffini, “ML-Based Handover Prediction Over a Real O-RAN Deployment Using RAN Intelligent Controller,” *IEEE Transactions on Network and Service Management*, vol. 22, no. 1, pp. 635–647, 2025, doi: 10.1109/TNSM.2024.3468910.
- [14] K. M. Istiaque, S. Sowad, M. M. S. Shafi and M. T. Kawser, “Reducing Handover and Ping-Pong Events in LTE Networks Via Q-Learning and Subtractive Clustering,” in *Proc. 2025 8th Int. Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, Oct. 2025. [Online]. Available: [researchgate.net/publication/396135987](https://researchgate.net/publication/396135987).
- [15] K. M. Istiaque, M. M. S. Shafi, S. S. Sowad and M. T. Kawser, “Adaptive Handover Optimization in LTE: Comparative Evaluation of Fuzzy Logic and DQN,” in *Proc. 10th IEEE Asia Pacific Conf. on Wireless and Mobile 2025 (APWiMob 2025)*, Oct. 2025. [Online]. Available: [researchgate.net/publication/396134205](https://researchgate.net/publication/396134205).
- [16] M. M. S. Shafi, K. M. Istiaque, S. S. Sowad and M. T. Kawser, “Handover Optimization in LTE Networks Using Contextual Bandit Reinforcement Learning and Real-World Data,” in *Proc. Int. Conf. on Computer Engineering, Network and Intelligent Multimedia (CENIM)*, Oct. 2025. [Online]. Available: [researchgate.net/publication/396132842](https://researchgate.net/publication/396132842).
- [17] X. Dong, K. Zhang, C. Sun, J. Zhang, A. Zhang, and L. Wang, “Towards 250-m Gigabits-Per-Second Underwater Wireless Optical Communication Using a Low-Complexity ANN Equalizer,” *Opt Express*, vol. 33, no. 2, p. 2321, Jan. 2025, doi: 10.1364/oe.549337.
- [18] A. Narwaria and A. P. Mazumdar, “SD-MoMa: ANN-Based Mobility Prediction and Adaptive Handover Management in SDWSN,” *Cluster Comput*, vol. 28, no. 6, Oct. 2025, doi: 10.1007/s10586-024-05006-0.
- [19] A. I. Bello, A. Y. Musa and H. S. Abdullahi, “Comparative Analysis of ANN and ANFIS in Forecasting GSM Handover Success Rate,” in *Proc. National Conf. on Engineering & Technology for Economic Transformation*, vol. 1, Sep. 2014. [Online]. Available: [researchgate.net/publication/283503306](https://researchgate.net/publication/283503306).
- [20] M. U. A. Khan, M. I. Babar, S. U. Rehman, D. Komosny, and P. H. J. Chong, “Optimizing Wireless Connectivity: A Deep Neural Network-Based Handover Approach for Hybrid LiFi and WiFi Networks,” *Sensors*, vol. 24, no. 7, Apr. 2024, doi: 10.3390/s24072021.
- [21] P. K. Sree, P. Chakrabarti, M. Margala, and S. S. N. Usha Devi, “Auto Encoders with Cellular Automata for Anomaly Detection,” *J. Electr. Syst.*, vol. 20, no. 2s, pp. 227–232, Apr. 2024, doi: 10.52783/jes.1131.
- [22] P. Wang, J. Zhang, Y. Li, Y. Guo and P. Li, “Breast Histopathological Image Classification Based on Auto-Encoder Reconstructed Domain Adaptation,” *Applied Sciences (Switzerland)*, vol. 14, no. 24, Dec. 2024, doi: 10.3390/app142411802.
- [23] S. J. Badashah, K. Moholkar, S. L. Bangare, G. Gupta, D. T., S. Francis, V. Hariram, B. Omarov, K. P. Rane and A. Raghuvanshi, “Enhanced Deep Auto Encoder Technique for Brain Tumor Classification And Detection,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 38, no. 3, p. 2031, Jun. 2025, doi: 10.11591/ijeecs.v38.i3.pp2031-2040.
- [24] P. V. Dinh, D. N. Nguyen, D. T. Hoang, Q. U. Nguyen and E. Dutkiewicz, “Multiple-Input Variational Auto-Encoder for Anomaly Detection in Heterogeneous Data,” Jan. 2025, [Online]. Available: <http://arxiv.org/abs/2501.08149>.
- [25] H. Powers and K. Edoh, “Outlier Detection Variational Autoencoder,” *Neural Comput Appl*, vol. 37, no. 21, pp. 16871–16882, Jul. 2025, doi: 10.1007/s00521-025-11357-5.
- [26] Z. Lu, Z. Chu, M. Zhu, and X. Dong, “Unsupervised Feature Learning Using Locality-Preserved Auto-Encoder with Complexity-Invariant Distance for Intelligent Fault Diagnosis of Machinery,” *Applied Intelligence*, vol. 55, no. 6, Apr. 2025, doi: 10.1007/s10489-025-06278-8.
- [27] A. A. Tafere, T. S. Hailemariam, and T. T. Debella, “Deep Learning-Powered Equalization with Autoencoders for Improved 5G Communication,” in *2025 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation, IATMSI 2025*, Institute of Electrical and Electronics Engineers Inc., 2025. doi: 10.1109/IATMSI64286.2025.10984954.
- [28] X. Tian, “A Deep Convolutional Autoencoder-Enabled Channel Estimation Method in Intelligent Wireless Communication Systems,” *International Journal of Intelligent Systems*, vol. 2024, no. 1, 2024, doi: 10.1155/2024/9343734.
- [29] P. Samarathunga, H. Rezaei, M. Lokumarambage, T. Sivalingam, N. Rajatheva, and A. Fernando, “An Autoencoder-Based Task-Oriented Semantic Communication System for M2M Communication,” *Algorithms*, vol. 17, no. 11, Nov. 2024, doi: 10.3390/a17110492.
- [30] N. A. Letizia and A. M. Tonello, “Capacity-Driven Autoencoders for Communications,” *IEEE Open Journal of the Communications Society*, vol. 2, pp. 1366–1378, 2021, doi: 10.1109/OJCOMS.2021.3087815.

- [31] B. Zhang, G. Zheng and N. Van Huynh, "A Hybrid Quantum-Classical Autoencoder Framework for End-to-End Communication Systems," Dec. 2024, [Online]. Available: <http://arxiv.org/abs/2412.20241>
- [32] Z. I. Tabi, T. X. Vu, M. Asad and H. Nishimura, "Quantum-Classical Autoencoder Architectures for End-to-End Radio Communication," *IEEE Access*, vol. 13, pp. 82181–82192, 2025, doi: 10.1109/ACCESS.2025.3566207
- [33] X. Sheng, Z. Shen and G. Xiao, "Contrastive Predictive Autoencoders for Dynamic Point Cloud Self-Supervised Learning," *Proceedings of the 37th AAAI Conference on Artificial Intelligence (AAAI-23)*, pp. 9802–9810, 2023, doi: 10.1609/aaai.v37i8.26170.
- [34] B. Xue *et al.*, "A Contrastive Autoencoder with Multi-Resolution Segment-Consistency Discrimination for Multivariate Time Series Anomaly Detection," *Applied Intelligence*, vol. 53, no. 23, pp. 28655–28674, Dec. 2023, doi: 10.1007/s10489-023-04985-8.
- [35] D. Luo, H. Zhou, J. Bae and B. Yun, "Combining Contrastive Learning with Auto-Encoder for Out-of-Distribution Detection," *Applied Sciences (Switzerland)*, vol. 13, no. 23, Dec. 2023, doi: 10.3390/app132312930.
- [36] C. Wu *et al.*, "Attack Detection Model For BCOT Based on Contrastive Variational Autoencoder and Metric Learning," *Journal of Cloud Computing*, vol. 13, no. 1, Dec. 2024, doi: 10.1186/s13677-024-00678-w.
- [37] E. Nikougoftar, "Outlier Detection in IoT using Trained Autoencoder and Contrastive Loss," *Preprint*, 19 August 2024. [Online]. Available: 10.21203/rs.3.rs-4657376/v1.
- [38] X. Zheng, Z. Wu, H. Meng and L. Cai, "Contrastive Auto-Encoder for Phoneme Recognition," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2014, pp. 2529–2533. doi: 10.1109/ICASSP.2014.6854056.
- [39] N. Liu, G. Wu, Y. Huang, X. Chen, Q. Li and L. Wan, "Unsupervised Contrastive Hashing with Autoencoder Semantic Similarity for Cross-Modal Retrieval in Remote Sensing," *IEEE J Sel Top Appl Earth Obs Remote Sens*, vol. 18, pp. 6047–6059, 2025, doi: 10.1109/JSTARS.2025.3538701.
- [40] H. M. Pham, A. Saeed and D. Ma, "C-MELT: Contrastive Enhanced Masked Auto- Encoders For ECG-Language Pre-Training," May 2025, [Online]. Available: [arxiv.org/html/2410.02131v1](https://arxiv.org/html/2410.02131v1)
- [41] M. M. S. Shafi, T. S. Ahona and Y. H. Saad, "A Weather-Aware Risk Stratification Framework for RLF Prediction Using Contrastive Autoencoders and Survival Analysis," in *Proceedings of the 10th IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob)*, 2025. [Online]. Available: [researchgate.net/publication/395962722](https://researchgate.net/publication/395962722)
- [42] S. Farthofer, M. Herlich, C. Maier, S. Pochaba, J. Lackner and P. Dorfinger, "An Open Mobile Communications Drive Test Data Set and Its Use for Machine Learning," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 1688–1701, 2022, doi: 10.1109/OJCOMS.2022.3210289.
- [43] R. H. Ratul, M. Iqbal, J.-Y. Pan, M. M. Al Deen, M. T. Kawser and M. M. Billah, "Performance Comparison Between VoLTE and non-VoLTE Voice Calls During Mobility in Commercial Deployment: A Drive Test-Based Analysis," in *Proceedings of the 2023 10th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, Palembang, Indonesia, 2023, pp. 446–452, doi: 10.1109/EECSI59885.2023.10295903.
- [44] S. Manas Kala *et al.*, "Architecture, Performance, and Usability of Mobile Cellular Network Monitoring Applications for Data-Driven Analysis," *IEEE Access*, vol. 12, pp. 88426–88444, 2024, doi: 10.1109/ACCESS.2024.3412752.
- [45] M. M. S. Shafi, K. M. Istiaque, S. S. Sowad, and M. T. Kawser, "Drive-Test-Based LTE Handover Dataset for Cellular Mobility Studies in Urban Bangladesh," *Mendeley Data*, vol. V1, 2025, doi: 10.17632/n2pvmtyjn2j.1.