Classification of DNA Sequence Using Machine Learning Techniques

Md. Ahsan Habib and Md. Motaleb Hossen Manik

August 4, 2022

Classification of DNA Sequence Using Machine Learning Techniques

Md. Ahsan Habib and Md. Motaleb Hossen Manik
Dept. of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna, Bangladesh
Email: mdnayan1507082@gmail.com and mkmanik557@gmail.com

Abstract— **DNA, the blueprint of life, a long repeating chain of nucleic acids, contains the genetic information of living organisms. Information extraction from DNA is an important research topic in genomics. The process of determining the order of base-pairs is called DNA sequencing and the activity of identifying whether or not an unlabeled sequence corresponds to an existing class is known as DNA sequence classification. This paper presents several machine learning techniques for DNA sequence classification using two public datasets. Promoters and splice datasets are used to assess the approaches' effectiveness and achieve noteworthy improvements in that datasets. Among all experimented schemes, only two of them have less than 90 percent accuracy in training the data sets and most of the techniques achieve more than 90 percent test accuracy. The results of the experiment reveal that several techniques outperform all other models.**

Keywords— ***DNA Sequence, DNA Sequence Classification, Machine Learning, K-Nearest Neighbour, Gaussian Processes, Decision Tree, Random Forest, AdaBoost, Support Vector Machine, Naive Bayes, Logistic Regression, Multi-Layer Perceptron.***

I. Introduction

DNA, Deoxyribonucleic acid, a long repeating chain of nucleic acids, contains the genetic information of living organisms [1]. DNA, the blueprint of life, is a crucial component of reproduction, as it allows genetic inheritance to be passed on from parent to offspring [1]. The information conveyed by DNA is stored in the form of a gene sequence, basic physical and functional unit of heredity, which is made up of many pieces of DNA. The method to determine the exact sequence of four base-pairs (A - Adenine, T - Thymine, C - Cytosine, or G - Guanine) in DNA molecule is called DNA sequencing. DNA sequencing knowledge is increasingly required for basic biological research as well as a variety of applied sectors such as biotechnology, medical diagnosis, virology, forensic biology and biological systematics. With the progression of sequencing tools and techniques, reading a DNA sequence has become fairly simple. DNA sequence data is also growing at an exponential rate. In December 2015, Genbank database had surpassed two billion base pairs [2]. It would be fantastic if we could combine these massive data sets with the computing power of today's computers to aid our understanding of DNA.

At present, DNA sequence classification is an important research focus in different prospects. Several studies are being carried out to classify DNA sequences. Different techniques were introduced for classification tasks such as Directed Acyclic Word Graphs (DAWGs) [3], Vector Space Classification [4], expectation-maximization algorithm along with neural network (NN) [5], variable order hidden Markov model with the continuous state: VOGUE [6], etc. Several researchers use different machine learning (ML) techniques for classification purposes [2], [7]–[9].

Recently, ML has brought attention to the genomics researcher. Dixit and Prajapati [10] analyze several ML algorithms with different contexts and datasets. Artificial Neural Network (ANN), Support Vector Machine (SVM) and Artificial NeuroFuzzy Inference System (ANFIS) techniques are considered in their study. The pros and cons of every method are mentioned in their study. SVM performs better with splice dataset for site classification but the appropriate kernel needs to be fixed out. In case of ANN, the performance increases with the growing number of neurons and is suitable for the identification of promoters only. ANFIS works well for gene classification of mediating cancer but to make significant groups it faces complexity which is the drawback of the algorithm. In the case of generic data, Collober et al. [11] first demonstrated that CNNs may be employed well for sequence analysis. Nguyen et al. [2] developed a technique for classifying DNA sequences by employing a convolutional neural network (CNN) and treating them as text input. As input to the model, they employed one-hot vectors to symbolise the sequences. They investigated that the proposed model using 12 DNA sequence datasets and achieved significant improvements in all of them. The minimum improvement in accuracy was nearly 1%, while the major improvement was more than 6%. A novel way is presented to classify pairwise sequence alignments for exact clustering of non-coding RNA (ncRNA) sequences using one-dimensional CNN [12]. Word2vec and one-hot coding were used to combine secondary-structure information unique to ncRNAs and with read-mapping profiles. The data used in their study was collected from RNA family databases such as HGNC and Rfam. The CNN-based approaches outperformed previous approaches on both Accuracy and F-value in 10-fold cross-validation. Giosue Lo Bosco and Mattia Antonino Di Gangi [9] proposed two DL models namely convolutional neural network (CNN) and long short-time memory network (LSTM) for DNA sequence classification. Multi-task learning variant is introduced, for both CNN and LSTM models, which affects both training time and performance of the model. 10-fold cross-validation is performed and 15 epochs are chosen for each fold.

This study aims to apply different ML techniques for DNA sequence classification and make comparisons among them. K-Nearest Neighbour (KNN), Gaussian Processes (GP), Decision Tree (DT), Random Forest (RF), AdaBoost, Naive Bayes variants (Gaussian Naive Bayes (GNB), MultiNomial Naive Bayes (MNB) and Bernoulli Naive Bayes (BNB)), Support Vector Machine (SVM) with different kernels, and Logistic Regression (LR) are applied as ML techniques for the classification task. On the other hand, Multi-Layer Perceptron (MLP), a DL technique, is also employed to classify DNA sequence. The performance of this study is measured using two well-known public datasets.

The rest of this paper is arranged as follows. The ML techniques are explained in Section II. The experimental studies and findings are presented in Section III. Finally, in Section IV, the conclusion is presented.

II. ML Techniques For DNA Sequence Classification

ML techniques have attracted the interest of genomic researchers because of technological advancements, even though there is a range of procedures for DNA classification. The following is a summary of each ML approach.

A. K-Nearest Neighbors

The K-Nearest Neighbors (KNN) is a simple supervised ML technique that may be used to address both classification and regression problems works based on similarity of data [13]. To find data similarity, many distance measuring techniques can be applied like Euclidean distance, Chebyshev distance, Cosine distance, etc. KNN algorithm is faster than other methods as there need no training before generating predictions. The algorithm becomes much slower as the size of the data in use increases which is the major disadvantage of KNN. The high dimensionality of data also hampers the calculation.

B. Gaussian Processes Classifier

The Gaussian Processes Classifier (GPC) is a generic supervised ML algorithm that uses the Gaussian probability distribution [14]. Gaussian processes, like SVMs, are a form of kernel approach, however, unlike SVMs, they can predict highly calibrated probabilities. It is worth noting that Gaussian Processes can be applied to solve both classification and regression problems. When data has a lot of dimensions, the classifier loses efficiency since it uses all of the features information to make a prediction which is the major disadvantage of GPC.

C. Decision Tree

Decision Tree (DT) is a supervised ML technique can be employed for both classification and regression problems. The algorithm works by building a training model that can predict the class or value of a target variable employing simple decision rules established from training data. The prediction starts comparison from the value of root node of the tree and continues until the terminal node. During pre-processing phase, this technique needs less effort for data preparation than other methods and the data does not need to be normalized. A minor change in data can result in a notable change in the DT's structure, producing instability. DT needs high time for training and sometimes the calculation goes far complex. [15]

D. Random Forest

Random Forest (RF) is a flexible and most used supervised ML algorithm because of its diversity and simplicity which can be applied for both classification and regression problems [16]. To produce a more stable and accurate estimation, the algorithm constructs multiple decision trees which are usually trained with the bagging method and combines them. This algorithm is a useful scheme as the default hyperparameters it employs frequently to produce accurate predictions and understanding them is simple. The major drawback of RF is that it can become too slow and ineffective for real-time classification if there are too many trees.

E. Adaptive Boosting (AdaBoost)

Several weak learning algorithms are combined to build the final strong predictive classification algorithm since when predicting a single classifier does not produce an accurate result. The outcome of those weaker algorithms is amalgamated into a weighted sum that makes the final output of the Adaboost classifier. Despite the individual learner's weakness, because the performance of each algorithm is slightly better than random guessing AdaBoost provides better results than other approaches [17]. Adaboost classifier doesn't allow overfitting problems. The limitation of the classifier is that the algorithm requires high-quality data and the classifier is very sensitive to outlier and noise.

F. Naive Bayes

The Naïve Bayes (NB) algoritm is a classification technique based on Bayes' theorem [18]. The model is simple to construct and is especially effective for the huge amount of data [19]. In comparison to numerical input variables, it performs well with categorical input variables. Gaussian Naive Bayes (GNB), Multinomial Naive Bayes (MNB), Bernoulli Naive Bayes (BNB), etc are variants available of NB. The classifier will give zero probability to any category variable in the test set that does not appear in the training data set, making prediction impossible. The smoothing technique can be employed to solve the zero-frequency problem. On the other hand, it is also known as a bad estimator which is another limitation of NB.

G. Support Vector Machine

The Support Vector Machine (SVM) is a supervised ML technique that can use to analyze data for classification and regression analysis [20], [21]. To find a hyperplane in an N-dimensional space is the objective of SVM classifier, whereas N is the number of features, that distinguishes between data points. The hyperplane's dimension depends upon the number of features. The data points closed to the hyperplane known as support vectors that influence the hyperplane's orientation and position. Those vectors are also used to maximize the margin of the hyperplane which is the objective of this algorithm. Whenever there is a vibrant margin of distinction between classes, SVM performs well. The algorithm is also effective in high-dimensional spaces [22], [23]. This method is not suitable for huge datasets or data that is more noisy. If the number of features for each data point exceeds the number of training data samples, this technique will underperform.

H. Logistic Regression

Despite the algorithm contains the word "regression" in its name, Logistic Regression (LR) is a supervised ML scheme that is usually employed to handle classification problems especially binary classification [24]. Logistic function is the fundamental function of this technique that also known as sigmoid function. It takes any real numerical value and transforms to a value between 0 and 1. This method is much easier to implement, interpret, and very efficient to train and performs well for linearly separable data [25]. LR should not be used if the number of observations is less than the number of features; otherwise, it may result in overfitting. Linear relationship assumption between dependent and independent variables is the major limitation of the model.

Table 1. Description of Datasets.

Sl. No	Dataset	Title of Dataset	No. of Samples	No. of Classes	Class Distribution	Length of Sequence
1	Promoters	E. coli promoter DNA sequences with associated imperfect domain theory	106	2	53;53	57
2	Splice	Primate splice-junction DNA sequences with associated imperfect domain theory	3190	3	767; 768; 1655	60

I. *Multi-Layer Perceptron*

A single perceptron effectively classifies linearly separable data. It encounters a severe problem with linearly inseparable data. Multi-Layer Perceptron (MLP), a DL technique, breaks this limitation and classify effectively the data that is not linearly separable [26]. The very simple architecture of an MLP contains an input layer, a hidden layer and an output layer. There may have multiple hidden layers in between input and output layers. Each layer can have one or more neurons. The dot product of inputs (from either input layer or previous layer) with the weights (exist between either hidden layer and input layer or two hidden layers or hidden layer and output layer) is pushed forward through the network. Non-linear activation functions like *tanh*, *sigmoid*, *relu*, etc can be used for calculation. MLP uses backpropagation algorithm, a supervised learning technique, during training. Figure 1 depicts a general architecture of the MLP neural network. The network contains *n* inputs and *k* outputs with *m* hidden layers. Hidden layer 1 has *t* neurons and hidden layer *m* consists of *r* neurons. For this study, only one hidden layer is used containing 100 neurons. *Relu* activation function is used for the hidden layer. *Adam* optimizer is employed for weight optimization. The solver iterates until convergence or default maximum of 200 iterations can occur. Figure 2 illustrates the applied MLP network for this study.

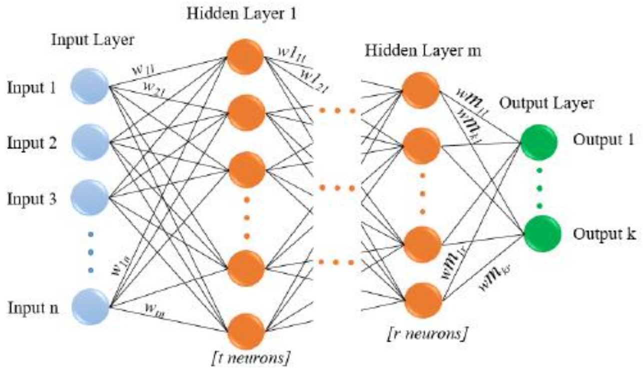


Fig. 1. An illustration of MLP Neural Network.

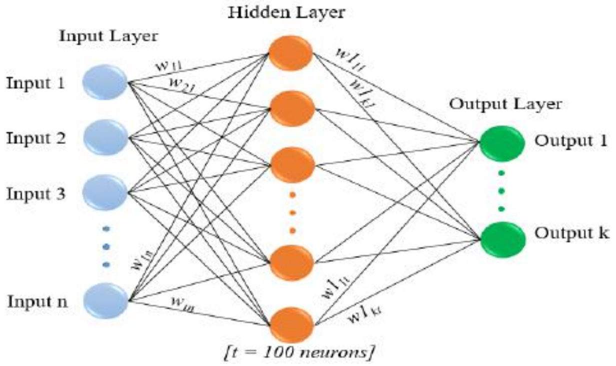


Fig. 2. Proposed MLP Neural Network.

III. EXPERIMENTAL STUDIES

This section describes data description and pre-processing, experimental setup, and classification accuracy using the aforementioned methodologies.

A. *Data Description and Pre-processing*

Table 1 shows the dataset description for this study. Two popular-public datasets are used namely promoters and splice. Promoters dataset contains 3 columns (class, id, seq – sequence) and 106 tuples. Each class is either positive (promoter) or negative (non-promoter) and each sequence contain 57 sequential nucleotides. There is no necessity for the id column for the classification task. The class distribution is exactly 50% for the promoters dataset. There are 3190 instances in the Splice dataset, which also comprises 3 columns (class, name, seq – sequence). The class column can have three possible categories (EI, IE, and N) and each sequence containing 60 sequential base-pairs. “name” column has no need here. EI, IE, and N each have 767, 768, and 1655 instances respectively among the 3190 total.

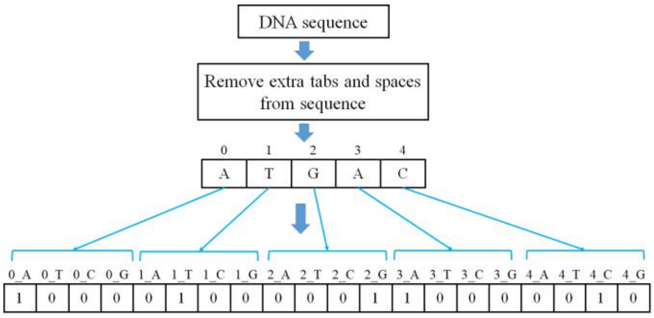
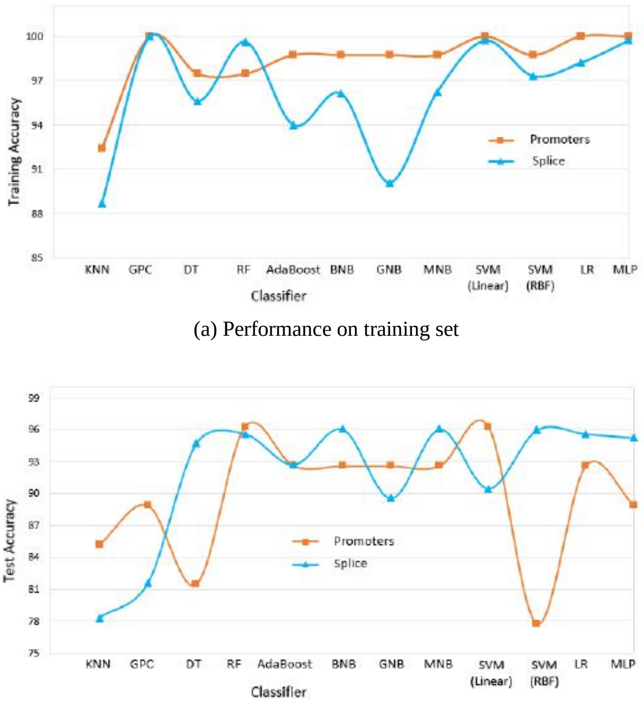

DNA sequence
Remove extra tabs and spaces from sequence
0 1 2 3 4
A T G A C
0_A 0_T 0_C 0_G 1_A 1_T 1_C 1_G 2_A 2_T 2_C 2_G 3_A 3_T 3_C 3_G 4_A 4_T 4_C 4_G
1 0 0 0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 1 0

Fig. 3. Data Preprocessing.

There are dummy spaces or tabs in both datasets that can be eliminated. Because ML techniques cannot be performed on string data, we must convert it to numerical data. Figure 3 shows the data preprocessing steps of a portion of DNA sequence for this study. At first, extra spaces and tabs are eliminated from the sequences. Then each base is divided into four bases. Set 1 in the corresponding base and put 0 to the other three bases. Positive and negative classes are represented by 1 and 0, respectively, in the promoters dataset.



(a) Performance on training set

(b) Performance on test set

Fig. 4. Classification accuracies for different methods.

The EI, IE, and N classes are represented by 1, 2, and 3 in the splicing dataset, respectively. For both datasets, 75% is used to train the models, while the remaining 25% is utilized as a test set to ensure that the models are generalizable.

B. *Experimental Setup*

The models and data analysis are implemented using the Python programming language. The experiment was carried out in online environment called www.kaggle.com with the use of a jupyter notebook. The experiment was carried out on a PC (Intel(R) Core(TM) i3-5005U, CPU @ 2.00 GHz, RAM 4 GB) running Windows10.

C. *Experimental Results and Analysis*

Figure 4 shows both training set and test set accuracy of different classification techniques. The training accuracy of both promoters and splice datasets is depicted in Fig. 4(a). Only two models have an accuracy rate of less than 90% on training data, and only a few approaches have been taught sufficiently enough to achieve near-perfect accuracy. The test set accuracies of both datasets, on the other hand, shows in Fig. 4(b). For promoters dataset, the best test set accuracy is 96.30%, which achieves both RF and linear SVM techniques. The largest test accuracy of splice dataset is 96.07% that achieves two variants of NB models, BNB and MNB. The lowest test accuracy achieved by the SVM algorithm with RBF kernel for promoters dataset is 77.78%. For splice dataset, 78.32% is the minimum test set accuracy experimented by KNN scheme. In every ML algorithm, more test set accuracy is desirable because it demonstrates the system's ability to generalize. Better performance in the test case demonstrates that using DNA sequence the suggested methods' capacity to correctly understand the proper sequence.

IV. CONCLUSIONS

The genetic information of almost all living organisms is encoded in DNA, a complex molecule. Four bases A, T, C and G are fundamental building blocks of DNA sequence. Sequence analysis and classification of DNA have significant standings in a variety of perspectives. In this study, several ML schemes are applied for DNA sequence classification using promoters and splice datasets. Since DNA sequencing can be useful in a variety of fields, these studied methods with satisfactory classification accuracy might be applicable in different prospects.

In this study, several ML techniques are applied for DNA sequence classification. Two public datasets are used for this classification task. This study opens the door to several future research directions. More datasets can be applied to observe the training and test set capability of those ML techniques, which remained as a future study. Moreover, tuning hyperparameters of different ML models may increase the performance of the models.

REFERENCES

[1] K. Vij and R. Biswas, "What is DNA?," in *Basics of DNA and Evidentiary Issues*, Jaypee Brothers Medical Publishers (P) Ltd., 2004, pp. 1–1.

[2] N. G. Nguyen *et al.*, "DNA Sequence Classification by Convolutional Neural Network," *J. Biomed. Sci. Eng.*, vol. 09, no. 05, pp. 280–286, 2016.

[3] S. Levy and G. D. Stormo, "DNA sequence classification using DAWGs," 1997, pp. 339–352.

[4] H.-M. Müller and S. E. Koonin, "Vector space classification of DNA sequences," *J. Theor. Biol.*, vol. 223, no. 2, pp. 161–169, Jul. 2003.

[5] Qicheng Ma, J. T. L. Wang, D. Shasha, and C. H. Wu, "DNA sequence classification via an expectation maximization algorithm and neural networks: a case study," *IEEE Trans. Syst. Man Cybern. Part C (Applications Rev.*, vol. 31, no. 4, pp. 468–475, 2001.

[6] M. J. Zaki, C. D. Carothers, and B. K. Szymanski, "VOGUE: a variable order hidden Markov model with duration based on frequent sequence mining," *ACM Trans. Knowl. Discov. Data*, vol. 4, no. 1, pp. 1–31, Jan. 2010.

[7] R. Ranawana and V. Palade, "A neural network based multi-classifier system for gene identification in DNA sequences," *Neural Comput. Appl.*, vol. 14, no. 2, pp. 122–131, Jul. 2005.

[8] N. A. Kassim and A. Abdullah, "Classification of DNA Sequences Using Convolutional Neural Network Approach," *Fac. Comput. Univ. Teknol. Malaysia (UTM), Malaysia*, 2017.

[9] G. Lo Bosco and M. A. Di Gangi, "Deep Learning Architectures for DNA Sequence Classification," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017, pp. 162–171.

[10] P. Dixit and G. I. Prajapati, "Machine Learning in Bioinformatics: A Novel Approach for DNA Sequencing," in *2015 Fifth International Conference on Advanced Computing & Communication Technologies*, 2015, pp. 41–47.

[11] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural Language Processing (almost) from Scratch," *J. Mach. Learn. Res.*, Mar. 2011.

[12] G. Aoki and Y. Sakakibara, "Convolutional neural networks for classification of alignments of non-coding RNA sequences," *Bioinformatics*, vol. 34, no. 13, pp. i237–i244, Jul. 2018.

[13] Z. Zhang, "Introduction to machine learning: k-nearest neighbors," *Ann. Transl. Med.*, vol. 4, no. 11, pp. 218–218, Jun. 2016.

[14] D. J. C. Mackay and M. N. Gibbs, "Variational Gaussian process classifiers," *IEEE Trans. Neural Networks*, vol. 11, no. 6, pp. 1458–1464, 2000.

[15] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Trans. Syst. Man. Cybern.*, vol. 21, no. 3, pp. 660–674, 1991.

[16] L. Breiman, "Random forests," *Mach. Learn.*, 2001.

[17] M. Collins, R. E. Schapire, and Y. Singer, "Logistic regression,

AdaBoost and Bregman distances," *Mach. Learn.*, 2002.

[18] S. Xu, "Bayesian Naïve Bayes classifiers to text classification," *J. Inf. Sci.*, vol. 44, no. 1, pp. 48–59, Feb. 2018.

[19] D. Berrar, "Bayes' theorem and naive bayes classifier," in *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, 2018.

[20] R. G. Brereton and G. R. Lloyd, "Support Vector Machines for classification and regression," *Analyst*, vol. 135, no. 2, pp. 230–267, 2010.

[21] M. Awad and R. Khanna, "Support Vector Machines for Classification," in *Efficient Learning Machines*, Berkeley, CA: Apress, 2015, pp. 39–66.

[22] M. Ramachandro and R. Bhramaramba, "Classification of gene expression data set using support vectors machine with RBF kernel," *Int. J. Recent Technol. Eng.*, 2019.

[23] Y. Nakayama, "Robust support vector machine for high-dimensional imbalanced data," *Commun. Stat. - Simul. Comput.*, vol. 50, no. 5, pp. 1524–1540, May 2021.

[24] S. Sperandei, "Understanding logistic regression analysis," *Biochem. Medica*, pp. 12–18, 2014.

[25] S. Wu, H. Jiang, H. Shen, and Z. Yang, "Gene Selection in Cancer Classification Using Sparse Logistic Regression with L1/2 Regularization," *Appl. Sci.*, vol. 8, no. 9, p. 1569, Sep. 2018.

[26] S. Potghan, R. Rajamenakshi, and A. Bhise, "Multi-Layer Perceptron Based Lung Tumor Classification," in *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 2018, pp. 499–502.