Predicting Human Essential Genes Using Deep Learning: MLP with Adaptive Data Balancing

Ahmed AbdElsalam[1], Mohamed Abdallah[2], Hossam Refaat[3]
Department of Information System-Faculty of Computers and Artificial Intelligence, University of Sadat City, Monifia, Egypt[1]
Department of Information System-Faculty of Computers and Informatics, Suez Canal University, Ismailia, Egypt-41522[2, 3]

***Abstract*—Artificial intelligence (AI) has transformed many scientific disciplines including bioinformatics. Essential gene prediction is one important use of artificial intelligence in bioinformatics since it is necessary for knowledge of the biological pathways needed for cellular survival and disease diagnosis. Essential genes are fundamental for maintaining cellular life as well as for the survival and reproduction of organisms. Understanding the importance of these genes can help one to identify the basic needs of organisms, point out genes connected to diseases, and enable the development of new drugs. Traditional methods for identifying these genes are time consuming and costly, so computational approaches are used as alternatives. In this study, a Multi-Layer Perceptron (MLP) model combined with ADASYN (adaptive synthetic sampling). Furthermore, using deep learning techniques to solve the restrictions of traditional machine learning techniques and raise forecast accuracy attracts a lot of interest. It was proposed to handle data imbalance. The model utilizes features from protein-protein interaction networks, DNA and protein sequences. The model achieved high performance, with a sensitivity of 0.98, overall accuracy of 0.94, and specificity of 0.96, demonstrating its effectiveness in data classification.**

Keywords*—*Artificial intelligence; bioinformatics; deep learning; Multi-Layer Perceptron (MLP); imbalanced-handling techniques; essential gene prediction; sequence characteristics

I. Introduction

Since they carry essential biological functions that cannot be replaced, essential genes are vital for the survival and procreation of life. Understanding the minimal biological requirements of organisms and identifying disease-associated genes depends on the ability to forecast these genes, therefore guiding a basic step in pharmacological research and therapeutic progress. Nevertheless, even if finding important genes is important, traditional laboratory methods remain expensive, time-consuming, and need specialist knowledge and a lot of work. Recent studies have therefore shifted to computational approaches using data from human cell lines and model organisms. Faster and more effective prediction of these genes is made possible by developments in machine learning and deep learning, allowing researchers to build more accurate and efficient models for evaluating the interactions between important genes and other biological characteristics.

This work presents a novel deep learning methodology combining numerous biological data sources including DNA sequence features, protein sequence attributes, and protein-protein interaction (PPI) network embeddings for anticipating important human genes. Unlike existing methods depending on network topology analysis or machine learning models using manually produced attributes, the proposed model offers many major contributions:

a) Combining several biological data sources to increase predictive precision: Three main categories of biological data are combined in the method to provide a more complete gene analysis: DNA sequence properties, codon frequency, GC content, and gene length. Features of protein sequences include the length of the protein and amino acid distribution. Node2Vec was used to build protein-protein interaction (PPI) network embeddings, therefore capturing network gene linkages [1] . This integration helps the model to expose more significant interactions among genes, hence improving the classification accuracy compared to previous methods.

b) Reducing class unbalance with ADASYN: Usually underrepresented in biological datasets, essential genes cause biased predictions favoring the majority class (non-essential genes). ADASYN (Adaptive Synthetic Sampling) was used to create synthetic samples for the minority class to handle this problem. This preserves a balanced dataset and considerably increases the model's ability to identify important genes [2].

c) Improved relative efficacy against conventional machine learning models: Support Vector Machine (SVM), Random Forest, AdaBoost, and Naïve Bayes were among the conventional machine learning methods used in the evaluation of the proposed Multi-Layer Perceptron (MLP) model. The results showed that the proposed model validated its effectiveness in important gene categorization since it obtained the best accuracy (94.38%), sensitivity (98.27%), and specificity (90.43%).

d) Improved model architectural design and regularization strategies: Several advanced techniques were used to ensure the best performance and reduce overfitting: batch normalization, which standardizes input distributions across layers, hence improving training efficacy. Dropout (0.03) helps to reduce too strong reliance on specific neurons, so enhancing generalization [3], [4]. Designed to systematically change the learning rate to improve convergence, cosine decay learning rate scheduling Early Stopping: if no improvement is found after 25 consecutive epochs, training is automatically stopped.

e) Prospective applications in genetic studies and biomedical research: With future uses in pharmaceutical discovery, disease gene identification, and functional

genomics. This work enhances the design of computer instruments for gene analysis. The proposed approach offers a strong framework for other species or genetic data utilization to further research.

Most past research relies on traditional machine learning techniques, which often face constraints such as manually acquired characteristics, thereby reducing the potential to find complex patterns in biological data. Inappropriate handling of unbalanced datasets that reduces the predicting accuracy for important genes. Data integration is limited since many studies focus just on either sequence-based traits or network structure, but rarely on both concurrently. On the other hand, our approach solves these challenges by using deep learning to identify hidden trends in biological data. The integration of DNA, protein, and PPI network features provides a whole understanding of gene essentiality. Equilibrating the dataset using ADASYN guarantees that the model is effectively trained on both important and non-essential genes. This work offers a more accurate and scalable approach for the investigation of genetic functions and their biological consequences, therefore reflecting significant progress in key gene prediction.

The remaining sections of this paper are structured as follows. Section II analyzes relevant literature, Section III provides the proposed model, Section IV provides a detailed description of proposed model, Section V provides Implementation Details, Section VI presents Results and discussion and finally, Section VII summarizes the most significant findings and conclusions.

II. Related Work

- Measures of Centrality in Network-Based Essential Gene Prediction

Examining the connection of important genes across biological networks is one approach to predict them. Research shows that compared to proteins with fewer contacts, those with more contacts inside a protein-protein interaction (PPI) network are more likely to be significant. Validated over several species, the idea is known as the centrality-lethality rule. Still, reliance just on network topology to determine gene essentiality has shown some degree of error. This restriction has several causes. PPI networks are less reliable and often insufficient and noisy. Second, several biological factors influence gene essentiality and cannot be explained by network connections by themselves. Recent studies have shown new centrality measures that combine network topology with additional biological data, therefore improving prediction accuracy and offering a more reliable and whole approach for identifying important genes. Various strategies have been developed to overcome the limitations of conventional methods by combining network architecture with additional biological data to improve the accuracy of fundamental gene prediction: CoEWC: This method synthesizes network topological characteristics with gene expression data, so enabling the identification of shared attributes of fundamental proteins in both date hubs and party hubs. Performance has been much improved by this integration compared to methods based only on protein-protein interaction (PPI) networks [5]. Zhang et al. presented an ensemble approach combining protein-protein interaction networks with gene expression data, therefore enhancing the predicted accuracy of widely used centrality measures [6]. Additionally presented was the OGN method, which uses orthologs in reference organisms, co-expression likelihood with nearby proteins, and network topology [7]. To better identify key genes, Li et al., created the GOS model, which combines gene expression, orthology, subcellular localization, and protein-protein interaction networks [8]. By combining protein domain properties with topological analysis of protein-protein interaction networks, UDoNC enhances fundamental protein prediction [9]. The fundamental dependence of centrality-based prediction methods is on a scalar score, which is derived either from biological networks or using the integration of several data sources. These approaches have produced progress, but they still lack enough accuracy in locating all important genes. Recent research provides rich new perspectives on centrality measurements and their relevance in forecasting critical genes and proteins [10].

- Methods of Machine Learning for Forecasting Gene Essentiality

One important method for estimating gene essentiality is using machine learning to combine several signals coming from many biological data sources. For this aim, Zhang et al. conducted an extensive evaluation of machine learning techniques highlighting the difficulties and possible directions for next research. Most machine learning-based predictive models have been assessed mostly on model organisms, therefore limiting their use in other settings. Conventional machine learning techniques usually need hand feature selection and extraction. This process calls for a thorough understanding of the biological field and knowledge of the relationship between gene essentiality and other kinds of biological data [11]. Using features taken from the λ-interval Z curve based on nucleotide sequence data, Guo et al. projected human gene essentiality using Support Vector Machines [12]. One main limitation of manually produced properties is their scope. Protein-protein interaction (PPI) networks [11] produce many topological metrics, including degree centrality, betweenness centrality, closeness centrality, subgraph centrality, and eigenvector centrality. Although studies on the link between these traits and gene essentiality in various organisms have been conducted, their predictive efficacy, either independently or integrated into machine learning methods, remains inferior to features derived automatically via deep learning frameworks [13]. Forecasting gene essentiality using machine learning combined with biological data has great potential. The challenges related to featuring extraction and the limited scope of manually selected characteristics underline the need for more advanced approaches, such as deep learning, to improve forecast accuracy.

- Deep learning approaches in bioinformatics

Deep learning has been a powerful tool in many fields of bioinformatics recently, including medical picture segmentation [14], drug-target prediction [15], and critical gene prediction [13], [16], [17] . Convolutional neural networks (CNNs) have shown to be helpful in the automatic feature extraction from image and sequence data [14], [15] thus, this overview emphasizes important field successes and

approaches. Zeng and colleagues used convolutional neural networks to identify notable trends in gene expression profiles of time-series. They converted these data into models of cell cycles, therefore enabling the prediction of key genes[13].

Time-series gene expression data were investigated by Zeng et al. using bidirectional Long Short-Term Memory (LSTM) cells. Emphasizing studies conducted using Saccharomyces cerevisiae [16], their approach combined gene expression data with subcellular localization information and protein-protein interaction (PPI) networks. Using manually obtained variables from sequence data, Hasan and colleagues built a neural network of six hidden layers to predict gene essentiality in microorganisms [17]. Deep learning-based network embedding methods have recently been presented to independently generate lower-dimensional representations for every node inside a network [1]. For every protein in a PPI network, Zeng et al. derived network properties using the node2vec technique [1]. They showed that more useful information is produced from this low-dimensional representation than from manually calculated traditional centrality metrics [13] , [16]. Deep learning approaches such as CNNs and LSTMs, as well as fresh ideas as network embedding, have greatly advanced bioinformatics and gene essentiality prediction. These techniques highlight how well deep learning might improve the accuracy and efficiency of biological data analysis.

Recent developments in CRISpen-Cas9 and gene-trap technologies have helped to identify important genes in many human cancer cell lines, therefore improving our knowledge of the requirements for maintaining basic biological functioning over many tumor types [18-20]. These important genes point to possible targets for the development of cancer treatments [21]. Together with other biological information sources, the availability of important gene data offers a chance to assess the hypothesis that computational techniques may exactly forecast human gene essentiality. Previous studies have indicated that omics experimental data's acquired properties are efficient tools for predicting gene essentiality. Still, for poorly studied species, such information is usually lacking. Therefore, various studies have focused on developing models that predict gene essentiality without using additional experimental data by depending just on attributes obtained from sequence data, including DNA and protein sequences [12], [17].

III. The Proposed Model

The proposed model integrates three main data sources to predict gene essentiality as shown in Fig. 1. Input Layer: DNA Sequence: Encodes the genetic information that defines protein synthesis inside the cell. Protein Sequence: Shows the structural make-up of the created proteins. Capturing interactions between proteins, Protein-Protein Interaction (PPI) Network offers understanding of gene roles and biological processes. These data kinds are handled to extract numerical features that feed the deep learning model. Deep Learning Architecture: The model is built on a Multilayer Perceptron (MLP) architecture, consisting of three layers: Layer 1 - Input Layer: Receives the extracted features from DNA, protein sequences, and the PPI network. Layer 2 - Hidden Layer: A non-linear transformation layer using activation functions like GELU to capture complex patterns in the data. Layer 3 - Output Layer: Predicts whether a gene is essential or non-essential using activation functions such as Sigmoid. Deep Learning and Data Balancing Strategies: Data-balancing methods were included to improve the performance of the model and lower bias towards the dominant class since important and non-essential genes are frequently skewed in datasets. This approach guarantees a more exact classification of gene essentiality and enhances prediction accuracy. Output Layer: Predicting both non-essential and necessary genes in humans is the last aim of this strategy. This integrated strategy improves the accuracy and robustness of key gene prediction by aggregating several biological data sources inside a deep learning framework and using data balancing strategies.

A. Model Selection

In this study, the goal is to classify essential genes based on numerical representations of biological features, such as codon frequencies and protein properties. Given the nature of the data, the model selection was guided by several key considerations:

- Lack of Spatial Pattern Extraction Requirement While Convolutional Neural Networks (CNNs) excel at extracting spatial patterns from image data, the data used in this study is represented numerically without spatial structure. Features such as codon frequencies and protein properties do not follow a spatial arrangement, making CNNs unnecessary. Therefore, a more suitable approach is the use of Multi-Layer Perceptron (MLP), which is capable of directly processing numerical data without relying on spatial patterns.
- No Need for Sequential Data Processing Recurrent Neural Networks (RNNs) are designed for sequential data where the temporal order is significant, such as time-series data or natural language processing. However, the features in this study are static and do not depend on temporal or sequential ordering. As such, RNNs were deemed unsuitable for this task, and MLP was preferred due to their ability to handle fixed, non-sequential data efficiently.
- Limitations of Graph Neural Networks (GNNs) Graph Neural Networks (GNNs) are highly effective in situations where the data is represented in graph form, such as protein-protein interaction (PPI) networks. However, the dataset in this study incorporates a combination of features from various sources, such as DNA sequence data, protein sequences, and statistical features, making the use of GNNs alone less effective. MLP, on the other hand, can easily integrate these diverse types of data and provide a more practical solution for gene classification.
- Computational Efficiency and Simplicity MLPs offer a simpler architecture compared to CNNs and GNNs, resulting in faster training and reduced computational cost. In the context of large-scale biological datasets, computational efficiency is crucial. MLP provides a balance of high classification accuracy

and low computational overhead, making them the ideal choice for this study

IV. Explanation of the Proposed Model

a) Features of DNA sequences: When we examine DNA sequence features, we are addressing the characteristics that improve our grasp of genes. Codon frequency is a measurement of the frequency with which each three-nucleotide codon appears in a gene. This frequency helps us to understand the conversion of genetic data into proteins. Frequent use of some codons implies that the gene might be more effective in expressing itself. Calculating the proportion of the cytosine (C) and guanine (G) nucleotides in the DNA sequence, GC content is another crucial aspect. Usually indicating a structural stability of the gene, a high GC content can affect the expression of the gene. Since it indicates the entire count of nucleotides in the gene, gene length is also important. More information included in longer genes influences their organization and expression. For a given organism, the codon adaptation index (CAI) gauges how well the codon sequence fits the preferred codons. Higher CAI values imply that the gene is more suited for these tastes, which can cause greater expression levels. The Maximal Relative Synonymous Codon Usage (RSCUmax) evaluates, at last, the usage of synonymous codons corresponding to the same amino acid in the gene. This clarifies the preferences of codon use of the gene [17], [22].

b) Features of protein sequences: Turning now to protein sequence characteristics, these center on protein physical and chemical characteristics. Amino acid frequencies which gauge the frequency of every amino acid in the protein sequence are one of main characteristics. Understanding the chemical makeup and interactions of the protein with other molecules depends on this knowledge. Defined as the total count of amino acids in the protein, protein length is another crucial consideration. The structure of the protein and its capacity for biological operations can be much influenced by its length [17], [22].

c) Protein-protein interaction (ppi) network characteristics: Regarding Protein-Protein Interaction (PPI) networks, these characteristics are essential for comprehending the interactions among several proteins. Node2Vec allows us to extract characteristics from the PPI network whereby every gene is expressed as a node [1]. This enables a thorough investigation of protein interactions, therefore illuminating information on the functional activities of genes inside the network. These associations allow us to extract roughly 64 features that mirror gene interactions. These aspects help to clarify the biological relevance of protein interactions as well as their consequences for gene activity.

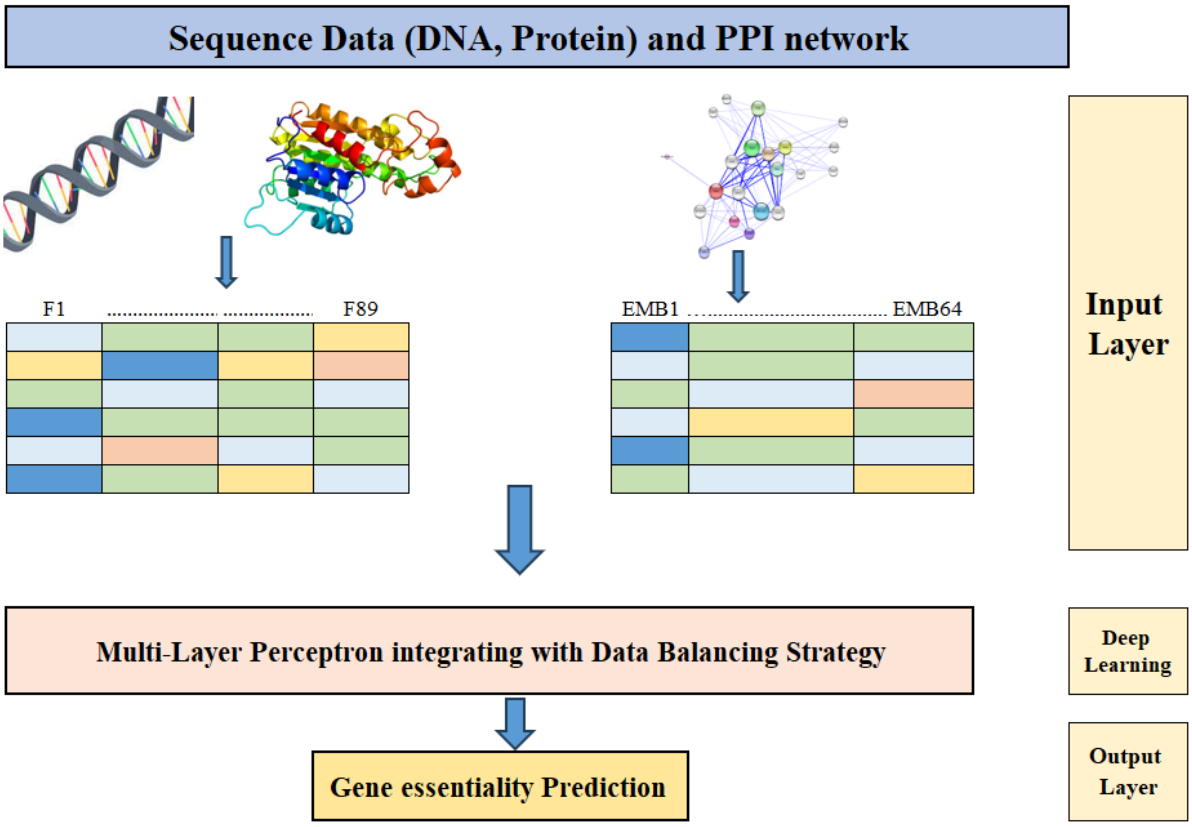

Fig. 1. Model architecture.

The dataset consists of several characteristics obtained from many kinds of biological data. There are five main elements to DNA sequence data: 64 codon frequencies, GC content, gene length, Codon Adaptation Index (CAI), and Maximal Relative Synonymous Codon Usage (RSCUmax), therefore generating 68 characteristics. With 22 characteristics, the protein sequence data consists in amino acid frequencies and protein length. Furthermore, automatically learning 64 features for every gene in the Protein-Protein Interaction (PPI) network is a network embedding technique known as Node2Vec. Comprising 153 characteristics in all, the dataset combines PPI network, DNA sequence, and protein sequence insights.

d) A deep learning method for handling imbalanced data using MLP and ADASYN: In classification challenges, imbalanced datasets provide a significant challenge since models often show bias towards the majority class, therefore compromising generalizing for the minority class. This work generates additional synthetic samples for the minority class by using a Multi-Layer Perceptron (MLP) model in combination with ADASYN (adaptive synthetic sampling) [2]. This approach increases classification efficiency and guarantees a fairer dataset. Data preparation, model architecture design, training with optimization strategies, and performance evaluation constitute part of the approach. The proposed model is meant to identify both non-essential and essential human genes. Along with characteristics derived from the Protein-Protein Interaction (PPI) network, feature extraction is done using DNA and protein sequences. Node2vec helps to automatically extract PPI features. Following their aggregation into a single feature vector with 153 attributes, the acquired features serve as the MLP model's input layer.

e) Data preprocessing and class balancing: First loaded and preprocessed is the dataset whereby the feature matrix separates from the target variable. The feature distribution is standardized using StandardScaler, therefore turning the data into zero, and its variance is one. The class imbalance in the dataset presents a major challenge that can produce biased predictions. This is treated with ADASYN. By a data-driven approach, ADASYN generates synthetic cases for the minority class unlike conventional oversampling techniques, therefore guaranteeing a more accurate distribution. Using stratified sampling to maintain class ratios, the resampled dataset is next split into training and testing sets (80% training, 20% testing).

f) MLP model architecture: The proposed MLP model consists of multiple layers meant to find complex trends in the data. An Input Layer of 153 neurons makes up the architecture and represents the count of retrieved features from PPI networks, protein sequences, and DNA sequences. There are 1024 hidden layers using GELU activation, 512 hidden layers using GELU activation, and 256 hidden layers using GELU activation. Batch Normalization to improve training stability and Dropout (0.03) to minimize overfitting follows each hidden layer. Given a binary classification job, a solitary neuron using Sigmoid activation. The model parameters are presented in Table I.

g) Optimization and regularization strategies: Several optimization techniques were used to reduce overfitting and enhance the training process: Optimizer: The model uses AdamW, an adaptive optimizer that combines weight decay to minimize strong weight changes. Learning rate scheduling is done using a Cosine Decay Learning Rate method, therefore enabling a slow drop in the learning rate throughout training periods rather than abrupt changes [3], [4]. This approach increases stability and convergence. Class Weights: The loss function is changed to give the minority class (class 0: 0.8, class 1: 1.5 more relevance to solve class imbalance.

h) Overfitting prevention and performance enhancement: Batch Normalization: Preserves a constant activation distribution, accelerating convergence and increasing generalizing ability, so improving the generalization of the model. Dropout (0.03): Randomly deactivates a portion during training to reduce reliance on specific neurons and hence prevent overfitting. Early Stopping: Checks validation loss and stops training should no improvement be observed after 50 epochs, therefore restoring the weight of the ideal model. Automatically lowers the learning rate by 50% once a validation loss plateaus, therefore enabling continuous improvement of model performance.

TABLE I. MODEL PARAMETERS

Component	Details
Input Layer	Number of features in the dataset (153)
Hidden Layer 1	1024 nodes, Activation: GELU, Dropout: 0.03
Hidden Layer 2	512 nodes, Activation: GELU, Dropout: 0.03
Hidden Layer 3	256 nodes, Activation: GELU, Dropout: 0.03
Output Layer	1 node, Activation: Sigmoid
Epochs	100
Early Stopping	Patience:25 epochs
Optimizer	AdamW with Cosine Decay Learning Rate

V. IMPLEMENTATION DETAILS

Using Python 3.8, TensorFlow and Keras for deep learning, and Scikit-learn for data preparation and evaluation, the proposed model was run. StandardScaler helped us standardize the data such that every feature fits a zero-mean, unit-variance distribution. ADASYN was used to generate synthetic samples for the minority class before stratified sampling split the dataset into 80% training and 20% testing, therefore helping to reduce class imbalance. Using GELU activation, the MLP model consisted of three hidden layers with 1024, 512, and 256 neurons correspondingly, succeeded by Batch Normalization and Dropout (0.03) to increase generalization and reduce overfitting. AdamW, combined with Cosine Decay Learning Rate Scheduling, helped to improve the model, guaranteeing training stability. Using Early Stopping (patience = 25) to prevent overfitting, the model ran through 100 epochs. Although the Quadro P620 helps CUDA acceleration, its computational capacity is less than that of high-end GPUs such the Tesla V100, which causes extended training times even if the experiments were conducted on a system with an Intel Core

i7-10750H CPU (2.60 GHz), 8GB RAM, and an NVIDIA Quadro P620. Still, careful change of batch size and learning rate helped to preserve training efficiency. The performance of the model in important gene categorization was shown by accuracy, sensitivity, specificity, and AUC-PR evaluation.

VI. Implementation and Results

a) Data collection: The Essential Genes Data (DEG) collection consists of twenty freely available sets of basic human genes [23]. To ensure complete coverage of important genes, we obtained and included all 20 data sets in our study [18-20], [24-26], [27-29] . We categorized essential genes found in a minimum of five independent datasets if around 10% of human genes are essential [20]. This criterion led us to identify 2162 essential genes, nearly 10% of the human genome. The genes not categorized as essential in the DEG database were labeled as non-essential. Table II presents Datasets Database. Protein-Protein Interaction (PPI) Network Data included only physically proven interactions among human proteins, derived from experiments. Eliminating self-interactions and many small, disconnected subgraphs helped to improve the dataset to produce a PPI network with 17,786 nodes and 355,646 edges. Embedding features reflecting the connectivity patterns of every gene within the network were obtained from this well-chosen interaction network. From the PPI network, each gene derived 64 embedding features overall. Essential genes: 2145; genes with both sequence features and network embedding. Non-essential genes: 7,680. There are 9,825 examples in the last dataset, and each one features 153 attributes.

b) Evaluation metrics: The Area Under the Receiver Operating Characteristic Curve (AUC-ROC) evaluates the model's performance in fit for balanced classification situations when all classes have roughly equal instance counts. When there is uneven classification, the Precision-Recall (PR) curve provides a more perceptive evaluation. The Area Under the Precision-Recall Curve (AP) is a more representative measure than the Area Under the Receiver Operating Characteristic Curve (AUC-ROC) since human essential gene prediction represents an unbalanced classification problem. We incorporate many statistical performance measures in addition to AUC and AP, namely Sensitivity, Specificity, Positive Predictive Value, and Accuracy, defined in Eq.(1) to (4).

$$\text{Sensitivity} = \frac{TP}{TP+FN} \quad (1)$$

$$\text{Specificity} = \frac{TN}{FP+TN} \quad (2)$$

$$\text{Positive Predictive Value} = \frac{TP}{TP+FP} \quad (3)$$

$$\text{Accuracy} = \frac{TP+TN}{TP+FN+TN+FP} \quad (4)$$

where, TP (True Positives): The number of correctly classified essential genes.TN (True Negatives): The number of correctly classified non-essential genes. FP (False Positives): The number of non-essential genes misclassified as essential. FN (False Negatives): The number of essential genes misclassified as non-essential. Especially in addressing the class imbalance inherent in essential gene prediction, these measures provide a complete evaluation of the classification performance of the model.

c) Ablation study: In this section, an ablation study was conducted to assess the impact of various components of the Multi-Layer Perceptron (MLP) model on the classification accuracy of essential genes. The primary objective was to identify the most influential factors within the model and evaluate how the exclusion or modification of specific parameters or inputs influences overall model performance.

To ensure the optimality of the selected model architecture (1024-512-256) and dropout rate (0.03), a series of controlled experiments were performed. The purpose of these experiments was to examine the effect of these variables on model performance, confirming that each design choice contributed positively to enhancing accuracy while also addressing challenges such as overfitting and class imbalance.

TABLE II. Datasets Database

Data	Database	File name
DNA and protein sequence data	Ensembl [30]	release 97, July 2019
PPI data	BioGRID [31]	release 3.5.181, February 2020
Essential genes data	DEG	Homo sapiens(DEG2006: DEG2032)

As shown in Table III and Fig. 2 presents the results of the ablation study, comparing the performance of various MLP architectures based on several evaluation metrics. The analysis includes accuracy, stability, tendencies toward overfitting, and the model's handling of class imbalance across the selected MLP architectures.

- 1024-512-256 Architecture: This architecture provides an effective balance between training and validation accuracy, showing a significant improvement in both test accuracy and Area Under the Curve (AUC).
- 512-256-128 Architecture: While this architecture yields good test accuracy, it is lower than that of the larger architecture. However, it achieves the best test loss among the configurations tested.
- 2048-1024-512 Architecture: This model achieves the highest training accuracy but is prone to overfitting. Nevertheless, it performs well on the validation data.

Table IV outlines the performance metrics for models with varying dropout rates, illustrating how train accuracy, validation accuracy, test accuracy, AUC scores, and loss are influenced by different dropout values:

- Dropout = 0.0 (No Dropout): In this configuration, overfitting is observed, as the model excels on the training data but struggles to generalize to validation and test data.
- Dropout = 0.03: This rate strikes an optimal balance between regularization and model performance. It reduces overfitting while maintaining high accuracy and AUC scores, yielding the best test accuracy (~94%) and the lowest test loss (~0.20).

- Dropout = 0.1: This configuration results in under fitting due to excessive regularization. It produces the lowest accuracy on the test data and the lowest AUC scores, although it achieves the lowest test loss (~0.19), suggesting some improvement in generalization.

In conclusion, a dropout rate of 0.03 provides the best trade-off between mitigating, overfitting and achieving high accuracy across training, validation, and test datasets

d) Performance evaluation

- Comparison of Traditional Machine Learning and Deep Learning Models in Classification

As shown in Fig. 4 and Table V, Deep Learning Models vs Conventional Machine Learning. Artificial intelligence applications depend on the proper model for classification tasks since model performance varies depending on data characteristics and class equilibrium. This paper evaluated and compared the performance of standard machine learning techniques, including AdaBoost, SVM, Random Forest, and Naïve Bayes with that of a Multi-Layer Perceptron (MLP) network coupled with ADASYN, a deep learning approach. Founded on fundamental performance measures— Sensitivity, Specificity, Positive Predictive Value, and Accuracy—the assessment sought to find the most effective model for the given dataset.

Conventional models showed significant performance variability; the Support Vector Machine (SVM) proved to be rather robust in identifying positive samples with a maximum sensitivity of 0.9693. With its best overall accuracy of 0.9015, it is a well-balanced choice between sensitivity and specificity. Random Forest showed a high sensitivity of 0.9776 yet a reduced specificity of 0.7585, therefore suggesting a higher false positive rate. AdaBoost achieved an overall accuracy of 0.8428 and showed a better-balanced performance than SVM in general efficacy, although it did not surpass SVM. Naïve Bayes had the lowest sensitivity of 0.6176, indicating poor identification of positive instances, and the highest specificity of 0.8932, therefore demonstrating its effectiveness in lowering false positives. Still, its general performance in categorization was worse than that of other models.

- Main Observation

The MLP running ADASYN exceeded all conventional models with the best accuracy of 94.38%. With a sensitivity of 96.93% and an accuracy of 90.15%, the Support Vector Machine (SVM) exceeded other traditional machine learning models. With high sensitivity (97.76%) but poor specificity (75.85%), the Random Forest model suggested a higher false positive rate. Naïve Bayes showed the lowest sensitivity (61.76%) but the highest specificity (89.32%) showing better performance in reducing false positives and less efficacy in discovering positive situations. AdaBoost showed a reasonable performance; however, it fell short of SVM or deep learning. Deep learning, especially when combined with data augmentation techniques such as ADASYN, clearly improves classification performance, so it is the most effective solution for this problem. With a sensitivity of 0.9827, an overall accuracy of 0.9438, and a specificity of 0.9643, the Multi-Layer Perceptron (MLP) with ADASYN model outperformed all other methods. These findings highlight its remarkable ability for exact data classification, particularly in view of imbalance-handling techniques like ADASYN. The deep learning model produced quite improved classification results by remarkably identifying complex patterns in the sample.

TABLE III. MLP ARCHITECTURES BASED ON SEVERAL EVALUATION METRICS

Architecture	Best Training Accuracy	Best Validation Accuracy	Test Accuracy	Best AUC	Test Loss	Number of Epochs
512-256-128	High, but lower than other architectures	Good, but lower than larger architectures	~ 91%	~ 0.97	Highest among the three	100
1024-512-256	Very high, with better stability	Very good with reduced fluctuations	~ 93%	~ 0.98	Relatively lower	100
2048-1024-512	Highest, but with overfitting	Very good performance with slight fluctuations	~ 92%	~ 0.98	Lower than the smaller architecture, but not much improved	100

TABLE IV. PERFORMANCE METRICS FOR MODEL WITH VARYING DROPOUT RATES

Dropout	Train Accuracy	Validation Accuracy	Test Accuracy	Train AUC	Validation AUC	Test AUC	Train Loss	Validation Loss	Test Loss
0	~99%	~95%	~93%	~1.00	~0.98	~0.97	Low	Higher	~0.22
0.03	~98%	~96%	~94%	~0.99	~0.98	~0.975	Moderate	Lower	~0.20
0.1	~97%	~94%	~92%	~0.98	~0.97	~0.965	Higher	Lower	~0.19

- Evaluation of Model Performance During Training and Testing

The training process was assessed over numerous epochs using accuracy, AUC, and loss measures as shown in Fig 5. During the first epoch, the training accuracy and AUC showed a rapid rise and then stabilized at about 1.0, indicating that the model efficiently absorbed the training data. With a final test accuracy exceeding 0.9 and a test AUC over 0.98, the validation accuracy and AUC show consistent enhancement, approaching the test performance, demonstrating strong classification skill. Whereas the validation loss showed volatility before steadying, the loss curves show that the

training loss fell sharply in the first epoch and stayed low. The test loss stayed constant, suggesting that the model fits fresh data rather well. These results support the durability and efficiency of the model in separating non-essential from essential genes.

TABLE V. PERFORMANCE COMPARISON OF MLP WITH ADASYN, ADABOOST, SVM, RANDOM FOREST, AND NAÏVE BAYES

Model	Sensitivity	Specificity	Positive Predictive Value	Accuracy
AdaBoost	0.8495	0.8359	0.8403	0.8428
Support Vector Machine (SVM)	**0.9693**	0.8327	0.8548	**0.9015**
Random Forest	0.9776	0.7585	0.8044	0.8689
Naïve Bayes	0.6176	**0.8932**	0.8546	0.7543
MLP + ADASYN (Deep Learning)	**0.9827**	**0.9043**	**0.9126**	**0.9438**

- Analysis of Training and Evaluation Curves
  - Fig. 3 presents the loss and accuracy curves during the training and evaluation phases across five folds using K-Fold Cross Validation.
  - The loss curve shows a rapid decrease in loss values during the early stages of training, reflecting the model's quick adaptation to the data. However, some fluctuations in the evaluation loss values are observed, which might indicate potential for slight overfitting.
  - The accuracy curve in Fig. 6 demonstrates that the model achieves a high accuracy rate exceeding 90% after a few training epochs, with the performance stabilizing afterward. The small gap between the training and evaluation curves suggests the model's stability and minimal overfitting.

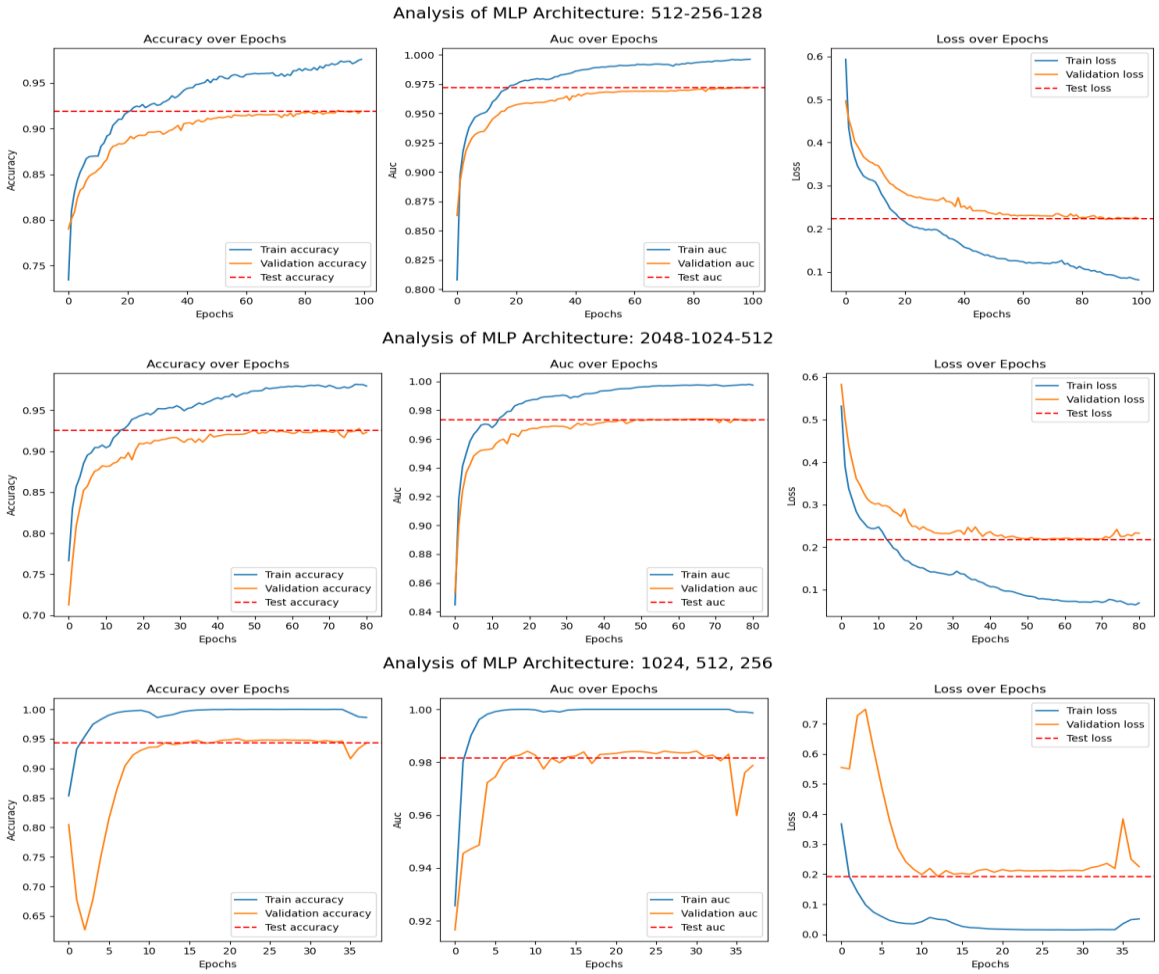

Fig. 2. Analysis of MLP architectures.

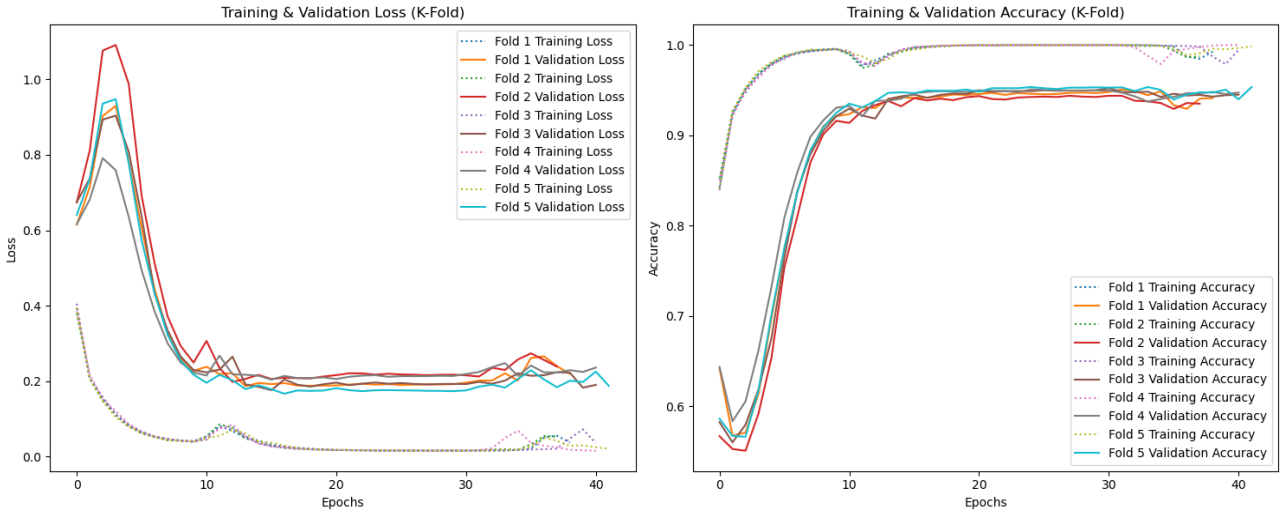

Fig. 3. Loss and accuracy curves.

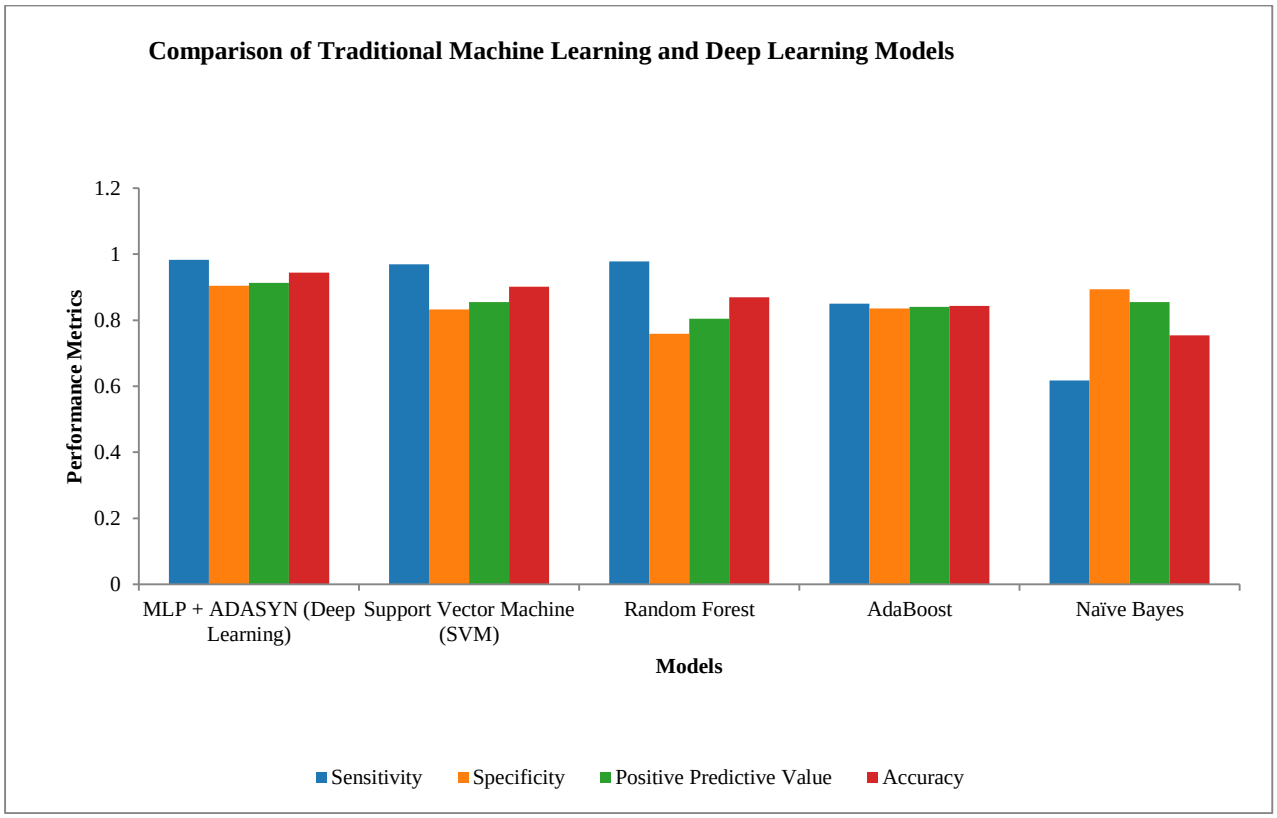

Fig. 4. Performance comparison of MLP with ADASYN, AdaBoost, SVM, random forest, and naïve bayes.

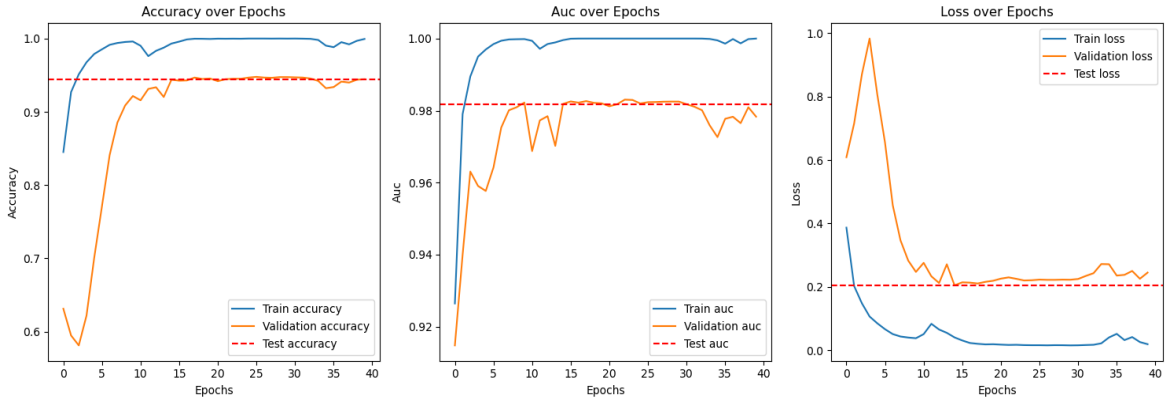

Fig. 5. Training and evaluation metrics over epoch.

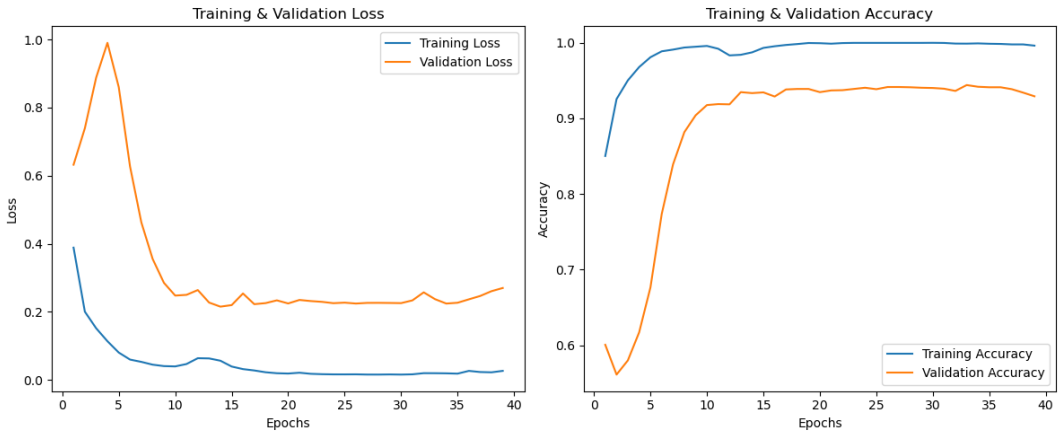

Fig. 6. Training and validation.

- Receiver Operating Characteristic (ROC) Curve and the Precision-Recall (PR) Curve

In Fig. 7, the ROC curve reflects the relationship between the True Positive Rate (TPR) and the False Positive Rate (FPR). The results show an AUC value of 0.98, indicating the model's high ability to discriminate between different classes.

The Precision-Recall curve illustrates the relationship between precision and recall, achieving a PR AUC value of 0.98. This indicates the model's effectiveness in maintaining a high balance between precision and recall, which is crucial in scenarios with imbalanced datasets.

The training and evaluation curves show a gradual improvement in model performance, while the high AUC values in both the ROC and Precision-Recall curves highlight the model's strong classification capability.

The model demonstrates high efficiency in classifying data, achieving high accuracy and excellent AUC values, which reflects its ability to effectively separate the target classes.

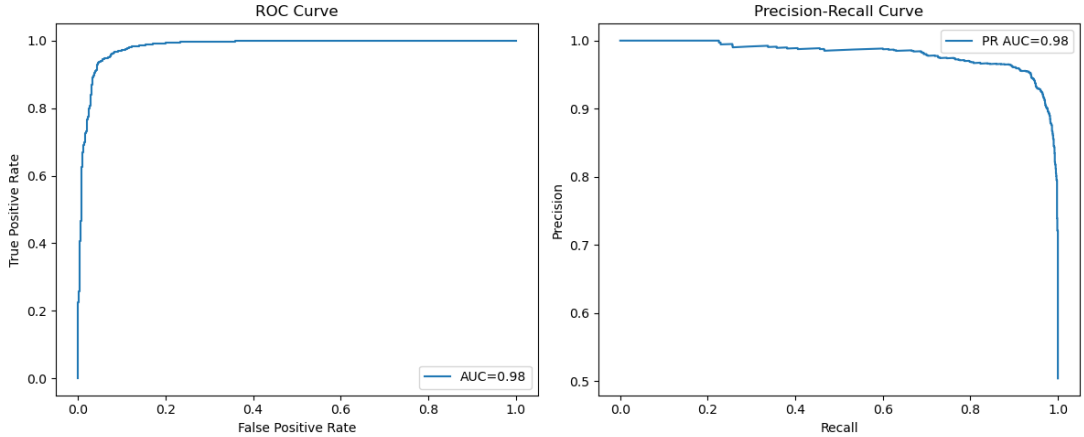

Fig. 7. Model Performance evaluation: ROC and precision-recall curves.

VII. CONCLUSION

a) Conclusion: By combining numerous biological data sources DNA sequence features, protein sequence attributes, and protein-protein interaction (PPI) this work developed a deep learning system for anticipating important human genes. With 94.38% accuracy, 98.27% sensitivity, and 90.43% specificity, the proposed MLP model using ADASYN showed improved performance relative to standard machine learning models. Especially in the management of imbalanced datasets and autonomously recognizing complex patterns in large-scale biological data, the results highlight the advantages of deep learning approaches in important gene prediction. This model is a potential tool for biological study since the combination of sequence-based properties and network topology produced a more complete and accurate classification of significant genes. Additionally, improving model generalization and stability were regularization techniques like Batch Normalization, Dropout, and Learning Rate Scheduling. The study underlined the need to balance class distribution and showed how much ADASYN enhanced model performance in predicting important genes in the minority class.

b) Future directions: Notwithstanding the positive results, there are still several paths for further improvement and research: Enhancing Feature Representation, combining epigenetic modifications, gene expression patterns, and functional annotations could increase the predictive power of the model. Examining graph-based embeddings outside Node2Vec including Graph Neural Networks (GNNs) may improve the representation of protein-protein interaction (PPI) networks. Application in Disease Gene Forecasting, since many important genes are linked to diseases, using this model to predict disease-related genes could have significant effects on pharmaceutical research and tailored therapy. For important gene prediction, the proposed deep learning system presents a strong, scalable, and well-performing approach. This work combines class-balancing methods, deep learning, and biological data to improve the biological relevance and accuracy of gene-categorizing algorithms.

DATA AVAILABILITY

All data used in this study are freely accessible from public databases:

Protein-protein interaction data are available from BioGRID database at http://thebiogrid.org/download.php.

Essential genes data and the corresponding sequence data from DEG database are available at

http://tubic.tju.edu.cn/deg/

DNA sequence and protein sequence data are available at https://useast.ensembl.org/Homo_sapiens/Info/Annotation

REFERENCES

[1] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855-864.

[2] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, 2008, pp. 1322-1328: Ieee.

[3] R. Moradi, R. Berangi, and B. J. A. I. R. Minaei, "A survey of regularization strategies for deep models," vol. 53, no. 6, pp. 3947-3986, 2020.

[4] I. Nusrat and S.-B. J. S. Jang, "A comparison of regularization techniques in deep neural networks," vol. 10, no. 11, p. 648, 2018.

[5] X. Zhang, J. Xu, and W.-x. J. P. o. Xiao, "A new method for the discovery of essential proteins," vol. 8, no. 3, p. e58763, 2013.

[6] X. Zhang, W. Xiao, M. L. Acencio, N. Lemke, and X. J. B. b. Wang, "An ensemble framework for identifying essential proteins," vol. 17, pp. 1-17, 2016.

[7] X. Zhang, W. Xiao, and X. J. P. o. Hu, "Predicting essential proteins by integrating orthology, gene expressions, and PPI networks," vol. 13, no. 4, p. e0195410, 2018.

[8] G. Li, M. Li, J. Wang, J. Wu, F.-X. Wu, and Y. J. B. b. Pan, "Predicting essential proteins based on subcellular localization, orthology and PPI networks," vol. 17, pp. 571-581, 2016.

[9] W. Peng *et al.*, "UDoNC: an algorithm for identifying essential proteins based on protein domains and protein-protein interaction networks," vol. 12, no. 2, pp. 276-288, 2014.

[10] X. Li, W. Li, M. Zeng, R. Zheng, and M. J. B. i. b. Li, "Network-based methods for predicting essential genes or proteins: a survey," vol. 21, no. 2, pp. 566-583, 2020.

[11] X. Zhang, M. L. Acencio, and N. J. F. i. p. Lemke, "Predicting essential genes and proteins based on machine learning and network topological features: a comprehensive review," vol. 7, p. 75, 2016.

[12] F.-B. Guo *et al.*, "Accurate prediction of human essential genes using only nucleotide composition and association information," vol. 33, no. 12, pp. 1758-1764, 2017.

[13] M. Zeng, M. Li, F.-X. Wu, Y. Li, and Y. J. B. b. Pan, "DeepEP: a deep learning framework for identifying essential proteins," vol. 20, pp. 1-10, 2019.

[14] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, 2015, pp. 234-241: Springer.

[15] H. Öztürk, A. Özgür, and E. J. B. Ozkirimli, "DeepDTA: deep drug–target binding affinity prediction," vol. 34, no. 17, pp. i821-i829, 2018.

[16] M. Zeng *et al.*, "A deep learning framework for identifying essential proteins by integrating multiple types of biological information," vol. 18, no. 1, pp. 296-305, 2019.

[17] M. A. Hasan and S. J. B. b. Lonardi, "DeeplyEssential: a deep neural network for predicting essential genes in microbes," vol. 21, pp. 1-19, 2020.

[18] V. A. Blomen *et al.*, "Gene essentiality and synthetic lethality in haploid human cells," vol. 350, no. 6264, pp. 1092-1096, 2015.

[19] T. Hart *et al.*, "High-resolution CRISPR screens reveal fitness genes and genotype-specific cancer liabilities," vol. 163, no. 6, pp. 1515-1526, 2015.

[20] T. Wang *et al.*, "Identification and characterization of essential genes in the human genome," vol. 350, no. 6264, pp. 1096-1101, 2015.

[21] A. J. C. s. Fraser, "Essential human genes," vol. 1, no. 6, pp. 381-382, 2015.

[22] X. Liu, B.-J. Wang, L. Xu, H.-L. Tang, and G.-Q. J. P. O. Xu, "Selection of key sequence-based features for prediction of essential genes in 31 diverse bacterial species," vol. 12, no. 3, p. e0174638, 2017.

[23] H. Luo, Y. Lin, F. Gao, C.-T. Zhang, and R. J. N. a. r. Zhang, "DEG 10, an update of the database of essential genes that includes both protein-coding genes and noncoding genomic elements," vol. 42, no. D1, pp. D574-D580, 2014.

[24] B. Georgi, B. F. Voight, and M. J. P. g. Bućan, "From mouse to human: evolutionary genomics analysis of human orthologs of essential genes," vol. 9, no. 5, p. e1003484, 2013.

[25] M. Lek *et al.*, "Analysis of protein-coding genetic variation in 60,706 humans," vol. 536, no. 7616, pp. 285-291, 2016.

[26] B.-Y. Liao and J. J. P. o. t. N. A. o. S. Zhang, "Null mutations in human and mouse orthologs frequently result in different phenotypes," vol. 105, no. 19, pp. 6987-6992, 2008.

[27] J. D. Arroyo *et al.*, "A genome-wide CRISPR death screen identifies genes essential for oxidative phosphorylation," vol. 24, no. 6, pp. 875-885, 2016.

[28] J. Bakke *et al.*, "Genome-wide CRISPR screen reveals PSMA6 to be an essential gene in pancreatic cancer cells," vol. 19, pp. 1-12, 2019.

[29] B. Mair *et al.*, "Essential gene profiles for human pluripotent stem cells identify uncharacterized genes and substrate dependencies," vol. 27, no. 2, pp. 599-615. e12, 2019.

[30] M. Ruffier *et al.*, "Ensembl core software resources: storage and programmatic access for DNA sequence and genome annotation," vol. 2017, p. bax020, 2017.

[31] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. J. N. a. r. Tyers, "BioGRID: a general repository for interaction datasets," vol. 34, no. suppl_1, pp. D535-D539, 2006.