Received: 6 August 2024 | Revised: 6 November 2024 | Accepted: 25 November 2024

Healthcare Technology Letters

DOI: 10.1049/htl2.12104

LETTER

Hybrid deep learning method to identify key genes in autism spectrum disorder

Naveen Kumar Singh[1] | Asmita Patel[1] | Nidhi Verma[2] | R. K. Brojen Singh[3] | Saurabh Kumar Sharma[1]

[1]School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India

[2]Department of Microbiology, Ram Lal Anand College, University of Delhi, New Delhi, India

[3]School of Computational and Integrative Sciences, Jawaharlal Nehru University, New Delhi, India

Correspondence
Saurabh Kumar Sharma, School of Computer and Systems Sciences, Jawaharlal Nehru University, Lab 204 (A), New Delhi 110067, India.
Email: saurabhsharma@jnu.ac.in

Nidhi Verma, Department of Microbiology, Ram Lal Anand College, University of Delhi, New Delhi 110021, India.
Email: vermanidhi5@gmail.com

R. K. Brojen Singh, School of Computational and Integrative Sciences, Jawaharlal Nehru University, New Delhi 110067, India.
Email: brojen@jnu.ac.in

Funding information
University Grants Commission (BSR Research Startup Grant), Grant/Award Number: F.30-553/2021

Abstract
Autism spectrum disorder (ASD) is a complex neurodevelopmental disorder with a strong genetic component. This research aims to identify key genes associated with autism spectrum disorder using a hybrid deep learning approach. To achieve this, a protein–protein interaction network is constructedand analyzed through a graph convolutional network, which extracts features based on gene interactions. Logistic regression is then employed to predict potential key regulatorgenes using probability scores derived from these features. To evaluate the infection ability of these potential key regulator genes, a susceptible–infected (SI) model, is performed, which reveals the higher infection ability for the genes identified by the proposed method, highlighting its effectiveness in pinpointing key genetic factors associated with ASD. The performance of the proposed method is compared with centrality methods, showing significantly improved results. Identified key genes are further compared with the SFARI gene database and the Evaluation of Autism Gene Link Evidence (EAGLE) framework, revealing commongenes that are strongly associated with ASD. This reinforces the validity of the method in identifying key regulator genes. The proposed method aligns with advancements in therapeutic systems, diagnostics, and neural engineering, providing a robust framework for ASD research and other neurodevelopmental disorders.

1 | INTRODUCTION

Autism spectrum disorder (ASD) is a complex neurodevelopmental disorder characterized by social and communication challenges, along with restricted and repetitive behaviours [1]. ASD presents with a wide range of symptoms and severity levels. Although the exact causes remain under investigation, the growing body of evidence points towards a strong genetic component in its development. Understanding the potential key regulator genes underlying ASD holds immense potential for improved diagnosis, targeted interventions, and potentially, future therapeutic strategies. Twin and family studies have consistently demonstrated a significant heritability for ASD, suggesting that genetic factors play a crucial role [2]. The advent of powerful genetic technologies, such as genome-wide association studies [3] and whole-exome sequencing [4], has revolutionized our understanding of the genetic basis of ASD. These studies have identified hundreds of genes associated with ASD risk, revealing a complex relationship of rare and common variants across the human genome [5]. Closeness centrality–graph convolutional network (CC-GCN) uses clustering coefficients and betweenness centrality (BC) to efficiently select key nodes, improving classification accuracy. Tested on ABIDE-I and ABIDE-II datasets, it performs well but struggles with highly disconnected or complete graphs, limiting its broader applicability [6]. Recent advancements in next-generation sequencing technologies continue to refine our understanding of the genetic architecture of ASD and uncover novel risk genes [7]. The identification of potential key regulator genes requires functional validation and network analysis [8]. Functional studies then elucidate the specific roles these genes play in brain development and function [9, 10].

This study proposes a two-step methodology to identify potential key regulator genes within an ASD gene network. This approach addresses the challenge of pinpointing key genes associated with ASD. First, we leverage GCNs [11], a deep learning technique, to analyse the network structure of ASD genes. GCNs excel at uncovering hidden patterns and relationships within complex networks, allowing them to extract informative features from gene interactions. These features capture the intricate interaction between genes within the network. Second, we employ logistic regression (LR) [12, 13], a well-established machine learning method, to predict potential key regulator genes based on the features extracted by the GCN. LR is chosen for its ability to output probabilities (0–1) for each gene, making it ideal for ranking gene associations. Its interpretability and effectiveness in binary classification also support clear and efficient predictions. The data for this study were obtained from the Autism Informatics Portal, a comprehensive resource dedicated to ASD research [14].

By strategically combining the feature extraction capabilities of GCNs with the classification ability of LR, this study seeks to illuminate potential key regulator genes within the ASD gene network. Identifying these potential key regulator genes can provide valuable insights into the biological pathways underlying ASD. This knowledge has the potential to pave the way for the development of novel therapeutic strategies and improved diagnostic tools, ultimately leading to better clinical outcomes for individuals with ASD. The major contribution of this research work is as follows:

1. **Leveraging GCNs**: A deep learning technique using GCNs is used to analyse network structures to extract informative features from gene interactions. Uncovering hidden patterns and relationships within intricate gene regulatory networks.
2. **Prediction of potential key regulator genes by linear regression**: The features extracted by the GCN are then fed into an LR layer. This model is designed to predict the probability scores for genes, with LR selected as the final layer due to its ability to output probabilities in the 0–1 range for each gene.
3. **Improved diagnosis and treatment**: Identifying influential genes can lead to improved diagnostic tools and targeted interventions. This approach supports advancements in therapeutic and diagnostic systems, healthcare information systems, and neural engineering, offering a robust framework for ASD research and other neurodevelopmental disorders.

The research paper construction is as follows: Section 2 explains the selection of the dataset, its curation process, and the method to identify influential genes in ASD network. Section 3 includes the results and discussion part of the research paper. And then, the article concludes in Section 4.

2 | MATERIALS AND METHODOLOGY

2.1 | Acquisition of autism spectrum disorder data and construction of PPI network

The Autism Informatics Portal [14] provided a comprehensive list of ASD genes (n = 1215). After rigorous data cleaning to eliminate duplicates, isolated, and redundant nodes, the final dataset contained 979 genes (N). Biological datasets, including protein–protein interaction (PPI) networks, often suffer from missing data and false positives due to experimental noise. To mitigate these challenges, we employed data preprocessing techniques that removed isolated nodes—genes with no known interactions—thereby refining the network to focus on biologically relevant connections. Furthermore, redundant and duplicate entries were carefully excluded to ensure the integrity of the dataset. The PPI network is constructed using the STRING database (https://string-db.org/), restricted to *Homo sapiens* genes, resulting in the identification of 9505 interactions (E) among the 979 ASD genes (N). In biological networks, gene interactions, often studied through PPIs, reveal how genes and their corresponding proteins influence each other and contribute to biological functions. In the context of ASD, understanding these interactions is essential, as they help identify key genetic contributors to the disorder.

2.2 | Key regulator gene identification in ASD

The methodology of the proposed method for predicting key genes in complex networks using GCNs with an LR layer involves a systematic process encompassing matrix creation and model training. The workflow of the proposed method for predicting potential key genes is illustrated in Figure 1.

Creating an undirected graph $G = (V, E, A)$, where $V = \{v_1, v_2, v_3, \dots .v_3\}$ represents the set of nodes, $E = \{e_1, e_2, e_3, \dots, e_m\}$ represents a set of edges, and $\mathbf{A} = \{a_{ij}\}$ is the adjacency matrix of the network defined as follows:

$$a_{ij} = \begin{cases} 1, (v_i, v_j) \in E \\ 0, (v_i, v_j) \notin E \end{cases} \tag{1}$$

An adjacency matrix (**A**) and a feature matrix (**X**) are created using the PPI network. The proposed model comprises four components: the input layer, feature processing layer, hidden layer, and output layer. The implementation details of the model are as follows:

1. **Input layer**: The ASD network G, consisting of 979 genes (nodes) and 9505 links (edges), is provided as input to the model. A classical matrix is selected to represent node features, where each node corresponds to a gene.
2. **Feature processing layer**: For each node $v \in G$, the model extracts its L neighbours and forms the adjacency matrix **A**, representing the connections between v and its neighbours. The feature matrix **X** is computed based on various

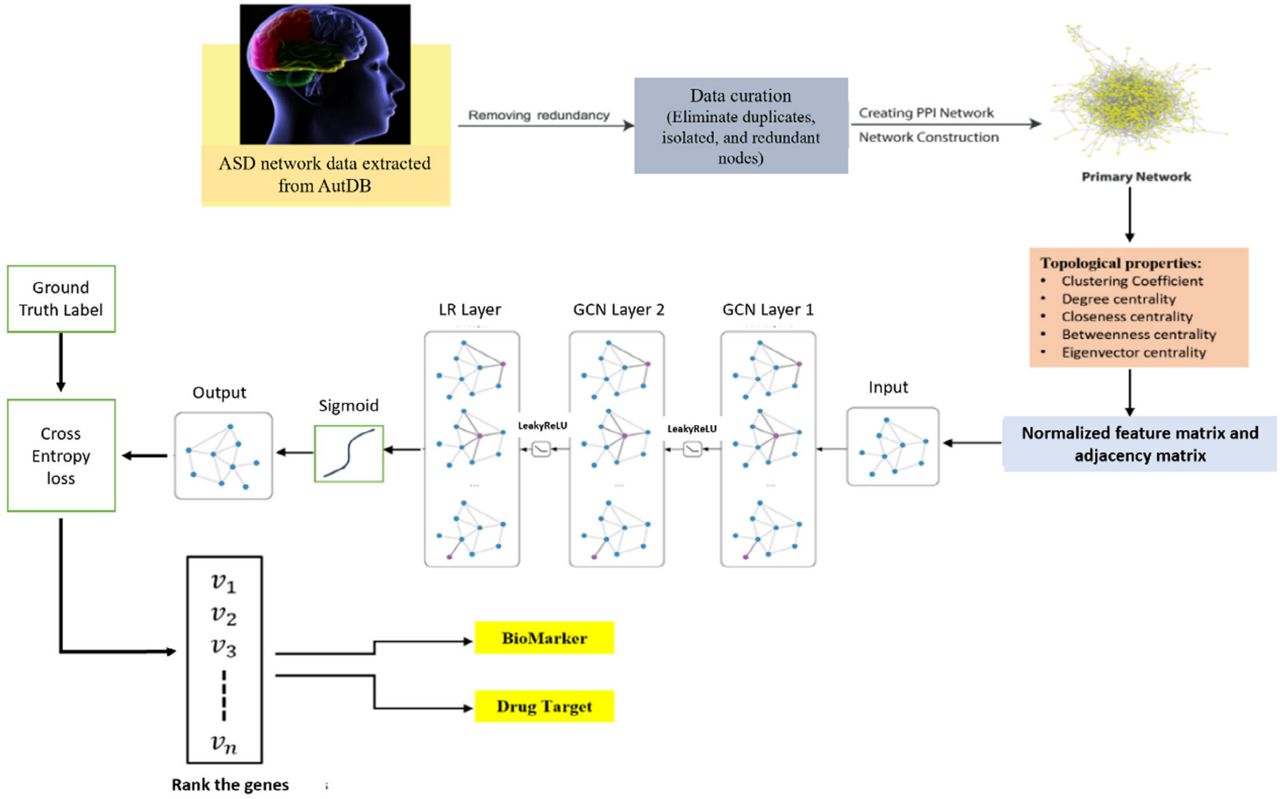


FIGURE 1 Workflow of the proposed method to identify potential key regulator genes in autism spectrum disorder (ASD).

topological properties of the graph. Each node is assigned feature values derived from centrality measures (i.e. degree centrality [DC], BC, CC, eigenvector centrality [EC]) and clustering coefficients as follows:

3. **Degree centrality**: This measures the fraction of nodes that a given node v_i is directly connected to [15]. The DC calculated as follows:

$$C_D(v_i) = \frac{deg\,(v_i)}{N-1} \tag{2}$$

where $deg\,(v_i)$ is degree of node $v_i \forall i = 1, 2, 3, \ldots, n$ and N is the total number of nodes in the graph.

Betweenness centrality: This measures the extent to which node v_i lies on the shortest paths among other nodes [16]:

$$C_B\,(v_i) = \sum_{s \neq v_i \neq t} \frac{\sigma_{st}\,(v_i)}{\sigma_{st}} \tag{3}$$

where σ_{st} is the number of shortest paths from node s to node t, and $\sigma_{st}\,(v_i)$ is the number of those paths that pass-through node v_i.

Closeness centrality: This measures how quickly information spreads from node v_i to all other nodes in the graph [17]:

$$C_c\,(v_i) = \frac{1}{\sum_j d\left(v_i, v_j\right)} \tag{4}$$

where $d(v_i, v_j)$ is the shortest path distance between nodes v_i and v_j.

Eigenvector centrality: This centrality measures the influence of a node based on the importance of its neighbours [18]:

$$C_E\,(v_i) = \frac{1}{\lambda} \sum_j \mathbf{A}_{ij} C_E\,(v_i) \tag{5}$$

where λ is a constant (the largest eigenvalue), and $\mathbf{A}_{ij}$ is the adjacency matrix element indicating a connection between nodes i and j.

Clustering coefficient: This measures the extent to which the neighbours of a node v_i are interconnected [19]:

$$C_{Clust}\,(v_i) = \frac{2.e\,(v_i)}{deg\,(v_i)\left(deg(v_i)\right) - 1)} \tag{6}$$

where $e(v_i)$ is the number of edges between the neighbours of v_i and $deg(v_i)$ is the degree of node v_i.

The feature matrix $\mathbf{X}$ is computed by combining these centrality and clustering coefficient values for all nodes:

$$\mathbf{X} = \begin{bmatrix} C_D\,(v_1) & C_B\,(v_1) & \ldots & C_C\,(v_1) & C_{Clust}\,(v_1) \\ C_D\,(v_2) & C_B\,(v_2) & & C_C\,(v_2) & C_{Clust}\,(v_2) \\ & \vdots & \ddots & \vdots & \\ C_D\,(v_n) & C_B\,(v_n) & \cdots & C_C\,(v_n) & C_{Clust}\,(v_n) \end{bmatrix} \tag{7}$$

ALGORITHM 1 Set of potential key regulator genes G_{top} from graph network.

```
Input: Graph network, $G = (V, E)$
Output: Rank of the influence of genes
1.  Load graph, $G = (V, E)$
2.  Graph preprocessing:
        Count, $K_{cc}$ = Find_Connected_Component ($G$)
        Remove all connected components, $K_{cc} < 10$
        Create an adj_matrix, $A_{N \times N}$.
        Adding identity matrix $\mathbf{I}$ to $\mathbf{A}$, new adj_mat $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$
        Data_preprocessing() function preprocesses the gene expression data
        Feature_matrix, $\mathbf{X}$ = data_preprocessing()
3.  Node Feature Extraction:
    for each $i \in N$ do
        Generate symmetric normalized features, $H = \mathbf{D}^{-1/2} \hat{A} \mathbf{D}^{-1/2} X$,
    where $\mathbf{D}$ is the degree matrix, to be used as input for the Graph
    Convolutional Network (GCN) model.
    end for
4.  Efficient Graph Convolutional Network (GCN):
    for each $h \in H$ do
        Train_gcn() function trains the GCN model
        To determine a significant node's feature, utilize efficient GCN layers
    to evaluate node features layer-by-layer.
        Extract_embeddings() function extracts node embeddings
        node_embeddings = extract_embeddings(gcn_model)
        train_logistic_regression() function trains the LR model
        Incorporate an LR layer to capture the probable influence score of
    nodes.
        Evaluate layer-wise node features with dropout to prevent overfitting.
    end for
5.  Sort the genes according to influence score:
        Feed the LR output into a sigmoid to generate probabilities
    representing the predicted influence of each node $G_{top}$
        sort_genes() function sorts genes based on influence scores
        influential_genes = sort_genes(influence_scores).
6.  Model Simulation:
        Simulate output with SI model
```

This matrix $\mathbf{X}$ contains topological features for all nodes in the graph, which will be input into the GCN model. These measures capture the importance of genes in the network based on their position and interactions. By this processing we get an adjacency matrix of $\mathbf{A}_{v \times v}$ and we have the feature matrix $\mathbf{X}_{v \times 5}$.

The algorithm of proposed method is as follows (Algorithm 1):

4. **Hidden layers**: Based on the feature matrix $\mathbf{X}_{v \times 5}$, the representation of node v is learnt through the hidden layers, which consist of two GCN layers and one LR layer. The GCN layers process the graph's structure using an adjacency matrix $\mathbf{A}$. To incorporate both the node's own features and those of its neighbours, a modified adjacency matrix $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ is used, where $\mathbf{I}$ is the identity matrix representing self-loops. This ensures that the node's own features are included in the learning process. At each GCN hidden layer, the input feature matrix $\mathbf{H}^i$ is transformed by multiplying it with the normalized adjacency matrix $\hat{\mathbf{A}}$ and a weight matrix $\mathbf{W}^i$, followed by an activation function σ. The normalization is achieved by multiplying $\hat{\mathbf{A}}$ by the inverse degree matrix $\hat{\mathbf{D}}^{-1}$, where $\hat{\mathbf{D}}$ is a diagonal matrix containing the degree of each node (i.e. the number of connections). This normalization ensures the stability of the network's output and prevents the exploding or vanishing gradient problem. The operation for each GCN layer is given by the following equation:

$$\mathbf{H}^{\mathrm{i}+1} = \sigma(\hat{\mathbf{D}}^{-1}\hat{\mathbf{A}}\mathbf{H}^{\mathrm{i}}\mathbf{W}^{\mathrm{i}}) \tag{8}$$

Here, the features of each node are aggregated with those of its neighbours and normalized by their respective degrees to maintain the output scale during training.

The GCN layer architecture consists of two convolutional layers, each followed by LeakyReLU activation functions [20], which is defined as

$$f(x) = \begin{cases} x, \text{if } x > 0 \\ \alpha x, \text{if } x \leq 0 \end{cases} \tag{9}$$

where x is the input to the activation function and α is a small positive constant (we take $\alpha = 0.01$), which defines the slope for negative values of x. In the first convolutional layer, the input channel size is $v \times v$, where v is the number of nodes, and the output channel size is $v \times 8$, using eight hidden channels. In the second convolutional layer, the input size is $v \times 8$ and the output size is $v \times 2$. The hidden layer output from the previous layer is computed using the propagation rule in the following equation:

$$\mathbf{H}^{(l+1)} = \sigma\left(\left(\hat{\mathbf{D}}_{ii}\right)^{-1/2}\hat{\mathbf{A}}\left(\hat{\mathbf{D}}_{jj}\right)^{-1/2}\mathbf{H}^{(l)}\mathbf{W}^{(l)}\right) \tag{10}$$

where $\mathbf{D}$ is the diagonal matrix of adjacency matrix ($\hat{\mathbf{A}}$). $\mathbf{H}^{(l)}$ is the output l^{th} hidden layer, and $\mathbf{W}$ is trainable weight matrix at l^{th} hidden layer. In this model, we use two GCN layers in which $\mathbf{W}_0 \in R^{v \times 8}$ and $\mathbf{W}_1 \in R^{8 \times 2}$, where v is number of nodes in graph G, 8 is the number of hidden channels, and 2 is the number of output channels.

After processing through the two GCN layers, embeddings for each node are generated. These embeddings are then fed into the LR layer to compute the influence probability for each node. The LR layer takes an input feature size of $v \times 2$ and outputs a feature size of $v \times 1$, effectively capturing influence scores for each node.

The model is trained using the binary cross-entropy loss function [21, 22] to optimize the weights of both the GCN and LR layers, with a learning rate of 0.1 over 100 epochs. The loss function is defined as follows:

$$\text{Loss} = -\frac{1}{n}\sum_{i=1}^{n}(y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)) \tag{11}$$

where y_i is the ground truth label, $\hat{y}_i$ is the predicted value of the model, and n is the number of nodes. After the GCN model predicts the probabilities, the loss is computed, and the model weights are updated during each training step using gradient descent [23]. The gradients of the loss function with respect to the weights (W_0, W_1, $W_{\text{regression}}$) are calculated, and the model is optimized using the gradient descent optimizer.

Output layer: The output of the GCN model is then passed through the LR layer to predict the influence score for each gene. The predicted influence score is converted into probabilities using the sigmoid function:

$$\text{Influence Score} = \sigma\left(\text{Out}_{\text{LR}}\right) \tag{12}$$

where Out_{LR} represents the output of the LR layer, and σ represents the sigmoid function [24], defined as $\sigma(x) = 1/1 + e^{-x}$, where x is the input to the function, and e is Euler's number. This activation function maps the output to the range between 0 and 1. This range corresponds to the infection capability (IC) of the nodes. Because the IC is constrained by $0 \leq \text{IC} \leq 1$, the sigmoid function is an appropriate choice for this task.

Influence score ranking: The influence scores are then sorted in descending order to produce a ranked list of the genes within the network G. These scores reflect the genes' potential to spread information or infection within the network, with higher scores indicating greater infection spreader.

The proposed model is trained on the ASD PPI network, where the labels for the nodes are derived using the susceptible–infected (SI) model [25]. These labels represent the IC of each node, and during training, they are used to minimize the loss function. The SI model simulates the spread of infection, where S represents the susceptible state and I represents the infected state. Each infected node has a probability $\beta = 0.2$ of infecting its immediate susceptible neighbours, continuing until no new infections occur, reaching a stable state. To label node $v \in G$, it is initialized as infected, whereas all other nodes remain susceptible. The infection capability (IC) of node v is defined (Equation 13) by the number of nodes infected when the network stabilizes. To ensure accuracy and minimize noise, the labels are averaged over 100 independent experiments:

$$\text{IC}\,(v) = \frac{\sum_{i=1}^{100} \text{Inf}_i}{N \times 100} \tag{13}$$

where Inf_i is the number of infected nodes in the ith experiment, and N is the total number of nodes. This provides a robust estimate of node $v's$ IC.

The proposed model generates predicted influence values for each node, which are then sorted to identify the potential key regulator nodes within the network. This step offers crucial insights into nodes with the highest potential to influence network dynamics, aiding informed decision-making and targeted interventions. The accompanying flow diagram (Figure 1) illustrates the seamless process from data preprocessing to potential key regulator node identification, harnessing the capabilities of GCN and LR to improve prediction accuracy. Applying the proposed method on the ASD dataset (as shown in Figure 2), the list of top 10 genes are (i.e. *ACTB, CTNNB1, MAPk3, EP300, ESR1, HRAS, NRXN1, GRIN2B, CREBBP, MAPT*) obtained.

3 | RESULTS AND DISCUSSION

To evaluate the model, a fivefold cross-validation approach is used [26]. The dataset is split into five parts, where the model is trained on four parts and tested on the remaining part. This process is repeated five times with a different test set in each fold. For each fold, the adjacency matrix **A** and feature matrix **X** are split into training and test sets. The GCN model is trained on the training set by applying forward propagation through two GCN layers, followed by an LR layer. Afterwards, the influence scores of the genes are calculated using the SI simulation. Then, the accuracy is evaluated for each fold.

3.1 | Accuracy

To compute accuracy, the models predicted probabilities are thresholded (threshold = 0.5) to convert them into the binary levels (key regulator gene or not). The accuracy score compares this prediction with ground truth labels. Figure 3 indicates that on average our model correctly predicts outcomes with 90.01% accuracy across different cross-validation folds. A higher accuracy implies that the model is performing well in terms of correct predictions.

3.2 | Tracing of potential key regulator genes within the proposed and existing methods

Several established centrality measures are widely used for identifying key regulator genes within complex networks, including DC, BC, CC, and EC. Table 1 presents a comparative analysis of the top 10 potential key regulator genes identified by the proposed method alongside those identified by these classical centrality measures. The results indicate an overlap, with eight common genes between the DC and the proposed method. Furthermore, comparisons with BC, CC, and EC reveal 9, 10, and 3 common genes, respectively. Notably, three genes—*ACTB, CTNNB1, and NRXN1*—are consistently identified as potential key regulators across all methods, as depicted in Figure 4.

However, the analysis was strengthened by incorporating more comprehensive datasets, such as SFARI Gene [27] and Evaluation of Autism Gene Link Evidence (EAGLE), to provide a more robust comparison and enhance the findings [28].

Table 2 provides an overview of genes evaluated using the EAGLE framework, which specifically assesses gene relevance to ASD, rather than broader neurodevelopmental conditions. EAGLE data, available via the SFARI Gene website [29], guided our selection process. We prioritized genes with EAGLE scores of 12 or higher, signifying a definitive association with ASD [30]. It highlights key genes like *CTNNB1, ACTB, EP300, HRAS, NRXN1, GRIN2B, and CREBBP*, which are strongly linked to ASD based on SFARI dataset. These genes fall under the

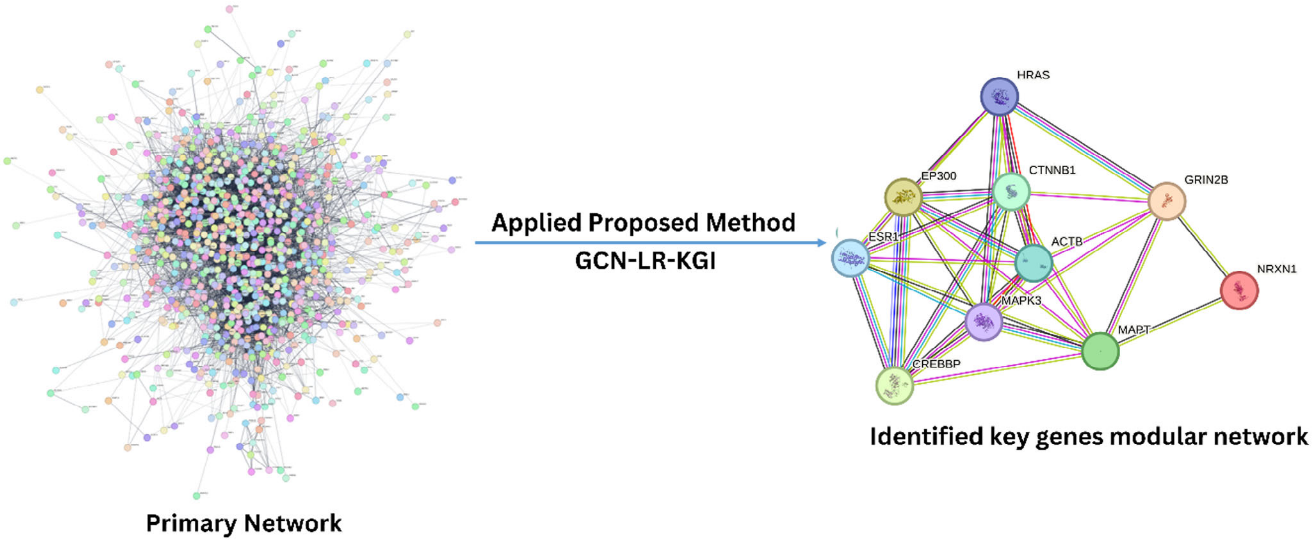


FIGURE 2 Identification of potential key regulator genes in the modular network using proposed method.

TABLE 1 Comparison of top 10 key genes in within degree centrality (DC), betweenness centrality (BC), closeness centrality (CC), eigenvector centrality (EC), and proposed method.

Rank	DC	BC	CC	EC	Proposed method
1	*CTNNB1 (169)*	*CTNNB1 (0.075868)*	*CTNNB1 (0.49443)*	*GRIN2B (0.172412)*	*ACTB*
2	*ACTB (157)*	*ACTB (0.064763)*	*ACTB (0.485574)*	*NRXN1 (0.158559)*	*CTNNB1*
3	*EP300 (132)*	*EP300 (0.034065)*	*HRAS (0.461686)*	*GRIN2A (0.153325)*	*MAPK3*
4	*NRXN1 (126)*	*ESR1 (0.032144)*	*MAPK3 (0.461468)*	*GRIA1 (0.151596)*	*EP300*
5	*GRIN2B (115)*	*HRAS (0.032013)*	*ESR1 (0.46125)*	*GRIA2 (0.151536)*	*ESR1*
6	*ESR1 (114)*	*MAPK3 (0.027962)*	*EP300 (0.45629)*	*GRM5 (0.149032)*	*HRAS*
7	*HRAS (114)*	*NRXN1 (0.027821)*	*GRIN2B (0.452063)*	*CTNNB1 (0.136654)*	*NRXN1*
8	*MAPK3 (110)*	*MAPT (0.021523)*	*NRXN1 (0.446071)*	*ACTB (0.133035)*	*GRIN2B*
9	*GRIA2 (100)*	*CREBBP (0.015861)*	*MAPT (0.442031)*	*SNAP25 (0.126040)*	*CREBBP*
10	*GRIA1 (99)*	*CHD8 (0.015162)*	*CREBBP (0.440435)*	*CAMK2A (0.124960)*	*MAPT*

TABLE 2 Comparison of gene rankings in autism spectrum disorder (ASD) using the proposed method, SFARI gene scores, and Evaluation of Autism Gene Link Evidence (EAGLE) classification.

Proposed method	SFARI curation	EAGLE curation (score)
ACTB	1	1
CTNNB1	1	32.75
MAPK3	2	
EP300	1	23.6
ESR1		
HRAS	1	
NRXN1	1	143.75
GRIN2B	1	29.65
CREBBP	1	31.35
MAPT	3	

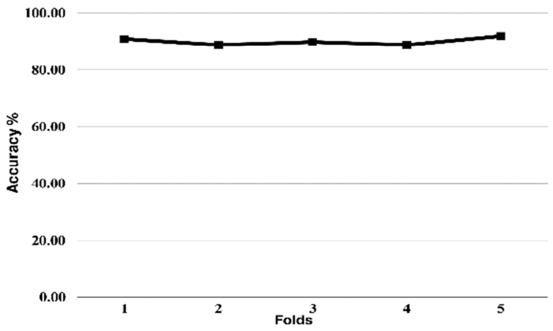
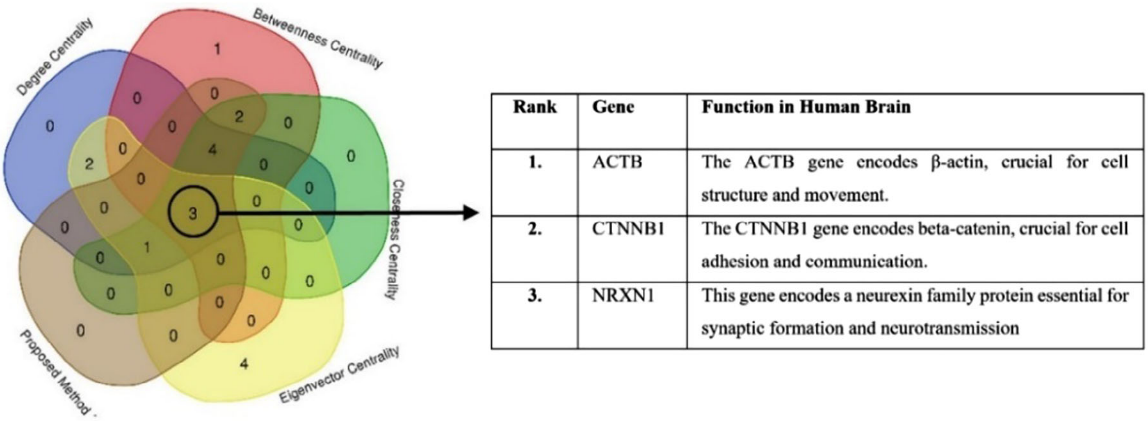


FIGURE 3 Average accuracy percentage across 100 epochs for each cross-validation fold.

'definitive' category in Dr. Scherer's classification, further reinforcing their significance in ASD research. However, broadening the comparison to include more recent gene data from



Rank	Gene	Function in Human Brain
1.	ACTB	The ACTB gene encodes β-actin, crucial for cell structure and movement.
2.	CTNNB1	The CTNNB1 gene encodes beta-catenin, crucial for cell adhesion and communication.
3.	NRXN1	This gene encodes a neurexin family protein essential for synaptic formation and neurotransmission

FIGURE 4 Visual representation of common genes between the centrality measures and proposed method. The adjacent table represents the common genes within all methods and their function to human brain.

EAGLE would provide a more comprehensive view. Among the top 10 genes identified, only *CTNNB1, EP300, NRXN1, GRIN2B, and CREBBP* show strong ASD associations, whereas *ACTB* requires further validation due to limited evidence. *NRXN1*, with the highest EAGLE score, encodes a neurexin protein essential for synaptic connections and brain functionality [31]. *CTNNB1* [32] is involved in cell communication, critical for brain architecture, whereas *CREBBP* [33] and *EP300* [34] regulate gene expression and play roles in brain development. *GRIN2B* is key for synaptic plasticity [35]. *MAPK3* and *MAPT*'s roles in ASD are moderate and limited, respectively. Incorporating both SFARI and EAGLE frameworks enriches the analysis.

3.3 | SHAP analysis for feature interpretation

SHAP (Shapley Additive Explanation) analysis is a powerful method used to interpret machine learning models by quantifying the contribution of each feature to the model's predictions [36]. By assigning a SHAP value to each feature for every prediction, the analysis reveals how each feature influences the model's output, whether positively or negatively. SHAP values facilitate the detection of complex interactions among features, offering a detailed and interpretable explanation of the model's decision-making process, which is essential for validating and refining predictive models in various applications.

The SHAP analysis presented in Figure 5 provides a detailed understanding of the features' impact on the model's predictions and their contributions to the final output. Feature 0 represents the combined impact of DC, BC, and CC, averaged to capture their collective influence as a potential key regulator. High values of this combined feature significantly increase the model's prediction, whereas low values decrease it, indicating a strong positive correlation with the model's output. This suggests that nodes with higher combined centrality scores are likely to play a more influential role in the network, aligning with the model's objectives. As also shown in Table 1, there are 8, 9, and 10 common genes between the proposed method and DC, BC, and CC, respectively. Feature 2 represents the EC exhibits a more complex relationship, with both high and low values affecting the prediction in varying directions, suggesting non-linear interactions. Feature 1 represents the clustering coefficient, but it has minimal impact, with its SHAP values clustered around 0, indicating it is less critical in the model's decision-making process. This comprehensive analysis enhances transparency and trust in the model's predictions, aiding in feature selection and engineering to potentially improve model performance.

As we analyse the results of our model, the high accuracy of 98.5% demonstrates that the model predicts correctly whether a gene is a key regulator or not. The precision of 99.0% measures the proportion of correctly predicted key regulator nodes among all nodes predicted as key regulators, indicating that most of the nodes predicted as key regulators are indeed key regulators according to the ground truth. By highlighting the importance of specific features and their contributions, SHAP analysis offers valuable insights into the underlying patterns and relationships within the data, essential for building more interpretable and reliable models. This level of precision and accuracy underscores the effectiveness of our model in identifying potential key regulator genes, providing a robust tool for advancing our understanding of genetic contributions to ASD.

3.4 | Evaluate model performance across various seed set sizes (*k*) and infection rates (F(t)) from SI model

In the context of evaluating the model's performance, the SI model represents a fundamental epidemiological model used to simulate the spread of infectious diseases within a population. By varying the seed set sizes *k* (initial number of infected individuals) and infection rates within the SI model, simulating different scenarios of disease spread, and observing how the model performs in predicting potential key regulator nodes or genes under these varying conditions. Initial infected nodes are taken by the proposed method. This evaluation helps in under-

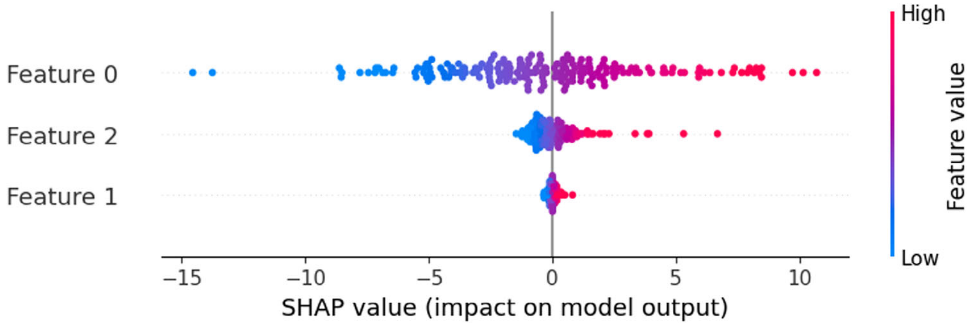


FIGURE 5 Feature contributions to model prediction using Shapley Additive Explanation (SHAP) analysis.

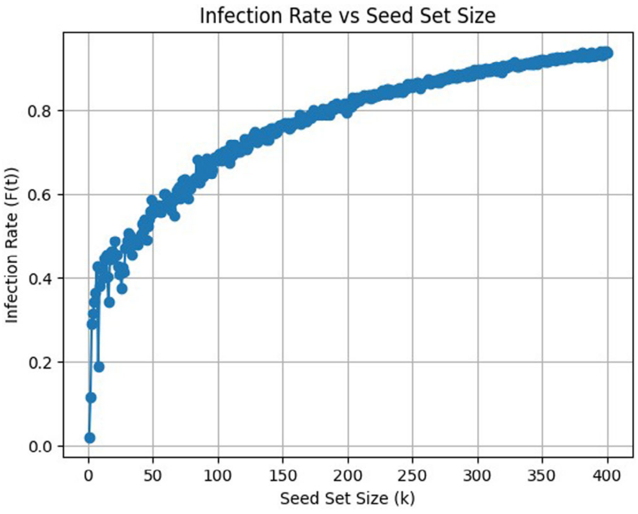


FIGURE 6 Relationship between seed set size (k) and infection rate ($F(t)$.

standing the robustness and effectiveness of the model across different disease propagation scenarios, providing insights into its real-world applicability and performance.

In Figure 6, the x-axis is labelled 'seed set size (k)' and the y-axis is labelled 'infection rate ($F(t)$)'. The graph shows a positive correlation between seed set size and infection rate. This means that as the number of initially infected genes increases, the overall infection rate in the network also increases. This is likely because each infected node has the potential to infect its neighbours, and so a larger initial set of infected nodes will lead to a larger cascade of infections. The specific shape of the curve also provides some insights. It appears to be a monotonically increasing curve, but it might also be reaching an asymptote (a horizontal line that the curve approaches but never intersects) as the seed set size gets very large. This suggests that there is a limit to how much the infection rate increases as the seed set size grows. This is because the network has a finite number of nodes, or because there are other factors that limit the spread of the infection, such as the recovery of nodes from infection or the existence of edges that are difficult to traverse. This analysis demonstrates that the list of potential key regulator genes identified using the proposed method effectively spreads the infection throughout the network within t time steps.

4 | CONCLUSION

We present a novel method for identifying key genes associated with ASD using a hybrid approach combining GCN and LR. Applied to a PPI network, the GCN extracts node embeddings, whereas the LR model predicts their influence. The SI model simulates infection rates to validate the effectiveness of identified key nodes, showing that our method outperforms traditional centrality measures. By integrating comprehensive datasets such as SFARI Gene and EAGLE, we ensure a robust comparison that strengthens our analysis. The key regulator genes categorized as 'definitive' in Dr. Scherer's classification reinforce their importance in ASD research. The use of SHAP analysis further improves interpretability by quantifying each gene's contribution, enhancing the reliability of the model's predictions.

The proposed model is further enhanced by utilizing advanced graph embedding techniques, such as node2vec or GraphSage, which can capture more detailed and nuanced structural information from the PPI network. The proposed GCN-LR framework could be improved further by adding more layers to the GCN architecture, enabling deeper feature extraction from the graph. Additionally, incorporating attention mechanisms, such as those in graph attention networks, could enhance the model's focus on critical gene interactions, leading to improved accuracy and interpretability. Exploring multi-modal data integration and dynamic graph representations could also provide a richer, more adaptive approach to understanding complex genetic interactions in ASD and other neurodevelopmental disorders. These advancements offer new opportunities for precision medicine, enhancing therapeutic and diagnostic systems while providing a robust framework for future research in the field.

AUTHOR CONTRIBUTIONS

Conceptualization: Naveen Kumar Singh, Asmita Patel, and Saurabh Kumar Sharma. *Computational works and preparation of the figures*: Naveen Kumar Singh, Asmita Patel and Nidhi Verma. *Methodology*: Naveen Kumar Singh and Asmita Patel. *Analysis of the results*: Naveen Kumar Singh, Asmita Patel, Nidhi Verma, R K Brojen Singh and Saurabh Kumar Sharma. *Manuscript writing*: Naveen Kumar Singh, Nidhi Verma and Asmita Patel. *Supervision*: Saurabh Kumar Sharma and R. K. Brojen Singh. *Review and* *editing*: Naveen Kumar Singh, Asmita Patel, Nidhi Verma, and R. K. Brojen Singh.

ACKNOWLEDGEMENTS

S.K.S. and R.K.B.S. are financially supported by the University Grants Commission under file no. F.30-553/2021 (BSR), India. R.K.B.S. also acknowledges research support from DBT, BIC Grant.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of the study are openly available in 'AutDB' and 'SFARI Gene' at https://doi.org/10.1093/nar/gkx1093, reference number [14], and https://doi.org/10.1186/2040-2392-4-36, reference number [29], respectively.

DISCLOSURE

We reiterate that this article has not been published anywhere and is not sponsored by any particular organization.

PERMISSION TO REPRODUCE MATERIALS FROM OTHER SOURCES

None.

ORCID

Naveen Kumar Singh https://orcid.org/0000-0002-9732-4532
Asmita Patel https://orcid.org/0009-0005-8699-1398
Nidhi Verma https://orcid.org/0000-0003-0643-8858
Saurabh Kumar Sharma https://orcid.org/0000-0002-3288-9620

REFERENCES

1. Hirota, T., King, B.H.: Autism spectrum disorder: A review. JAMA, J. Am. Med. Assoc. 329(2), 157–168 (2023)
2. Kainer, D., Templeton, A.R., Prates, E.T., Jacboson, D., Allan, E.R., Climer, S., Garvin, M.R.: Structural variants identified using non-Mendelian inheritance patterns advance the mechanistic understanding of autism spectrum disorder. Hum. Genet. Genomics Adv. 4(1), 100150 (2023)
3. Uffelmann, E., Huang, Q.Q., Munung, N.S., De Vries, J., Okada, Y., Martin, A.R., Martin, H.C., Lappalainen, T., Posthuma, D.: Genome-wide association studies. Nat. Rev. Methods Primers. 1(1), 59 (2021)
4. Backman, J.D., Li, A.H., Marcketta, A., Sun, D., Mbatchou, J., Kessler, M.D., Benner, C., Liu, D., Locke, A.E., Balasubramanian, S., Yadav, A., Banerjee, N., Gillies, C.E., Damask, A., Liy, S., Bai, X., Hawes A., Maxwell E., Gurski L., Watanabe, K., Kosmichi, J.A., Rajagopal, V., Mighty, J., Center, R.G., Discov, E.H.R. Jones, M., Mitnaul, L., Stahl, E., Coppola, G., Jorgenso, E., Habegger, L., Salerno, W.J., Shuldiner, A.R., Lotta, L.A., Overton, J.D., Cantor, M.N., Reid, G.R., Yancopoulos, G., Kang, H.M., Marchini, J., Baras, A., Abecasis, G.R., Ferreira, M.A.: Exome sequencing and analysis of 454,787 UK Biobank participants. Nature. 599(7886), 628–634 (2021)
5. Sanders, S.J., He, X., Willsey, A.J., Ercan-Sencicek, A.G., Samocha, K.E., Cicek, A.E., Murtha, M.T., Bal, V.H., Bishop, S.S.L, Dong, S., Goldberg, A.P., Jinlu, C., Keaney, J.F., Klei, L., Mandekk, J.D., Morena-De-Luca, D., Poultney, C.S., Robinson, E.B., Smith, L., Solli-Nowlan, T., State, M.W: Insights into autism spectrum disorder genomic architecture and biology from 71 risk loci. Neuron. 87(6), 1215–1233 (2015). https://doi.org/10.1016/j.neuron.2015.09.016
6. Rohilla, S., Jadeja, M., Pilli, E.S., Vyas, P., Gehlot, P.: CC-GCN: A novel graph-based approach for identification and detection of autism spectrum disorder. In: Multimedia tools and applications, Springer, Berlin (2024). https://doi.org/10.1007/s11042-024-20111-3
7. Sanders, S.J.: Next-generation sequencing in autism spectrum disorder. Cold Spring Harb. Perspect. Med. 9(8), a026872 (2019)
8. Sener, E.F., Canatan, H., Ozkul, Y.: Recent advances in autism spectrum disorders: Applications of whole exome sequencing technology. Psychiatry Investigation 13(3), 255 (2016)
9. Asif, M., Martiniano, H.F., Vicente, A.M., Couto, F.M.: Identifying disease genes using machine learning and gene functional similarities, assessed through Gene Ontology. PLoS One. 13(12), e0208626 (2018)
10. Duda, M., Zhang, H., Li, H.D., Wall, D.P., Burmeister, M., Guan, Y.: Brain-specific functional relationship networks inform autism spectrum disorder gene prediction. Transl. Psychiatry. 8(1), 56 (2018)
11. Chen, M., Wei, Z., Huang, Z., Ding, B., Li, Y.: Simple and deep graph convolutional networks. In: International conference on machine learning, pp. 1725–1735. ACM, New York (2020)
12. Schober, P., Vetter, T.R.: Logistic regression in medical research. Anesthesia Analg. 132(2), 365–366 (2021)
13. Zabor, E.C., Reddy, C.A., Tendulkar, R.D., Patil, S.: Logistic regression in clinical studies. Int. J. Radiat. Oncol. Biol. Phys. 112(2), 271–277 (2022)
14. Pereanu, W., Larsen, E.C., Das, I., et al. AutDB: A platform to decode the genetic architecture of autism. Nucleic Acids Res.. 46(D1), D1049–D1054 (2018). http://doi.org/10.1093/nar/gkx1093
15. Yum, S.: Social network analysis for coronavirus (COVID-19) in the United States. Soc. Sci. Q. 101(4), 1642–1647 (2020)
16. Lee, J., Lee, Y., Oh, S.M., Kahng, B.: Betweenness centrality of teams in social networks. Chaos: An Interdisciplinary J. Nonlinear Sci. 31(6), 061108 (2021)
17. Evans, T.S., Chen, B.: Linking the network centrality measures closeness and degree. Commun. Phys. 5(1), 172 (2022)
18. Merelo, J.J., Molinari, M.C.: Self-loops in social networks: Behavior of eigenvector centrality. In: Italian Workshop on Artificial Life and Evolutionary Computation, pp. 366–380.Springer Nature Switzerland, Cham (2024)
19. Zhu, G., Li, Y., Wan, L., Sun, C., Liu, X., Zhang, J., Liang, Y., Liu, G., Yan, H., Li, R., Yang, G.: Divergent electroencephalogram resting-state functional network alterations in subgroups of autism spectrum disorder: A symptom-based clustering analysis. Cereb. Cortex. 34(1), bhad413 (2024)
20. Hu, J., Cao, L., Li, T., Liao, B., Dong, S., Li, P.: interpretable learning approaches in resting-state functional connectivity analysis: The case of autism spectrum disorder. Comput. Math. Meth. Med. 2020(1), 1394830 (2020)
21. Preethi, S., Arun Prakash, A., Ramyea, R., Ramya, S., Ishwarya, D.: Classification of autism spectrum disorder using deep learning. In: Intelligent Systems: Proceedings of ICMIB 2021, pp. 247–255. Springer Nature Singapore, Singapore (2022)
22. Mao, A., Mohri, M., Zhong, Y.: Cross-entropy loss functions: Theoretical analysis and applications. In: International conference on Machine learning, pp. 23803–23828. ACM, New York (2023)
23. Chandra, K., Xie, A., Ragan-Kelley, J., Meijer, E.: Gradient descent: The ultimate optimizer. Adv. Neural Inf. Proces. Syst. 35, 8214–8225 (2022)
24. Chowdhury, K., Iraj, M.A.: Predicting autism spectrum disorder using machine learning classifiers. In: 2020 International conference on recent trends on electronics, information, communication & technology (RTEICT), pp. 324–327. IEEE, Piscataway, NJ (2020)
25. Kosmidis, K., Macheras, P.: A fractal kinetics SI model can explain the dynamics of COVID-19 epidemics. PLoS One. 15(8), e0237304 (2020)
26. Zhang, F., Wei, Y., Liu, J., Wang, Y., Xi, W., Pan, Y.: Identification of autism spectrum disorder based on a novel feature selection method and variational autoencoder. Comput. Biol. Med. 148, 105854 (2022)
27. Larsen, E., Menashe, I., Ziats, M.N., Pereanu, W., Packer, A., BanerjeeBasu, S.: A systematic variant annotation approach for ranking genes associated with autism spectrum disorders. Mol. Autism. 7, 44 (2016). http://doi.org/10.1186/s13229-016-0103-y
28. Schaaf, C.P., Betancur, C., Yuen, R.K.C., et al.: A framework for an evidence-based gene list relevant to autism spectrum disorder. Nat. Rev.

Genet. 21(6), 367–376 (2020). http://doi.org/10.1038/s41576-020-0231-2

29. Abrahams, B.S., Arking, D.E., Campbell, D.B., et al. SFARI gene 2.0: A community-driven knowledgebase for the autism spectrum disorders (ASDs). Mol. Autism. 4, 36 (2013). https://doi.org/10.1186/2040-2392-4-36
30. Vorstman, J.A.S., Scherer, S.W.: Contemplating syndromic autism. Genet. Med. 25(10), 100919 (2023)
31. Cooper, J.N., Mittal, J., Sangadi, A., Klassen, D.L., King, A.M., Zalta, M., Mittal, R., Eshraghi, A.A.: Landscape of NRXN1 gene variants in phenotypic manifestations of autism spectrum disorder: A systematic review. J. Clin. Med. 13(7), 2067 (2024)
32. Zhuang, W., Ye, T., Wang, W., Song, W., Tan, T.: CTNNB1 in neurodevelopmental disorders. Front. Psychiatry. 14, 1143328 (2023)
33. Thudium, S., Palozola, K., L'Her, É., Korb, E.: Identification of a transcriptional signature found in multiple models of ASD and related disorders. Genome Res. 32(9), 1642–1654 (2022)
34. Rahnama, M., Tehrani, H.A., Mirzaie, M.: Identification of key genes and convergent pathways disrupted in autism spectrum disorder via comprehensive bioinformatic analysis. Inf. Med. Unlocked. 24, 100589 (2021)
35. Sabo, S.L., Lahr, J.M., Offer, M., Weekes, A.L., Sceniak, M.P.: GRIN2B-related neurodevelopmental disorder: Current understanding of pathophysiological mechanisms. Front. Synaptic Neurosci. 14, 1090865 (2023)
36. Cakiroglu, C., Demir, S., Ozdemir, M.H., Aylak, B.L., Sariisik, G., Abualigah, L.: Data-driven interpretable ensemble learning methods for the prediction of wind turbine power incorporating SHAP analysis. Expert Syst. Appl. 237, 121464 (2024)

How to cite this article: Singh, N.K., Patel, A., Verma, N., Singh, R.K.B., Sharma, S.K.: Hybrid deep learning method to identify key genes in autism spectrum disorder. Healthc. Technol. Lett. 12, e12104 (2025). https://doi.org/10.1049/htl2.12104