2024 International Conference on Advances in Computing, Communication, Electrical, and Smart Systems (iCACCESS), 8-9 March, Dhaka, Bangladesh

2024 International Conference on Advances in Computing, Communication, Electrical, and Smart Systems (iCACCESS) | 979-8-3503-5028-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICACCESS61735.2024.10499483

Cracking the Genetic Codes: Exploring DNA Sequence Classification with Machine Learning Algorithms and Voting Ensemble Strategies

Arifur Rahman
Department of Computer Science and Engineering.
Khulna University of Engineering & Technology.
Khulna, Bangladesh.
rarifkhan652@gmail.com

Sakib Zaman
Department of Computer Science and Engineering.
Khulna University of Engineering & Technology.
Khulna, Bangladesh.
sakibzaman169@gmail.com

Dola Das
Department of Computer Science and Engineering.
Khulna University of Engineering & Technology.
Khulna, Bangladesh.
dola.das@cse.kuet.ac.bd

***Abstract*—In the domain of bioinformatics, DNA sequence classification is an indispensable tool that spans various scientific disciplines, contributing to scientists' understanding of biology, aiding in the identification of genes, regulatory elements, and the functional significance of distinct genomic regions. Moreover, it plays a vital role in disease diagnosis, treatment strategies, drug discovery, evolution, agriculture, forensic identification, environmental monitoring and more. The classification process involves the intricate mapping of DNA sequences to distinct classes based on the arrangement of nucleotides. A fractional mutation in the sequence corresponds to a nuanced shift in the assigned class. Every numerical instance, serving as a depiction of a particular class, is closely associated with a specific gene lineage. In this study, for the DNA sequence preprocessing, both K-mer counting and count vectorization were used respectively. Afterwards, we utilized a variety of classifier models, encompassing Multinomial naive bayes (MNB) , Logistic regression (LR), Random forest (RF), LightGBM (LGMB), XGBoost (XGB), K-nearest neighbors (KNN) and Decision tree (DT) algorithm on three types of DNA sequence datasets (Human, Chimpanzee & Dog) to identify each of sequence's corresponding gene class (0, 1, 2, 3, 4, 5, & 6). Then, the highest three and highest five classifier models were picked based on their accuracy scores. Afterwards, both soft voting and hard voting ensemble methods were implemented on this cluster of fundamental models to effectively leverage their collective predictive strength. The soft voting ensemble on the best three models consistently reached the highest accuracy across all three datasets. Employing this ensemble method, the human, chimpanzee, and dog datasets exhibited highest performance metrics i.e. accuracy, precision, recall, and f1-scores of (98.42%, 98.41%, 98.40%, 98.40%), (92.28%, 92.40%, 92.30%, 92.10%), and (70.12%, 73.10%, 70.10%, 69.20%) respectively.**

***Index Terms*—Bioinformatics, DNA sequence classification, K-mer counting, CountVectorizer, BoW (Bag of Words), classifier models, soft voting ensemble, hard voting ensemble.**

I. Introduction

DNA sequence classification is a cornerstone in genomics, playing a pivotal role in advancing our understanding of life processes, genetics, comparative genomics, agricultural applications, in identification of genetic variations associated with diseases, facilitating drug target identification etc. The double-helix structure precisely represents the chemical structure of DNA. The arrangement comprises two spiraled nucleotide chains, connected by hydrogen bonds, and navigating in different orientations [1]. Comprising four nitrogen bases—Adenine (A), Thymine (T), Guanine (G), and Cytosine (C)—DNA forms nucleotides, linking together via hydrogen bonds in various orders [2], [3], [4]. The two threads of the double helix balance each other, following a simple rule: if one thread has A, the other must have T, and similarly, C always pairs with G [5]. DNA sequencing is the process of determining the order of nucleotides in DNA, revealing the sequence of nucleic acid bases through various identification techniques. Gene prediction methods in machine learning can be grouped into two techniques, one of them is similarity-based approach and another one is content-based approach [6]. These methodologies leverage several sequence attributes, encompassing GC content, sequence length & codon usage. Academic researchers were pioneers in tracing the DNA sequence in the early 1970s. Afterwards, the implementation of fluorescence-based sequencing methods took place, utilizing a DNA sequencer [7].

II. Literature Review

Using feature descriptors from different physiochemical properties and six classifiers, the authors [8] created a stacked ensemble model to identify enhancers. The model outperformed previous methods in accuracy, specificity, sensitivity, and correlation coefficient. The researchers [9]used machine learning to classify DNA sequences using label and k-mer encoding, distinguishing infected and normal genes. Juneja et al. [10] used a classification algorithm to classify three datasets by gene class, where they split the sequences into defined-length substrings for analysis. Mathur et al. [11] proposed a hot vector matrix and machine learning-based DNA sequence feature extraction classifier that represents word pairs as a binary matrix of nucleotide positions.

The study [12] elucidates DNA sequences to distinguish between regular and disease-affected genes using ML techniques, particularly AdaBoost and Random forest classifier for bagging and detection, respectively. Furthermore, an identification cascade structure reduced false-positive results and enhanced reliability. In the paper [13], the authors predicted gene

families using human, chimpanzee, and dog DNA sequences using SVM and classification. Combining machine learning techniques with a pattern-matching algorithm, the study [14] suggests a model incorporating SVM Linear, and Naive Bayes to execute DNA sequence classification. Vedanshee et al. [15] predicted genetic defects in 22083 patient records using human, chimpanzee, and dog DNA. They tagged, correlated, and analyzed using five classifier models like SVC Classifier, Gradient Boosting, Cat-Boost etc.

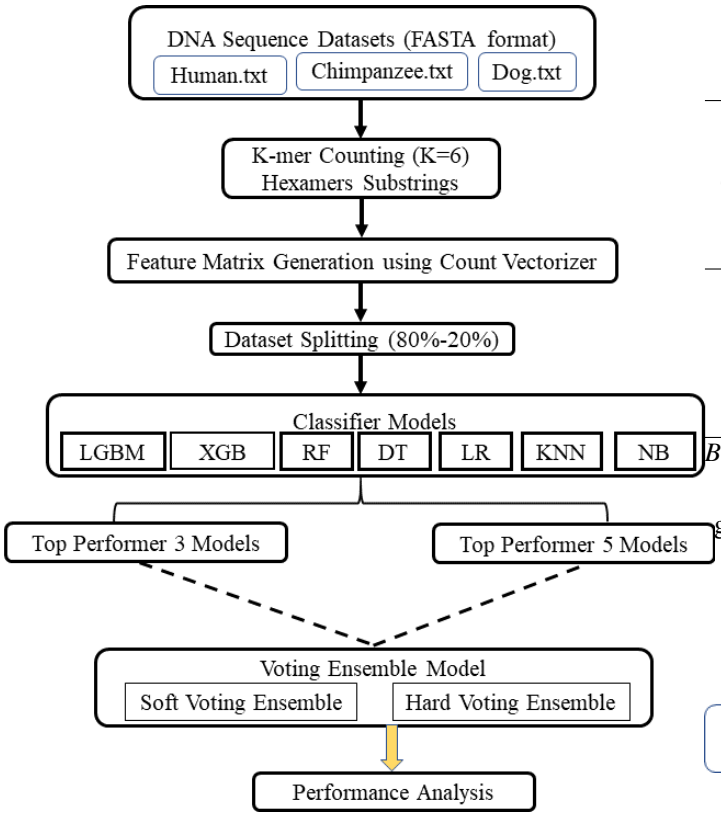


Fig. 1: Overview of the suggested ensemble approaches on the three datasets.

III. Methodology

A. Dataset Insight

In our research, we procured the comprehensive datasets of gene sequences sourced from the publicly available DNA sequence repository on Kaggle. These datasets are available for download through the following link: *https://www.kaggle.com/code/khalidmostafa/dna-sequence-classification-using-machine-learning/input*. There are three types of datasets are present, including Human Dataset, Chimpanzee dataset, & Dog Dataset (Fig. 1). These datasets are present in FASTA format. In the realm of bioinformatics, the FASTA format emerges as a pivotal text-based encoding method employed for the representation of nucleotide or amino acid sequences. This format stands as a standardized approach for conveying biological information, where nucleotides or amino acids are denoted by succinct single-letter codes, such as [A, C, G, T, N]. In this intricate encoding system, each letter signifies a distinct biological entity: A for Adenosine, C for Cytosine, G for Guanine, T for Thymidine, and N serving as a wildcard for any of the aforementioned entities. Tab. I depicts the overview of the three datasets.

TABLE I: **Frequency count of each gene class for the three datasets.**

Dataset name	Dataset size	Training set size	Testing set size	Class label	Count
Human	4380	3504 (80%)	876 (20%)	0	531
				1	534
				2	349
				3	672
				4	711
				5	240
				6	1343
Chimpanzee	1682	1346 (80%)	336 (20%)	0	234
				1	185
				2	144
				3	228
				4	261
				5	109
				6	521
Dog	820	656 (80%)	164 (20%)	0	131
				1	75
				2	64
				3	95
				4	135
				5	60
				6	260

B. Feature Matrix Generation

Both K-mer counting and CountVectorizer was utilized to generate the feature matrix.

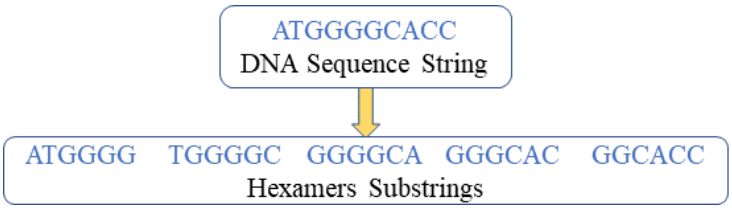


Fig. 2: Hexamer substrings (K=6).

1) K-mer Counting: K-Mer counting converted the DNA sequence strings into k-mer words with a **K** value of 6, known as hexamers (K=6). Fig. 2 depicts the generated hexamer substrings for the DNA sequence **ATGGGGCACC**. The conversion of DNA sequences into k-mers serves the purpose of breaking down the genetic information into smaller, overlapping units. These k-mers serve as the elemental vocabulary in deciphering the genetic language encoded in DNA. The k-mers act as the primary features.

2) CountVectorizer: The CountVectorizer was applied to establish a BoW (Bag of words) model, concentrating on the counts of 4-grams (tetragrams). The resulting string sentence formulated by K-mer counting served as input for the count vectorizer, allowing for the creation of a comprehensive bag-of-words model that encapsulated the unique genetic features encoded in the original DNA sequences. By utilizing the BoW approach, the count vectorizer constructed a sparse matrix where each entry represents the count of a specific 4-gram in a given genetic sequence. In Tab. II we showed a portion of the generated sparse matrix.

TABLE II: **A portion of sparse matrix generated by Count Vectorization technique.**

(index, column)	value
(0, 181326)	1
(0, 178989)	1
(0, 55066)	1
(0, 217067)	1
(0, 171189)	1
(0, 216740)	1
(0, 169929)	1
(0, 211678)	1
(0, 151026)	1
(0, 135341)	1
(0, 74165)	1
(0, 58623)	1
(0, 231147)	1
(0, 227458)	1

The sparse matrix representation of a genetic sequence, demonstrated the utilization of the CountVectorizer to convert raw genetic text data into a structured and numerical format suitable for subsequent analysis and machine learning tasks.

C. Dataset splitting

After the sparse matrix generation, the dataset was segmented into two parts, one part was for training purpose with 80% of the data, while another one part was for testing purpose with 20% of the data.

D. Classifier models

In this study, various classifier model was employed to train the model including Multinomial naive bayes (MNB) , Logistic regression (LR), Random forest (RF), LightGBM (LGMB), XGBoost (XGB), K-nearest neighbors (KNN) and Decision tree (DT). To obtain the highest accuracy by K-nearest neighbors model, a loop was implemented to iteratively evaluate the performance of the K-Nearest Neighbors (KNN) classifier with varying values of number of neighbors, ranging from 1 to 199. Fig. 3 depicts the generated plots to visualize how the accuracy of the KNN classifier changes with different values of K for the human, chimpanzee & Dog dataset. In Tab. III we also represented the best K value with their corresponding accuracies for the three datasets.

TABLE III: **Best K value with corresponding accuracies for the three datasets**

Dataset name	Best K Value	Accuracy
Human	K=1	85.84%
Chimpanzee	K=1	84.87%
Dog	K=3	51.22%

E. Ensemble model

1) Soft voting ensemble: The soft voting ensemble model is a sophisticated technique in machine learning that amalgamates the predictions of multiple base models by considering their weighted average probabilities, resulting in a consensual decision. Mathematically, let $P_{i,j}$ denote the predicted probability of the i-th sample belonging to the j-th class according to the i-th base model. The soft voting ensemble prediction $P_{\text{ensemble},j}$ for the j-th class is computed as follows:

$$P_{\text{ensemble},j} = \frac{\sum_{i=1}^{N} P_{i,j}}{N}$$

After picking the best three and best five models, we utilized soft voting ensemble on these sets of models.

2) Hard voting ensemble: Hard voting is an ensemble technique in machine learning that combines the predictions of multiple base models by selecting the class label that receives the majority of votes. Mathematically, let M represent the number of base models in the ensemble, and C denote the number of classes in the classification task. For each input sample i, the hard voting ensemble prediction $E_{\text{hard},j}$ for the j-th class is determined as follows:

$$E_{\text{hard},j} = \text{argmax}_c \sum_{m=1}^{M} I(y_{m,i} = c)$$

After choosing the best three and best five models, we employed hard voting ensemble on these sets of models.

IV. Experimental Results

In the analysis of Human.txt, Chimpanzee.txt, and Dog.txt datasets, it was evident that the soft voting ensemble, incorporating with the top three classifier models consistently provided the highest accuracy, as shown in Fig. 4, Fig. 5, Fig. 6, and Tab. IV

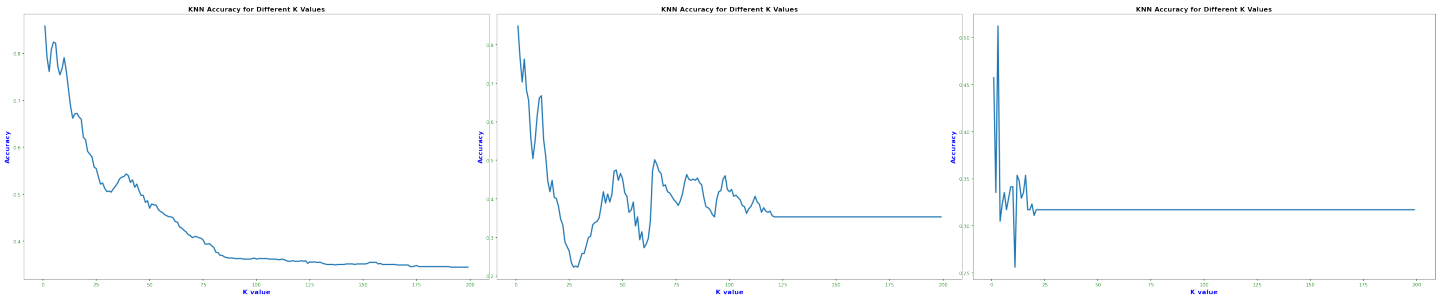
Human dataset **–** When considering the human dataset, Multinomial Naive Bayes, Logistic Regression, Random Forest, LightGBM, XGBoost, K-Nearest Neighbors, and Decision Tree recorded accuracy percentages of 98.40%, 93.95%, 92.24%, 91.21%, 89.84%, 85.84%, and 81.15%, respectively. Among classifier models, Multinomial Naive Bayes took the lead, achieving the highest levels of accuracy, precision, recall, and f1-score, all at an impressive 98.40%. Logistic Regression earned the second-highest accuracy, showcasing impressive precision, recall, and f1-score at 94.80%, 93.90%, and 94.00%. The soft voting ensemble on the top three models (MNB+LR+RF) showcased the highest accuracy among all proposed models for the human dataset, with significant values for accuracy, precision, recall, and f1-score of 98.42%, 98.41%, 98.40%, and 98.40%, respectively. In contrast, the soft voting ensemble incorporating the top five models (MNB+LR+RF+LGBM+XGB) attained the second-best rank among all ensemble models, presenting an impressive accuracy of 96.23%. Besides, this model delivered an outstanding precision, recall and f1-score of 96.50%, 96.20%, and 96.20% respectively.

Chimpanzee dataset **–** For the Chimpanzee dataset, the performance of various models was evaluated, with Multinomial Naive Bayes, Logistic Regression, LightGBM, XGBoost, K-Nearest Neighbors, Random Forest, and Decision Tree achieving accuracy scores of 91.39%, 89.91%, 88.13%, 86.05%, 84.87%, 84.27%, and 79.23% respectively. Among these, Multinomial Naive Bayes exhibited the highest accuracy, precision, recall, and f1-score at 91.39%, 91.80%, 91.40%, and 91.40%. The soft voting ensemble on the top three models (MNB+LR+LGBM) stood out as the best-performing model for the Chimpanzee dataset, showcasing significant accuracy, precision, recall, and f1-score values of 92.28%, 92.40%, 92.30%, and 92.10% respectively. This ensemble demonstrated superior classification capabilities, leveraging the strengths of the individual models.

Dog dataset **–** Finally for the Dog dataset, a comprehensive

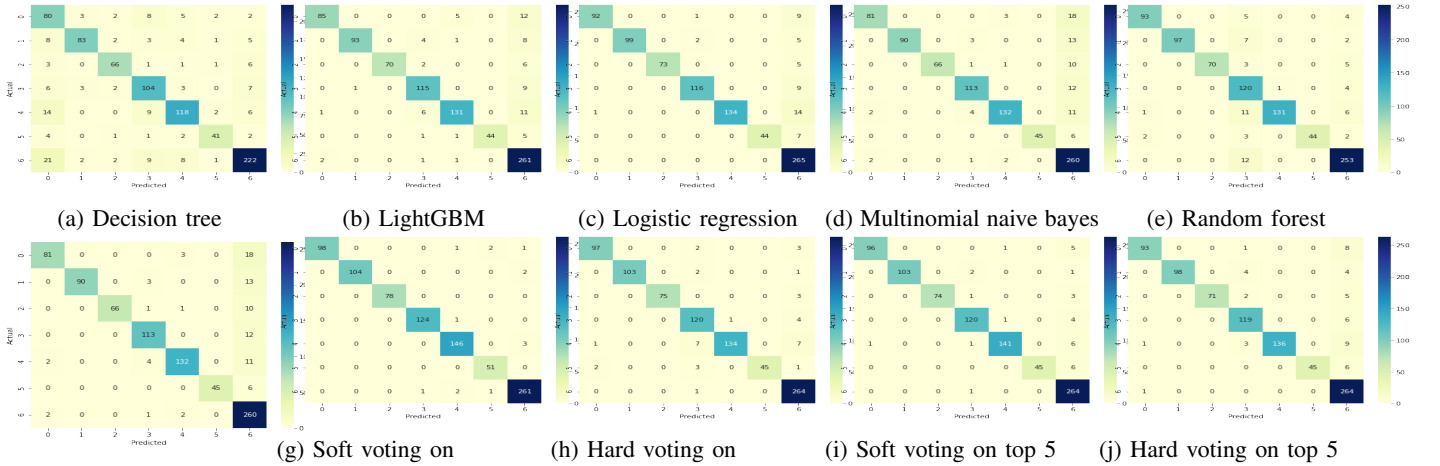
TABLE IV: Performance evaluation of all recommended models across the Human, Chimpanzee, and Dog datasets.

Dataset type	Model name	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	MAE (%)	MSE (%)	RMSE (%)	RAE (%)	RRSE (%)	Best performer model
Human dataset	MNB	98.40%	98.40%	98.40%	98.40%	5.10%	20.90%	45.70%	1.50%	13.00%	Soft voting on top 3 models (MNB+LR+RF)
	LR	93.95%	94.80%	93.90%	94.00%	19.60%	80.60%	89.80%	5.60%	25.50%	
	RF	92.24%	93.40%	92.20%	92.40%	20.50%	69.60%	83.40%	5.80%	23.70%	
	LGBM	91.21%	92.00%	91.20%	91.20%	29.10%	12.34%	11.11%	8.30%	31.60%	
	XGB	89.84%	91.20%	89.80%	90.00%	37.60%	169.1%	130.0%	10.70%	37.00%	
	KNN	85.84%	92.60%	85.80%	87.40%	32.90%	87.20%	93.40%	9.40%	26.60%	
	DT	81.15%	82.80%	81.50%	81.90%	59.80%	242.9%	155.9%	17.00%	44.40%	
	Soft voting on top 3 models (MNB+LR+RF)	**98.42%**	98.41%	98.40%	98.40%	7.30%	28.30%	53.20%	2.10%	15.10%	
	Hard voting on top 3 models (MNB+LR+RF)	95.67%	95.90%	95.70%	95.70%	14.70%	60.80%	78.00%	4.20%	22.20%	
	Soft voting on top 5 models (MNB+LR+RF +LGBM+XGB)	96.23%	96.50%	96.20%	96.20%	11.00%	36.10%	60.10%	3.10%	17.10%	
	Hard voting on top 5 models (MNB+LR+RF +LGBM+XGB)	94.29%	94.80%	98.40%	98.40%	16.40%	65.30%	80.80%	4.70%	23.00%	
Chimpanzee dataset	MNB	91.39%	91.80%	91.40%	91.40%	21.40%	72.40%	85.10%	5.60%	22.20%	Soft voting on top 3 models (MNB+LR+LGBM)
	LR	89.91%	91.10%	89.90%	89.70%	30.00%	119.6%	109.4%	7.80%	28.60%	
	RF	84.27%	87.30%	84.30%	84.10%	54.60%	238.0%	154.3%	14.30%	40.30%	
	LGBM	88.13%	89.10%	88.10%	87.90%	34.10%	127.9%	113.1%	8.90%	29.60%	
	XGB	86.05%	86.90%	86.10%	85.80%	38.60%	147.2%	121.3%	10.10%	31.70%	
	KNN	84.87%	89.40%	84.90%	85.00%	47.20%	197.9%	140.7%	12.30%	36.80%	
	DT	79.23%	79.70%	79.20%	79.20%	54.90%	189.0%	137.5%	14.40%	35.90%	
	Soft voting on top 3 models (MNB+LR+LGBM)	**92.28%**	92.40%	92.30%	92.10%	20.20%	74.80%	86.50%	5.30%	22.60%	
	Hard voting on top 3 models (MNB+LR+LGBM)	91.10%	91.80%	91.10%	90.80%	26.70%	104.5%	102.2%	7.00%	26.70%	
	Soft voting on top 5 models (MNB+LR+LGBM +XGBC+KNN)	89.91%	91.60%	89.90%	89.80%	28.20%	110.7%	105.2%	7.40%	27.50%	
	Hard voting on top 5 models (MNB+LR+LGBM +XGBC+KNN)	89.90%	91.20%	89.90%	89.70%	27.00%	100.0%	100.0%	7.10%	26.10%	
Dog dataset	MNB	70.10%	73.20%	70.10%	69.40%	101.2%	430.5%	207.5%	29.40%	60.30%	Soft voting on top 3 models (MNB+LGBM+RF)
	LR	59.77%	71.20%	59.80%	57.60%	145.7%	650.6%	255.1%	42.40%	74.20%	
	RF	56.71%	64.40%	56.70%	53.40%	162.8%	756.7%	275.1%	47.30%	80.00%	
	LGBM	64.63%	68.10%	64.60%	63.40%	108.5%	420.7%	205.1%	31.60%	59.60%	
	XGB	59.76%	63.50%	59.80%	58.90%	131.1%	538.4%	232.0%	38.10%	67.50%	
	KNN	51.22%	67.50%	51.20%	45.50%	179.9%	839.6%	289.8%	52.30%	84.30%	
	DT	53.66%	53.30%	53.70%	52.50%	142.7%	545.1%	233.5%	41.50%	67.90%	
	Soft voting on top 3 models (MNB+LGBM+LR)	**70.12%**	73.10%	70.10%	69.20%	100.6%	425.0%	206.2%	29.30%	59.90%	
	Hard voting on top 3 models (MNB+LGBM+LR)	66.50%	71.90%	66.50%	64.60%	109.8%	458.5%	214.1%	31.90%	62.30%	
	Soft voting on top 5 models (MNB+LGBM+LR +XGBC+RF)	67.10%	69.70%	67.10%	65.60%	134.8%	601.8%	245.3%	39.20%	71.30%	
	Hard voting on top 5 models (MNB+LGBM+LR +XGBC+RF)	62.20%	67.60%	62.20%	60.10%	134.1%	576.8%	240.2%	39.00%	69.80%	



(a) K value vs Accuracy (Human). (b) K value vs Accuracy (Chimpanzee). (c) K value vs Accuracy (Dog).

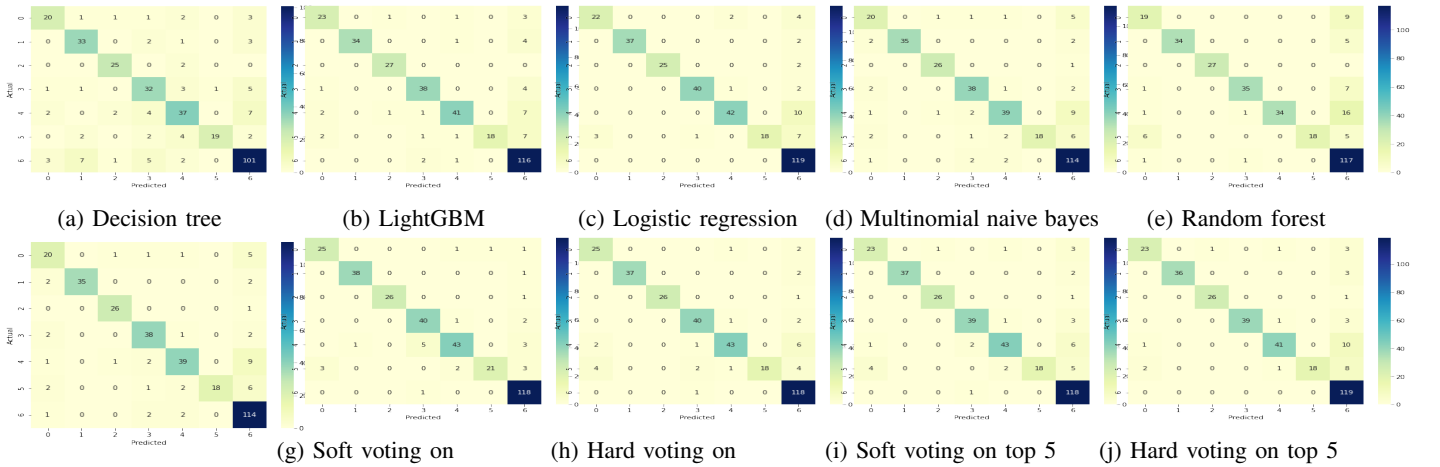
Fig. 3: Evaluation of a KNN classifier (accuracy metric) with the changes of number of nearest neighbors for the 3 datasets.



(a) Decision tree (b) LightGBM (c) Logistic regression (d) Multinomial naive bayes (e) Random forest

(f) XGBoost (g) Soft voting on top 3 (MNB+LR+RF) (h) Hard voting on top 3 (MNB+LR+RF) (i) Soft voting on top 5 (MNB+LR+RF+LGBM+XGBC) (j) Hard voting on top 5 (MNB+LR+RF+LGBM+XGBC)

Fig. 4: Exhibit the confusion matrix results for the **Human Dataset** through all our suggested models.

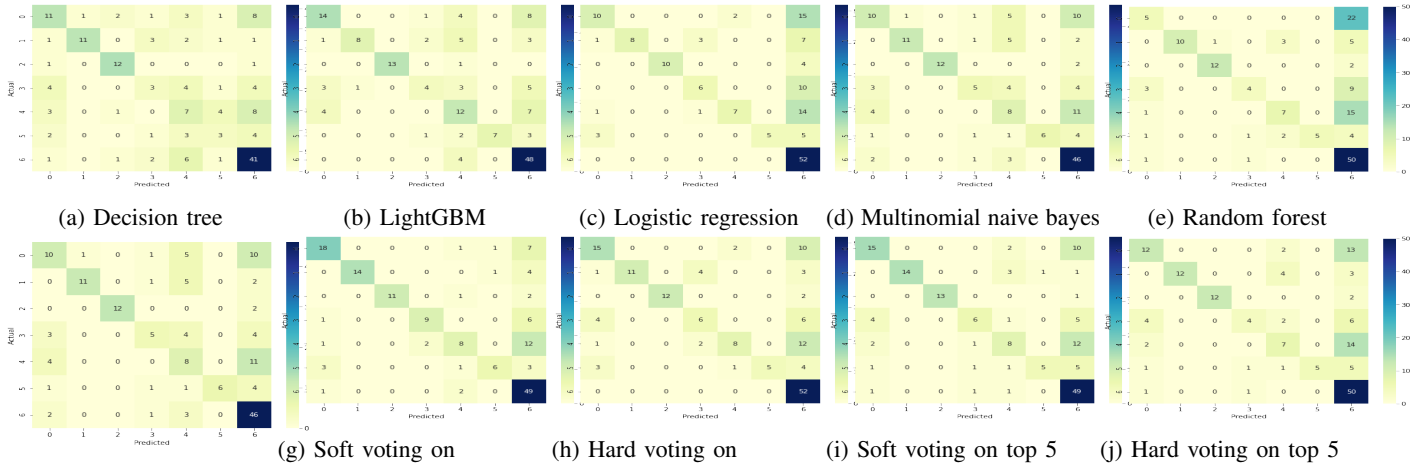


(a) Decision tree (b) LightGBM (c) Logistic regression (d) Multinomial naive bayes (e) Random forest

(f) XGBoost (g) Soft voting on top 3 (MNB+LR+LGBM) (h) Hard voting on top 3 (MNB+LR+LGBM) (i) Soft voting on top 5 (MNB+LR+LGBM+XGBC+KNN) (j) Hard voting on top 5 (MNB+LR+LGBM+XGBC+KNN)

Fig. 5: Exhibit the confusion matrix results for the **Chimpanzee Dataset** through all our suggested models.

evaluation of various models was conducted, including Multinomial Naive Bayes, Logistic Regression, Random Forest, LightGBM, XGBoost, K-Nearest Neighbors, and Decision Tree, which achieved accuracy scores of 70.10%, 59.77%, 56.71%, 64.63%, 59.76%, 51.22%, and 53.66%, respectively. Out of the classifiers, Multinomial Naive Bayes model showcased the best accuracy. In addition, it secured noteworthy precision, recall, and f1-score figures of 73.20%, 70.10%, and 69.40%. Moreover, the soft voting ensemble on the top three models accomplished a notable accuracy of 70.12%, distinguishing it as the best model among all suggested models. Besides, it presents noteworthy precision, recall, and



(a) Decision tree (b) LightGBM (c) Logistic regression (d) Multinomial naive bayes (e) Random forest

(f) XGBoost (g) Soft voting on top 3 (MNB+LGBM+LR) (h) Hard voting on top 3 (MNB+LGBM+LR) (i) Soft voting on top 5 (MNB+LGBM+LR+XGBC+RF) (j) Hard voting on top 5 (MNB+LGBM+LR+XGBC+RF)

Fig. 6: Exhibit the confusion matrix results for the **Dog Dataset** through all our suggested models.

f1-scores of 73.10%, 70.10%, and 69.20%. Additionally, due to lack amount of data, we noted a substantial variation in the accuracy scores of the dog dataset compared to other two.

V. Discussion and Conclusion

Our proposed both soft and hard voting ensemble model was employed to all of the three species datasets to assess its cross-species performance. In this study, the choice of value **K** in k-mer counting was significant, as it determines the length of the subsequences considered. This parameter is crucial in capturing specific patterns and characteristics within the genetic data. We explored the models' performance through adjustments to the **K** value, spanning from 1 to 6. Our observations indicate that the recommended algorithms deliver superior performance at K=6, emphasizing the importance of a substring length comprising 6 nucleotides. Notably, deviations beyond this value result in a decline in performance. In addition, the CountVectorizer was applied to establish a BoW (Bag of Words) model, concentrating on the counts of 4-grams. In the case of tetragram vectorization, we attained the top accuracy, leading us to opt for tetragram tokenization. Besides, We noticed that the soft voting ensemble consistently gave a small advantage in accuracy than hard voting ensemble. We also observed a consistent trend where the ensemble accuracies with the top three models consistently surpassed those with the top five models. So according to this study, we can conclude that the increment of classifier models could also degrade the performance of voting ensemble models.

References

[1] Chou KC, Shen HB. Predicting eukaryotic protein subcellular location by fusing optimized evidence-theoretic K-Nearest Neighbor classifiers. J Proteome Res. 2006 Aug;5(8):1888-97. doi: 10.1021/pr060167c. PMID: 16889410.

[2] Akhtar, M., Epps, J., & Ambikairajah, E. (2007). On DNA Numerical Representations for Period-3 Based Exon Prediction. 2007 IEEE International Workshop on Genomic Signal Processing and Statistics, 1-4.

[3] M. Akhtar, J. Epps and E. Ambikairajah, "Signal Processing in Sequence Analysis: Advances in Eukaryotic Gene Prediction," in IEEE Journal of Selected Topics in Signal Processing, vol. 2, no. 3, pp. 310-321, June 2008, doi: 10.1109/JSTSP.2008.923854.

[4] Ramachandran P, Lu WS, Antoniou A. Filter-based methodology for the location of hot spots in proteins and exons in DNA. IEEE Trans Biomed Eng. 2012 Jun;59(6):1598-609. doi: 10.1109/TBME.2012.2190512. Epub 2012 Mar 9. PMID: 22410955.

[5] W. Kinsner, "Towards cognitive analysis of DNA," 9th IEEE International Conference on Cognitive Informatics (ICCI'10), Beijing, China, 2010, pp. 6-7, doi: 10.1109/COGINF.2010.5599728.

[6] Wang Z, Chen Y, Li Y. A brief review of computational gene prediction methods. Genomics Proteomics Bioinformatics. 2004 Nov;2(4):216-21. doi: 10.1016/s1672-0229(04)02028-5. PMID: 15901250; PMCID: PMC5187414.

[7] Olsvik O, Wahlberg J, Petterson B, Uhlén M, Popovic T, Wachsmuth IK, Fields PI. Use of automated sequencing of polymerase chain reaction-generated amplicons to identify three types of cholera toxin subunit B in Vibrio cholerae O1 strains. J Clin Microbiol. 1993 Jan;31(1):22-5. doi: 10.1128/jcm.31.1.22-25.1993. PMID: 7678018; PMCID: PMC262614.

[8] B. A. Mir, M. U. Rehman, H. Tayara, and K. T. Chong, "Improving enhancer identification with a multi-classifier stacked ensemble model," Journal of Molecular Biology, vol. 435, no. 23, p. 168314, 2023.

[9] S. Sarkar, K. Mridha, A. Ghosh, and R. N. Shaw, "Machine learning in bioinformatics: New technique for dna sequencing classification," in Advanced Computing and Intelligent Technologies: Proceedings of ICACIT 2022. Springer, 2022, pp. 335–355.

[10] S. Juneja, A. Dhankhar, A. Juneja, and S. Bali, "An approach to dna sequence classification through machine learning: Dna sequencing, k mer counting, thresholding, sequence analysis," International Journal of Reliable and Quality E-Healthcare (IJRQEH), vol. 11, no. 2, pp. 1–15, 2022.

[11] G. Mathur, A. Pandey, and S. Goyal, "A comprehensive tool for rapid and accurate prediction of disease using dna sequence classifier," Journal of Ambient Intelligence and Humanized Computing, vol. 14, no. 10, pp. 13 869–13 885, 2023.

[12] S. S. Kanumalli, S. Swathi, K. Sukanya, V. Yamini, and N. Nagalakshmi, "Classification of dna sequence using machine learning," in Soft Computing for Security Applications: Proceedings of ICSCS 2022. Springer, 2022, pp. 723–732.

[13] J. Rexie, K. Raimond, D. Brindha, and A. K. Prabavathy, "K-mer based prediction of gene family by applying multinomial na¨ıve bayes algorithm in dna sequence," in AIP Conference Proceedings, vol. 2914, no. 1. AIP Publishing, 2023.

[14] B. A. Hamed, O. A. S. Ibrahim, and T. Abd El-Hafeez, "Optimizing classification efficiency with machine learning techniques for pattern matching," Journal of Big Data, vol. 10, no. 1, p. 124, 2023.

[15] V. Upadhyay, S. Harbhajanka, S. Pangaonkar, and R. Gunjan, "Exploratory data analysis and prediction of human genetic disorder and species using dna sequencing," in Proceedings of the Future Technologies Conference. Springer, 2023, pp. 197–213.