Journal of Network and Systems Management (2026) 34:28
https://doi.org/10.1007/s10922-025-10002-9

RAHL: A Residual-based Adaptive Huber Loss for Robust CQI Prediction in 5G Networks

Mina Kaviani[1] · **Jurandy Almeida**[1] · **Fábio L. Verdi**[1]

Received: 8 April 2025 / Revised: 17 October 2025 / Accepted: 28 October 2025

Abstract

The Channel Quality Indicator (CQI) is an essential metric in 5G networks, supporting adaptive infrastructure optimization to ensure a high level of Quality of Service (QoS). Recent studies have explored enhancing CQI estimation using machine learning techniques. A key factor in training an accurate model is selecting an appropriate loss function. Two widely used loss functions are Mean Squared Error (MSE) and Mean Absolute Error (MAE). Generally, MSE emphasizes outliers more heavily, whereas MAE prioritizes the majority of the data points. In this work, we highlight the advantages of the Huber loss function for CQI prediction, as it effectively integrates the strengths of both MSE and MAE. The Huber loss achieves this by smoothly transitioning between the two, governed by a user-defined hyperparameter called delta. However, manually selecting the optimal delta to balance the sensitivity to minor errors (as in MAE) and robustness to outliers (as in MSE) is a challenging task. To overcome this limitation, we introduce a novel loss function called Residual-based Adaptive Huber Loss (RAHL). RAHL incorporates a learnable residual into the delta parameter, allowing the model to adapt dynamically to the error distribution in the data. This approach enhances robustness against outliers while preserving precision for inlier data. Additionally, we leverage various features to improve CQI prediction accuracy. To evaluate RAHL's effectiveness, we evaluate its performance across multiple architectures, including Long Short-Term Memory (LSTM), parallel 1D Convolutional Neural Networks (CNN1D), and the Informer model. Experimental results confirm that RAHL consistently enhances performance across these models, demonstrating its adaptability and reliability. These findings establish RAHL as a promising solution for improving CQI prediction in 5G networks.

Keywords Channel quality indicator (CQI) · 5G networks · Loss functions (MSE, MAE, Huber, Residual-based Adaptive Huber Loss) · Time series forecasting models (LSTM, CNN1D, Informer) · Adaptive optimization in wireless networks

Extended author information available on the last page of the article

Published online: 15 November 2025

1 Introduction

The deployment of fifth-generation (5G) networks marks a significant advancement in telecommunications, introducing three key service categories: enhanced mobile broadband (eMBB), ultra-reliable low-latency communications (uRLLC), and massive machine-type communications (mMTC). The effectiveness of these networks relies on the efficient management of the 5G Core, where Network Functions (NFs) play a crucial role. Ensuring optimal communication requires continuous monitoring of radio signal quality indicators, which are essential for managing 5G links. These indicators, including the Channel Quality Indicator (CQI), Signal-to-Noise Ratio (SNR), Reference Signal Received Quality (RSRQ), Reference Signal Received Power (RSRP), and Received Signal Strength Indicator (RSSI), provide valuable insights into link quality. User Equipment (UE) collects these metrics and transmits them to the evolved Node B (eNB), the base station responsible for communication. The eNB's radio network controller then adjusts channel modulation based on these data, optimizing communication links for UEs.

The reactive nature of this approach presents challenges since the gathered indicators represent past events. Depending only on reactive operations may not be sufficient for optimal performance, particularly in 5G networks that rely on short-range, high-frequency radio signals and mobile UEs. To address this issue and enable more proactive network management, predictive analytics, and machine learning algorithms are becoming increasingly essential. These technologies can process historical data, detect patterns, and anticipate potential network problems or changes. Using predictive insights allows network operators to take preemptive actions, improving channel modulation, resource distribution, and overall network efficiency [1–5].

Modern Long-Term Evolution (LTE) and 5G networks have become too complex for traditional management, prompting the use of Machine Learning-driven automation in Self-Organizing Networks to improve efficiency and reduce costs. A major challenge is load balancing in RANs, which requires adaptive solutions that consider multiple factors. This paper proposes a novel Clipped Double Q-Learning algorithm that combines resource utilization, latency, and CQI for smarter handover decisions, showing significant performance improvements over traditional methods in simulations [6]. Accurate downlink link adaptation in uRLLC is challenged by unpredictable interference from neighboring cells. To tackle this, researchers present enhancements to the 5G New Radio(NR) CQI measurement and reporting processes aimed at better estimating the lower percentiles of user channel quality [7]. CQI plays a pivotal role in 5G resource allocation, particularly in machine-type communication scenarios, highlighting its importance as a core metric in intelligent network management [8]. A machine learning framework deployed at the network edge uses CQI, jitter, and other metrics for real-time traffic classification and load prediction. This enables dynamic slice reallocation, reducing latency, improving throughput, and enhancing energy efficiency while continuously adapting to changing network conditions [9]. A combined method using Support Vector Machines (SVM) and Neural Networks (NN) was developed to adapt the CQI reporting frequency according to predicted channel stability. Findings indicate that Neural Networks provide more accurate channel dynamic predictions than SVM, effectively lowering reporting overhead [10]. In the context of UAV communications, deep learning methods have been applied to predict CQI using historical feedback, addressing the issue of CQI aging [11]. A CQI-based traffic control strategy in Open RAN was shown to improve QoS by adaptively managing VR traffic. This highlights CQI's key role as both a predictive and actionable metric for learning-driven 5G network optimization [12].

In communication system design, CQI plays a vital role. It represents the channel condition, allowing base stations to dynamically adjust service quality according to real-time variations. This adaptability improves communication efficiency [1]. However, predicting CQI accurately is challenging due to its fluctuating nature, spanning values from 0 to 15 and being influenced by multiple environmental factors. Inaccurate predictions can significantly impact 5G network performance, as CQI guides modulation decisions and resource allocation. Since base stations depend on CQI feedback to optimize bandwidth distribution, errors in estimation can lead to reduced Quality of Experience (QoE) for users, lower application data rates, and inefficient utilization of network resources.

Machine learning models, despite their effectiveness, struggle to predict CQI with high accuracy due to sudden variations and fluctuations, which can result in suboptimal performance. Consequently, exploring alternative approaches becomes essential. Selecting a suitable error metric, such as Mean Squared Error (MSE) or Mean Absolute Error (MAE) to assess the CQI signal quality is challenging, as each metric has its own limitations. In addition, the unique conditions that influence the accuracy of CQI within the 5G environment further complicate this decision.

To address this challenge, we investigate the Huber loss function. The Huber loss [13] effectively combines the benefits of MSE and MAE by introducing a hyperparameter, delta, which controls the transition between these two loss behaviors. However, selecting an appropriate delta value manually is not straightforward, as it requires balancing sensitivity to small errors (MAE) and resistance to outliers (MSE). To overcome this limitation, we propose the Residual-based Adaptive Huber Loss (RAHL), where delta is treated as a trainable parameter rather than a fixed value. This adaptive approach enables the model to dynamically adjust its robustness to outliers during training, optimizing the trade-off between generalization and precision [13, 14].

The prediction of CQI is inherently a time-dependent task, as the quality of a communication channel is influenced by previous states and fluctuates over time. Given the temporal nature of this, it is crucial to use models that effectively capture these time-based patterns. In this study, we selected three models, Long-Short-Term Memory (LSTM), CNN-1D, and Informer, each offering distinct advantages for handling time series data. Previous research has already demonstrated the effectiveness of LSTM, CNN-1D, and Informer for various time series prediction tasks, such as stock price prediction, energy consumption prediction, and traffic flow modeling [15–17]. These studies highlight the potential of such models to capture temporal dependencies and identify patterns in sequential data. Building on this existing work, we aim to use these models for CQI prediction, applying them to our data set to predict channel quality in a time-dependent manner.

LSTMs are well suited to capture long-term dependencies, which makes them ideal for modeling CQI fluctuations over time. CNN-1D efficiently extracts local features and short-term patterns, reducing computational cost and maintaining predictive accuracy. Informer, designed for long-sequence forecasting, uses self-attention mechanisms to model both global and local dependencies efficiently. Using these models, we ensure a comprehensive approach to CQI prediction, balancing short- and long-term pattern recognition while optimizing computational efficiency. Through this investigation , we systematically evaluated the impact of RAHL in model training, comparing its performance with alternative loss functions such as Huber, MSE, and MAE. using the Mean Absolute Percentage Error (MAPE) as the evaluation metric, our results revealed that RAHL consistently produced lower MAPE values across all models compared to other loss functions, indicating improved model accuracy. This research contributes to a deeper understanding and a broader application of machine learning models in the forecasting of signal quality indicators, ultimately leading to improved 5G network performance.

This paper is structured as follows. In Sect. 2, we explore previous studies that capture current advances in the field. Section 3 describes the features and operational aspects of the proposed ML models and RAHL. The key quantitative outcomes in various scenarios are discussed in Sect. 4. Finally, Sect. 5 summarizes the findings and provides recommendations for future investigations.

2 Related Works

In the realm of network and system management, the emergence of advanced applications operating on 5G networks, such as AR/VR, cloud gaming, robotics, and autonomous driving, has introduced a host of new challenges. These applications demand ultra-low latency and high bandwidth, placing unprecedented strain on telecom infrastructures. Managing, controlling and allocating resources across the entire end-to-end pipeline, from the RAN to the transport infrastructure and core network, has become increasingly complex. To meet these stringent requirements, intelligent ML solutions are gaining significant traction. In particular, models that leverage real-time network indicators, such as the CQI, enable predictive and adaptive resource allocation strategies. By anticipating network conditions, these models support more efficient scheduling decisions and improve QoS for latency-sensitive and bandwidth-intensive applications.

The study in [6] proposed a clipped double Q learning algorithm that improves the reward mechanism by incorporating several performance indicators, such as CQI, delay, jitter, and resource utilization, leading to better performance in RAN slicing and mobility management. Several works have highlighted the critical role of the CQI in enabling adaptive and predictive resource allocation in 5G. In [7], the authors focused on improving downlink link adaptation for uRLLC services by proposing improvements to CQI reporting formats and filtering methods for estimating the tail SINR, significantly reducing latency in system-level evaluations. The work in [8] also demonstrated the usage of CQI for the allocation of radio resources in machine-type communication scenarios, further highlighting its role as a foundational metric in intelligent 5G resource management systems.

Another promising direction involves the use of edge intelligence to support real-time, autonomous network operations. The study in [9] proposed a machine learning framework deployed at the network edge that performs real-time traffic classification and load prediction using CQI, jitter, and performance metrics. This enables dynamic and context-sensitive slice reallocation, resulting in lower latency, better throughput, and improved energy efficiency. In addition, continuous online learning ensures that the model adapts to changing network conditions.

In parallel, recent research has explored machine learning-based optimization of CQI reporting and prediction to improve both performance and signaling efficiency. For example, [10] introduced a hybrid approach using SVM and NN to dynamically adjust the CQI reporting frequency based on predicted channel stability, thereby reducing overhead without compromising accuracy. The results showed that NN outperforms SVM in forecasting channel dynamics. Likewise, [11] proposed a deep learning-based CQI prediction mechanism tailored to UAV communications, addressing the effect of CQI aging by forecasting future channel states using historical feedback data. Furthermore, [12] demonstrated a CQI-informed traffic control strategy in Open RAN, where adaptive handling of VR traffic leads to better QoS. These findings collectively underscore the centrality of CQI as both a predictive and an actionable metric, making it a natural and impactful target for learning-based approaches aimed at optimizing 5G network performance.

In the field of anticipatory networking, where accurate forecasting of wireless channel quality is essential, researchers have traditionally relied on past channel data to inform future predictions.

However, a recent study by [2] addresses the challenge of predicting cross-channel quality. Their novel transfer learning framework leverages both Parera CNN-1D and LSTM to predict channel quality for specific frequency carriers. Importantly, the study uses two different model architectures, both trained with the RMSE loss function, and demonstrates the superior performance of LSTMs compared to other algorithms.

In their research, [18] presents Informer, an efficient Transformer-based model designed for long-sequence time-series forecasting. They introduce several innovations, including ProbSparse self-attention, self-attention distilling, and a generative-style decoder, to overcome the challenges of high complexity and long inference times typically associated with traditional Transformers.

Numerous challenges in the literature on learning, optimization, and statistics [19, 20] highlight the necessity of robust solutions, requiring models to be trained or optimized in ways that reduce their sensitivity to outliers. This ensures that the models remain resilient against the influence of outliers while being less affected by the impact of inliers (that is, typical data points) [21, 22]. This approach is commonly applied in tasks such as parameter estimation and learning, especially when using a robust loss function like absolute error, which is more advantageous than using non-robust losses like quadratic error due to its resistance to large errors. In such cases, it is important to go beyond conventional outlier detection techniques [23, 24] and focus on designing loss functions that inherently resist the impact of outliers. Among these, the Huber loss function stands out as an effective choice, as it balances the mean squared error and absolute error, providing robustness to outliers while still being sensitive to inliers. Its flexible nature makes it particularly suitable for situations where a balance between these two extremes is critical for optimal model performance and generalization [13].

The Huber loss, as proposed by Jadon et al. [25], is a robust combination of the MSE and MAE loss functions. Introduces a hyperparameter delta (δ) that determines the transition point between the two functions. When the absolute difference between predicted and true values is smaller than δ, Huber loss behaves like MSE. Otherwise, it transitions to a linear MAE-based function. This dual nature helps balance outliers sensitivity and stability in gradient-based optimization, and the Huber loss function offers the advantage of limiting the impact of outliers through linearity above the threshold δ, unlike MSE.

Adaptive Huber regression offers a robust and data-driven approach to managing outliers and heavy-tailed distributions in large datasets, in contrast to traditional methods. This technique dynamically adjusts its parameters to strike a balance between bias and robustness and has been shown to be effective in a range of data scenarios, including those with heavy-tailed distributions [26].

In [27], the authors introduce a method for optimizing the exact Huber loss in scalar regression within a semi-supervised learning framework. Their approach integrates multiview learning, which utilizes information from various data perspectives, along with manifold regularization. Furthermore, they propose a data-driven adaptation of the Huber loss threshold and actively manage the use of labeled data to minimize the effects of noisy or inconsistent annotations during the training process.

In contrast to [26] and [27], which focus on learning the hyperparameter delta directly, RAHL instead aims to learn a residual that is added to it. This approach simplifies deep model training, resulting in enhanced performance.

Drawing on previous research, Table 1 provides a comparative overview of the main features of each method. In this study, our main focus is on predicting the CQI,

Table 1 Summary listing the main characteristics of the works cited

Work	5G	CQI	LSTM	CNN-1D	Huber	Adaptive Huber	Informer
Bakri et al. [10]	✓	✓					
Pocovi,et al. [7]	✓	✓					
Beshley et al. [8]	✓	✓					
Tsourdinis et al. [9]	✓	✓					
Casparsen et al. [12]	✓	✓					
Iturria-Rivera et al. [6]	✓	✓					
Bartoli et al. [11]	✓	✓					
Parera et al. [2]	✓	✓					
Yin et al. [1]	✓	✓	✓				
Vankayala and Shenoy [3]	✓	✓		✓			
Gokcesu and Gokcesu [13]					✓		
Sun et al. [26]						✓	
Cavazza and Murino [27]					✓		
Jadon et al. [25]					✓		
Zhou et al. [18]							✓
RAHL	✓	✓	✓	✓	✓	✓	✓

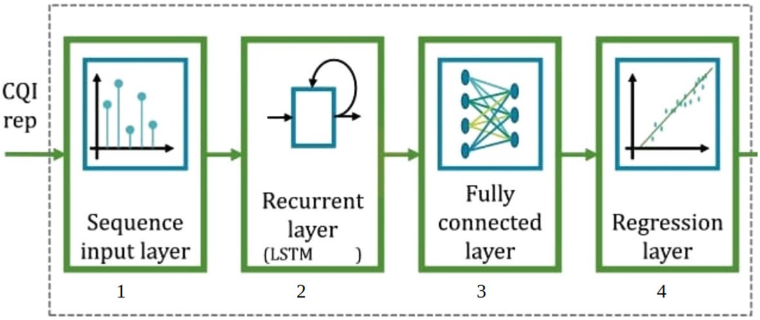


Fig. 1 Considered LSTM model

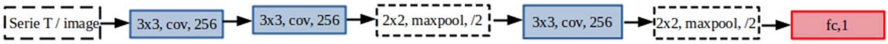


Fig. 2 Considered CNN 1D model

a vital parameter in 5G networks. Although earlier work has explored similar directions, our method stands out due to its unique optimization strategy. Considering the highly dynamic nature of CQI values in 5G environments, our goal is to enhance the prediction accuracy by tailoring the loss function accordingly. Specifically, we propose a novel loss function, RAHL, designed to adapt to CQI fluctuations during training and contribute significantly to performance improvement.

3 System Model

In this paper, we present RAHL for the prediction of 5G characteristics. Our model effectively uses temporal dependencies, spectral features, and historical data to improve its predictive performance, with the goal of achieving optimal CQI estimation.

3.1 The Deep Learning-Based CQI Prediction Models

Building on previous work, we employ LSTM, CNN1D, and the Informer model due to their ability to capture long-term dependencies and efficiently process sequential data, making them suitable for accurately predicting CQI values. It is not our intention to delve deeply into the architectures, but rather to provide a brief introduction to their functionality.

Figure 1 [11] depicts the LSTM architecture, which is designed to process input sequences with one-dimensional features. The sequence is initially passed through an LSTM layer with 64 hidden units, allowing it to capture sequential patterns. This is followed by a fully connected layer with 64 units that applies non-linear transformations to the LSTM output. The final regression layer produces a single output value, which corresponds to the prediction of the model for the given sequence.

The Parera CNN-1D model as shown in Fig. 2 [2], originally developed to predict CQI in 4 G networks at future time moments. Its architecture consists of three one-dimensional convolutional layers that act as feature extractors, followed by a Fully Connected (FC) output layer designed for the regression task (with one neuron). Due to the use of one-dimensional convolutional layers, the model takes as input finite sequences of one-dimensional data, such as vectors or time series, and outputs a predicted CQI value for a future time instant.

The Informer model shown in Fig. 3 [18] efficiently processes long-sequence data using a ProbSparse self-attention mechanism in the encoder to focus on critical information, with self-attention distillation reducing the network size while maintaining robustness through stacked layers. The decoder generates output sequentially by padding target elements with zeros, computing weighted attention from feature maps, and predicting output in a generative style, enabling scalability and accuracy in long-sequence forecasting.

3.2 Residual-based Adaptive Huber Loss (RAHL): The Improved Huber Loss for 5G Features Prediction

In regression tasks, two widely utilized loss functions are the absolute loss, $L_1(y, f_\theta(x)) = |y - f_\theta(x)|$, and the squared loss, $L_2(y, f_\theta(x)) = (y - f_\theta(x))^2$, each offering distinct benefits that require thoughtful selection based on their characteristics. The squared loss, with its strong convexity, tends to enable faster convergence, while the absolute loss is preferred for its resistance to outliers. This contrast underscores the value of combining the advantages of both loss functions, resulting in models that can efficiently handle outliers while achieving rapid convergence and precise model fitting.

The Huber loss is a widely used approach that combines the characteristics of quadratic and absolute losses, providing a robust function that ensures quick con-

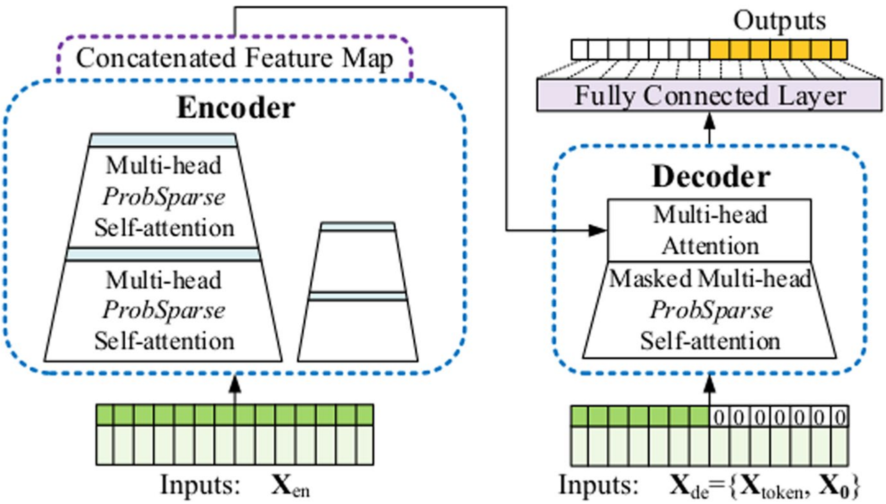


Fig. 3 Considered Informer model

vergence and resistance to outliers [23]. It is commonly applied in regression tasks, especially when the data includes outliers or noise. By integrating the benefits of MSE (quadratic loss) and MAE (absolute loss), Huber loss delivers greater resilience against extreme values compared to using either loss function on its own. The key transition point in the Huber loss defines the shift from quadratic to absolute loss, making this parameter a crucial hyperparameter that significantly influences model performance. However, determining the optimal transition parameter can be difficult, often requiring substantial hyperparameter tuning to identify the most effective value [28].

Formally, the Huber loss is expressed as (Eq. 1):

$$H(y, f_\theta(x)) = \begin{cases} \frac{1}{2}(y - f_\theta(x))^2, & \text{if } |y - f_\theta(x)| \leq \delta \\ \delta|y - f_\theta(x)| - \frac{1}{2}\delta^2, & \text{if } |y - f_\theta(x)| > \delta \end{cases}, \tag{1}$$

In this framework, y represents the true value, while $f_\theta(x)$ denotes the predicted value of the model, with θ as the training parameters. The hyperparameter δ is essential as a scaling factor, which guides the change in loss penalty from L_2 to L_1. This parameter plays a vital role in striking a balance between model accuracy and its ability to manage outliers. Selecting the right δ is crucial, as it determines when the loss function switches from focusing on precision to prioritizing robustness. A smaller δ results in behavior similar to L_1, enhancing precision but reducing the resilience of the model to outliers. On the other hand, a larger δ makes the loss function more like L_2, improving robustness at the expense of some precision.

Manually tuning the hyperparameter δ poses various challenges when using the Huber loss in the regression model training. These challenges include subjectivity, dataset dependence, increased computational cost, potential overfitting, and difficulty maintaining an optimal balance between accuracy and robustness. These issues emphasize the importance of using a more systematic approach, incorporating advanced hyperparameter optimization methods for improved regression performance. To address these challenges, we introduce the RAHL [29],[1] a novel technique that enables the model to automatically select the optimal penalty scheme.

Figure 4 [29] illustrates a comparison between RAHL and other commonly used loss functions, highlighting the influence of outliers on regression models. In this figure, outliers are depicted as red points, while inliers are shown as blue points. With the MSE loss function, the resulting model (purple line) is heavily affected by outliers, causing significant deviations. On the other hand, the MAE loss produces a model (red line) that fits the inliers well but ignores the outliers. The Huber loss offers a balance by allowing manual tuning of the δ hyperparameter. A large δ makes the Huber loss behave similarly to MSE (black line), while a small δ makes it more similar to MAE (brown line). Therefore, the choice of δ is critical, as it controls the robustness of the model to outliers and influences its overall performance in regression tasks.

RAHL has been specifically designed to address this challenge by dynamically adjusting the penalty function, effectively balancing the model's precision on inliers

[1] This work is an extension of a previous paper published in a Brazilian conference.

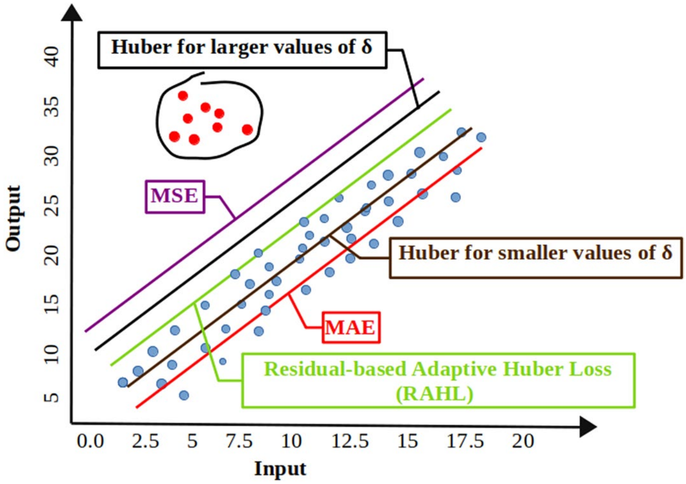


Fig. 4 Exploring how different loss functions shape regression models

(blue points) with its resistance to outliers (red points). As shown by the green line, RAHL adapts to the underlying data distribution, optimizing the trade-off between accuracy and robustness. This adaptive mechanism boosts the model's reliability, leading to better performance in regression tasks that are impacted by outliers.

Mathematically, RAHL follows the same formulation as the Huber loss; however, rather than relying on a fixed hyperparameter δ, it dynamically determines δ based on the data. This adaptive adjustment allows RAHL to optimize the trade-off between inlier sensitivity and robustness to outliers. The calculation of δ is given by (Eq. 3 [29]):

$$RAHL(y, f_\theta(x)) = \begin{cases} \frac{1}{2}(y - f_\theta(x))^2, & \text{if } |y - f_\theta(x)| \le \delta \\ \delta|y - f_\theta(x)| - \frac{1}{2}\delta^2, & \text{if } |y - f_\theta(x)| > \delta \end{cases}, \tag{2}$$

$$\delta = \alpha + \mathrm{ELU}(\beta), \tag{3}$$

In this formulation, α is a positive hyperparameter that establishes the initial value of δ, while β is a learnable parameter that is added to α. To ensure that δ remains positive and within a bounded range, the Exponential Linear Unit (ELU) function [30] is used in β. ELU is an activation function that exhibits linear behavior for positive inputs, while applying an exponential transformation to negative inputs. The mathematical definition of ELU is given by (Eq. 4):

$$ELU(x) = \begin{cases} x, & \text{if } x \ge 0 \\ \alpha(e^x - 1), & \text{if } x < 0 \end{cases}, \tag{4}$$

In this case, α is a constant that controls the smoothness of the function when the inputs are negative, typically set to 1.0. By selecting this constant to match the value of the hyperparameter α of Eq. 3 (which represents the initial value of δ), we constrain the output to the range $[-\alpha, +\infty)$, thus ensuring that δ remains positive.

Figure 5 [29], illustrates how different values of δ influence the Huber loss compared to the absolute (MAE) and quadratic (MSE) losses. Each curve shows the relationship between the residual (x-axis), which is the difference between the actual value (y) and the model's prediction ($\hat{y}$), and the loss (y-axis). As δ increases, the Huber loss tends to mimic MSE, while decreasing δ makes it closer to MAE. The key benefit of RAHL is its flexibility to adapt according to the error distribution, allowing the model to adjust its behavior to resemble MSE or MAE depending on the characteristics of the data.

4 Experimental Evaluation

This section reveals the intricate details of our data collection and methodology, meticulously describing each step and ensuring a clear understanding of the research framework.

4.1 Data Collection and Preparation

The data set comprises 83 records of Internet transmissions recorded by G-NetTrack v18.7 on a Samsung S10 connected to an Irish mobile operator. It includes 3142 min of transmission logs, organized into three services (File Download, Amazon Prime, and Netflix) and two mobility patterns (Static and Vehicular). The logs, limited by an 80 GB data plan, are stored in a CSV file with fixed features and variable data points. The data set provides detailed attributes such as timestamp, geographical coordinates,

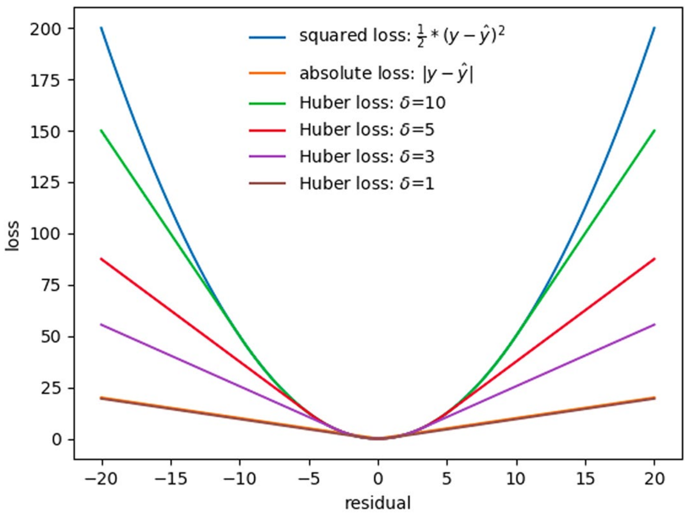


Fig. 5 The Huber loss for various values of δ, moving between MSE and MAE

node velocity, mobile operator (anonymized), cell ID, network mode, bitrates, device state, and various signal quality indicators for primary and neighboring cells [31].

We tackle missing data challenges by leveraging NaN (Not a Number) for placeholder values, a standard practice in Python's numerical domain. Through preprocessing techniques such as *MinMaxScaler*, we achieve uniformity and improve the performance of the model. Recognizing the importance of time order, we employ a sliding window approach (w = 32, shifted T times) to preserve the temporal context within our analysis. Figure 6 [2] shows this procedure in detail. In the Informer model, we use a "secondly" frequency (denoted by s) for time feature encoding to capture fine-grained temporal patterns, enabling the model to better detect high-frequency variations and subtle changes crucial for accurate prediction.

4.2 Training Procedure

To take advantage of the inherent time dependence of our data, we strategically partition them by timestamps, reflecting real-world dynamics, where past information heavily influences future predictions. Each CSV data set is meticulously divided, dedicating 80% to training and reserving 20% for rigorous testing.

In our previous work [29], we predicted CQI based solely on historical CQI values using an LSTM model with different loss functions. However, CQI is influenced by multiple factors, including signal strength and network conditions. To improve prediction accuracy and capture a more complete representation of the wireless environment, we expanded our feature set to include SNR, RSRP, and RSRQ in addition to CQI. These additional features provide valuable information about signal quality and network reliability, allowing the model to make more informed and robust predictions. CQI serves as a high-level indicator of channel quality, while SNR captures the signal's strength relative to noise, providing critical insight into reliability. RSRP contributes information about signal power, allowing the model to assess channel strength, and RSRQ offers additional context by considering interference and overall signal quality. This multidimensional approach enhances predictive accuracy and increases the model's robustness against noise and incomplete data, making it well-suited for dynamic and challenging environments.

However, incorporating multiple features also presents certain challenges. The preprocessing requirements are more complex, as each feature must be normalized

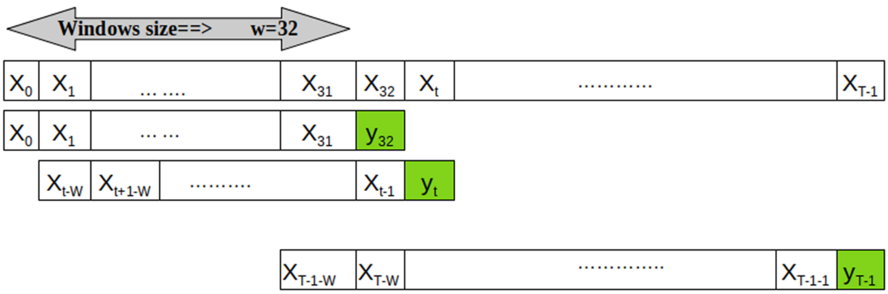


Fig. 6 Sliding windows for time series forecasting

and aligned in time, increasing the risk of introducing errors during data preparation. Additionally, a larger input space can lead to higher model complexity, requiring careful regularization to prevent overfitting, especially when the dataset is small. Computational costs also increase, in terms of both training and inference time, as the model must process more inputs.

To assess the impact of RAHL on prediction accuracy, we incorporated these features into the model to examine how RAHL adapts to varying error distributions. The findings indicated that RAHL contributed to improved model performance, demonstrating its robustness and improving the precision of CQI prediction under different test conditions.

4.3 Implementation Details

We implemented the entire project in Python 3.10, using PyTorch 2.1 as the primary deep learning framework. The experiments were initially prototyped using Google Colab Pro, and the final training was performed on the SDumont supercomputer at the Laboratório Nacional de Computação Científica (LNCC/MCTI), Brazil [32], which offers access to high performance GPU clusters.

In addition, we defined the network architecture and the training hyperparameters for the models, which are summarized in Tables 2, 3, 4.

All source code, datasets, data preprocessing scripts, training configurations are publicly available in our GitHub repository.[2]

This ensures complete reproducibility and allows future extensions of our work.

4.4 Performance Metrics

To facilitate an equitable evaluation of model performance across various loss functions, we employ the Mean Absolute Percentage Error (MAPE). In our methodology, models are trained using different loss functions, with MAPE serving as the main

Table 2 Network architecture and training hyperparameters for the LSTM model

(a) Network architecture	
Input size	4
Hidden size	64
Number of stacked layers	1
Number of FC layers	1
FC hidden size	64
Activation function	ReLU
Output size	1
(b) Training hyperparameters	
Number of epochs	300
Mini-batch size	24
Window size	36
Initial learning rate	0.001
Optimizer	Adam

[2] https://github.com/dcomp-leris/RAHL_CQI_Prediction.git.

Table 3 Network architecture and training hyperparameters for the CNN1D model

(a) Network architecture	
Input size	4
Hidden size (conv layers)	256
Number of convolutional layers	3
Number of pooling layers	2
Number of FC layers	1
FC hidden size	None
Activation function for FC layers	ReLU
Dropout probability	0.2
Output size	1
(b) Training hyperparameters	
Number of epochs	300
Mini-batch size	24
Window size	36
Initial learning rate	0.001
Optimizer	Adam

Table 4 Network architecture and training hyperparameters for the Informer model

(a) Network architecture	
Input size	4
Encoder size	512
Decoder size	512
Number of encoder layers	3
Number of decoder layers	3
Attention heads	8
Hidden size in feedforward	2048
Activation function	ReLU
Output size	1
(b) Training hyperparameters	
Number of epochs	100
Mini-batch size	24
Initial learning rate	0.001
Optimizer	Adam

metric for both validation and testing. Commonly applied in statistical analysis and time series forecasting, MAPE quantifies error as a percentage, as illustrated in Eq. 5, where smaller values correspond to improved accuracy [33].

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{Y_i - \hat{Y}_i}{Y_i}\right| \times 100\% \tag{5}$$

4.5 Experimental Results

Initially, as shown in Table 5(a), (b) and (c), we trained models using the Huber loss function, where we manually selected the hyperparameter δ. To determine the best

Table 5 MAPE values for CNN 1D, LSTM, and Informer across different loss functions compared with RAHL

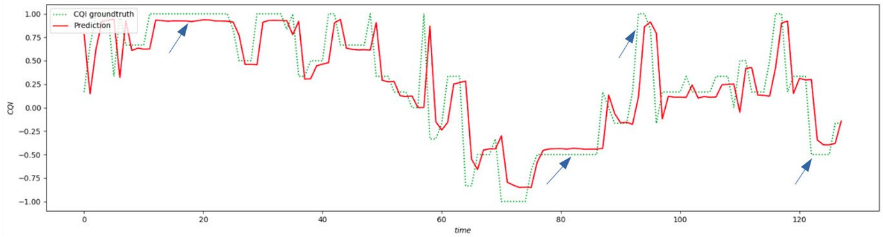
Dataset	$\delta = 1$	$\delta = 1.5$	$\delta = 2$	$\delta = 2.5$	$\delta = 3$	RAHL	Huber (best δ)	MAE	MSE
(a) MAPE values for CNN 1D across different loss functions compared with RAHL									
Download driving	17.90	21.89	25.43	26.64	23.46	**16.14**	17.90	16.90	21.19
Amazon prime driving	29.03	27.35	24.99	38.40	30.56	**22.59**	24.99	28.46	28.70
Netflix driving	30.63	28.11	30.47	30.47	28.17	**21.66**	28.11	25.14	30.47
Download static	22.43	21.70	23.11	23.11	22.87	**17.46**	21.70	17.69	21.38
Amazon prime static	27.67	37.31	27.41	32.05	26.22	**15.03**	26.22	24.05	34.52
Netflix static	11.91	10.60	10.76	9.10	10.21	**7.40**	9.10	7.58	9.83
(b) MAPE values for LSTM across different loss functions compared with RAHL									
Download driving	18.09	17.80	17.54	18.10	18.10	**11.11**	17.54	12.21	18.09
Amazon prime driving	26.73	20.85	21.08	21.80	21.08	**20.68**	20.85	20.80	25.61
Netflix driving	24.35	25.98	26.02	26.02	26.02	**22.97**	24.35	20.94	25.09
Download static	18.13	18.16	18.16	18.16	18.16	**14.48**	18.13	14.92	18.14
Amazon prime static	13.52	24.25	21.54	21.54	21.54	**5.33**	13.52	8.82	18.08
Netflix static	11.88	9.18	8.54	8.54	8.54	**5.25**	8.54	5.51	9.16
(c) MAPE values for Informer across different loss functions compared with RAHL									
Download driving	30.33	13.00	20.75	31.36	27.35	**9.82**	13.00	16.45	32.81
Amazon prime driving	12.85	24.81	26.43	20.26	20.22	**12.45**	12.85	20.06	17.03
Netflix driving	32.28	34.97	29.98	39.29	36.11	**21.97**	29.98	33.43	33.56
Download static	9.19	11.42	11.87	7.53	10.71	**4.77**	7.53	6.46	9.03
Amazon prime static	30.36	29.71	30.42	26.50	29.70	**3.20**	26.50	11.03	25.15
Netflix static	3.65	3.22	3.32	3.31	3.27	**3.22**	3.22	2.45	3.63

value for δ, we experimented with different values ranging from 1 to 3, increasing by 0.5 at each step. After extensive calculations, we recorded the minimum error achieved for each tested δ. In Table 5, the results corresponding to the RAHL loss function are highlighted in bold. However, further analysis showed that the performance of the Huber loss is highly sensitive to the choice of δ, meaning that small changes in δ can significantly impact the results. Furthermore, it is important to note that the lowest recorded error does not necessarily indicate the optimal choice of δ.

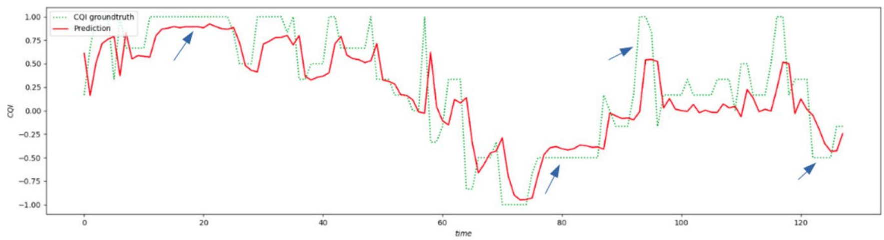
The models were then trained using RAHL and other loss functions, with results compared to Huber loss based on the best values δ from the previous experiment. Unlike Huber loss, RAHL treats δ as a trainable parameter, eliminating the need for manual selection. For a complete evaluation, we also included the results for MSE and MAE. Consistently in all datasets, the RAHL loss function produced MAPE values between those of MAE and MSE, and in more than 90% of cases, RAHL MAPE resulted in a value lower than MAE, demonstrating its stable performance compared to other loss functions. However, in about 10% of cases, RAHL yielded higher MAPE than MAE. The reason may be because the dataset can contain very small residuals and few or no outliers, where the simplicity of MAE, which directs minimizing absolute errors without relying on an adaptive parameter like δ can sometimes lead to better performance. Thus, while RAHL is generally more stable, its adaptability may provide less benefit in clean, low-noise settings.

To highlight the benefits of performing CQI prediction using a model trained with RAHL, we compare the ground truth value and the model's prediction for some samples from our datasets. Since our data set includes six distinct categories, we have selected these comparison charts from different categories for various models.

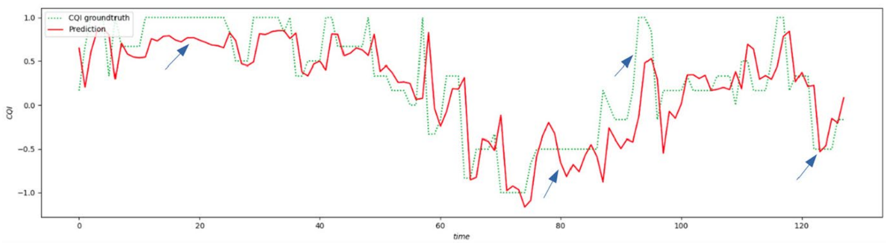
For this analysis, we selected samples from three distinct categories: Netflix Static, Download Static and Netflix Driving, which were trained sequentially using the CNN1D, LSTM and Informer models. Figures 7, 8 and 9 compare the actual



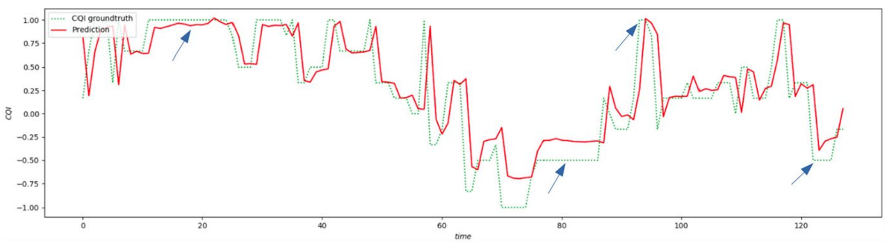
(a) Residual-based Adaptive Huber Loss (RAHL) - MAPE=7.4



(b) Huber loss (best δ) - MAPE=9.1

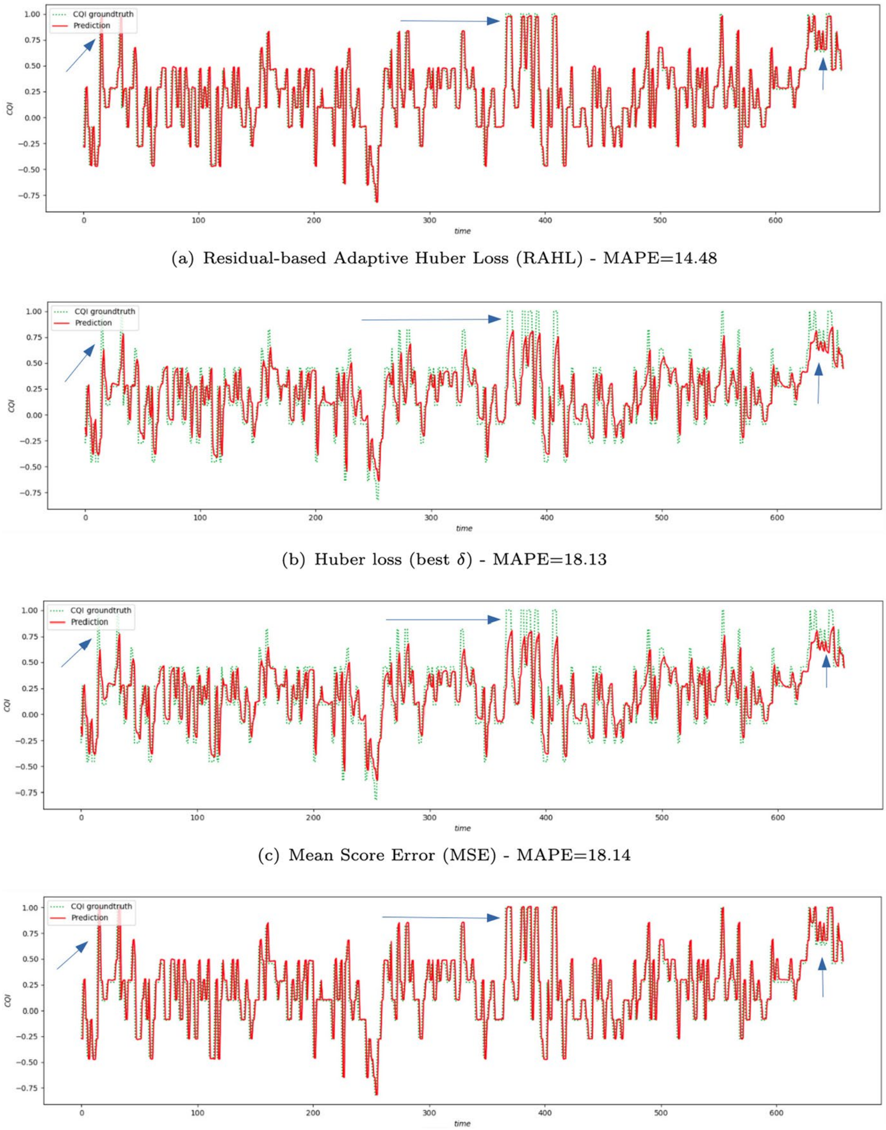


(c) Mean Score Error (MSE) - MAPE=9.83



(d) Mean Absolute Error (MAE) - MAPE=7.58

Fig. 7 Comparison of the ground truth value and the CNN1D model's prediction for a sample from the Netflix Static category



(a) Residual-based Adaptive Huber Loss (RAHL) - MAPE=14.48

(b) Huber loss (best δ) - MAPE=18.13

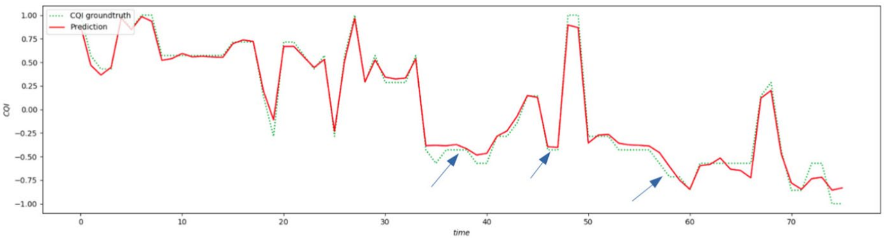
(c) Mean Score Error (MSE) - MAPE=18.14

(d) Mean Absolute Error (MAE) - MAPE=14.92

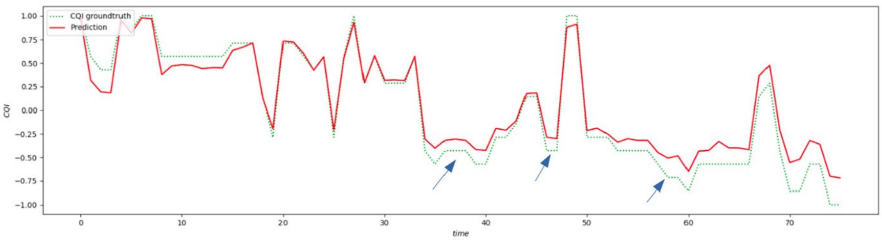
Fig. 8 Comparison of the ground truth value and the LSTM model's prediction for a sample from the Download Static category

CQI values (green line) with the predictions (red line) for each of the chosen samples, made by different models trained with various loss functions.

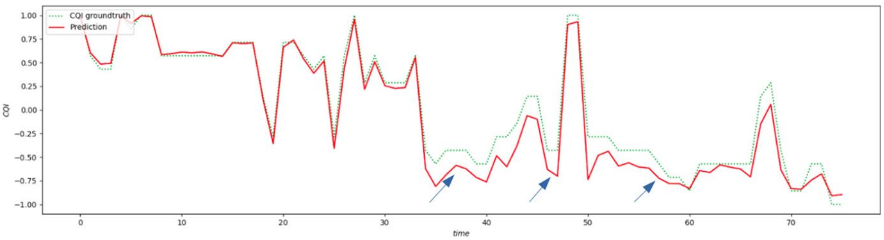
As the sub-sections of each figure reveal, the MAPE value for RAHL is the lowest in three samples. Furthermore, predictions for models trained with RAHL consistently follow the actual CQI values more closely, regardless of sample size (time



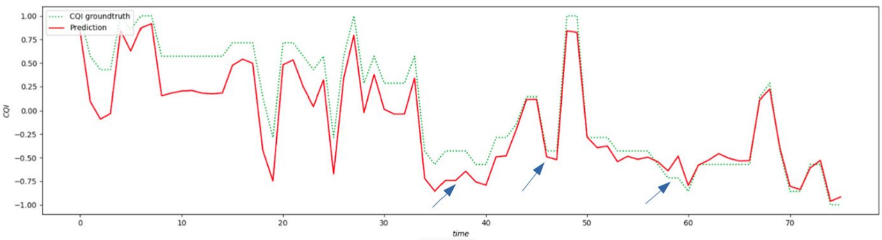
(a) Residual-based Adaptive Huber Loss (RAHL) - MAPE=21.97



(b) Huber loss (best δ) - MAPE=29.9



(c) Mean Score Error (MSE) - MAPE=33.56



(d) Mean Absolute Error (MAE) - MAPE=33.43

Fig. 9 Comparison of the ground truth value and the Informer model's prediction for a sample from the Netflix Driving category

duration). This indicates the robustness of RAHL against outliers and the overall superiority in achieving accurate results for CQI forecasting.

It is essential to understand that the CQI serves as a key metric for assessing the quality of the radio link between a User Equipment (UE) and the base station. The CQI allows the base station to dynamically adjust the modulation scheme for each UE, ensuring an optimized data rate. Consequently, inaccurate CQI predictions can lead to improper modulation configurations and inefficient resource allocation within the 5G network, ultimately impacting overall network performance.

As you can see in Figs. 7, 8 and 9, we have added arrows to highlight the differences in prediction behavior when using different loss functions. These visual markers emphasize how RAHL outperforms other loss functions by providing smoother and more stable predictions, especially across data sets of varying sizes and conditions. The curves clearly demonstrate that RAHL reduces fluctuations and enhances reliability, making it a superior choice for accurate forecasting.

For example, analyzing the predictions made by the CNN1D model trained with MSE in Fig. 7, we can observe that, starting from time 80, the prediction fluctuates incorrectly, suggesting a possible network issue. However, the actual CQI values indicate that the network connection remains stable most of the time. Similarly, in Fig. 8, the predictions made by the LSTM model between times 300 and 400, using either Huber loss or MSE, show significant differences from the actual values, increasing and decreasing. Furthermore, in Fig. 9, around times 37 and 56, we observe non-uniformity between the predicted values and the ground truth, similar to the previous cases.

Figure 10 presents another view on the results showing the accumulation of absolute percentage error over time. This method allows us to examine the error trends of models trained using different loss functions. As expected, the cumulative error for the RAHL loss function increases at a slower rate compared to others, highlighting its improved performance in predicting CQI.

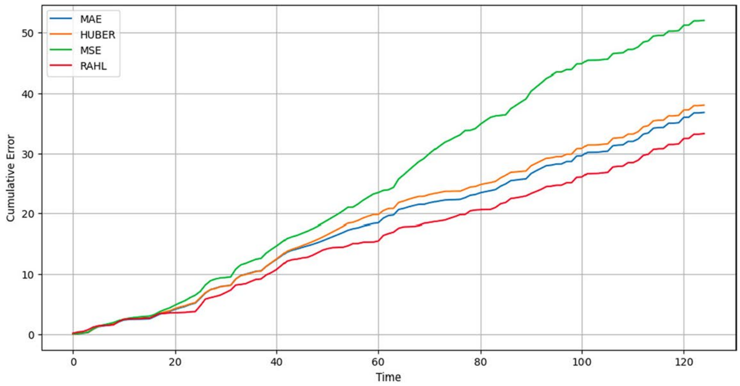
5 Conclusion

CQI is a crucial metric for evaluating the quality of a 5G channel. The base station relies on CQI for key decisions, including resource allocation, modulation, and coding. Since channel quality has a direct impact on data rates and network capacity utilization, it ultimately influences the user's Quality of Experience (QoE). When applying machine learning models for the prediction of CQI, achieving a lower MAPE leads to a better user experience.

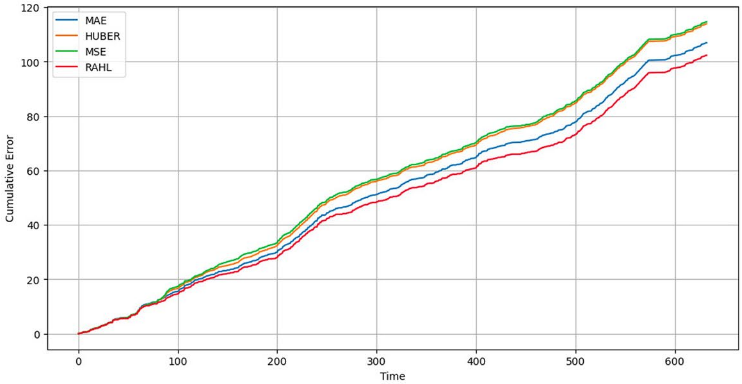
In this paper, we introduce the RAHL method as an innovative approach to reducing error rates. Our experiments, illustrated in Figs. 7, 8, and 9, highlight the superior performance of RAHL in the prediction of CQI compared to traditional methods such as Huber loss, MSE, and MAE. These findings have significant practical implications, as accurate CQI predictions are essential in real-world applications.

A key advantage of RAHL is that it eliminates the need for manual hyperparameter tuning, addressing challenges such as subjectivity, data dependency, computational costs, overfitting risks, and the trade-off between accuracy and robustness. Although our study provides valuable information, we recognize its limitations. Future research could investigate the effectiveness of RAHL in different datasets and neural network architectures.

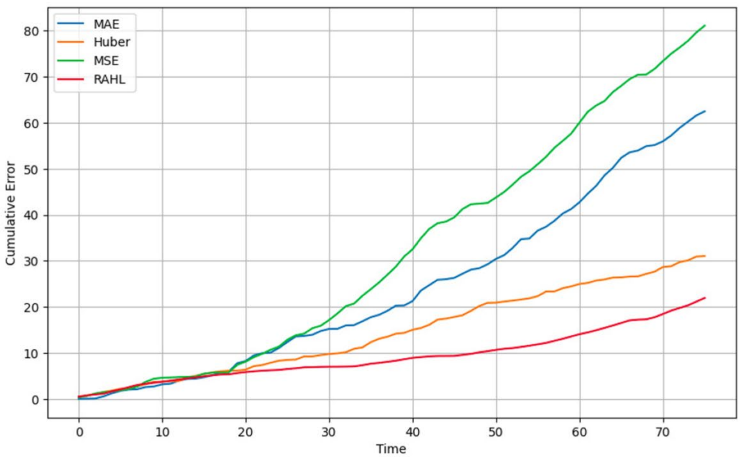
Furthermore, given the evolving nature of loss functions, our work paves the way for exploring adaptive methods that can enhance performance across various tasks. The contributions of RAHL not only improve prediction accuracy, but also simplify



(a) CNN1D - Netflix Static



(b) LSTM - Download Static



(c) Informer - Netflix Driving

Fig. 10 Absolute percentage error accumulated over time for different losses and models hyperparameter selection, making it a useful addition to the machine learning field. In general, RAHL presents a flexible and effective approach to minimizing error rates, offering practical recommendations to researchers and practitioners.

Acknowledgements This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. This research was also supported by São Paulo Research Foundation - FAPESP (Grant 2023/17577-0), National Council for Scientific and Technological Development - CNPq (Grants 315220/2023-6 and 420442/2023-5), and LNCC via resources of the SDumont supercomputer of the IDeepS project.

Author Contributions M.K implemented, evaluated, and wrote the paper. J.A and F.L.V contributed to the design, idea conception, writing, and advising.

Data Availability The dataset used in this study is publicly available and was originally published by Raca et al. It includes 83 records of Internet transmissions with 3142 min of logs, recorded using G-NetTrack v18.7 on a Samsung S10. The dataset contains various 5G channel and context metrics. Further details can be found in: Raca et al. [31]. Beyond throughput, the next generation: a 5G dataset with channel and context metrics. In: Proceedings of the 11th ACM Multimedia Systems Conference, pp. 303–308. The dataset is directly accessible at: https://github.com/uccmisl/5Gdataset.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

1. Yin, H., Guo, X., Liu, P., Hei, X., Gao, Y.: Predicting channel quality indicators for 5G downlink scheduling in a deep learning approach. arXiv:2008.01000 (2020)
2. Parera, C., Redondi, A.E., Cesana, M., Liao, Q., Malanchini, I.: Transfer learning for channel quality prediction. In: 2019 IEEE International Symposium on Measurements & Networking (M &N), pp. 1–6. IEEE (2019)
3. Vankayala, S.K., Shenoy, K.G.: A neural network for estimating CQI in 5G communication systems. In: 2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), pp. 1–5. IEEE (2020)
4. Sakib, S., Tazrin, T., Fouda, M.M., Fadlullah, Z.M., Nasser, N.: A deep learning method for predictive channel assignment in beyond 5G networks. IEEE Netw. **35**(1), 266–272 (2020)
5. Kimura, B.Y.L., Almeida, J., et al.: Deep learning in beyond 5G networks with image-based time-series representation. arXiv:2104.08584 (2021)
6. Iturria-Rivera, P.E., Erol-Kantarci, M.: QoS-aware load balancing in wireless networks using clipped double Q-learning. In: 2021 IEEE 18th International Conference on Mobile Ad Hoc and Smart Systems (MASS), pp. 10–16. IEEE (2021)
7. Pocovi, G., Esswie, A.A., Pedersen, K.I.: Channel quality feedback enhancements for accurate URLLC link adaptation in 5G systems. In: 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), pp. 1–6. IEEE (2020)
8. Beshley, H., Beshley, M., Medvetskyi, M., Pyrih, J.: QoS-aware optimal radio resource allocation method for machine-type communications in 5G LTE and beyond cellular networks. Wirel. Commun. Mob. Comput. **2021**(1), 9966366 (2021)
9. Tsourdinis, T., Chatzistefanidis, I., Makris, N., Korakis, T., Nikaein, N., Fdida, S.: Service-aware real-time slicing for virtualized beyond 5G networks. Comput. Netw. **247**, 110445 (2024)
10. Bakri, S., Bouaziz, M., Frangoudis, P.A., Ksentini, A.: Channel stability prediction to optimize signaling overhead in 5G networks using machine learning. In: ICC 2020-2020 IEEE International Conference on Communications (ICC), pp. 1–6. IEEE (2020)

11. Bartoli, G., Marabissi, D.: CQI prediction through recurrent neural network for UAV control information exchange under URLLC regime. IEEE Trans. Veh. Technol. **71**(5), 5101–5110 (2022)
12. Casparsen, A., Soret, B., Nielsen, J.J., Popovski, P.: Near real-time data-driven control of virtual reality traffic in open radio access network. In: GLOBECOM 2023-2023 IEEE Global Communications Conference, pp. 3481–3486. IEEE (2023)
13. Gokcesu, K., Gokcesu, H.: Generalized Huber loss for robust learning and its efficient minimization for a robust statistics. arXiv:2108.12627 (2021)
14. Dong, H., Yang, L.: Training robust support vector regression machines for more general noise. J. Intell. Fuzzy Syst. **39**(3), 2881–2892 (2020)
15. Moodi, F., Jahangard Rafsanjani, A., Zarifzadeh, S., Zare Chahooki, M.A.: Advanced stock price forecasting using a 1D-CNN-GRU-LSTM model. J. AI Data Min. **12**(3), 393–408 (2024)
16. Chung, J., Jang, B.: Accurate prediction of electricity consumption using a hybrid CNN-LSTM model based on multivariable data. PLoS ONE **17**(11), 0278071 (2022)
17. Petersen, N.C., Rodrigues, F., Pereira, F.C.: Multi-output bus travel time prediction with convolutional LSTM neural network. Expert Syst. Appl. **120**, 426–435 (2019)
18. Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., Zhang, W.: Informer: beyond efficient transformer for long sequence time-series forecasting. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 11106–11115 (2021)
19. Cesa-Bianchi, N., Lugosi, G.: Prediction, Learning, and Games. Cambridge University Press, Cambridge (2006)
20. Portnoy, S., He, X.: A robust journey in the new millennium. J. Am. Stat. Assoc. **95**(452), 1331–1335 (2000)
21. Hastie, T., Tibshirani, R., Wainwright, M.: Statistical Learning with Sparsity: The Lasso and Generalizations. CRC Press, Boca Raton (2015)
22. Huber, P.J.: Robust Statistics, vol. 523. Wiley, Hoboken (2004)
23. Gokcesu, K., Neyshabouri, M.M., Gokcesu, H., Kozat, S.S.: Sequential outlier detection based on incremental decision trees. IEEE Trans. Signal Process. **67**(4), 993–1005 (2018)
24. Delibalta, I., Gokcesu, K., Simsek, M., Baruh, L., Kozat, S.S.: Online anomaly detection with nested trees. IEEE Signal Process. Lett. **23**(12), 1867–1871 (2016)
25. Jadon, A., Patil, A., Jadon, S.: A comprehensive survey of regression-based loss functions for time series forecasting. In: International Conference on Data Management, Analytics & Innovation, pp. 117–147. Springer (2024)
26. Sun, Q., Zhou, W.-X., Fan, J.: Adaptive Huber regression. J. Am. Stat. Assoc. **115**(529), 254–265 (2020)
27. Cavazza, J., Murino, V.: Active regression with adaptive Huber loss. arXiv:1606.01568 (2016)
28. Meyer, G.P.: An alternative probabilistic interpretation of the Huber loss. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5261–5269 (2021)
29. Kaviani, M., Almeida, J., Verdi, F.L.: Residual-based Adaptive Huber Loss (RAHL)—design of an improved Huber loss for CQI prediction in 5G networks. Anais do XLII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos **12**(3), 602–615 (2024). https://doi.org/10.5753/sbrc.2024.1446. (**Sociedade Brasileira de Computação (SBC)**)
30. Clevert, D., Unterthiner, T., Hochreiter, S.: Fast and accurate deep network learning by exponential linear units (ELUS). In: 4th International Conference on Learning Representations (ICLR'16) (2016)
31. Raca, D., Leahy, D., Sreenan, C.J., Quinlan, J.J.: Beyond throughput, the next generation: a 5G dataset with channel and context metrics. In: Proceedings of the 11th ACM Multimedia Systems Conference, pp. 303–308 (2020). The dataset is directly accessible at: https://github.com/uccmisl/5Gdataset
32. Laboratório Nacional de Computação Científica (LNCC/MCTI): SDumont Supercomputer. https://sdumont.lncc.br (2024). Accessed 18 Dec 2024
33. De Myttenaere, A., Golden, B., Le Grand, B., Rossi, F.: Using the mean absolute percentage error for regression models. In: ESANN (2015)

Mina Kaviani is a PhD student in Computer Science at UFSCar – Campus Sorocaba, Brazil. She received her M.Sc. in Computer Science from the same university, where her research focused on CQI prediction in 5G networks. She holds a B.Sc. in Mathematics. Her current research interests include multitask learning and its applications in network optimization.

Jurandy Almeida is an Associate Professor in the Department of Computing at the Federal University of São Carlos, campus of Sorocaba, Brazil. He used to hold a position as an Assistant Professor in the Institute of Science and Technology at the Federal University of São Paulo, campus of São José dos Campos, Brazil (2013-2022) and as an Associate Researcher in the Institute of Computing at the University of Campinas, Brazil (2011-2013). Jurandy received his B.Sc. in Computer Science (2004) from São Paulo State University, Brazil, and his M.Sc. (2007) and Ph.D. (2011) degrees in Computer Science from University of Campinas, Brazil. He is a productivity research fellow (2018-present) of CNPq (the Brazilian National Council for Scientific and Technological Development), and a member of IAPR and SBC (the Brazilian Computer Society).

Fábio L. Verdi is a Full Professor in the Computer Science Department of the Federal University of São Carlos (UFSCar) Sorocaba campus. He has been working with data centers, cloud computing, SDN, and dataplane programmability, leading several national and international projects.

Authors and Affiliations

Mina Kaviani[1] · Jurandy Almeida[1] · Fábio L. Verdi[1]

✉ Mina Kaviani
mina.kaviani@estudante.ufscar.br

Jurandy Almeida
jurandy.almeida@ufscar.br

Fábio L. Verdi
verdi@ufscar.br

[1] Department of Computer Science, Federal University of São Carlos (UFSCar) - Campus Sorocaba, SP-264, Km 110, Itinga, Sorocaba, São Paulo 18052-780, Brazil