Available online at http://scik.org

Commun. Math. Biol. Neurosci. 2026, 2026:8

https://doi.org/10.28919/cmbn/9400

ISSN: 2052-2541

CLASSIFICATION OF ESSENTIAL AND NON-ESSENTIAL GENES IN HUMAN GENOME SEQUENCE DATA USING ENSEMBLE MACHINE LEARNING

SRI KARNILA[1], FAVORISEN ROSYKING LUMBANRAJA[2,*], AKMAL JUNAIDI[2], WARSONO[3]

[1]Doctoral Program of Mathematics and Natural Sciences, Faculty of Mathematics and Natural Sciences, University of Lampung, 35145, Indonesia

[2]Department of Computer Science, Faculty of Mathematics and Natural Sciences, University of Lampung, Lampung, 35145, Indonesia

[3]Department of Mathematics, Faculty of Mathematics and Natural Sciences, University of Lampung, Lampung, 35145, Indonesia

Abstract: DNA (Deoxyribonucleic Acid), RNA (Ribonucleic Acid), and proteins are basic biochemical molecules essential for cellular organization. DNA serves as the primary store of genetic information encoded in genes, and humans are estimated to have 20,000 to 30,000 genes. This genetic information is represented in a chemical code. Essential genes are fundamental to various biological mechanisms, and when disrupted or removed, they may cause genetic defects, mutations, or, in extreme conditions, result in organismal fatality. Identifying these genes through experimental methods requires large resources and is often inefficient. Computational methods, especially those involving machine learning, offer a more efficient and effective solution to this challenge. This research explores the ability of machine learning techniques to build classification models for human gene sequence data. Two data sets, Cellular Essential Gene (CEG) and Organism Essential Gene (OEG), were analyzed, with genes categorized as essential or non-essential. The study was structured through multiple phases, such as data acquisition, cleaning, feature engineering, and dividing the dataset into subsets for model training and evaluation. Model construction followed this phase, where various ensemble learning techniques were applied. These included algorithms like Decision Tree, Support Vector Machine, Random Forest, Extreme Gradient Boosting, and Adaptive Boosting. The best overall results

*Corresponding author

E-mail address: favorisen.lumbanraja@fmipa.unila.ac.id

Received June 04, 2025

were achieved using the SVM model with 5-mers, reaching a sensitivity of 0.81 and ROC AUC of 0.81 on the CEG dataset, and a PR AUC of 0.46, sensitivity of 0.69, and ROC AUC of 0.80 on the OEG dataset. The consistently high accuracy results indicate that these models effectively distinguish essential and non-essential genes. These machine learning-based classification models can potentially be valuable tools in the healthcare field, contributing to a deeper understanding of normal gene function in organisms.
Keywords: sequence data; genes; k-mer; genetic information; classification
2020 AMS Subject Classification: 92C40.

1. INTRODUCTION

Deoxyribonucleic acid (DNA) functions as a carrier of hereditary information that is vital for the development, survival, and reproductive processes of living organisms. DNA sequencing techniques are used to identify the precise arrangement of nucleotide bases within a strand of DNA. The basis of DNA sequencing is to pass on the information required by cells to construct RNA and synthesize proteins [1]. DNA is considered the master template in all organisms that stores complete genetic instructions for cellular operation [2]. Genes are structured into extended strands known as chromosomes, with the human genome comprising 23 chromosome pairs, totaling 46 in number [3]. The number of genes in each human is about twenty thousand to thirty thousand [3]. Every nucleotide type binds to its complement on the opposing DNA strand: Adenine (A) links with Thymine (T), and Cytosine (C) connects with Guanine (G) [4]. Genes and proteins are biological data, each gene encodes a specific protein or group of proteins associated with a specific biological function [5], [6], [7].

Genes store genetic information, which is the arrangement of the four DNA bases—adenine (A), cytosine (C), guanine (G), and thymine (T) [4]. This genetic information is often processed in genomic research to identify gene function, genetic mutations, or genetic relationship to disease. Essential genes are those required to sustain life at the cellular or organismal level [8]. Genes play a role in regulating central metabolism, gene translation, DNA replication, forming basic cellular structures, and facilitating intracellular and extracellular transport [8]. The biological function of essential genes is very important because damage or deletion in the gene can cause mutations, even potentially resulting in the organism's death [9], [10]. Information about essential genes is used in various scientific studies, especially to determine potential molecular targets for drug development, such as cancer therapy or insecticide targets, as well as to design minimal genomes in biological synthesis [11], [12]. Two approaches can be implemented to understand the basic mechanisms of

life of cellular organisms in identifying essential genes, which are experimental and computational methods. Experimental methods for identifying essential genes include strategies like gene knockout, antisense RNA application, insertional mutagenesis using transposons, and CRISPR-based genome editing [13], [14].

The process of essential gene identification through experimental approaches involves techniques such as single gene deletion, antisense RNA, transposon mutagenesis, and Clustered Regularly Interspaced Short Palindromic Repeats (CRISPR). These techniques are complex, costly, and time-consuming [15], [16]. For this reason, a computational approach is required, Machine Learning (ML) methods present computational algorithms to facilitate the identification and classification [17], [18], [19].

This study focuses on enhancing the application of artificial intelligence algorithms to efficiently recognize and extract meaningful patterns from the input genetic sequence information. The problem-solving approach in this research with Ensemble Machine Learning is a classification model development method starting with: genome data collection, preprocessing, feature extraction, training, and testing to train the classification model [20].

To support the classification process, this study implements gene representation through feature engineering using the k-mer technique [21], [22]. A k-mer refers to a substring of DNA composed of k nucleotide bases. Once the DNA sequence is segmented into these k-length fragments, the number of times each unique k-mer appears is counted. These frequency values are then translated into numerical feature vectors, effectively encoding the original DNA sequence based on its k-mer distribution. The process converts DNA sequence into a numeric format for use by machine learning models, utilizing the pattern of k-mers occurrence as a feature that describes the sequence. Feature extraction from DNA sequence data serves to calculate gene frequencies (dinucleotide, trinucleotide). Code to create a dictionary that maps all possible k-mers of DNA codes ('A', 'C', 'G', 'T'), with k-mer length that can be determined according to the needs of genomic analysis. K-mer 3 will result in 4^3 = 64 mapped dictionaries, and k-mer 5 will result in 4^5 = 1024 mapped dictionaries. The result of k-mer 7 is 4^7 = 16384 mapped dictionaries. This study applies several machine learning algorithms—namely Decision Tree, SVM, Random Forest, XGBoost, and AdaBoost (Adaptive Boosting)—to construct the classification framework. The model is developed using a designated portion of the dataset and later assessed using a separate evaluation subset. To minimize the risk of the model memorizing patterns instead of learning generalizable

structures, validation is carried out using both 5-fold and 10-fold schemes. The success of each algorithm is then measured through key performance indicators such as classification accuracy, recall (sensitivity), and the model's ability to correctly reject negative instances (specificity) [23], [24], [25]. The accuracy of the ensemble method is very good on average greater than > 90%. The ML ensemble model for classification is quite effective and better than previous studies.

2. RELATED WORKS

Another investigation made use of Random Forest and Extreme Gradient Boosting (XGBoost) methods [11], employing both CEG and OEG datasets, which contained annotated records of essential and non-essential genes [26], [27]. Specifically, the CEG dataset comprised 1,127 essential and 10,320 non-essential genes, whereas the OEG dataset included 246 essential and 271 non-essential entries. The analysis showed that for the CEG dataset, Random Forest attained a ROC-AUC of 84%, PR-AUC of 41%, sensitivity of 54%, and specificity of 88%. On the other hand, XGBoost delivered a ROC-AUC of 83%, PR-AUC of 40%, sensitivity of 55%, and specificity of 86%.

When evaluated using the OEG data, Random Forest achieved impressive scores with both ROC-AUC and PR-AUC reaching 92%, supported by a sensitivity and specificity of 82% each. XGBoost also demonstrated strong performance with a ROC-AUC of 91%, PR-AUC of 90%, sensitivity of 81%, and specificity of 85%.

These collective results highlight that this area has already gained substantial attention, affirming the necessity for continued exploration. Leveraging computational techniques is seen as a powerful way to accelerate the processing and analysis of gene sequence data, enhancing accuracy in detection, classification, and modeling tasks [28], [29], [30], [31], [32], [33], [34]. In the context of this study, the emphasis lies in utilizing human gene sequence information to distinguish essential genes from non-essential ones, with a novel aim to fine-tune machine learning models that can perform reliably even under data scarcity or incompleteness. The resulting classification model holds potential for future use in studying gene functions and could also serve as a foundation for therapeutic target discovery and pharmaceutical research.

3. RESEARCH METHODOLOGY

This research focuses on the role of machine learning in creating a classification model of human

gene sequence data, with essential and non-essential gene labels. The process begins with the application of machine learning techniques, starting from analyzing the gathered sequence data, continuing through the stages of data cleaning, transformation into relevant features, and culminating in the partitioning of the dataset into subsets for model training and evaluation. Human gene data is sourced from public repository sources [12]. The stages of this research are shown in the following diagram:

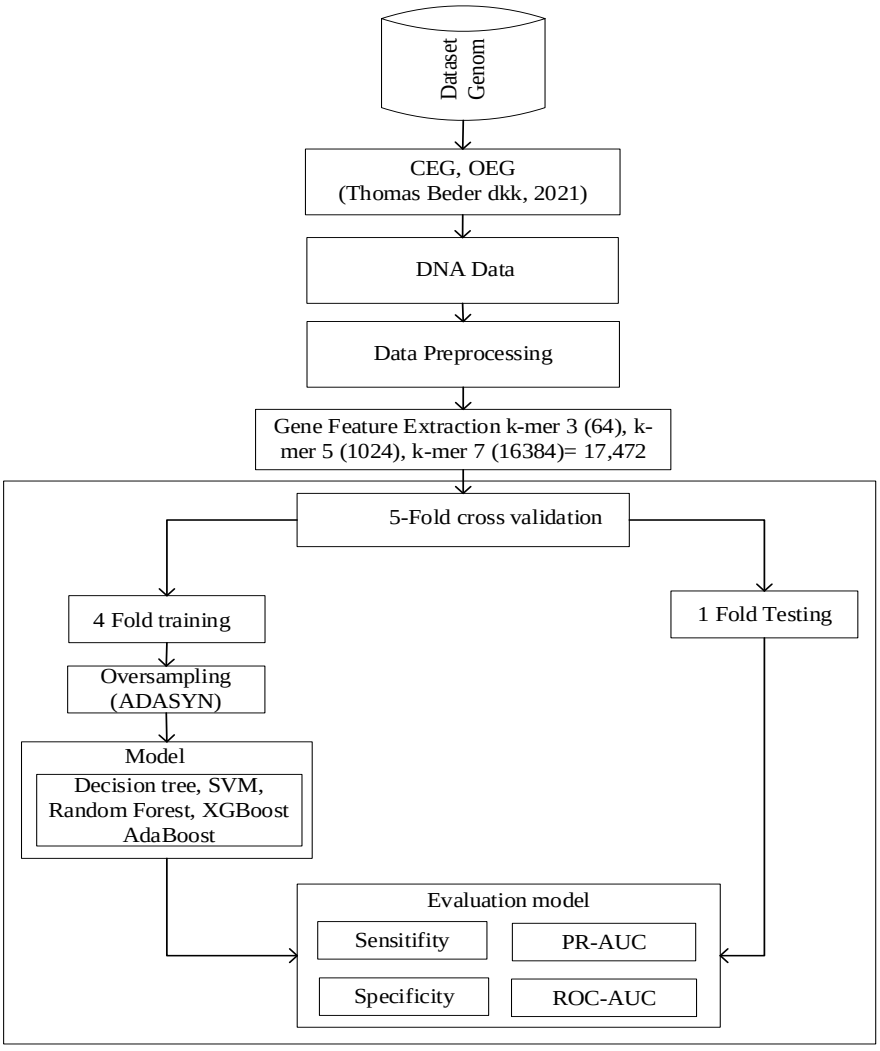

Figure 1. Methodological Framework.

3.1 Data Collection

The data used is obtained from datasets in research by[11], using fasta datasets and CSV datasets from human DNA sequences, which are human DNA sequence data with essential and non-essential gene labels. Cellular Essential Genes (CEG) represent gene sets required to sustain

fundamental cellular viability and core biological functions. These genes are typically identified through cell-based experimental assays and are known to participate in critical processes such as macromolecular biogenesis and cell-cycle progression. Owing to their indispensable role at the cellular level, CEGs are generally expected to contribute substantially to organism-level survival as well. Organismal Essential Genes (OEG) denote genes whose disruption compromises the survival or proper development of multicellular organisms. These genes are commonly associated with higher-order biological processes, including embryonic development, neural morphogenesis, and organism-level regulatory pathways. OEG identification is conducted through in vivo experiments aimed at determining gene functions essential for maintaining whole-organism viability [11].

In the OEG dataset, 15.672 genes were identified, consisting of 2.828 essential gene data and 12.844 non-essential data. OEG has the shortest sequence data totaling 60 and the longest is 34.626. For the CEG dataset, 14.579 genes were identified, consisting of 833 essential gene data and 13.743 non-essential data. The CEG dataset has the shortest sequence length of 192 and the longest of 2.304.997. An example of DNA sequence data is shown in Figure 2.

```
>ENSG00000273542
ATGTCTGGCCGCGGCAAAGGCGGGAAGGGTCTTGGCAAAGGCGGCGCTAAGCGCCACCGTAAA
GTACTGCGCGACAATATCCAGGGCATCACCAAGCCGGCCATCCGGCGCCTTGCTCGCCGCGGC
GGCGTGAAGCGCATCTCCGGCCTCATCTACGAGGAGACTCGCGGGGTGCTGAAGGTGTTCCTG
GAGAACGTGATCCGGGACGCCGTGACCTATACAGAGCACGCCAAGCGCAAGACGGTCACCGCC
ATGGATGTGGTCTACGCGCTCAAGCGCCAGGGCCGCACCCTCTACGGTTTCGGTGGTTG
```

Figure 2. ENSG00000273542 Gene Sequence Name Example in Fasta Format.

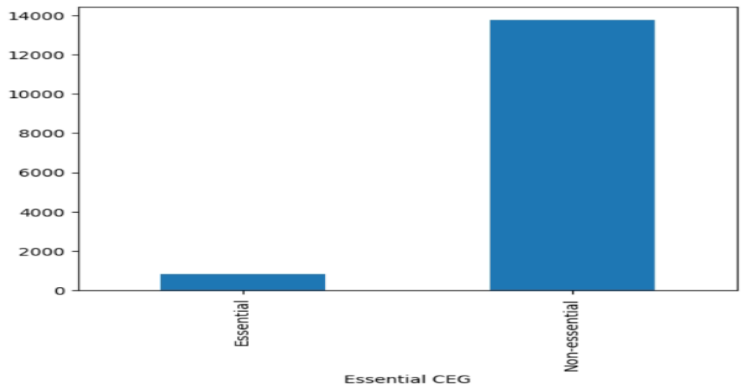

Figure 3. Distribution of Essential and Non-Essential Genes in CEG Dataset.

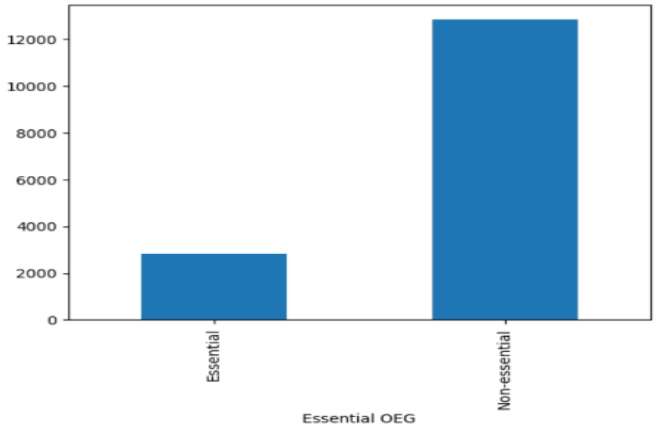

Figure 4. Class Composition of Essential and Non-Essential Genes in the OEG Dataset.

Figures 3 and 4 reveal a notable disparity in class representation within both the OEG and CEG datasets, where non-essential genes vastly outnumber essential genes, highlighting a clear imbalance in the dataset distribution. This disproportion is a prevalent issue in biological datasets, often leading to biased predictions and reduced classification model effectiveness. To mitigate this, the ADASYN (Adaptive Synthetic Sampling) technique is applied at the data level, generating synthetic samples for the minority class in an adaptive manner to enhance model learning.

3.2 Data Preprocessing

The preprocessing stage is a crucial step in preparing data before developing a machine-learning model. At this stage, the preprocessing process ensures that the data used has optimal quality and can be effectively processed by the model. The following are the preprocessing steps:

1. Data Cleaning

   At this stage, invalid data, including missing values, duplicates, and noise, is removed. Missing values are addressed using methods like mean or median, except in research where they are discarded. Duplicates are eliminated to prevent bias, and noisy data is filtered out to ensure data integrity for model training.

2. Data Normalization

   After data cleaning, normalization is performed to ensure that all features are in a uniform range. Normalization is essential to avoid the dominance of features with larger values over features with smaller values, which can cause machine learning models to be biased.

3.3 Feature Extraction

Feature extraction identifies key attributes from sequence data to predict essential and non-essential genes. This study caried out with *k-mer* frequency (k=3, 5, 7) to capture nucleotide

patterns. DNA sequences are converted into feature vectors, ensuring meaningful information for machine learning models and enhancing analysis relevance through systematic feature extraction.

3.4. Dataset Splitting

Following preprocessing, the dataset is randomly divided into training and testing subsets, maintaining the original ratio of essential to non-essential genes. The training subset is used to develop the machine learning model, while the testing subset assesses its ability to perform accurately on new, unseen data. This approach helps ensure that the model can generalize effectively beyond the training samples.

3.5 k-fold

This process divides the dataset into k parts (folds). k-fold cross-validation helps reduce bias in model evaluation as all data is used for both training and validation [35]. k-fold is performed at the training stage after splitting the data and before the final evaluation on the Testing data. The following is an explanation of the k-fold process:

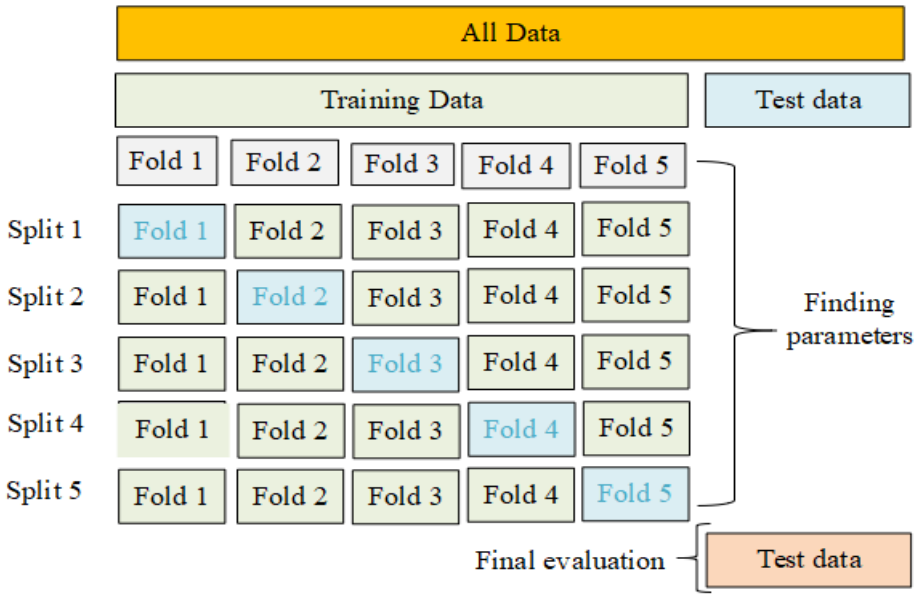

Figure 5. 5-Fold Cross Validation.

3.6 Oversampling ADASYN

Oversampling is a machine learning approach that addresses class imbalance by generating additional instances for the underrepresented class [36]. ADASYN tackles the imbalance issue by adaptively creating synthetic samples for the minority class, focusing more on those data points that are harder for the model to learn, guided by distribution-based weighting. This strategy helps minimize bias caused by uneven class distributions [37].

3.7 Model

This study utilizes five classification algorithms, namely Decision Tree, Support Vector Machine (SVM), Random Forest, XGBoost, and Adaptive Boosting (AdaBoost). A Decision Tree is easy to interpret and has the ability to handle non-linear relationships between features. SVM can effectively separate high-dimensional data through appropriate kernel functions. Random Forest improves prediction stability and accuracy by combining multiple Decision Trees. XGBoost maximizes the efficiency of the gradient boosting process and adapts to previous prediction errors. Meanwhile, AdaBoost can gradually strengthen weak models to achieve more accurate classification results. The combination of these five models enables comprehensive analysis of human genomic data, thereby improving reliability in distinguishing essential and non-essential genes.

3.8 Evaluation

Confusion matrix is a tool used to evaluate the effectiveness of classification models in machine learning. It is a table that summarizes how many predictions a classifier made correctly and incorrectly. The matrix has dimensions of N by N, where N corresponds to the total number of classes being predicted [38]. This table compares the predicted labels against the true labels of the data. Within the confusion matrix, four key metrics are identified: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

The Receiver Operating Characteristic (ROC) curve illustrates a classification model's performance over a range of threshold values. Visually, it depicts the trade-off between sensitivity (true positive rate) and specificity (true negative rate). The curve is plotted using these two measures, showing how the model's ability to distinguish between classes changes with different decision thresholds.

True Positive Rate (TPR), also known as sensitivity or recall, is calculated in Equation 1.

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

TP = *True Positive*: the count of positive cases correctly predicted by the model.

FN = *False Negative*: the count of actual positive cases that were wrongly classified as negative.

The True Positive Rate (TPR) is determined by taking the number of correctly identified positive cases and dividing it by the total number of actual positive cases, which includes both true positives and false negatives. Also referred to as sensitivity or recall, TPR measures how effectively the

model detects positive instances. It indicates the model's capability to correctly recognize positive examples.

The False Positive Rate (FPR) is calculated using the following formula:

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

FP = *False Positive*: the number of negative instances incorrectly classified as positive.

TN = *True Negative*: the number of correctly predicted negative instances.

FPR is obtained from the false positive value divided by the sum of TN and FP.

Specificity, also known as the true negative rate (TN) is given by :

$$Specificity = \frac{TN}{FP+TN} \quad (3)$$

Specificity is a measure of model goodness that is useful for measuring how well the model correctly predicts the testing data in the negative class. ROC-AUC is said to be good if the TPR value is higher and the FPR value is lower. The higher TPR means that many positive classes are correctly classified by the model. The smaller the FPR value, the smaller the error of the model predicting the negative class, and the more negative classes the model correctly predicts. The Area Under the Curve (AUC) is a metric used to evaluate the effectiveness of a classification model by measuring the area beneath the ROC curve. This value summarizes the model's overall ability to distinguish between classes. AUC scores range from 0 to 1, where values closer to 1 signify superior performance. Unlike sensitivity and specificity, which can be misleading when dealing with imbalanced classes, AUC provides a more balanced assessment of the model's accuracy.

4. MAIN RESULTS

4.1 Preprocessing

The data analysis collected was human genome data from public repositories The dataset is divided into two categories and described in Table 1.

Table 1. Preprocessing Data

Description	Gene	
	Organismal essential gene (OEG)	*Cellular essential gene* (CEG)
Number of label data before cleaning	19.914	19.914
Number of label data after cleaning	15.672	14.576
Number of sequence data before cleaning	20.727	20.727
Number of sequence data after cleaning	20.727	20.727
Merge label and sequence data	15.672	14.576

Table 1 explains the amount of label data before cleaning and after cleaning is reduced by about 0.3% because there are values that are nan or empty. The amount of sequence data before and after the cleaning process also decreased because there was duplicate data. Meanwhile, when the label data and sequences are combined, there is a decrease because there are sequences that are not labeled. Label data is stored in .csv format files while sequence data is stored in fasta format files. In the initial stage, we separated the label data for CEG and the label data for OEG. Next, cleaning is done to remove NaN-valued data labels. From the cleaning process, CEG data labels amounted to 15.672 (reduced by 20.85% from the amount before cleaning), while OEG amounted to 14.576 (reduced by 26.81% before cleaning). Sequence data is also cleaned to remove any duplicate data. The number of sequences before and after cleaning has not changed, indicating that there is no duplicate data. Then the label and sequence data were merged based on gene ID. The number of sequences before cleaning is different from the number of data after combining with labels because there are sequences that do not have labels.

4.2 k-mers Feature Extraction

k-mer mapping is a process that assigns a unique number to every k-mer, which is a sequence of k nucleotides. This mapping helps convert these nucleotide sequences into numerical values that computers can process. For instance, if k equals 7, then the sequences being mapped are exactly 7 nucleotides long. The function to convert a sequence into k-mers :

```
This function divides a DNA sequence into k-mers of length k.
Parameters:
seq: String containing the DNA sequence.
k: The length of the k-mer to be generated.
Process:
Retrieve a k-long piece of DNA sequence starting from position
i using seq[i:i+k].
Loop using range(len(seq) - k + 1) to generate all k-mers from
the beginning to the end of the sequence.
```

This function extracts all possible k-mers from a given DNA sequence. It takes two parameters: seq (the DNA sequence) and k (the desired k-mer length). The function loops from index 0 to len(seq) - k + 1, ensuring that every possible k-mer of length k is generated. In each iteration, it extracts a substring of length k using seq[i:i+k], where i is the starting position. Examples of the

results from k-mers 3, 5, and 7 is shown in Table 2.

Table 2. Example of Gene Feature Extraction Result.

K- mers Types	DNA	GENE	K-mers
3	ATTCCGCTTCCGGCATCTGGCTCAG TTCCGCCATGGCCTCCTTGGA...	Non-essential	[ATT, CCT, TCC, GTC, CGC, CCG, CTC, GCT, TGC, TTC, CTT...
5	ATTCCGCTTCCGGCATCTGGCTCAG TTCCGCCATGGCCTCCTTGGA...	Non-essential	[ATTCC, TTCCG, TCCGC, CCGCT, CGCTT, GCTTC, CTT...
7	ATTCCGCTTCCGGCATCTGGCTCAG TTCCGCCATGGCCTCCTTGGA...	Non-essential	[ATTCCTT, CCGTCCG, CCCGCTC, GCTTGCT…

4.3 Data Splitting

This method involved dividing the full dataset into training and testing portions through a 5-fold stratified cross-validation technique. The data was separated into five equally sized groups, ensuring that each group maintained the same proportion of classes as the original dataset. During each cycle, the model was trained on four groups and evaluated on the remaining one, rotating this process until every group had been used for testing.

4.4 Classification and Evaluation Results

This study employs an ensemble of machine learning algorithms to classify essential and non-essential genes within human genomic datasets, achieving robust accuracy and reliability. The classification performance was evaluated using k-mer representations of lengths 3, 5, and 7 on both Cellular Essential Genes (CEG) and Organism Essential Genes (OEG) datasets. Five machine learning methods—Decision Tree, Support Vector Machine (SVM), Random Forest, XGBoost, and AdaBoost—were systematically compared, demonstrating varied effectiveness in distinguishing gene essentiality.

For feature extraction, the k-mer approach with a length of three nucleotides (k = 3) was utilized to numerically encode gene sequences, facilitating model input. The comparative performance metrics of models based on the 3-mer encoding are detailed in Table 3.

Table 3. Evaluation Results (3-Mers)

Data	Model	Accuracy	PR AUC	ROC AUC	Sensitivity	Specificity
CEG	Decision Tree	0.83	0.26	0.60	0.35	0.86
	SVM	0.69	0.17	0.78	0.72	0.68
	Random Forest	0.91	0.19	0.79	0.29	0.95
	XGBoost	0.87	0.21	0.80	0.45	0.89
	AdaBoost	0.72	0.17	0.77	0.68	0.72
OEG	Decision Tree	0.71	0.45	0.62	0.49	0.76
	SVM	0.73	0.43	0.78	0.68	0.74
	Random Forest	0.79	0.42	0.79	0.52	0.85
	XGBoost	0.76	0.42	0.78	0.62	0.79
	AdaBoost	0.65	0.41	0.78	0.80	0.62

Table 3 represents the evaluation results of various machine learning models on two datasets, CEG and OEG, using 3-mers as features. Key evaluation metrics include Accuracy, PR AUC, ROC AUC, Sensitivity, and Specificity. For CEG dataset, the Random Forest model achieved the best Accuracy (0.91) and Specificity (0.95), indicating strong overall and negative class prediction performance, but its PR AUC (0.19) and Sensitivity (0.29) are relatively low—suggesting poor detection of the positive class. In contrast, XGBoost on CEG achieves a better balance between Sensitivity (0.45) and Specificity (0.89), making it more favorable for identifying essential genes despite slightly lower Accuracy.

In the OEG dataset, although the overall performance of models is lower than in CEG, AdaBoost achieves the highest Sensitivity (0.80), which is beneficial in detecting positive cases (essential genes), even though it comes with the lowest Accuracy (0.65) and Specificity (0.62). SVM and Random Forest show the highest PR AUC (0.43 and 0.42), indicating better precision-recall trade-offs for the imbalanced data. Random Forest also maintains decent overall performance (Accuracy = 0.79, ROC AUC = 0.79, Sensitivity = 0.52, Specificity = 0.85), making it a balanced choice for OEG. Overall, the results highlight trade-offs between sensitivity and specificity, where models like AdaBoost prioritize sensitivity, and Random Forest leans toward overall balanced accuracy and specificity.

Table 4. Evaluation Results (5-Mers)

Data	Model	Accuracy	PR AUC	ROC AUC	Sensitivity	Specificity
CEG	Decision Tree	0.82	0.23	0.58	0.30	0.85
	SVM	0.67	0.20	0.81	0.81	0.66
	Random Forest	0.94	0.21	0.81	0.10	0.99
	XGBoost	0.92	0.21	0.81	0.23	0.96
	AdaBoost	0.78	0.19	0.78	0.60	0.79
OEG	Decision Tree	0.70	0.43	0.61	0.47	0.75
	SVM	0.74	0.46	0.80	0.69	0.76
	Random Forest	0.80	0.44	0.80	0.49	0.87
	XGBoost	0.79	0.44	0.79	0.52	0.85
	AdaBoost	0,.68	0.39	0.78	0.76	0.66

The results in the table highlight how various models perform on CEG and OEG datasets using 5-mer features. In the CEG dataset, Random Forest leads in terms of accuracy (0.94) and specificity (0.99), but this comes at the cost of an extremely low sensitivity (0.10), indicating that it fails to effectively identify essential genes. Similarly, XGBoost follows closely in accuracy (0.92) and specificity (0.96), but again suffers from low sensitivity (0.23). AdaBoost, though not as high in accuracy (0.78), offers a better balance with sensitivity at 0.60, showing more promise for recognizing essential genes. While Decision Tree and SVM achieve decent ROC AUC scores, their lower PR AUC and sensitivity suggest limited utility in applications requiring precise positive class identification.

On the OEG side, model performances are more balanced. SVM stands out with the highest PR AUC (0.46) and sensitivity (0.69), demonstrating its effectiveness in handling imbalanced data. Random Forest and XGBoost both show solid accuracy (0.80 and 0.79 respectively) and specificity (0.87 and 0.85), making them reliable for overall classification, though their sensitivities remain moderate. AdaBoost, meanwhile, offers the best sensitivity at 0.76 but has lower accuracy (0.68) and specificity (0.66), indicating its focus on capturing more true positives even if it misclassifies more negatives. These patterns suggest that while some models prioritize overall accuracy and negative class prediction, others—like SVM and AdaBoost—offer more favorable results when identifying essential genes is the main objective.

Table 5. Evaluation Results (7-Mers)

Data	Model	Accuracy	PR AUC	ROC AUC	Sensitivity	Specificity
CEG	Decision Tree	0.82	0.19	0.55	0.25	0,85
	SVM	0.78	0.10	0.69	0.36	0,81
	Random Forest	0.94	0.12	0.72	0.01	0,99
	XGBoost	0.94	0.14	0.74	0.06	0,99
	AdaBoost	0.84	0.10	0.66	0.27	0,88
OEG	Decision Tree	0.68	0.40	0.59	0.44	0.74
	SVM	0.80	0.45	0.77	0.50	0.87
	Random Forest	0.80	0.41	0.77	0.41	0.89
	XGBoost	0.81	0.42	0.75	0.34	0.92
	AdaBoost	0.74	0.32	0.67	0.44	0.81

5. CONCLUSION

Based on the evaluation results across all k-mer representations, it can be concluded that the 5-mers configuration combined with the Support Vector Machine (SVM) model yields the most balanced and reliable performance for both CEG and OEG datasets. Although it does not always achieve the highest accuracy, SVM with 5-mers consistently demonstrates superior sensitivity and precision-recall performance, which are critical for effectively identifying essential genes. For the CEG dataset, this combination achieved a high sensitivity of 0.81 and ROC AUC of 0.81, indicating a strong ability to differentiate essential genes from non-essential ones. Similarly, in the OEG dataset, SVM with 5-mers recorded the highest PR AUC (0.46), along with competitive sensitivity (0.69) and ROC AUC (0.80), highlighting its robustness in handling class imbalance.

These findings suggest that SVM with 5-mers is particularly well-suited for biological datasets where detecting the minority class (essential genes) is more important than overall classification accuracy. High sensitivity ensures fewer false negatives, which is crucial in genomics applications to avoid missing potentially vital genes. Additionally, the strong PR AUC values reinforce the model's ability to maintain reliable precision even in imbalanced settings. Overall, this method provides a more biologically meaningful classification outcome, making it a favorable choice for gene essentiality prediction tasks.

ACKNOWLEDGMENT

This work was supported by the Directorate of Research, Technology and Community Service,

Ministry of Education, Culture, Research and Technology, Republic of Indonesia, through the Doctoral Desertion Research Program (PDD), Grant No. 057/E5/PG.02.00.PL/2024.

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interests.

REFERENCES

[1] B.A. Hamed, O.A.S. Ibrahim, T. Abd El-Hafeez, Optimizing Classification Efficiency with Machine Learning Techniques for Pattern Matching, J. Big Data 10 (2023), 124. https://doi.org/10.1186/s40537-023-00804-6.

[2] S.M. Abd–Alhalem, E.M. El-Rabaie, N.F. Soliman, S.E.S.E. Abdulrahman, N.A. Ismail, et al., DNA Sequences Classification with Deep Learning: A Survey, Menoufia J. Electron. Eng. Res. 30 (2021), 41-51. https://doi.org/10.21608/mjeer.2021.146090.

[3] K. Nandhini, G. Tamilpavai, An Optimal Stacked ResNet-BiLSTM-Based Accurate Detection and Classification of Genetic Disorders, Neural Process. Lett. 55 (2023), 9117-9138. https://doi.org/10.1007/s11063-023-11195-3.

[4] A. El-Tohamy, H.A. Maghwary, N. Badr, A Deep Learning Approach for Viral DNA Sequence Classification Using Genetic Algorithm, Int. J. Adv. Comput. Sci. Appl. 13 (2022), 530-538. https://doi.org/10.14569/ijacsa.2022.0130861.

[5] C. Lu, C. Liu, X. Sun, P. Wan, J. Ni, et al., Bioinformatics Analysis, Codon Optimization and Expression of Ovine Pregnancy Associated Glycoprotein-7 in HEK293 Cells, Theriogenology 172 (2021), 27-35. https://doi.org/10.1016/j.theriogenology.2021.05.027.

[6] Y. Sun, Z. Zhu, Z. You, Z. Zeng, Z. Huang, et al., FMSM: A Novel Computational Model for Predicting Potential MiRNA Biomarkers for Various Human Diseases, BMC Syst. Biol. 12 (2018), 121. https://doi.org/10.1186/s12918-018-0664-9.

[7] G. Vasiliev, I. Chadaeva, D. Rasskazov, P. Ponomarenko, E. Sharypova, et al., A Bioinformatics Model of Human Diseases on the Basis of Differentially Expressed Genes (of Domestic Versus Wild Animals) That Are Orthologs of Human Genes Associated with Reproductive-Potential Changes, Int. J. Mol. Sci. 22 (2021), 2346. https://doi.org/10.3390/ijms22052346.

[8] Y. Guo, Y. Ju, D. Chen, L. Wang, Research on the Computational Prediction of Essential Genes, Front. Cell Dev. Biol. 9 (2021), 803608. https://doi.org/10.3389/fcell.2021.803608.

[9] R.K. Rout, S. Umer, M. Khandelwal, S. Pati, S. Mallik, et al., Identification of Discriminant Features from Stationary Pattern of Nucleotide Bases and Their Application to Essential Gene Classification, Front. Genet. 14 (2023), 1154120. https://doi.org/10.3389/fgene.2023.1154120.

[10] W. Hu, M. Li, H. Xiao, L. Guan, Essential Genes Identification Model Based on Sequence Feature Map and Graph Convolutional Neural Network, BMC Genom. 25 (2024), 47. https://doi.org/10.1186/s12864-024-09958-w.

[11] T. Beder, O. Aromolaran, J. Dönitz, S. Tapanelli, E.O. Adedeji, et al., Identifying Essential Genes Across

Eukaryotes by Machine Learning, NAR Genom. Bioinform. 3 (2021), lqab110. https://doi.org/10.1093/nargab/lqab110.

[12] O. Aromolaran, T. Beder, M. Oswald, J. Oyelade, E. Adebiyi, et al., Essential Gene Prediction in Drosophila Melanogaster Using Machine Learning Approaches Based on Sequence and Functional Features, Comput. Struct. Biotechnol. J. 18 (2020), 612-621. https://doi.org/10.1016/j.csbj.2020.02.022.

[13] T.T.H. Yen, D.T. Linh, P.T. Minh Hue, The Application of Microfluidics in Preparing Nano Drug Delivery Systems, VNU J. Sci.: Med. Pharm. Sci. 35 (2019), 1-10. https://doi.org/10.25073/2588-1132/vnumps.4150.

[14] S. Nandi, P. Ganguli, R.R. Sarkar, Essential Gene Prediction Using Limited Gene Essentiality Information–An Integrative Semi-Supervised Machine Learning Strategy, PLOS ONE 15 (2020), e0242943. https://doi.org/10.1371/journal.pone.0242943.

[15] O. Aromolaran, D. Aromolaran, I. Isewon, J. Oyelade, Corrigendum to: Machine Learning Approach to Gene Essentiality Prediction: A Review, Briefings Bioinform. 23 (2021), bbab419. https://doi.org/10.1093/bib/bbab419.

[16] O. Aromolaran, D. Aromolaran, I. Isewon, J. Oyelade, Machine Learning Approach to Gene Essentiality Prediction: A Review, Briefings Bioinform. 22 (2021), bbab128. https://doi.org/10.1093/bib/bbab128.

[17] S. Nurk, S. Koren, A. Rhie, M. Rautiainen, A.V. Bzikadze, et al., The Complete Sequence of a Human Genome, Science 376 (2022), 44-53. https://doi.org/10.1126/science.abj6987.

[18] S. Medina, R. Dominguez-Perles, J. Gil, F. Ferreres, A. Gil-Izquierdo, Metabolomics and the Diagnosis of Human Diseases -A Guide to the Markers and Pathophysiological Pathways Affected, Curr. Med. Chem. 21 (2014), 823-848. https://doi.org/10.2174/09298673206613119124056.

[19] M.Q. Pham, K.B. Vu, T.N. Han Pham, L.T. Thuy Huong, L.H. Tran, et al., Correction: Rapid Prediction of Possible Inhibitors for SARS-Cov-2 Main Protease Using Docking and FPL Simulations, RSC Adv. 12 (2022), 35778-35778. https://doi.org/10.1039/d2ra90114e.

[20] K. Pal, B.V. Patel, Data Classification with K-Fold Cross Validation and Holdout Accuracy Estimation Methods with 5 Different Machine Learning Techniques, in: 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), IEEE, 2020, pp. 83-87. https://doi.org/10.1109/ICCMC48092.2020.ICCMC-00016.

[21] A. Saavedra, H. Lehnert, C. Hernandez, G. Carvajal, M. Figueroa, Mining Discriminative K-MERS in DNA Sequences Using Sketches and Hardware Acceleration, IEEE Access 8 (2020), 114715-114732. https://doi.org/10.1109/access.2020.3003918.

[22] J.S. Samagh, D. Singh, A Machine Learning Model for Predicting Heart Disease Using Ensemble Methods, Int. J. Adv. Comput. Sci. Appl. 13 (2022), 558-565. https://doi.org/10.14569/ijacsa.2022.0130965.

[23] T.L. Campos, P.K. Korhonen, R.B. Gasser, N.D. Young, An Evaluation of Machine Learning Approaches for the Prediction of Essential Genes in Eukaryotes Using Protein Sequence-Derived Features, Comput. Struct. Biotechnol. J. 17 (2019), 785-796. https://doi.org/10.1016/j.csbj.2019.05.008.

[24] S. Levy, G. Sutton, P.C. Ng, L. Feuk, A.L. Halpern, et al., The Diploid Genome Sequence of an Individual Human, PLoS Biol. 5 (2007), e254. https://doi.org/10.1371/journal.pbio.0050254.

[25] B.D. Heavner, N.D. Price, Comparative Analysis of Yeast Metabolic Network Models Highlights Progress, Opportunities for Metabolic Reconstruction, PLOS Comput. Biol. 11 (2015), e1004530. https://doi.org/10.1371/journal.pcbi.1004530.

[26] G. Liu, M.Y.J. Yong, M. Yurieva, K.G. Srinivasan, J. Liu, et al., Gene Essentiality Is a Quantitative Property Linked to Cellular Evolvability, Cell 163 (2015), 1388-1399. https://doi.org/10.1016/j.cell.2015.10.069.

[27] Z. Zhang, Q. Ren, Why Are Essential Genes Essential? - the Essentiality of Saccharomyces Genes, Microb. Cell 2 (2015), 280-287. https://doi.org/10.15698/mic2015.08.218.

[28] A.A. Parikesit, Kontribusi Aplikasi Medis dari Ilmu Bioinformatika Berdasarkan Perkembangan Pembelajaran Mesin (Machine Learning) Terbaru, Cermin Dunia Kedokteran 45 (2018), 700-703.

[29] N.Q.K. Le, D.T. Do, T.N.K. Hung, L.H.T. Lam, T. Huynh, et al., A Computational Framework Based on Ensemble Deep Neural Networks for Essential Genes Identification, Int. J. Mol. Sci. 21 (2020), 9070. https://doi.org/10.3390/ijms21239070.

[30] Z. Abdellah, A. Ahmadi, S. Ahmed, et al. Finishing the Euchromatic Sequence of the Human Genome, Nature 431 (2004), 931-945. https://doi.org/10.1038/nature03001.

[31] P. Chen, A.H. Michel, J. Zhang, Transposon Insertional Mutagenesis of Diverse Yeast Strains Suggests Coordinated Gene Essentiality Polymorphisms, Nat. Commun. 13 (2022), 1490. https://doi.org/10.1038/s41467-022-29228-1.

[32] K. Plaimas, R. Eils, R. König, Identifying Essential Genes in Bacterial Metabolic Networks with Machine Learning Methods, BMC Syst. Biol. 4 (2010), 56. https://doi.org/10.1186/1752-0509-4-56.

[33] F. Nainu, Review: Penggunaan Drosophila Melanogaster Sebagai Organisme Model Dalam Penemuan Obat, J. Farm. Galen. (Galen. J. Pharm.) 4 (2018), 50-67. https://doi.org/10.22487/j24428744.2018.v4.i1.9969.

[34] Silvia, Mengenal Google Colab, Sebagai Aplikasi Penunjang Machine Learning dan Artificial Intelligence, https://www.jetorbit.com/blog/mengenal-google-colab-sebagai-aplikasi-penunjang-machine-learning-dan-artificial-intelligence.

[35] H. Azis, P. Purnawansyah, F. Fattah, I.P. Putri, Performa Klasifikasi K-Nn Dan Cross Validation Pada Data Pasien Pengidap Penyakit Jantung, LKOM J. Ilmiah 12 (2020), 81-86. https://doi.org/10.33096/ilkom.v12i2.507.81-86.

[36] H. Shi, C. Wu, T. Bai, J. Chen, Y. Li, et al., Identify Essential Genes Based on Clustering Based Synthetic Minority Oversampling Technique, Comput. Biol. Med. 153 (2023), 106523. https://doi.org/10.1016/j.compbiomed.2022.106523.

[37] H. Marlisa, N. Satyahadewi, N. Imro'ah, N.N. Debataraja, Application of Adasyn Oversampling Technique on K-Nearest Neighbor Algorithm, BAREKENG: J. Ilmu Mat. Ter. 18 (2024), 1829-1838. https://doi.org/10.30598/barekengvol18iss3pp1829-1838.

[38] Z. Karimi, Confusion Matrix, 2021. https://www.researchgate.net/publication/355096788.