Journal of Ambient Intelligence and Humanized Computing
https://doi.org/10.1007/s12652-022-04099-y

ORIGINAL RESEARCH

A comprehensive tool for rapid and accurate prediction of disease using DNA sequence classifier

Garima Mathur[1] · **Anjana Pandey**[2] · **Sachin Goyal**[2]

Received: 10 December 2021 / Accepted: 6 June 2022

Abstract
In the current pandemic situation where the coronavirus is spreading very fast that can jump from one human to another. Along with this, there are millions of viruses for example Ebola, SARS, etc. that can spread as fast as the coronavirus due to the mobilization and globalization of the population and are equally deadly. Earlier identification of these viruses can prevent the outbreaks that we are facing currently as well as can help in the earlier designing of drugs. Identification of disease at a prior stage can be achieved through DNA sequence classification as DNA carries most of the genetic information about organisms. This is the reason why the classification of DNA sequences plays an important role in computational biology. This paper has presented a solution in which samples collected from NCBI are used for the classification of DNA sequences. DNA sequence classification will in turn gives the pattern of various diseases; these patterns are then compared with the samples of a newly infected person and can help in the earlier identification of disease. However, feature extraction always remains a big issue. In this paper, a machine learning-based classifier and a new technique for extracting features from DNA sequences based on a hot vector matrix have been proposed. In the hot vector representation of the DNA sequence, each pair of the word is represented using a binary matrix which represents the position of each nucleotide in the DNA sequence. The resultant matrix is then given as an input to the traditional CNN for feature extraction. The results of the proposed method have been compared with 5 well-known classifiers namely Convolution neural network (CNN), Support Vector Machines (SVM), K-Nearest Neighbor (KNN) algorithm, Decision Trees, Recurrent Neural Networks (RNN) on several parameters including precision rate and accuracy and the result shows that the proposed method gives an accuracy of 93.9%, which is highest compared to other classifiers.

Keywords DNA sequence · Classifier · Feature extraction · Convolution neural network (CNN) · Support vector machines (SVM) · K-nearest neighbor (KNN) algorithm · Decision trees · Recurrent neural networks (RNN)

1 Introduction

Earlier identification of viruses and diseases is the biggest concern for everyone after the covid-19 outbreak as their earlier identification can save more lives and can help in the earlier designing of drugs (Herath et al. 2022; Shadab et al. 2020; Herath et al. 2021a, b). However, due to the shortage of patterns available for various diseases, it becomes difficult to identify them. The classification of DNA sequence can be considered one of the solutions and this is the reason why the classification of DNA sequence plays an important role in computational biology.

NCBI has the largest collection of genome sequences, that's why it is named Genbank. It consists of millions of distinct DNA sequences with billions of nucleoid bases (Benson et al. 2010). DNA sequences are the blueprints of any organism. DNA stands for deoxyribonucleic acid which is made up of four nucleotides namely A, C, G, and T i.e. Adenine, cytosine, guanine, and thymine. The DNA sequence of every disease and virus is unique which helps to extract unique patterns (Momenzadeh et al. 2020). The nucleotides

✉ Garima Mathur
garima41mathur@gmail.com

Anjana Pandey
anjanapandey@rgtu.net

Sachin Goyal
sachingoyal@rgtu.net

[1] Department of Computer Science and Engineering, UIT, RGPV, Bhopal, India

[2] Department of Information Technology, UIT, RGPV, Bhopal, India

Published online: 25 June 2022

in any form are always bound in double-stranded with its harmonious pair (A-T and C-G pair) as shown in Fig. 1.

In this paper, a framework has been proposed in which whenever a patient (infected from any virus or any other disease) visits a doctor his/her DNA samples are collected and matched with the database to identify the occurrence of the disease. However, due to the shortage of patterns available for various diseases, it becomes difficult to identify them. This paper has presented a solution in which samples collected from NCBI are used for the classification of DNA sequences. DNA sequence classification will in turn gives the pattern of various diseases; these patterns are then compared with the samples of a newly infected person and can help in the earlier identification of disease. As we know the size of FASTA-based DNA sequences is too big and complex which means this data cannot be given directly for feature extraction. For this data needed to be converted to some equivalent numerical form, in this paper a new hot vector-based numerical representation is introduced, where the position of each nucleotide is reserved by using binary 0 or 1. This hot vector matrix is then given as an input to traditional CNN for feature extraction. The models are then trained and evaluated on test data. Finally, the model can be applied to unseen data (Unknown DNA sequence) for disease prediction.

Various ML (machine learning) techniques are there that can be used for classification purposes. For analysis of our work, we have taken 37,272 COVID samples, 1418 Middle East Respiratory Syndrome (MERS), 6503 Severe Acute Respiratory Syndrome (SARS), 1886 DENGUE, 8226 HEPATITIS, and 10,848 INFLUENZA samples. These samples are firstly used for training and then testing our model on trained datasets. To train machine models for various kind of disease, feature extraction of some well-known sequences are done. In this work, a new neural network approach has been proposed that takes a hot vector matrix as input to convolution layers for extracting features from given input data. The position of each nucleotide of the DNA sequence is represented using a hot vector and the previous layer's extracted features are used by the convolution layer's neurons for extracting high-level abstraction features (Fig. 2).

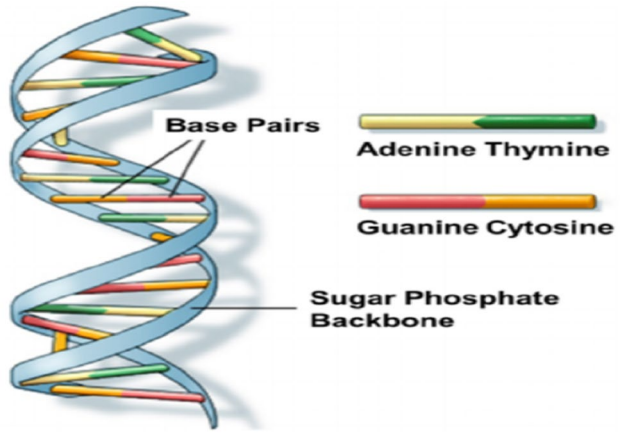


Fig. 1 DNA base pair structure with the sugar-phosphate backbone

2 Background

In this paper Sect. 2.1 highlights the DNA sequence in FASTA file format. Section 2.2 gives a detailed description of what is classification, types of classifiers, and their basic introduction.

2.1 DNA sequence in FASTA file format

In this paper, we have also studied DNA sequences in FASTA file format, which is a text-based format used to denote sequences in bioinformatics (Hach et al. 2014; Mohammed et al. 2012). It consists of base pairs using the single-letter code as shown in Table 1.

The layout of organisms depends upon the order in which they are stored. The format allows sequences of preceding sequence names and comments. FASTA file format sequence starts with a one-line description of identifier along with DNA data sequence.

>AR147821.1 Sequence 1 from patent US 6225051
GGCATCTGAGACCAGTGAGAA

Fig. 2 Number of samples in a dataset

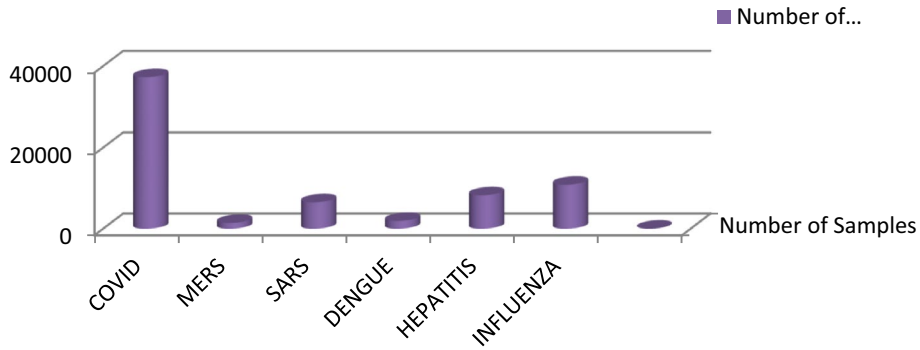


Table 1 DNA base pairs

A	Adenosine
C	Cytosine
G	Guanine
T	Thymidine
N	Can be any of these (A, C, G, T)

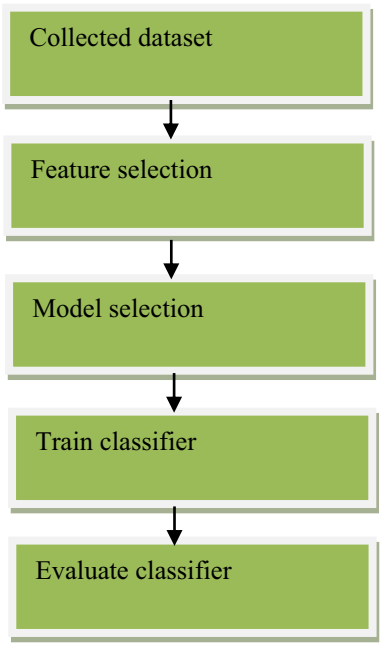


Fig. 3 Pattern recognition model

The details about the identifier are separated from the sequence of data by the '>' sign. The identifier of the sequence is the word after the">" symbol and the remaining part represents the description which is an optional part that separates the identifier from a white space or tab. The data sequence will start from the next line after the text line and the second line begins with another ">" sign indicating the ending of the sequence and also the beginning of another sequence.

2.2 Pattern recognition

For the identification of various kinds of disease, pattern recognition is considered an important factor. It consists of three primary elements which are data perception, extraction of features, and data classification (Fig. 3) (Sathish kumar et al. 2005; Jain et al. 2004).

2.2.1 Preprocessing

To increase the efficiency of the work, the dataset should be pre-processed instead of giving direct input of the raw dataset to selected classifiers; the raw dataset is preprocessed in different ways to overcome different issues like training overhead, and classifier confusion, false alarms, and detection rate ratios. Separating feature spaces from one another is very necessary and arranged in vector. CNN can be used for various textual data problems such as for classification and categorization purposes, but the only difference is that digital pictures are 2-dimensional matrixes whereas textual data is a 1-dimensional form consisting of letters. That's why we need to convert it to its equivalent numerical value with the goal that it can be now given as input to CNN.

2.2.2 Features selection

Feature selection is an important factor in any kind of classification either supervised or unsupervised. Since the large numbers of features can be monitored taking into account the large variety of possible values especially for a continuous feature even for a small network.

2.2.3 Final cluster data

As per the final updated population of the cluster center session, the proposed work can use these sessions only for training the neural network. Here, this will increase the learning capacity of the work. As more patterns from the same class increase the confusion of the system. This can be said as small input learning cluster training data improving the detection rate of the proposed work.

2.2.4 Classification

The proposed model uses the latest deep learning neural network for the training as a finding of the subclass is possible by using these networks. Based on this neurons of the network will adjust their weight. The feature vector is grouped during the feature collection steps of the different type of class which is matched, in the network. Finally, TN (Trained neural network) is obtained. For the classification of text data, CNN is proved to give the best results, so in this work, we are using CNN for the classification of textual DNA data.

3 Classification techniques

The most common way of perceiving, comprehending, and gathering thoughts and items into preset categories or "sub-populaces" is known as classification. ML uses several algorithms and some pre-categorized datasets i.e. training

datasets to classify the new dataset. These training datasets are then used to predict which sequence will come into which pre-categorized set. One simplest example of classification is to filter mails into spam or nonspam categories or we can say that "It is a form of pattern recognition where classification algorithm is used along with training dataset to check the patterns such as words with similar sentiments etc. for future dataset".

3.1 Machine learning-based systems

In a machine, learning-based text classification system past experiences/observations are used for making the system learn anything (Ikonomakis et al. 2005). The first step for training an ML-based system is feature extraction, where the textual data is converted into its equivalent number form and represented in the form of a vector. One such method is to represent a word with its frequency of appearing, this approach is known as a "bag of words". Suppose we have a dictionary of words- {I, have, has, dog, cat, a} and we want to represent the text "I have a dog" in vector form then the vector representation of the text will be {1, 1, 0, 1, 0, 1}.To generate a classification model, a training dataset is given as an input to the ML algorithm that consists of feature set pairs as well as tags (Fig. 4).

Once the ML model gets trained with several training data samples it becomes ready to make exact predictions. This feature extractor can now be used in the transformation of unseen text into some feature set and these feature sets are then given as an input to the classification model to get accurate predictions on tags.

Classifying text using ML is very much accurate and precise as compared to when performed by any man-made system, it makes learning new examples very much easier.

Text classification algorithms based on ML: two text classifiers are listed below:

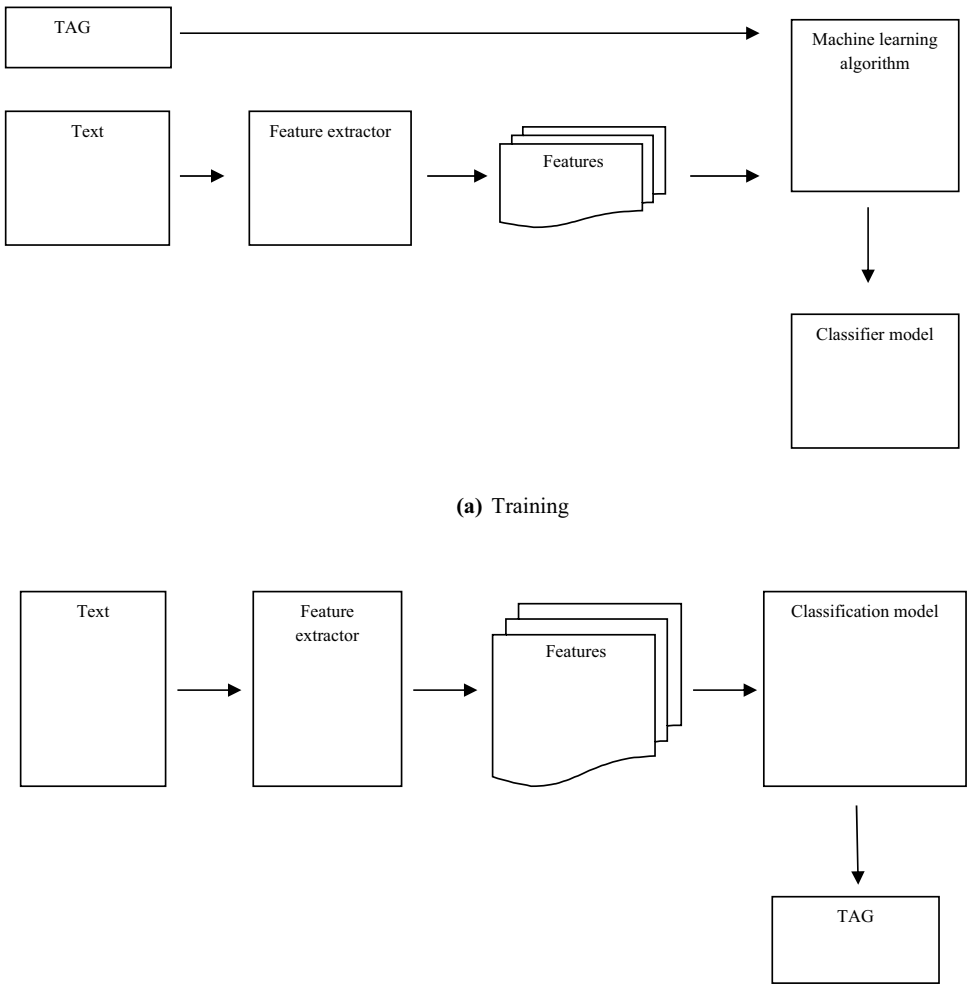


Fig. 4 **a** Training. **b** Prediction

(a) Training

(b) Prediction

3.2 SVM (support vector machines)

SVM stands for support vector machine, which is well known for its more precise and even fastest results than any other ML algorithm without much training (Shanahan et al. 2003). However, it requires more resources for performing its computational work. We can say it sketches a hyperplane that separates the subspace into two parts, the first part consists of vectors that belong to some group, and the second part consists of remaining vectors which do not belong to any group. Hyperlane can be defined as the longest distance between each tag. Figure 5a shows the example of a hyperplane in 2 dimensions.

Vectors represent the training text and the group represents tags used for tagging texts. As the complexity of data increases, it becomes difficult to categorize/classify vectors and tags into 2 subgroups only.

However, even when the data becomes more complicated, SVM gives more accurate and precise results, which is the best thing about any classification technique. Figure 5c shows the best hyperplane in 3 dimensions (in circle form).

3.3 Deep learning

Deep learning consists of an algorithm that was inspired by the phenomena of how a human brain works, which is well known as neural networks. For textual classification, deep learning provides the best and most accurate results with the lowest computations. Two well-known text classification deep learning techniques are:

- Convolutional neural networks (CNN)
- Recurrent neural networks (RNN).

3.3.1 RNN (recurrent neural network)

In this variety of neural networks, the output of the previous layer is feedback as an input to the current layer. It can help in predicting the output of the layer. Here every neuron has some memory that they kept before going to the next step. One of the most popular applications of RNN is in text-to-speech conversion (Ikonomakis et al. 2005).

3.3.2 CNN (convolution neural network)

A deep learning model with a vital thought of utilizing convolutional layers to separate features from input information is a convolution neural network. It was enlivened by the mechanism of living creatures, where the previous layer's extracted features are used by the convolution layer's neurons for extracting high-level abstraction features (Kassim et al. 2017). It consists of several artificial neurons in multiple layers. These neurons are used for computing the sum (weighted) of its input and giving activation value as an output.

Weight defines a neuron's behavior and by applying it with pixels CNN's neurons can extract the features. When an input image is given to CNN then an activation map is generated by each layer, which will highlight the image features. Every neuron now picks the pixels of an image and multiplies the colored value with the weight, adds them, and finally, the activation function runs them. Lines such as vertical, horizontal, and diagonal or we can say simple shapes are recognized through the first layer. Complicated shapes for example curves and line endpoints are recognized by second layer neurons. The ear, nose, and mouth are recognized by the third layer. Finally, the complete human face is recognized by the last layers of neurons. One such example is shown in Fig. 6b.

3.3.3 K-nearest neighbors

KNN is another popular algorithm used for recognizing patterns with the help of a training dataset. It always finds the kth close relative for this purpose. While using KNN for classification we always have to find a place for data to put it in its nearby neighbor's category (Lim 2004). For example, if the value of k is 1 then we have to place the data in a class that is near 1.

3.3.4 Decision tree

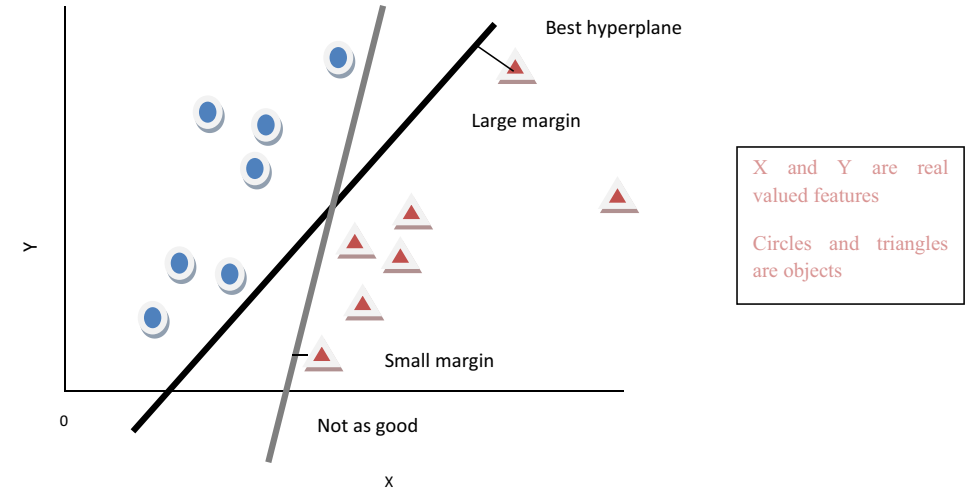
Another method used for text classification is a decision tree that can order the class on some level. The functionality of the decision tree is more similar to that of the flow chart; it separates the dataset into two categories of similar type i.e. from tree trunk to branches to leaves (D. E. Johnson. et al. 2002). It allows nested categorization without any supervision by a human. Figure 7 shows the example of a decision tree when applied to a sports dataset.

4 Literature review

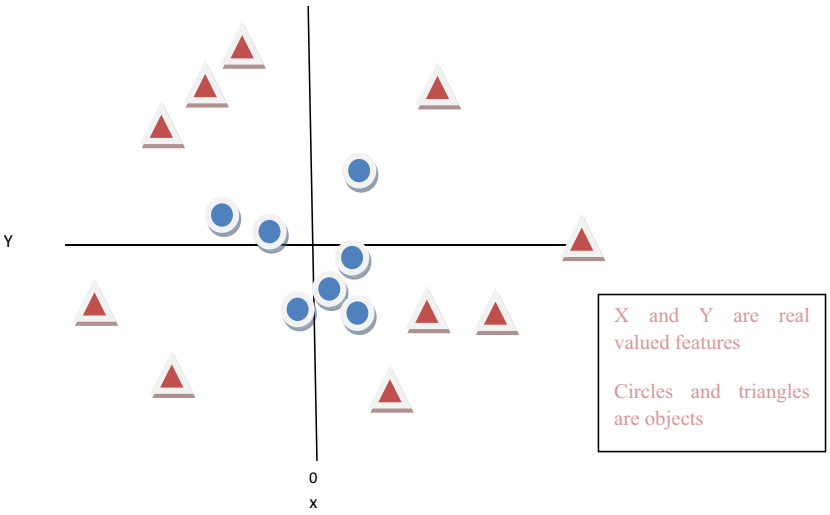
In this section a survey of related work in the field of DNA Sequence classification has been carried out, Table 2 shows the research work done by several authors and answers questions such as why there is a need for DNA sequence classification and what are the methods that can be used for classification purpose.

CNN has already proved its excellent performance in various fields and is now for DNA sequence classification. In a paper (Bosco. et al. 2016), where an author has compared CNN with other machine learning methods and the results proved that it also gives the best results when applied to

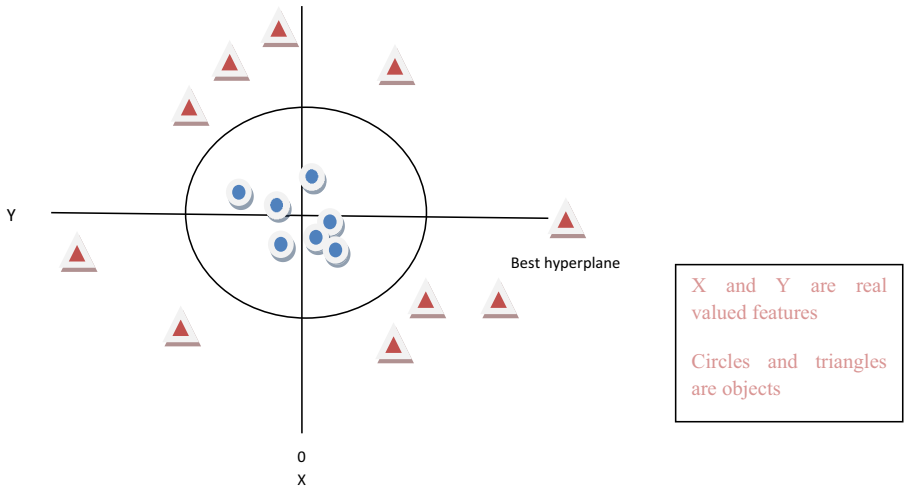
Fig. 5 **a** Hyperlane in 2D. **b** Classification of vector and tags. **c** Best hyperplane



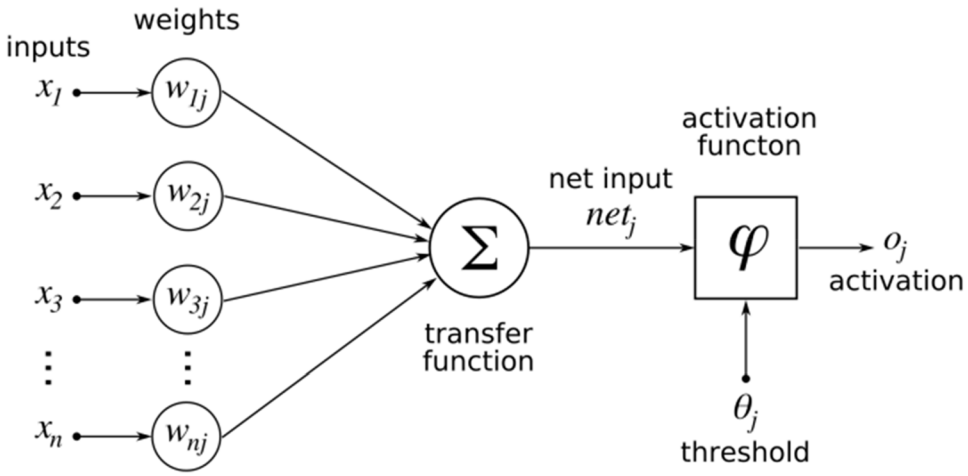
(a). Hyperlane in 2D



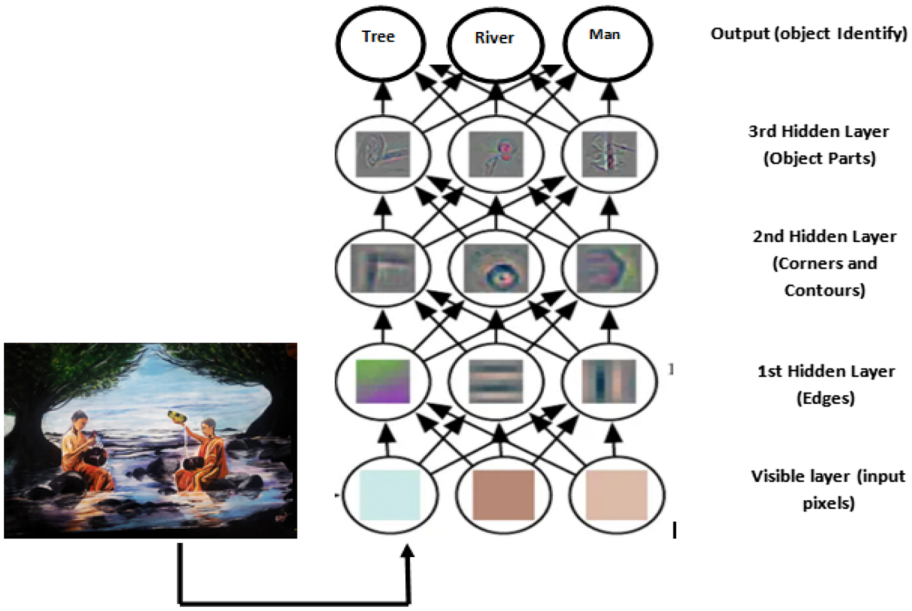
(b). Classification of vector and tags



(c). Best hyperplane



(a) Structure of an artificial neuron



(b) Feature extraction by CNN

Fig. 6 **a** Structure of an artificial neuron. **b** Feature extraction by CNN

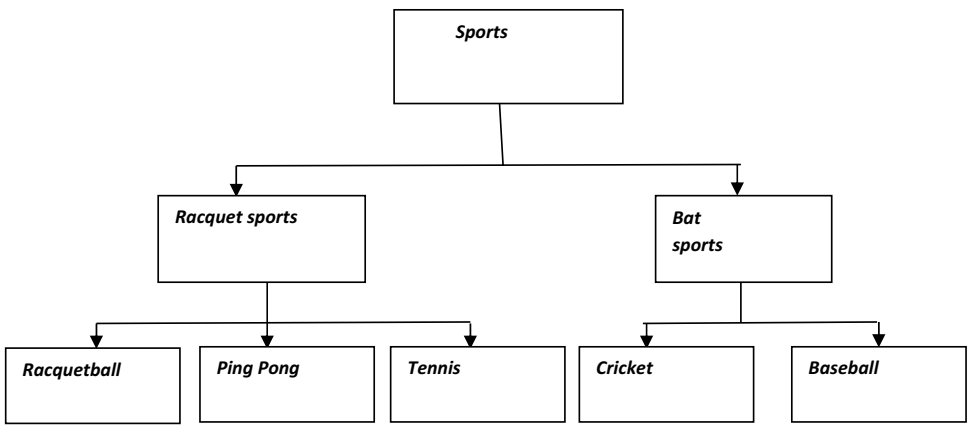


Fig. 7 Decision Tree

Table 2 The research was done in the field of DNA sequence classification

S. no.	Paper	Description
1	DNA sequence classification via an expectation–maximization algorithm and neural networks (Wang. et al. 1996)	In the paper, to recognize E. Coli promoters found in DNA and also to determine whether the given DNA sequence is E. Coli promoters or not a new technique has been introduced. This paper uses the EM algorithm for locating binding sites in the E. Coli promoter sequence which is better than previous algorithms as it reduces the Probability distribution of lengths. After locating the binding sites, feature selection will be done in each sequence according to the information content available and then represented it using the orthogonal encoding method. Finally, for promoter recognition, features are inserted into a neural network
2	Vector space classification of DNA sequences (Müller. et al. 2003)	This paper highlights the issues associated with the identification of intron and exon. For DNA sequence classification they use PCA. To represent word content, sequences are converted into document vectors. Finally, these word contents are used for the classification of sequences. This approach has been tested over many data sets of DNA for the classification of intron–exon and gives the highest accuracy as compared to other approaches
3	DNA sequence classification using DAWGs (Levy. et al. 1997)	Assigning words or substrings in a given sequence to generate a unique sequence class is said to be known as DNA sequence classification. To classify an unknown sequence, all its words are compared in the dictionary so that they can be represented in DNA's basic three classes. In this work, efficiency is increased by the construction of DAWGs i.e. directed acyclic word graphs. With this method, it is possible to identify 94% of a test set and only 4% of failures were there
4	Classification of gene expression data using fuzzy logic (Ohno-Machado et al. 2002)	Technologies like microarray have permitted the estimation of numerous qualities of gene expression levels at the same time. These generated levels can be further utilized for the classification of tissues into a prognostic category or diagnostic category. As estimations from various microarray innovations are made on various scales, it can be very useful to create a simple and easy-to-understand classification scheme that is technology-independent. This paper highlights how fuzzy logic can be useful to capture issues involved in the classification of gene expression data with 2 examples. However, in terms of classification performance, fuzzy inference performance is the same as that of other classifiers but it is simple and easy to understand
5	Complementary classification approaches for protein sequences (Wang et al. 1996)	In this paper, they have studied 5 methods of protein classification and have applied them to proteins that are related to each other and belong to the PROSITE catalog. Out of them, 4 methods are based on a search into the database (block-based approach) and the remaining are based on repeatedly appearing motifs present in a protein family. They have concluded that when talking about amino acids that occur in blocks, the block-based approach is considered the most suitable method, and using all these 5 methods can give a good classification result
6	Review on the Application of Machine Learning Algorithms in the Sequence Data Mining of DNA (Yang et al. 2020)	This paper focuses on the basic definition of DNA and the use of machine learning techniques for the mining of DNA sequences. They have also analyzed the basic concepts of data mining, various ML algorithms, and problem faced by them during sequence mining. A review has been done on how DNA sequencing technologies are growing exponentially, the structure of the sequence, and their similarity

Table 2 (continued)

S. no.	Paper	Description
7	DNA sequence classification by convolutional neural network (Nguyen et al. 2016)	In this work, a convolution neural network that uses a convolution layer for extracting features from given input data is used. The previous layer's extracted features are used by the convolution layer's neurons for extracting high-level abstraction features. In this method, DNA data is treated as text and then CNN is applied to them, just like CNN is applied to the text. They have used twelve datasets of DNA sequences and proved that CNN provides the best solution for solving a sequence classification problem
8	Deep learning architectures for DNA sequence classification (Bosco et al. 2016)	Generic computation is used for medical-related data analysis, DNA classification is considered one of the important tasks and ML (machine learning) techniques are successful in doing this task. However, the problem that still exists is feature selection. Machine learning methods highly depends on feature selection but the selection of meaningful feature is another important task. Deep learning models have already proven themselves in extracting useful features from given input patterns. This work highlights 5 different classification methods done on public DNA sequence data and also introduces 2 deep learning models
9	A neural network-based multi-classifier system for gene identification in DNA sequences (Ranawana et al. 2005)	Intending to identify promoter sequence (E.coli), this article proposes a multi-classifier system that is based on a neural network. This is because before every gene there is a promoter seq. so, for successful identification of DNA sequence gene, it is important to locate the E.coli promoter. This multi classifier system has been tested over different promoter as well as non-promoter sequences and the result shows that it gives the best prediction than other systems that have been developed so far

nucleotides. Therefore, in this work for feature extraction and classification CNN is used as a basic model.

5 Model

CNN can be used for various textual data problems such as for classification and categorization purposes, but the only difference is that digital pictures are 2-dimensional matrixes whereas textual data is a 1-dimensional form consisting of letters. That's why we need to convert it to its equivalent numerical value with the goal that it can be now given as input to CNN. A lookup table is maintained to match each letter with its numerical value known as a word vector. However using the lookup table have some limitation, so one new approach of using hot vectors to represent words has been proposed. Figure 8 shows an example, where dictionary D has some words with their hot vector representation.

Now suppose we have 5 words in a dictionary I; car; bike; bought; a; with their respective hot vectors in the 2 D matrix. To represent the sentence "I bought a car" in a 2 D numerical matrix we take a region of size two and write a hot vector for each pair of words. After converting all words into hot vectors we got a 2-dimensional matrix that can be directly given as input to traditional CNN for further classification. This model is well known as the CNN model for sequences or sequential convolutional neural networks.

5.1 Proposed method

As we know FASTA based DNA sequences are in textual format, but they are the sequences of continuous letters with no space, which means there is no concept of word in it. That's why we need to convert the sequences into words so it can work the same as CNN on text. The conversion is done in such a way that each nucleotide holds its position in the DNA sequence. This concept is better understood with the help of the below figure. Here we are using a fixed size window of 4 which slides with fixed strides 1. Every time a window reads some continuous sequence of DNA and considered it as a word. Finally, we have derived sequences of words from DNA data. Now textual-based CNN technique can be applied to this data.

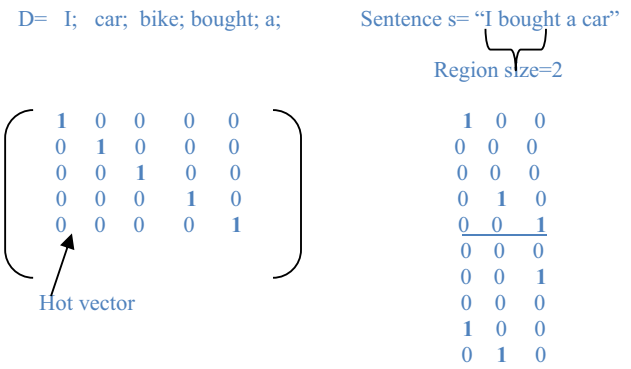
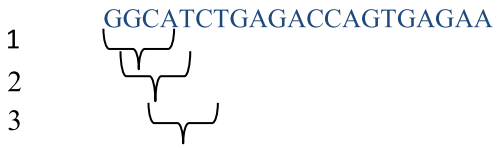


Fig. 8 Hot vector representation of words

As we know that in the FASTA DNA file format there are only 4 letters i.e. A, C, G, and T (Garima et al. 2020, 2021a, b). If the chosen window size is 4 then we have …256 distinct words in a dictionary. That means each word can be represented by using a 256-size hot vector. Finally, from this generated words of DNA sequence, a 2 D matrix has been generated where the position of every nucleotide is maintained, the resultant matrix is now given as an input to CNN for classification purposes.



In the above example every time 4 consecutive words are picked with the slide of stride 1, such as GGCA, GCAT, CATC, ATCT …and added to the destination sequence. Now we have a dictionary of 256 words (4^4) and each word can be represented using its corresponding hot vector to make a 2 D matrix of numerical values (Fig. 9).

Now using this dictionary hot vector is generated for given DNA sequences (Fig. 10).

5.2 Proposed framework

Earlier identification of deadly viruses like corona can prevent the outbreaks that we are facing currently as well as can help in the earlier designing of drugs. This is the reason why the classification of DNA sequences plays an important role in computational biology. In this paper, a framework has been proposed in which whenever a patient (infected from any virus or any other disease) visits a doctor his/her DNA samples are collected and matched with the database

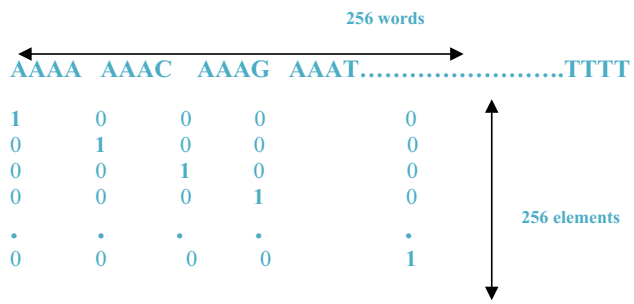


Fig. 9 Dictionary of words

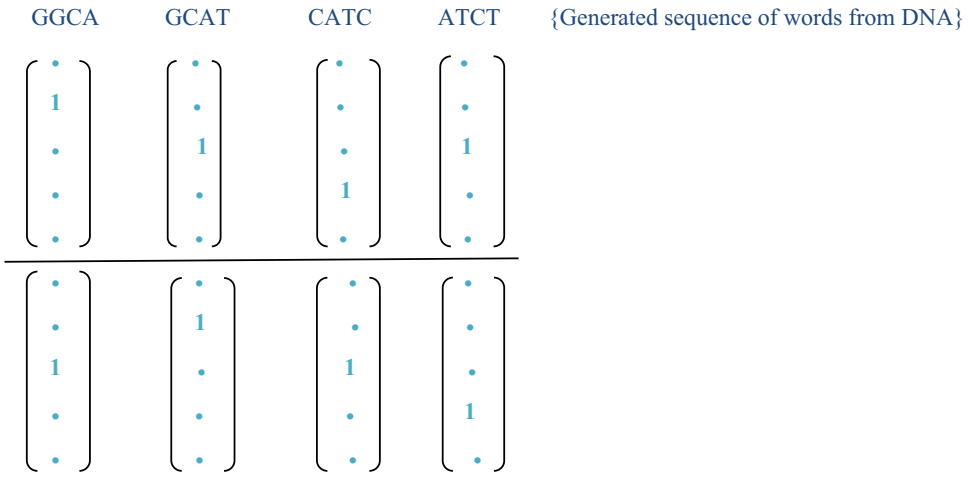


Fig. 10 Hot vector representation of FASTA DNA sequence

to identify the occurrence of the disease. However, due to the shortage of patterns available for various diseases, it becomes difficult to identify them. This paper has presented a solution in which samples collected from NCBI are used for the classification of DNA sequences. DNA sequence classification will in turn gives the pattern of various diseases; these patterns are then compared with the samples of a newly infected person and can help in the earlier identification of disease. As we know the size of FASTA-based DNA sequences is too big and complex which means this data cannot be given directly for feature extraction. For this data needed to be converted to some equivalent numerical form, in this paper a new hot vector-based numerical representation is introduced, where the position of each nucleotide is reserved by using binary 0 or 1. This hot vector matrix is then given as an input to traditional CNN for feature extraction. The models are then trained and evaluated on test data. Finally, the model can be applied to unseen data (Unknown DNA sequence) for disease prediction (Fig. 11).

6 Result analysis and discussion

For evaluation purposes, we have used a python based tool that can perform feature extraction as well as classification of DNA sequences. The overall architecture used for result analysis is shown in Fig. 12. In this work, we have compared 5 well-known classifiers namely Convolution neural network (CNN), Support Vector Machines (SVM), K-Nearest Neighbor (KNN) algorithm, Decision Trees, and Recurrent Neural Networks (RNN) on several parameters. The Kmer feature descriptor with a Kmer size of 3, the K-means method for

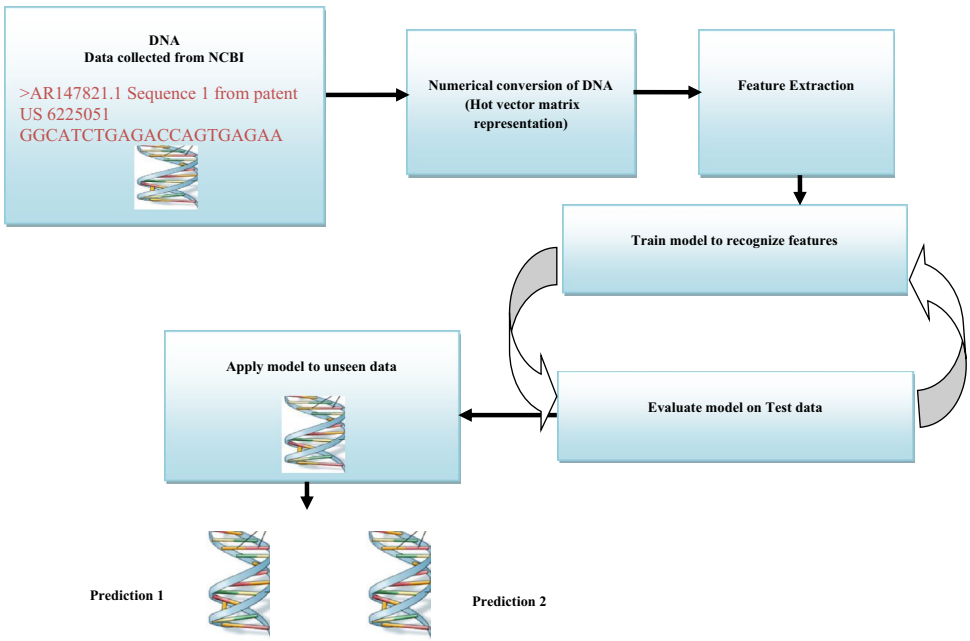


Fig. 11 Proposed framework

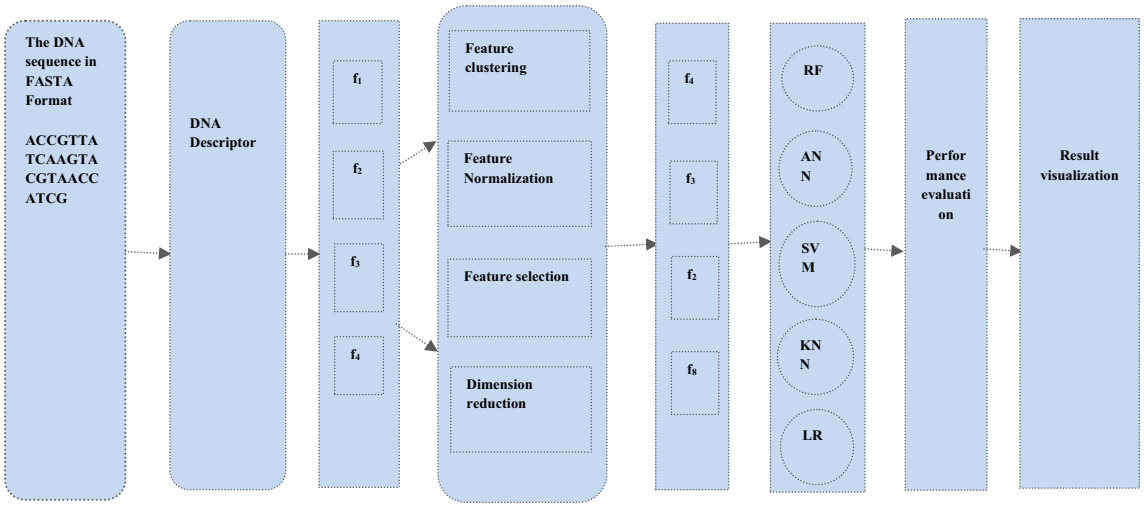


Fig. 12 The overall architecture used in result visualization

clustering of the dataset where the number of clusters is 3, for feature normalization Zscore method has been used, 100 features using the Chi-square selection method have been selected. The calculation and comparison of 7 different parameters, including sensitivity (Sn), specificity (Sp), accuracy (Acc), Matthews correlation coefficient (MCC), Recall, Precision, F1-score, the area under the receiver operating characteristic (ROC) curve (AUROC) and the area under the Precision-Recall (PRC) curve (AUPRC) (Liu 2017; Song et al. 2018; Liu et al. 2016, 2018; Li et al. 2018), has been done.

They are defined as

$$\text{Sensitivity} = 1 - \frac{\text{false negatives}}{\text{true positives}} \tag{1}$$

$$\text{Specificity} = 1 - \frac{\text{false positives}}{\text{true negatives}} \tag{2}$$

$$\text{Accuracy} = 1 - \frac{\text{false negatives} + \text{false positives}}{\text{true positives} + \text{true negatives}} \tag{3}$$

$$\text{Matthews correlation coefficient} = \frac{\text{Accuracy}}{\sqrt{\left[1 + \frac{\text{false positives} - \text{false negative}}{\text{true Positives}}\right]\left[1 + \frac{\text{false negatives} - \text{false positive}}{\text{true Negatives}}\right]}} \tag{4}$$

$$\text{Precision} = 1 - \frac{\text{false positives}}{\text{true positives} + \text{false positives}} \tag{5}$$

$$\text{F1 - score} = 1 - \frac{\text{false positives} + \text{false negatives}}{2 * \text{true positives} + \text{false positives} + \text{false negatives}} \tag{6}$$

Here, true positive, true negative, false positive, and false negative terms are defined as follows. The correct positive (+ ve) outcome predictions done by the model are termed "true + ve". The correct negative (-ve) outcome predictions done by the model is termed "true -ve". The incorrect positive (+ ve) outcome predictions done by the model are termed "false + ve". The incorrect negative (-ve)) outcome predictions done by the model are termed as "false -ve".

In the case of multiclass classification accuracy is defined as:

$$\text{Accuracy} = 1 - \frac{(\text{false negatives(i)} + \text{false positives(i)})}{(\text{true positives(i)} + \text{true negatives(i)})} \tag{7}$$

Here, false negatives(i), false positives(i),true positives(i), true negatives(i) represents the samples in ith class. The Naive Bayes algorithm is a popularly used multiclass algorithm. The AUROC value is calculated based on the ROC curve, and takes values between 0 and 1, while the AUPRC value is calculated based on the precision-recall curve. We have compared our model with other test data classification approaches based on these parameters. For estimation and comparison of the proposed model with other machine learning models k- fold cross-validation is used, where k = 4.

Table 3 The evaluation result of CNN

	Parameters	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	Average
CNN	Sn	60.0	60.0	45.0	45.0	50.0	52.0
	Sp	95.0	95.0	90.0	95.0	100.0	95.0
	Pre	92.31	92.31	81.82	90.0	100.0	91.288
	Acc	77.5	77.5	67.5	70.0	75.0	73.5
	MCC	0.5871	0.5871	0.3919	0.4619	0.5774	0.5211
	F1	0.7273	0.7273	0.5806	0.6	0.6667	0.6604
	AUROC	0.7575	0.8525	0.8375	0.765	0.795	0.8015
	AUPRC	0.827	0.8702	0.836	0.7893	0.8574	0.836

Table 4 The evaluation result of the Decision Tree

	Parameters	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	Average
Decision tree	Sn	50.0	70.0	55.0	70.0	65.0	62.0
	Sp	55.0	65.0	70.0	60.0	65.0	63.0
	Pre	52.63	66.67	64.71	63.64	65.0	62.53
	Acc	52.5	67.5	62.5	65.0	65.0	62.5
	MCC	0.0501	0.3504	0.2529	0.3015	0.3	0.251
	F1	0.5128	0.6829	0.5946	0.6667	0.65	0.6214
	AUROC	0.525	0.675	0.625	0.65	0.65	0.625
	AUPRC	0.6382	0.7582	0.711	0.7432	0.7375	0.7176

Table 5 The evaluation result of MLP

	Parameters	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	Average
MLP	Sn	70.0	80.0	80.0	85.0	80.0	79.0
	Sp	90.0	70.0	95.0	55.0	75.0	77.0
	Pre	87.5	72.73	94.12	65.38	76.19	79.184
	Acc	80.0	75.0	87.5	70.0	77.5	78.0
	MCC	0.6124	0.5025	0.7586	0.4193	0.5507	0.5687
	F1	0.7778	0.7619	0.8649	0.7391	0.7805	0.7848
	AUROC	0.8075	0.845	0.8975	0.755	0.8225	0.8255
	AUPRC	0.8183	0.8608	0.9232	0.7375	0.8339	0.8347

Table 6 The evaluation result of the RNN

	Parameters	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	Average
RNN	Sn	45.0	60.0	55.0	30.0	25.0	43.0
	Sp	95.0	95.0	85.0	100.0	100.0	95.0
	Pre	90.0	92.31	78.57	100.0	100.0	92.176
	Acc	70.0	77.5	70.0	65.0	62.5	69.0
	MCC	0.4619	0.5871	0.4193	0.4201	0.378	0.4533
	F1	0.6	0.7273	0.6471	0.4615	0.4	0.5672
	AUROC	0.815	0.885	0.8975	0.785	0.7575	0.828
	AUPRC	0.8662	0.8647	0.8864	0.8125	0.7998	0.8459

6.1 Comparison of evaluation metrics

From the above-calculated result, CNN gives an accuracy of 73.5, Decision tree with an average accuracy of 62.5, MLP with 78.0, RNN with 69.0, SVM with 50.0, and proposed method with 93.9, so we can say that the proposed method gives the highest accuracy as compared to other classification methods (Tables 3, 4, 5, 6, 7, 8).

Table 7 The evaluation result of the SVM

	Parametrs	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	Average
SVM	Sn	0.0	0.0	0.0	0.0	0.0	0.0
	Sp	100.0	100.0	100.0	100.0	100.0	100.0
	Pre	0.0	0.0	0.0	0.0	0.0	0.0
	Acc	50.0	50.0	50.0	50.0	50.0	50.0
	MCC	0.0	0.0	0.0	0.0	0.0	0.0
	F1	0.0	0.0	0.0	0.0	0.0	0.0
	AUROC	0.1225	0.115	0.1	0.2475	0.195	0.156
	AUPRC	0.324	0.3222	0.3203	0.3607	0.3427	0.334

Table 8 The evaluation result of the Proposed Method

	Parameters	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	Average
Proposed method	Sn	60.0	60.0	45.0	45.0	50.0	52.0
	Sp	95.0	95.0	90.0	95.0	100.0	95.0
	Pre	92.31	92.31	81.82	100.0	100.0	93.288
	Acc	93.5	92.5	97.5	91.0	95.0	93.9
	MCC	0.5871	0.5871	0.3919	0.4619	0.5774	0.5211
	F1	0.7273	0.7273	0.5806	0.6	0.6667	0.6604
	AUROC	0.7575	0.8525	0.8375	0.765	0.795	0.8015
	AUPRC	0.827	0.8702	0.836	0.7893	0.8574	0.836

6.2 Clustering

As per the final updated population of the cluster center session, the proposed work can use these sessions only for training the neural network. Here, this will increase the learning capacity of the work. As more patterns from the same class increase the confusion of the system. This can be said as small input learning cluster training data improving the detection rate of the proposed work. Clustering results of all given classification algorithm along with the proposed algorithm is shown below (Fig. 13).

Table 9 shows that the proposed method gives a significant improvement over the previous best results in terms of both precision and accuracy. The improvement in precision is nearly 1.112% and the improvement in terms of accuracy is about 15.9%. The proposed method's improvement is very high as compared to other methods because we have used an improved method for sequence representation (hot-vector representation) and also we have chosen the best feature selection method. The results have proved that features extracted by the proposed method are very useful for classifying the sequence into a true category (Fig. 14).

7 Conclusion

After the covid-19 outbreak, earlier identification of disease is the willingness of every person as their earlier identification can save more lives. Various ML (machine learning) techniques are there that can be used for classification purposes. This paper has presented a solution in which samples collected from NCBI are used for the classification of DNA sequences. DNA sequence classification will in turn gives the pattern of various diseases; these patterns are then compared with the samples of a newly infected person and can help in the earlier identification of disease. However, the size of these DNA sequences is too big and complex which means this data cannot be given directly for feature extraction and needed to be converted to some equivalent numerical form. In this paper a new hot vector-based numerical representation is introduced, where the position of each nucleotide is reserved by using binary 0 or 1. This hot vector matrix is then given as an input to traditional CNN for feature extraction. A comparison of about 7 classifiers like Convolution neural network (CNN), Support Vector Machines (SVM), K-Nearest Neighbor (KNN) algorithm, Decision Trees, Artificial Neural Networks (ANN), and proposed method on 7 different parameters, including sensitivity, specificity, accuracy, Matthews correlation coefficient,

Table 9 Comparison with the previous best results

Parameters	Previous best result	Av (proposed)	Increased
Precision	92.176	93.288	1.112
Accuracy	78.0	93.9	15.9

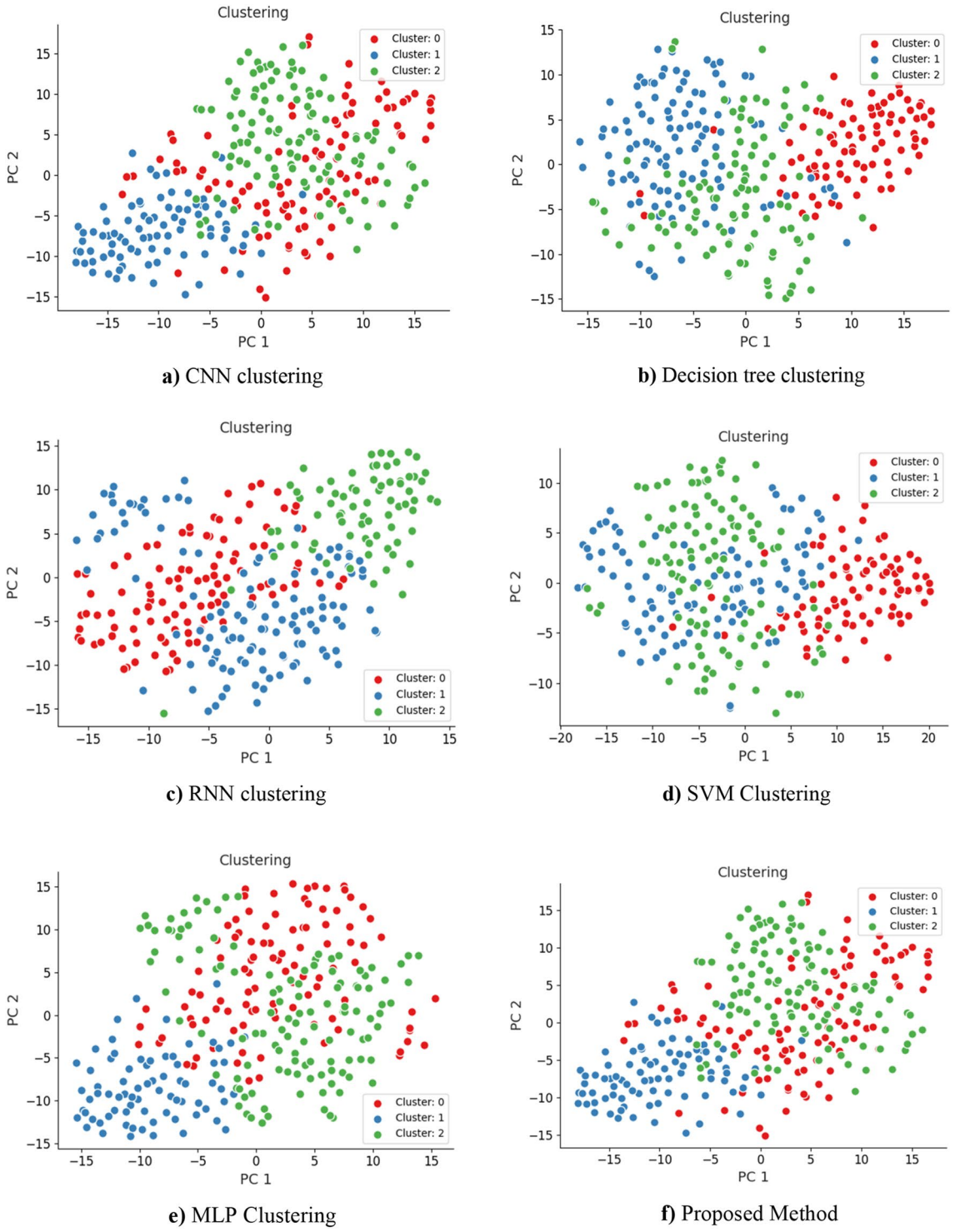


Fig. 13 **a** CNN clustering. **b** Decision tree clustering. **c** RNN clustering. **d** SVM Clustering. **e** MLP clustering. **f** Proposed method

Recall, Precision, F1-score, the area under the receiver operating characteristic (ROC) curve (AUROC), and the area under the Precision-Recall (PRC) curve (AUPRC), has been done and the result shows that proposed method gives highest accuracy of 93.9%, which is highest compared to other classifiers.

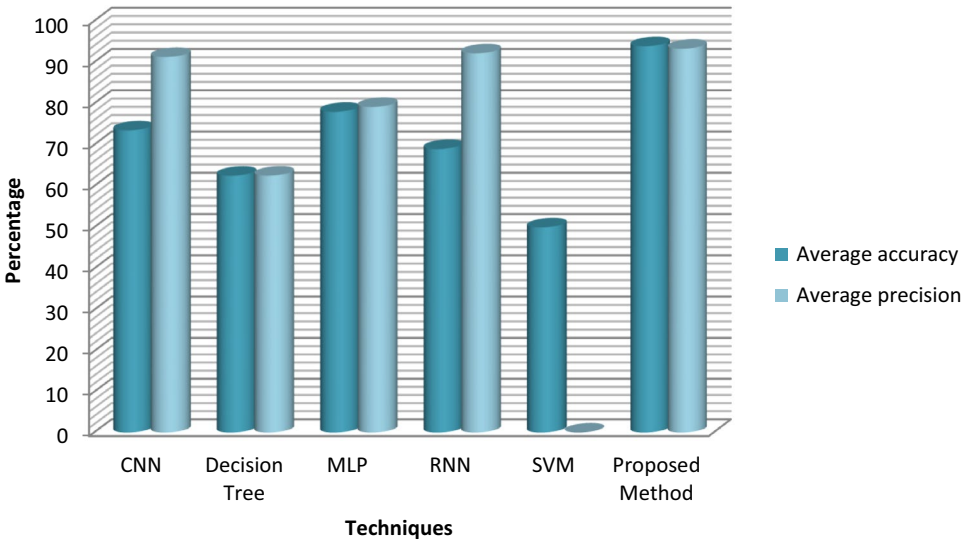


Fig. 14 Average accuracy comparison of classification techniques

Data availability The datasets generated during and/or analyzed during the current study are not publicly available due to [security reasons] but are available from the corresponding author on reasonable request.

References

Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Sayers EW (2010) GenBank. Nucleic Acids Research. vol. 38. Supplement 1:46–51

Bosco GL, Di Gangi MA (2016) Deep learning architectures for DNA sequence classification. In: Proceedings of the international workshop on fuzzy logic and applications. Springer, Cham, pp 162–171. https://doi.org/10.1007/978-3-319-52962-2_14

Garima M, Anjana P, Sachin G (2020) Immutable DNA sequence data transmission for next-generation bioinformatics using blockchain technology. In: 2nd international conference on data, engineering, and applications (IDEA)

Garima M, Anjana P, Sachin G (2021a) An approach to compress human genome sequence by delta computation and secure storage by Blockchain. DE. Pp 7130–7144

Garima M, Anjana P, Sachin G (2021b) Blockchain-based healthcare information exchange systems for the security of healthcare data. Turk Online J Qual Inquiry (TOJQI) 12(8):4498–4507

Hach F, Numanagic I, Sahinalp SCD (2014) Reference-based compression by local assembly. Nat Methods 11:1082–1084

Herath HMKKMB, Karunasena GMKB, Madhusanka BGDA, Priyankara HDNS (2021a). Internet of medical things (IoMT) enabled TeleCOVID system for diagnosis of COVID-19 patients. In: Agrawal R, Mittal M, Goyal LM (eds) Sustainability measures for COVID-19 pandemic. Springer, Singapore

Herath HMKKMB, Karunasena GMKB, Herath HMWT (2021b) Development of an IoT based systems to mitigate the impact of COVID-19 pandemic in smart cities. In: Ghosh U, Maleh Y, Alazab M, Pathan ASK (eds) Machine intelligence and data analytics for sustainable future smart cities. Studies in Computational Intelligence, vol 971. Springer, Cham

Herath HMKKMB, Karunasena GMKB, Madhusanka BGDA (2022) Early detection of COVID-19 pneumonia based on ground-glass opacity (GGO) features of computerized tomography (CT) angiography. 5G IoT and Edge Computing for Smart Healthcare Intelligent Data-Centric Systems, pp 257–277

https://monkeylearn.com/blog/classification-algorithms/

Ikonomakis M, Kotsiantis S, Tampakas V (2005) Text classification using machine learning techniques. WSEAS Trans Comput 4(8):966–974

Jain AK, Duin RPW (2004) Introduction to pattern recognition. In: The Oxford companion to the mind, second edition, Oxford University Press, Oxford, UK, pp 698–703

Johnson DE, Oles FJ, Zhang T, Goetz T (2002) A decision-tree-based symbolic rule induction system for text categorization. IBM Syst J

Kassim NA, Abdullah A (2017) Classification of DNA sequences using convolutional neural network approach. UTM Comput Proc Innov Comput Technol Appl 2:1–6

Levy S, Stormo GD (1997) DNA sequence classification using DAWGs. Struct Logic Comput Sci. https://doi.org/10.1007/3-540-63246-8_21

Li F, Li C, Marquez-Lago TT et al (2018) A comprehensive tool for rapid and accurate prediction of kinase family-specific phosphorylation sites in the human proteome. Bioinformatics 34:4223–4231

Lim H (2004) Improving kNN based text classification with well estimated parameters. LNCS 3316:516–523

Liu B, Fang L, Long R et al (2016) A two-layer predictor for identifying enhancers and their strength by pseudo k tuple nucleotide composition. Bioinformatics 32:362–369

Liu B, Yang F, Huang DS et al (2018) A two-layer predictor for identifying promoters and their types by multi-window-based PseKNC. Bioinformatics 34:33–40

Liu B (2017) BioSeq-analysis: a platform for DNA, RNA and protein sequence analysis based on machine learning approaches. Brief Bioinform. https://doi.org/10.1093/bib/bbx165

Ma Q, Wang JTL, Shasha D, Wu CH (2001) DNA sequence classification via an expectation maximization algorithm and neural networks: a case study. IEEE Trans Syst 31:468–475. https://doi.org/10.1109/5326.983930

Mohammed MH, Dutta A, Bost T, Chadaram S (2012) DELIMINATE—A fast and efficient method for lossless compression of genomic sequences. Bioinformatics 28:2527–2529

Momenzadeh M, Sehhati M, Rabbani H (2020) Using hidden Markov model to predict recurrence of breast cancer based on sequential patterns in gene expression profiles. J Biomed Inf 111

Müller HM, Koonin SE (2003) Vector space classification of DNA sequences. J Theor Biol 223:161–169. https://doi.org/10.1016/S0022-5193(03)00082-1

Nguyen N, Tran V, Ngo D, Phan D, Lumbanraja F, Faisal M, Abapihi B, Kubo M, Satou K (2016) DNA sequence classification by

convolutional neural network. J Biomed Sci Eng 9:280–286. https://doi.org/10.4236/jbise.2016.95021

Ohno-Machado L, Vinterbo S, Weber G (2002) Classification of gene expression data using fuzzy logic. J Intell Fuzzy Syst 12(1):19–24

Ranawana R, Palade V (2005) A neural network-based multi-classifier system for gene identification in DNA sequences. Neural Comput Appl 14:122–131. https://doi.org/10.1007/s00521-004-0447-7

Sathish kumar S, Duraipandian N (2005) Int J Comput Technol 4(2c2):722–730. https://doi.org/10.24297/ijct.v4i2c2.4190

Shadab S, Alam Khan MT, Neezi NA, Adilina S, Shatabda S (2020) DeepDBP: deep neural networks for the identification of DNA-binding proteins. Inf Med Unlock 19:100318

Shanahan J, Roma N (2003) Improving SVM text classification performance through threshold adjustment. LNAI 2837:361–372

Song J, Li F, Takemoto K et al (2018) an integrative approach for inferring catalytic residues using sequence, structural, and network features in a machine-learning framework. J Theor Biol 443:125–137

Wang JTL, Marr TG, Shasha D, Shapiro BA, Chirn G, Lee TY (1996) Complementary classification approaches for protein sequences. Protein Eng 9(5):381–386

Yang A, Zhang W, Wang J, Yang K, Han Y, Zhang L (2020) Review on the application of machine learning algorithms in the sequence data mining of DNA. Front Bioeng Biotechnol. https://doi.org/10.3389/fbioe.2020.01032