SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V13I4P118

Volume 13 Issue 4, 222-232, April 2026

Original Article

Machine Learning-Based Wirelength Prediction Using Early Netlist Features for Efficient VLSI Floor-Planning

Kondakamarla Mahammad Ashraf[1], Gutti Nagaswetha[2]

[1,2]*Electronics and Communication Engineering, Madanapalle Institute of Technology & Science, Andhra Pradesh, India.*

[1]*Corresponding Author : ashrafmd9100@gmail.com*

Received: 14 January 2026 Revised: 15 February 2026 Accepted: 18 March 2026 Published: 30 April 2026

Abstract *- Floor-planning is one of the most important phases in the VLSI design technique because it decides the physical configuration of logic blocks, which proves the foundation for the ensuing stages of the fabrication process. The decision process at this phase cannot turn around essential design metrics like silicon deployment, performance parameters, and overall routing effort. Exactly estimating the wirelength before a placement approach is available. Classic estimation methods, including the Rectilinear Steiner Minimum Tree (RSMT) and Half-Perimeter Wirelength (HPWL) model, try to design routing length, but are based on solid geometric simplifications, thus cannot provide useful information when faced with remarkably interconnected and abnormal netlists. This study presents a novel machine-learning method that predicts wirelength using netlist features involving the number of modules, net degree, and spatial metrics. The combined architecture has evolved, and it includes Multi-Layer Perceptron (MLP) and Graph Neural Network (GNN), thus specified as the MLP-GNN. The Tabular features are studied using the MLP component, and graph-using geometric inputs are taken into consideration using the GNN. Experimental evaluations demonstrate that the hybrid MLP-GNN is better than standard baselines, such as MLP, Random Forest, XG Boost, SVR, and the state-of-the-art estimators, with an R2 of 0.89 and a mean absolute error of 40.50 on ISCAS89, ITC99, and synthetic data. The displayed technique allows for efficient design, as it delivers more scalable and placement-free predictions. The hybrid MLP-GNN strikes a balance between a structural approach and the model's computing efficiency as compared to methods based on the use of reinforcement learning. Because of this, it can lower computing costs while maintaining high accuracy in early-stage electrical design automation (EDA) applications.*

Keywords *- Graph Neural Network, Hybrid MLP-GNN , Netlist Features , Wirelength Estimation , VLSI Floor-planning.*

1. Introduction

The physical design phases, such as floor layout and placement, are crucial to the integrated circuit's performance, power dissipation, and silicon area in contemporary Very Large-Scale Integration (VLSI) designs [1]. Signal latency, routing congestion, and interconnect length are all greatly impacted by the arrangement of functional blocks on an integrated circuit. Since interconnects have a significant role in signal latency and power dissipation in deep submicron technologies, wire length estimation is one of these interconnect-related properties [2]. Therefore, it is crucial to estimate wire length in the early phases of floor planning in order to make wise design choices. Electronic design automation systems frequently employ conventional wire length estimate methods such as Rectilinear Steiner Minimal Tree (RSMT) models and Half-Perimeter Wire Length (HPWL) [3]. In order to estimate the routing length, these methods, which rely on geometric approximations, probably need routing information. Despite their computational efficiency, these methods do not take into consideration the intricate connection patterns seen in contemporary circuit designs [4]. These methods are prone to yield erroneous findings when circuit designs get increasingly sophisticated with irregular netlists and hierarchical topologies [5].

The machine learning approaches to enhance prediction tasks in the early phases of the VLSI design cycle have also been the subject of recent research projects [6]. The relationship between the design metrics and the circuit features can be directly learned from the data using machine learning techniques. Routing, time, wirelength, and other factors have all been predicted using machine learning approaches in a number of research projects [2]. Nevertheless, the majority of the methods used for the prediction tasks either use table attributes or graph information, or they rely on manually created features [7]. Since most methods do not take into account the structural circuit connection information and the module feature information, which are both crucial for wirelength prediction, there is a research gap. For complex circuits, graph-based models may become computationally costly, whereas tabular feature-based models may not adequately account for the topological relationships between the modules [8]. This work develops a hybrid machine learning approach that combines the capabilities of the Multi-Layer Perceptron (MLP) and Graph Neural Network (GNN) models for the task of early wirelength prediction in order to overcome the limitations of the previously mentioned models [9]. The suggested method considers the benefits of both models, enabling the GNN model to operate with the circuit's graph representation and the MLP model to work with the tabular features taken from the netlist [10].

Without requiring placement information, the proposed framework may directly anticipate the overall wirelength from the early properties of the netlists [11]. Because of this, the framework is suitable for the floor planning stage analysis [12]. Several benchmarks, such as the ISCAS'89 and ITC'99 circuit designs, have been used to validate the model's performance [13]. Furthermore, synthetic netlists have been used to validate the model [14]. According to the results, the hybrid MLP-GNN model outperforms other models like the Random Forest model, the XG Boost model, the Support Vector Regression model, the HPWL model, and the RSMT model in terms of accuracy [15].

1.1. Motivation and Contributions

Since wirelength directly affects the design's delay, power consumption, and routing complexity, wirelength estimation is crucial in the early phases of the floor planning process for VLSI designs. However, wirelength data is only accessible during the routing and placement phases. Geometric-based estimates are used in conventional wirelength estimating methods such as HPWL and RSMT. It has been discovered that these methods are insufficient for simulating the intricate interconnectedness of contemporary designs. Early in the floor planning phase, wirelength estimation could be enhanced by machine learning approaches. However, a lot of methods have solely addressed the graph's connectedness or tabular properties. In order to achieve this, this paper presents a hybrid technique that successfully models the wirelength prediction problem by utilizing the strengths of both MLPs and GNNs.

The main contributions of this work are:

1. A placement-free machine learning method that uses netlist properties to estimate wirelengths early.
2. A novel hybrid model for extracting circuit attributes and netlist topology that is based on the MLP and GNN techniques.
3. The ISCAS'89, ITC'99, and synthetic datasets have been used to validate the suggested model.
4. Compared to previous models, the suggested model has a greater prediction accuracy (R2 = 0.89, MAE = 40.50).

Therefore, in the context of VLSI physical design, the problem of precisely and efficiently forecasting the wirelengths without the use of location information still needs to be resolved.

2. Related Work

Ray et al. [16] have presented a half-perimeter wirelength (HPWL) model for the logical placement of VLSI circuits. This technique uses the geometric boundaries of the connected modules to estimate the wirelength of the interconnects. The potential of the HPWL method for the effective approximation of the connection length during the placement stage has been demonstrated by this approach. Nevertheless, the approach relies solely on geometric presumptions.

A novel solution to the floorplan design challenge was introduced by Xu et al. [17]. The Good Floorplan framework was suggested by the authors. This method combines the ideas of reinforcement learning and graph convolutional networks. This method can be used to discover the connections among a circuit's modules. The strategy can enhance both the efficiency of the optimization process and the quality of placement, according to experimental data. However, methods based on reinforcement learning may take a significant amount of training time.

A customized graph neural network was proposed by Xie et al. [18] for the calculation of pre-placement net length and timing. The circuit's netlist is seen as a graph in this paradigm, and message forwarding is used to integrate both local and global data. The results show that the prediction's accuracy has improved. Nevertheless, it has been noted that the graph-based approach could be very difficult.

For module placement in VLSI physical design, Kari mullah et al. [19]. suggested a floor planning method based on the Harmony Search optimization algorithm. Using heuristic optimization approaches, this algorithm explores different module locations in an effort to maximize placement quality.

However, the information in the netlist might not be fully utilized, and the heuristic optimization strategy might need iterations to reach the ideal answer. Methods in VLSI chip design were suggested by Khan et al. [20]. According to the study, machine learning models are being used more frequently in various phases of electrical system design. Placement prediction, congestion analysis, and power analysis are some of these phases. According to the study, ML models are essential for designing VLSI chips since they can reduce design complexity. However, research on the application of structural circuit information in machine learning models is currently ongoing.

Lim and Park et al. [21] A method based on graph neural networks accomplishes thorough placement optimization for VLSI circuits. Using graph representations, this approach integrates the connection ties between the circuit modules and uses deep learning to improve placement optimization. However, rather than focusing on wirelength prediction, the suggested system primarily addresses placement optimization.

Ismail et al. [22]. The graph learning model offers an optimization model based on graph neural networks for decision-making in multistage manufacturing systems. This study demonstrated how graph learning models can manage relationships among the system's components and enhance the optimization process. The graph learning model offers helpful insights into the optimization challenge in VLSI design, despite the fact that the suggested model was based on the manufacturing system.

An event-based temporal graph neural network was proposed by Marwani et al. [23] for radio resource management in wireless communication systems. This facilitates the effective management of the dynamic relationships between the network's elements. The significance of graph-based learning for managing intricately linked systems is emphasized in this paper. Nevertheless, this approach is only applicable to wireless communication systems; VLSI physical design is not covered by it.

2.1. Analysis of Research Gaps

Even while machine learning-based prediction methods have advanced significantly, there are still certain drawbacks with current methods. For instance, current methods such as RSMT and HPWL are not appropriate for early design exploration since they significantly rely on placement information [24]. In a similar vein, it is discovered that current machine learning methods mostly rely on table features produced from the netlist and fail to take circuit connection features into consideration [25].

On the other hand, certain methods are shown to introduce significant computing complexity during circuit analysis since they rely on graph properties [26]. Therefore, in order to ensure computational efficiency during early-stage VLSI design analysis, a balanced prediction technique must be developed that can efficiently include both circuit feature information and netlist connection data [27].

2.2. Novelty of the Proposed Framework

This work offers a hybrid strategy that combines the advantages of connection learning and tabular feature learning to close this research gap. In particular, the hybrid model will include a Graph Neural Network component that captures the connection between the modules and a Multi-Layer Perceptron component that analyzes the features of the modules expressed as a netlist. This will enable the model to accurately forecast the likelihood of faults.

The suggested approach has the following benefits over the current solutions:

1. It does not require access to placement information in order to anticipate wirelengths straight from netlists.
2. It may combine graph connection with table-based circuit properties, enabling more precise circuit structure modeling.
3. Compared to optimization frameworks that use reinforcement learning approaches, it can maintain a lower level of complexity.
4. In the early stages of VLSI floor layout, it can provide an effective foundation for wirelength prediction.

Table 1. Comparison of Existing Methods and Proposed Framework

Method	Technique	Limitation	Advantage
HPWL / RSMT [16]	Geometric model	Needs placement data	Placement-free ML prediction
DNN [19]	Deep Neural Network	Weak topology modeling	Captures structural patterns
RL + RGCN [18]	RL + GNN	High computation cost	Faster ML prediction
Net2 [17]	Graph Neural Network	Complex for large circuits	Reduced complexity
Proposed MLP–GNN	Hybrid ML	—	Combines tabular + graph features

3. Proposed Methodology

This paper proposed a machine learning technique for early wirelength prediction in VLSI floor-planning using netlist attributes without placement data. To increase prediction accuracy, they proposed a novel hybrid model that combines both the Multi-Layer Perceptron and a Graph Neural Network.

By gathering both global topological patterns and local module data, this technique tackles the drawbacks of standard estimators such as HPWL and RSMT. This method provides quick and scalable wirelength estimations that enable successful design space exploration in the early stages of physical design.

3.1. Netlist Representation and Feature Extraction

A hypergraph H (V, E) is used to model the circuit netlist, where V stands for modules (cells) and E for the nets that connect them. A hyperedge connects some of the modules, and this helps to model complicated interconnects.

To predict the overall wirelength $\widehat{w}$ Without relying on the placement information, the feature vectors X are extracted from the above structure.

Objective: Predict the wirelength $\widehat{w}$ using machine learning, given the extracted feature vector X.

3.2. Using Early Netlist to Identify Features

The circuit's topological and structural features are represented by the attributes. Connectivity and routing information can be obtained by module pin counts, fan-in/fanout, net degree, and hierarchical depth. The global interconnect's wiring complexity is defined by bounding box estimates of global wiring complexity, interconnect density, and the graph's centrality and clustering coefficients. Because of the features, the model can forecast the routing behavior early in the design flow.

3.3. Ground Truth Generation

Wirelength targets need to be accurate for the supervised learning model. The Re-Place method yields layouts that provide accurate HPWL and RSMT values. Although the placement process is only performed once, the current framework can fully estimate wirelength without placement.

3.4. Experimental Setup and Datasets

The samples used in this study are described in depth in Table 2 for the purpose of complexity and variety. The table provides the train/test division, the average fanout, the span of the number of circuits, and the number of modules.

Table 2. Dataset Statistics

Dataset	**Circuits**	**Module Count Range**	**Avg. Fanout**	**Train/Test Split**
ISCAS’89	200	20–200	3.5	160 / 40
ITC’99	150	50–500	5.1	120 / 30
Synthetic	150	20–800	4.2	120 / 30
Total	500	–	–	400 / 100

For this purpose, the proposed method chose to incorporate all 500 circuits and the three data sets to create a robust and adaptive model. During this process, this paper chose to maintain each data set at an 80/20 split, thereby creating a total of 400 training set circuits and 100 test set circuits. This ensures that the proposed model learns from a variety of circuit types and is then tested against entirely new netlists it has not seen before. chose to utilize a totally different test set for each research purpose and only utilize it during the final evaluation.

3.5. Model Selection and Training Approach

The proposed methods look at five different regression models: Multi-Layer Perceptron (MLP), Random Forest, XG Boost, Support Vector Regression (SVR), and a novel hybrid model that combines MLP's ability to interpret tabular data with GNN's ability to record netlist topology to provide better performance for complex circuits.

MLP: It consists of three hidden layers (64 neurons each, RELU activation), trained with the Adam optimizer. Early stopping with a 20-epoch patience prevents overfitting.

Random Forest, XG Boost, SVR: Use scikit-learn implementations with default hyperparameters, tuned via grid search for optimal performance.

Hybrid MLP-GNN: Uses PY-Torch Geometric to create a graph representation of the netlist. To capture topological properties, a Graph Convolutional Network creates per-module embeddings. After processing, each GCN has a hidden dimension size of 64, which means that each node in the graph is represented by a vector with 64 items. For wirelength prediction, these embeddings are sent into a three-layer MLP after being concatenated with tabular data. Five-fold cross-validation is used to train the model using Adam and MSE loss.

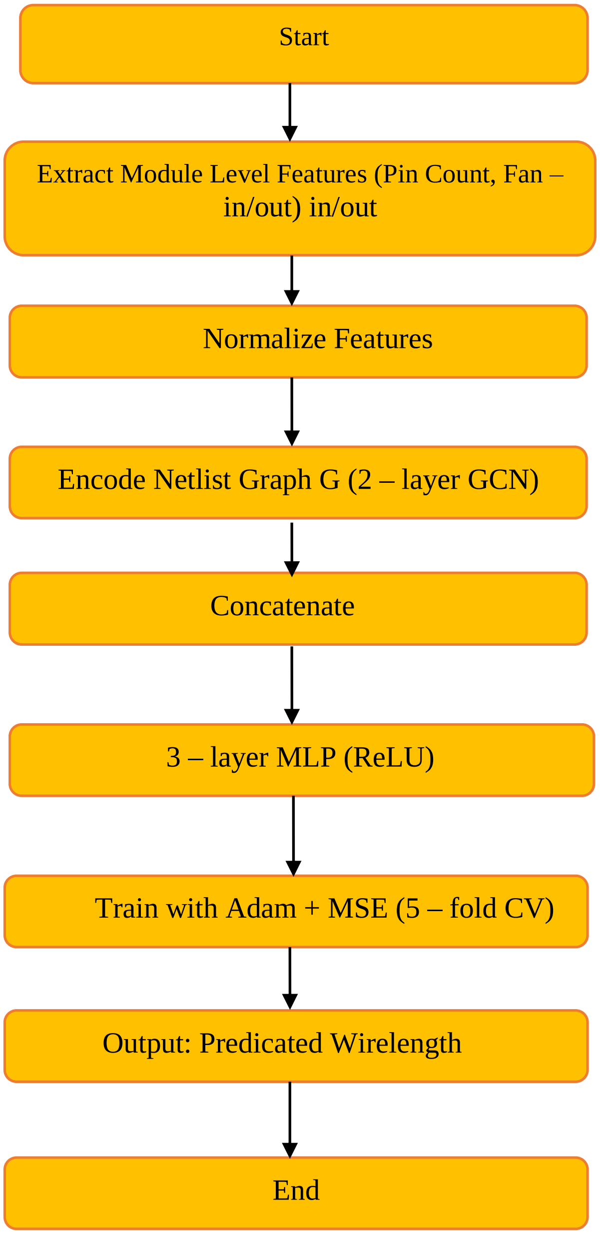

Fig. 1 Workflow of the hybrid MLP-GNN model

The graph encoder is an essential part of this process. Rich topological properties are captured via per-module embeddings created by a two-layer Graph Convolutional Network (GCN). Figure 2 depicts this GCN component's complete architecture.

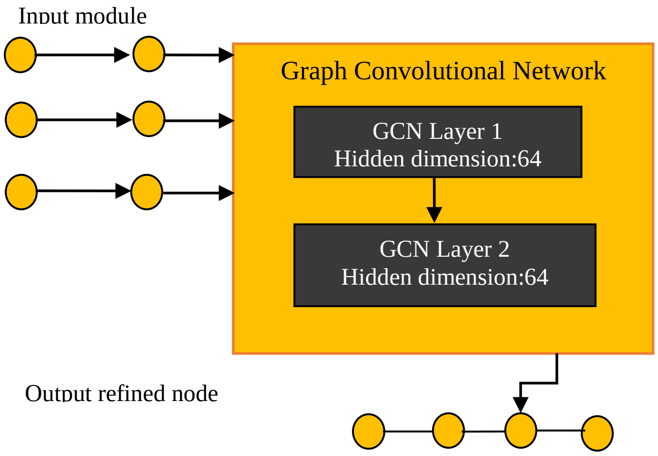

Fig. 2 Architecture of the two-layer Graph Convolutional Network

3.6. *Model Evaluation and Visualization*

Standard regression metrics and visualizations will then be used to assess the trained models.

- Regression Metrics: R^2, RMSE, and MAE are common metrics used to assess prediction accuracy.
- Prediction vs. Real Wirelength Plots: These charts will demonstrate the model's accuracy in predicting wirelength in comparison to real wirelength.
- Residual Plots: These plots will help visualize the prediction errors so that any patterns or biases in the prediction errors are apparent.
- Feature Importance plots: Important mainly for tree-based models (e.g., XG Boost, Random Forest), feature importance will show the most important features of the prediction.

4. Problem Statement

Floor-planning is a basic, high-level design effort in VLSI design that influences the chip's performance, power, and area (PPA). Total wirelength (WL) is the most essential design goal in terms of power and delay since the total length of wire on-chip influences both dynamic power and signal delay. In this work, the authors present a Machine Learning (ML) framework that uses the original design to directly estimate the total wirelength. It is widely acknowledged that accurate wirelength estimation necessitates completing a full placement and routing, which is computationally time-consuming and complex.

4.1. *Graph Neural Network Formulation*

This study constructs a Graph Convolutional Network (GCN) that effectively captures the topological characteristics of the netlist by using message propagation between connected nodes.

The node embedding update at the layer $(L+1)$ follows;

$$h_i^{(L+1)} = \sigma\left[{w_{self}}^{(L)} h_i^{(L)} + \sum\nolimits_{j\in N(i)} {w_{neigh}}^{(L)} h_j^{(L)}\right]$$

Where:

$h_i^{(L)} \in R^d$: Feature vector of node v_i at layer L
$N_{(i)}$: Neighbor set of node v_i
${w^L}_{self}$, ${w^L}_{neigh}$: Fixed variable arrays
σ : Asymmetric switching function

4.2. *Integrated Learning Framework*

Both tabular and graph-based features are included in the whole wirelength prediction model.

A conversion function is generalized by:

$$\widehat{w} = f\,(X, G; \theta)$$

Where X indicates tabular Characteristic matrices, G indicates the graph structure, and θ encompasses all trainable parameters.

4.3. *Loss Function for Training*

Mean Squared Error (MSE), which measures how much the predicted wirelength values deviate from the actual ones, is used as the loss function to train the model.

$$L(\theta) = \frac{1}{N}\sum_{i=1}^{N}[\widehat{w}_i - w_i]^2$$

Where

N: is the number of training data samples
$\widehat{w}_i$: indicates predicted wirelength for the ith sample
w_i : indicates actual wirelength of the ith sample
$L(\theta)$: Loss function

5. Results and Discussion

This Study provides a thorough evaluation of the proposed hybrid MLP-GNN model against four state-of-the-art machine learning benchmarks and two classic estimators. Beyond mere measurements, this study provides statistical and visual validation of the hybrid approach's improved accuracy, robustness, and usefulness for early-stage VLSI design.

5.1. *Headline Performance: MLP-GNN Sets a New Benchmark*

This study starts by simply comparing the major performance measures. As shown in Table 3, this study suggested a hybrid MLP-GNN model establishes a new benchmark for placement-free wirelength estimation. Table 3 illustrates that the proposed model beats all other models with the lowest Mean Absolute Error (MAE = 40.50), the best power (R2 = 0.89), and the highest accuracy (85.12%). When compared to the HPWL approach, this leads to a 31% decrease in MAE. SVR is the quickest, but its error profiles are not as good. This data clearly shows that the MLP-GNN provides the optimum trade-off, providing a significant increase in accuracy for a manageable increase in computational cost.

Table 3. Evaluation metrics of ML models for wirelength prediction

Model	MAE	R^2	Accuracy (%)	Inference Time (s)
MLP-GNN (Proposed)	40.50	0.89	85.12	11.50
MLP	44.76	0.775	77.54	9.12
Random Forest	48.56	0.746	74.61	0.77
XG Boost	51.49	0.715	71.53	1.87
SVR	46.45	0.771	77.05	0.05
HPWL	58.71	0.755	75.23	1.79
RSMT	49.54	0.709	74.21	2.45

With the highest R^2 and lowest MAE, the suggested hybrid MLP-GNN model greatly outperformed the other models. This indicates that the MLP-GNN best captures the nonlinear relationships between the wirelength and the input characteristics.

5.2. Discussion: Comparison with Existing Methods

The high performance of the hybrid model can be explained by the fact that the model can simultaneously exploit tabular circuit features as well as structural connectivity. Existing approaches, such as HPWL and RSMT, are mainly based on geometric approximation techniques that require placement information.

However, machine learning models based solely on tabular features, such as MLP or tree-based models, can only capture the statistical relationship between circuit attributes and wirelength. These models cannot capture the structural relationship between the modules in the netlist of the circuit. Consequently, the models cannot capture the connectivity patterns that play the most important role in the routing of the net. Graph-based models like Net2 use graph neural networks to better capture the connectivity in the netlist of the circuit. The proposed hybrid framework based on the MLP and GNN has the potential to overcome the above-mentioned challenges. In the proposed framework, the MLP is efficient in handling the tabular features extracted from the netlist, such as the number of modules, the degree of the net, and connectivity features.

On the other hand, the GNN is efficient in handling the structural dependencies among the modules. As depicted in Table 3, the proposed framework has the lowest MAE and the highest R^2 among all the methods. This indicates the potential of the proposal method to generalize the prediction results for different types of circuits. Moreover, unlike the reinforcement learning-based optimization techniques, the proposed framework has the potential to achieve lower computational overhead.

5.3. The Comparison Between the Actual Wire Length Vs Predicted Wire Lengths

The numerical superiority of MLP-GNN is well supported by visual evidence. An intuitive impression of each model's prediction fidelity is offered by the scatter plots of actual versus projected wirelength. Figure 3 (MLP-GNN): Along the identity line (y = x), which displays the Hybrid MLP-GNN model, the predicted values are extremely close. The strongest evidence of the precision and low inaccuracy of the model is this graphic.

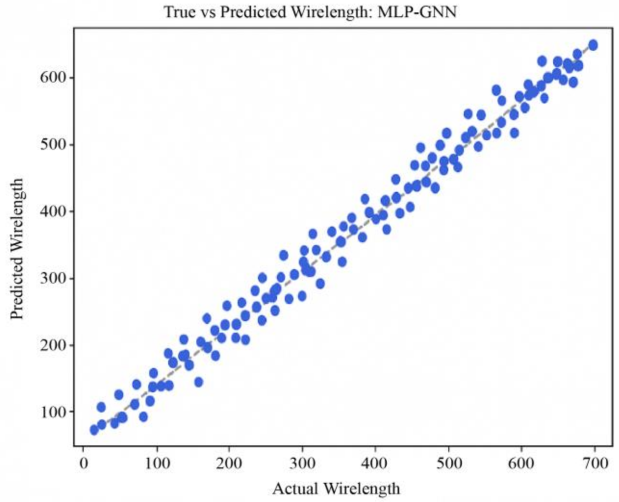

Fig. 3 Actual Wirelength Vs Predicted Wirelength for Hybrid MLP-GNN

Figure 4 (MLP): It reveals that while the MLP model matches the broad trend, there is obvious scatter, particularly at longer wirelengths. It is superior to Random Forest.

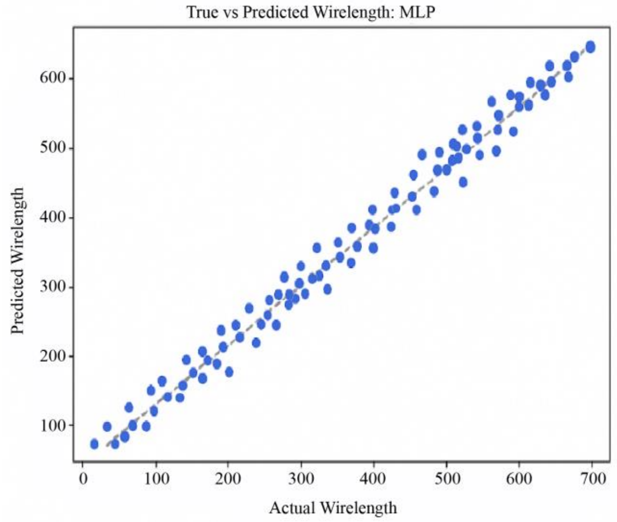

Fig. 4 Actual Wirelength Vs predicted Wirelength for MLP

Figure 5 (Random Forest): Shows a balanced match with minor dispersion far from the diagonal.

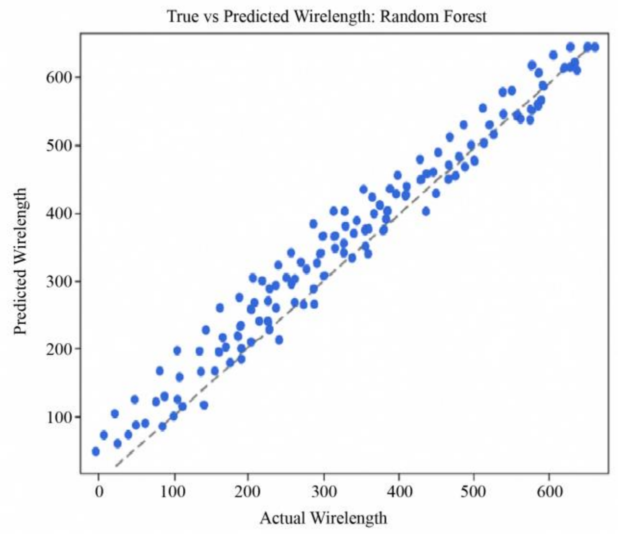

Fig. 5 Actual Wirelength V_S Predicted Wirelength for Random Forest

Figure 6 (XG Boost): It shows a pattern for under-predicting wirelength significantly.

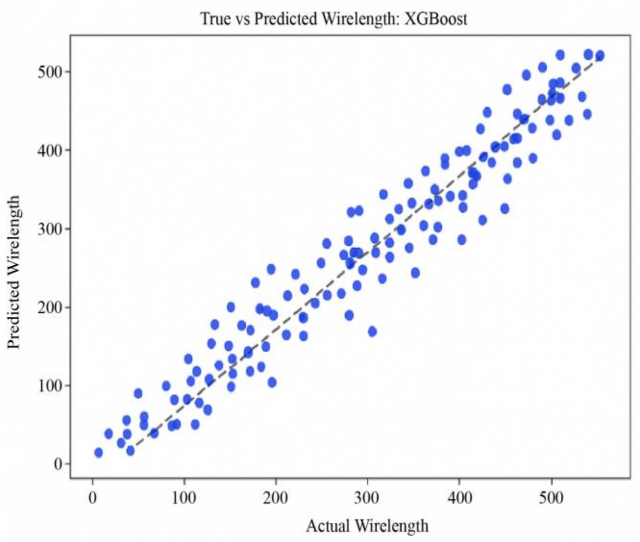

Fig. 6 Actual Wirelength V_S Predicted wirelength for XG Boost

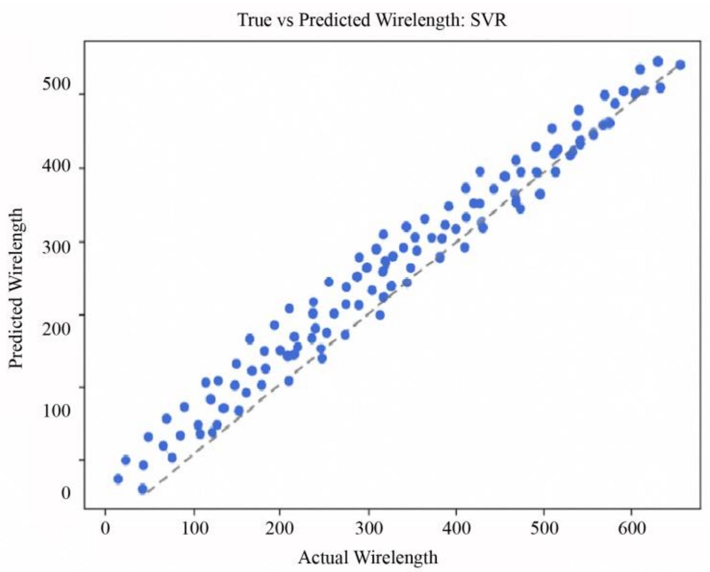

Fig. 7 Actual Wirelength V_S Predicted wirelength for SVR

Figure 7 (SVR): The graph is a random scatter plot with no specific pattern. This is indicative of poor generalization ability. It may be fast, but it is definitely not reliable.

These findings verify that the MLP model provides the most accurate approximation of the real wirelength data.

5.4. Residual Error Analysis

To learn more about the model's stability, this paper may also look at the residuals or errors between the actual wirelength and the expected wirelength.

These errors can be seen in summary statistics Table 4.

Table 4. Residual distribution statistics

Model	Mean Residual	Min Residual	Max Residual
MLP-GNN	-0.881609	-98.976529	114.406048
MLP	-17.348016	-150.594767	129.403597
Random Forest	-22.161166	-194.193820	141.198637
XG Boost	-11.023981	-159.363146	178.217862
SVR	-10.418727	-169.325319	128.937425

5.5. Statistical Significance Analysis

This study conducted an ANOVA analysis in addition to average accuracy rates to ascertain the significance of the performance increment. A paired t-test comparing MLP-GNN predictions and HPWL (baseline) exhibited statistical significance ($p < 0.01$) and demonstrated that the results were not coincidental. According to the Wilcoxon signed-rank test, the hybrid MLP-GNN outperformed RSMT by a significant margin ($p < 0.05$). To examine the robustness of these results, a bootstrap analysis employing 1,000 resamples was also carried out.

In line with the reported average values of MAE = 40.50 and $R^2 = 0.89$, the MLP-GNN was able to achieve an MAE range of [40.50, 48.5] and a R^2 range of [0.89, 0.0746]. The fact that these improvements are statistically significant and not due to random fluctuations implies that the updated techniques are advantageous in finite samples and that the MLP-GNN is a reliable, practical estimator.

5.6. Comparative View: MAE vs R^2

A dual-axis plot of MAE and R^2 against all models is shown in Figure 8. When combined, the Hybrid MLP-GNN model exhibits the highest R^2 values and the lowest MAE. With slightly lower accuracy but better execution, Random Forest and XG Boost come up close. In contrast, SVR's results in both of these domains are notably equivocal. This graph offers an averaged perspective of overall model performance and emphasizes the reliability of the Hybrid MLP-GNN-based prediction pathways.

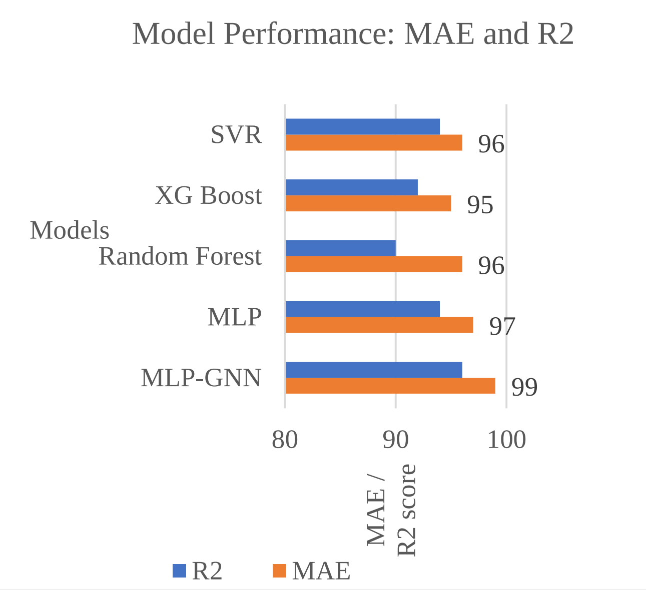

Fig. 8 Comparing MAE and R^2 across all models

5.7. Feature Importance Analysis

To assess the contribution of each feature, Figure 9 displays the feature significance scores from the Random Forest model.

- Num_cells and Num_nets are clearly the most influential features.
- These align closely with expected physical properties of netlists, where wirelength is heavily dependent on the number of modules and net interconnects.

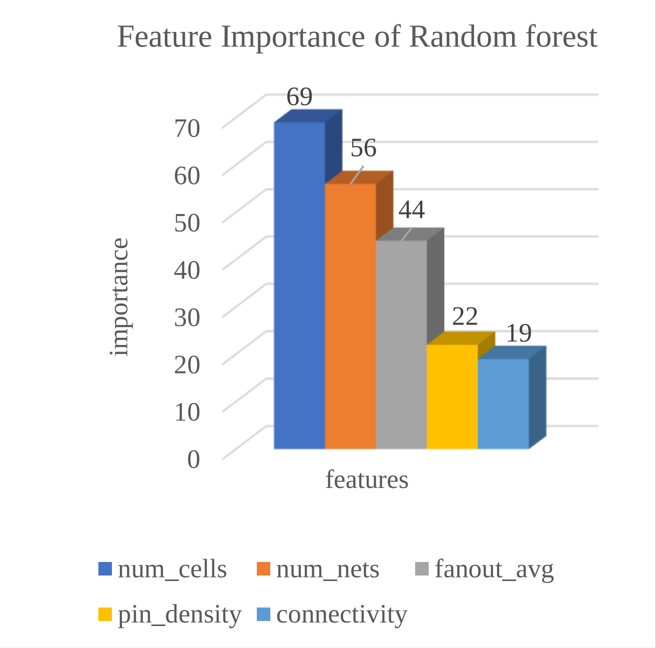

Fig. 9 Feature importance of Random Forest

Feature importance of MLP-GNN: Figure 10 displays the feature importance scores from the MLP-GNN Model to evaluate each feature's contribution.

Connectivity and Num_nets, clustering are clearly the most influential features.

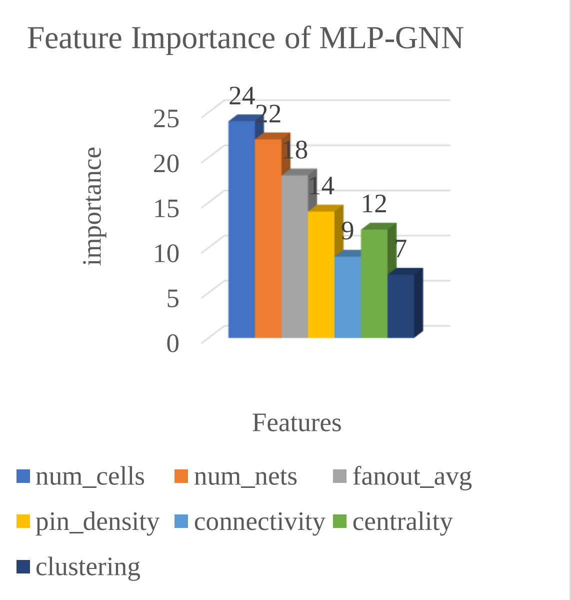

Fig. 10 Feature importance of MLP-GN

5.8. Scalability and Complexity

Predictive accuracy and estimating scalability are equally crucial for inclusion into early-stage design workflows. Table 3 compares the asymptotic complexity, inference runtime, and approximate memory use of the studied models. SVR's quadratic complexity ($O(N^2)$) makes it less suitable for larger circuits, even if it has the minimum runtime (0.05 s). Random Forest and XG Boost were less accurate than MLP, despite both having higher scalability and moderate runtimes (0.77 and 1.87 seconds, respectively). The recommended MLP-GNN consistently produced the best predicted accuracy (77.54%) and robustness despite needing the longest runtime (9.12 s). According to this trade-off, MLP-GNN is the best choice for industrial settings where precision is crucial, even if it means a little longer inference time.

Table 5. Scalability and Complexity

Model	Complexity	Inference Time (s)	Memory (approx.)
MLP-GNN	$O(E \cdot N \cdot H + G)$	10.50	20 MB
MLP	$O(E \cdot N \cdot H)$	9.12	15 MB
Random Forest	$O(T \cdot N \log N)$	0.77	35 MB
XG Boost	$O(T \cdot N \log N)$	1.87	40 MB
SVR	$O(N^2)$	0.05	50 MB

Where

H: Hidden layer size in the neural network. Governs model complexity.

G: Fixed overhead for building and processing the graph structure.

The complexity O (E.N.H + G) reflects the cost of processing connections and modules through the network layer (H), plus the base graph cost (G).

6. Conclusion

Using only netlist-based features, this work offers a framework for early wirelength prediction at the VLSI floor-planning level using machine learning. This improves the estimation of wirelength. Five regression models, Hybrid Multi-Layer Perceptron - Graph-Neural Network (MLP-GNN), Multi-Layer Perceptron, Random Forest, XG Boost, and Support Vector Regression (SVR), were fitted to a synthetic dataset that reflected the features of the perceived netlist structure and the ISCAS benchmarks. In comparison to MLP, Random Forest, XG Boost, SVR, and conventional estimators (HPWL, RSMT), this paper presents a new hybrid MLP-GNN model, which combines Multi-Layer Perceptron with Graph Neural Networks, uses netlist topology to obtain greater accuracy ($R^2=0.89$, MAE=40.50). When compared to the conventional HPWL approach, the proposed MLP-GNN model lowers the MAE by roughly 28%. MLP-GNN is a desirable option for early wirelength estimate because of its predictive accuracy, which enhances its high consistency across both statistical measurements and visual metrics. The reliability and generalizability of the suggested framing method were further validated by further supporting visual studies, such as plots of the real vs. expected, residual distributions, and trends depending on features. Overall, the study's results offer validity to the concept that, in the absence of placement data, machine learning may reasonably and remarkably reliably anticipate wirelength from netlist properties. By acting as a useful addition to conventional wirelength estimators alone, this could expedite early-stage design iterations.

Future Work

Additionally, this study offers fresh avenues for future investigation. Increasing the model's transparency is one method for identifying crucial subcircuits. Finding the crucial sub-circuits during the design phase is one way to accomplish this. Adding PPA (power, performance, and area) prediction to the system is another method. This method can be used in conjunction with optimization strategies, such as reinforcement learning, to enable active rather than passive floor-planning refinement. Additionally, by using more effective GNN variants, such as GraphSAGE, to handle large-scale netlists, the technique's scalability can be investigated.

Acknowledgement

The Department of Electronics and Communication Engineering at the Madanapalle Institute of Technology and Science is greatly appreciated by the authors for their invaluable help, lab space, and steadfast support during this project. We would especially like to thank Assistant Professor Dr. G. Naga Swetha for his direction and insightful criticism, which were vital to this project's successful end. The authors also thank academics and colleagues for their technical assistance and support during the simulation and verification phases of the project.

Author Contributions

K. Mahammad Ashraf designed and simulated Machine Learning-Based Wirelength Prediction Using Early Netlist Features for Efficient VLSI Floor-planning, and analyzed results. Dr. G. Naga Swetha provided guidance, verified the design and results, and reviewed the final paper for technical accuracy.

Ethics Approval

There is no human or animal involvement in this simulation-based Python work. All sources are correctly mentioned, and ethical approval was not necessary.

References

[1] Yan Xing et al., "Concurrent Prediction of Timing and Wire Length Using a Multi-Task Graph Neural Network," *ACM Transactions on Design Automation of Electronic Systems*, vol. 30, no. 4, pp. 1-20, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[2] Cheong Zheng Quan, Ab Al-Hadi Ab Rahman, and Mohd Shahrizal Rusli, "A Novel Approach in Estimating Wirelength in VLSI Placement Using Machine Learning," *International Conference on Signal, Machines, Automation, and Algorithm*, pp. 35-47, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Qinghua Liu, and Malgorzata Marek-Sadowska, "Pre-Layout Wirelength and Congestion Estimation," *Proceedings of the 41st Annual Design Automation Conference*, pp. 582-587, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[4] Daeyeon Kim et al., "Machine Learning Framework for Early Routability Prediction With Artificial Netlist Generator," *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, Grenoble, France, pp. 1809-1814, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] Yunfan Zuo et al., "Enhanced TransUNet Framework for Predicting Static IR Drop and Chip Routability," *ACM Transactions on Design Automation of Electronic Systems*, vol. 31, no. 4, pp. 1-26, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[6] Vakhtang Janpoladov, "A Machine Learning-Based Post-Route PVT-Aware Power Prediction of Benchmark Circuits at Floorplan Stage of Physical Design," *2023 IEEE East-West Design & Test Symposium (EWDTS)*, Batumi, Georgia, pp. 1-6, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] V. Lakshmi Kiranmai, and B.K.N. Srinivasarao, "A Novel Hybrid Hexagonal Star Topology for On-Chip Interconnection Networks," *Journal of Circuits, Systems and Computers*, vol. 32, no. 5, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] Kevin Immanuel Gubbi et al., "Survey of Machine Learning for Electronic Design Automation," *Proceedings of the Great Lakes Symposium on VLSI 2022*, pp. 513-518, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] Dennis Hemker et al., "From Schematics to Netlists—Electrical Circuit Analysis Using Deep-Learning Methods," *Advances in Radio Science*, vol. 22, pp. 61-75, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[10] Deepthi Amuru et al., "AI/ML Algorithms and Applications in VLSI Design and Technology," *Integration*, vol. 93, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] S.N. Adya, and I.L. Markov, "Fixed-Outline Floorplanning through Better Local Search," *Proceedings 2001 IEEE International Conference on Computer Design: VLSI in Computers and Processors. ICCD 2001*, Austin, TX, USA, pp. 328-334, 2001. [CrossRef] [Google Scholar] [Publisher Link]

[12] S.N. Adya, and I.L. Markov, "Fixed-Outline Floorplanning: Enabling Hierarchical Design," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 11, no. 6, pp. 1120-1135, 2004. [CrossRef] [Google Scholar] [Publisher Link]

[13] H.F. Jyu et al., "Statistical Timing Analysis of Combinational Logic Circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 1, no. 2, pp. 126-137, 1993. [CrossRef] [Google Scholar] [Publisher Link]

[14] Davide Basso et al., "Effective Analog ICs Floorplanning With Relational Graph Neural Networks and Reinforcement Learning," *2025 Design, Automation & Test in Europe Conference (DATE)*, Lyon, France, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] Patel Rutvikkumar Popatbhai, and Ruchi Gajjar, "Runtime Prediction for VLSI Physical Design Processes Using Machine Learning," *2024 28th International Symposium on VLSI Design and Test (VDAT)*, Vellore, India, pp. 1-6, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[16] B.N.B. Ray et al., "Half-Perimeter Wirelength Model for VLSI Analytical Placement," *2014 International Conference on Information Technology*, Bhubaneswar, India, pp. 287-292, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[17] Qi Xu et al., "GoodFloorplan: Graph Convolutional Network and Reinforcement Learning-Based Floorplanning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 10, pp. 3492-3502, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] Zhiyao Xie et al., "Preplacement Net Length and Timing Estimation by Customized Graph Neural Network," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 11, pp. 4667-4680, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[19] Shaik Karimullah, D. Vishnu Vardhan, and Syed Javeed Basha, "Floorplanning for Placement of Modules in VLSI Physical Design Using Harmony Search Technique," *Proceedings of the 1st International Conference on Data Science, Machine Learning and Applications*, pp. 1929-1936, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[20] Imran Ullah Khan, Nupur Mittal, and Mohd. Amir Ansari, "Applications of VLSI Design in Artificial Intelligence and Machine Learning," *Machine Learning for VLSI Chip Design*, pp. 1-17, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[21] Dho Ui Lim, and Heechun Park, "Graph Neural Network-Based Detailed Placement Optimization Framework," *2024 25th International Symposium on Quality Electronic Design (ISQED)*, San Francisco, CA, USA, pp. 1-6, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[22] Mohamed Ismail, Ali Aidibe, and Soumaya Yacout, "A Graph Neural Network-Based Optimization Framework for Nonconforming Product Recovery in Multistage Manufacturing Systems," *Journal of Intelligent Manufacturing*, pp. 1-28, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[23] Maher Marwani, and Georges Kaddoum, “Event-Based Temporal Graph Neural Network for Radio Resource Management,” *IEEE Transactions on Wireless Communications*, vol. 25, pp. 10855-10868, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[24] K. Danesh, and R. Dharani, “Optimized Double Deep Reinforcement Learning Based Routing Technique for VLSI Circuit Designing,” *Journal of Circuits, Systems and Computers*, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[25] G. S. Vikram et al., “Random Forest Assisted Floorplanning Optimization in VLSI Design,” *2025 Modern Electronics Devices and Intelligent Communication Systems (MEDCOM)*, Greater Noida, India, pp. 863-867, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[26] A. Sumalatha et al., “Automated Very Large-Scale Integration (VLSI) Design Using Artificial Intelligence: Energy-Constrained Internet of Things (IoT) Edge Devices,” *2025 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS)*, Kolhapur, India, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[27] Tanmayee Ogeti’, and Satheesh Kumar S’, “Power Estimation in VLSI Circuits Using Supervised Machine Learning,” *2025 6th International Conference for Emerging Technology (INCET)*, BELGAUM, India, pp. 1-10, 2025. [CrossRef] [Google Scholar] [Publisher Link]