Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Exploring Explainable AI Technique SHAP and RNN: For predicting Promotor Region

G. Aruna Arumugam[1], Dr. M. Mohamed Divan Masood[2*]

[1]*Research Scholar, Department of Computer Applications, B.S.Abdur Rahman Crescent Institute of Science and Technology, Vandalur, Tamilnadu, India, arunanet23@gmail.com*
[2]*Assistant Professor, Department of Computer Applications, B.S.Abdur Rahman Crescent Institute of Science and Technology, Vandalur, Tamilnadu, India, divan@crescent.education*

A critical DNA segment found upstream of a gene in humans and other eukaryotes is recognized as the promoter region. Regulatory the expression of a gene requires it. The promoter region is the area of a gene that is located upstream of the transcription start site (TSS). It reaches between 20 and 200 base pairs upstream of the TSS. Finding genetic variants or mutations inside promoter regions that are associated with a disease can be a useful method in medical genetics and disease research. In this work, we evolved an explainable AI model SHAP for feature extraction and trained the model using a Recurrent neural network for classifying promotor regions. By comparing with other algorithms like support vector machine, Random Forest, and Naive Bayse our method can achieve an improved performance on promotor or non-promotor classification. This work yielded profound accurate identification of promoter areas and offered analysis for both precision medicine and further biological research.

Keywords: DNA-Protein interaction, Transcription Factor Binding Site (TFBS), SHAP, Recurrent Neural Network (RNN), Precision Medicine.

1. Introduction

A vital regulatory factor found in the upstream section of genes, DNA promoters are found in several species, including humans. In directive to start transcription the process by which a gene's information is translated from RNA to protein the promoter region must have specific sequences and other elements. Promoters serve as binding sites for various proteins and enzymes that control the transcriptional onset. This method uses a gene's information to produce a complementary RNA molecule. After that, this RNA molecule might be used as a template to make a protein. The complementary strand is referred to as the "antisense" strand, and the DNA strand that is transcribed as the "sense" strand[1].

Promoters are usually made up of different components and sequences that have different functions when it comes to starting transcription. In eukaryotic promoters, three essential components are frequently present. 1. Core promotors (Transcription Start Site (TSS)-

CCAAT Box (CCAAT or ATTGG)), TATA Box- TATAAA/TATAAT), 2. Proximal Promoter Elements (CAAT Box- (CCAAT/ GGCCAATCT), GC Box- (GGGCGG/GGGCGGGG)) 3. Enhancer Elements (E Boxes)- (CACGTG/ CANNTG).[2] Together, these components govern transcription initiation, which in turn controls gene expression. The combinations of promoter elements found in various genes can vary, and the timing and intensity of gene expression can be affected by the presence or lack of certain components. Promoters can have a wide range of specialised components, depending on the gene and the organism[3]. It is necessary to comprehend the composition and role of promoters in order to make sense of the mechanisms underlying gene regulation in eukaryotic cells. Scholars continue to explore the intricate mechanisms behind promoter activity to gain further insights into genomic functioning and the potential for therapeutic interventions.

The term "core promoter" refers to the section of DNA that regulates the initiation of transcription. The general transcription factor initiates the recruitment of RNA poly II, which in turn initiates the commencement of transcription, by binding to these core promotor regions. Upstream promotor elements, such as the TATA box and BRE, are found in the region of DNA that is upstream of the transcription start site (TSS), or towards the 5' end of the molecule. Promotor elements like DCEI and DPE that are downstream of the transcription start site (TSS) are located in the downstream (or 3' end) region of the DNA as shown in the figure 1.

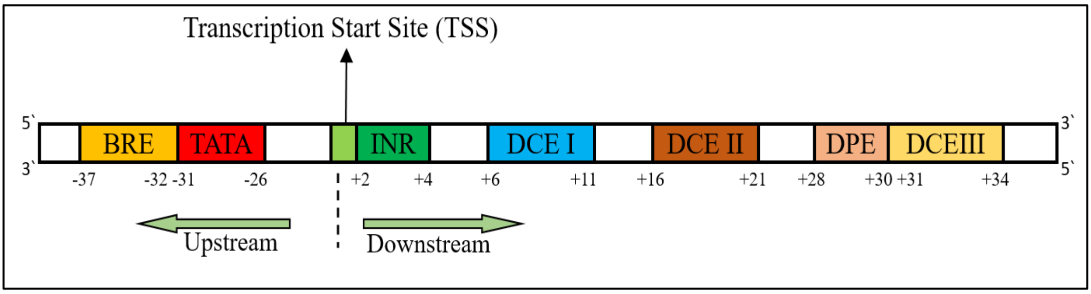


Figure 1. Upstream and downstream promotor region

Identifying the promotor region could involve the subsequent actions. 1) Coordinates with the appropriate chromosome and gene locus, 2) It has a length of roughly base pairs (desired length), 3) Takes into account the most recent human genome reference sequence (e.g., GRCh38), 4) Minimises the presence of repetitive elements, and 5) Includes the core promoter elements, such as TATA box, CAAT box, and initiator elements.

2. Literature Survey

The DNA promoter regions comprehend binding sites for RNA polymerase, which is accountable for initiating transcription, and these regions are essential for gene expression. Detecting these regions precisely is vital for understanding the causes of diseases, gene regulatory mechanisms, and developing new treatment methods. Traditional computational techniques such as sequence alignment, motif identification, and statistical representations have been broadly used for predicting promoters in research[2]. However, these approaches often struggle to handle the difficulty and range of promoter sequences, especially when associating sequences from different species.

Deep learning has renovated the analysis of genomic data by providing powerful tools to understand the complex patterns within DNA sequences. Recurrent neural networks (RNNs), including Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRUs), have proven to be effective in representing sequential data due to their ability to capture long-range dependencies. In genomic sequence analysis, RNNs have been employed for many tasks, such as gene prediction, splice site identification, and the discovery of promoter regions[3]. These models can study the temporal correlations amongst nucleotides, which is crucial for precisely identifying promoter areas that are often categorized by specific sequence motifs and patterns.

The growing complexity and accuracy of deep learning models have made interpretability increasingly important, especially in sensitive fields like genetics, where understanding the decision-making processes of the models is critical. Explainable AI (XAI) techniques are focused on enhancing the transparency of decision-making in machine learning models, which is crucial for uncovering biological pathways and establishing trust in model predictions in genomics.

The combination of RNNs with XAI techniques such as SHAP offers a practical and comprehensible approach to predicting promoter regions. While RNNs effectively capture the intricate patterns and sequential relationships found in DNA sequences, SHAP allows researchers to analyze the predictions made by these models and identify the significant features. This merging of methods addresses a key obstacle in deep learning in genomics: the opaque nature of these models[4]. By examining RNN predictions at the nucleotide level using SHAP, researchers can determine the specific sequences that the model predicts as promoter areas. This improves the accuracy of forecasts and advances our understanding of the molecular processes that govern gene regulation.

In various research efforts, XAI methods have been explored in the field of genomics. DeepLIFT and LIME, for instance, have been utilized to interpret deep learning models for tasks such as predicting protein-DNA binding sites and enhancers. These techniques facilitate the identification of new regulatory motifs by visually representing each nucleotide's impact on the model's predictions. The fusion of RNNs and explainable AI methods like SHAP signifies a significant progress in genomic sequence analysis. This approach enhances the accuracy of predicting promoter regions and offers valuable insights into the biological processes, benefiting computational biologists and bioinformaticians. As this field evolves, the convergence of deep learning and XAI is expected to play a crucial role in unveiling novel genetic discoveries.

The promotor region's significance

With the promoter region, transcription starts. The initial step of molecular biology is the transcription. The information found in DNA is translated into messenger RNA (mRNA). A few DNA sequences located in the promoter region serve as binding sites for RNA polymerase, an enzyme that synthesises RNA from DNA templates, and gene transcription factors. The activity of the promoter region controls the quantity and timing of a gene's transcription [4]. The promoter is home to numerous regulatory elements, such as enhancers, silencers, and transcription factor binding sites, which regulate the enrolment of transcriptional machinery and the rate at which transcription initiation occurs [5].

Transcription factors and regulatory proteins recognise certain sequence motifs found in the discrete promoter region of each gene. Only specific genes are transcribed in a specific cell type or under a specific set of conditions as a result of this choosiness. Response elements, which are often located in promoter regions, allow genes to be activated or repressed in response to a variety of environmental stimuli, including hormones, nutrition, diseases, or shocks. Due to the varying activity of these regulatory components, cells are able to respond to changing physiological conditions and maintain balance by constantly altering their gene expression profiles [6].

Figure 2 shows the proximal control elements, core promotor region and the transcription start site (TSS).

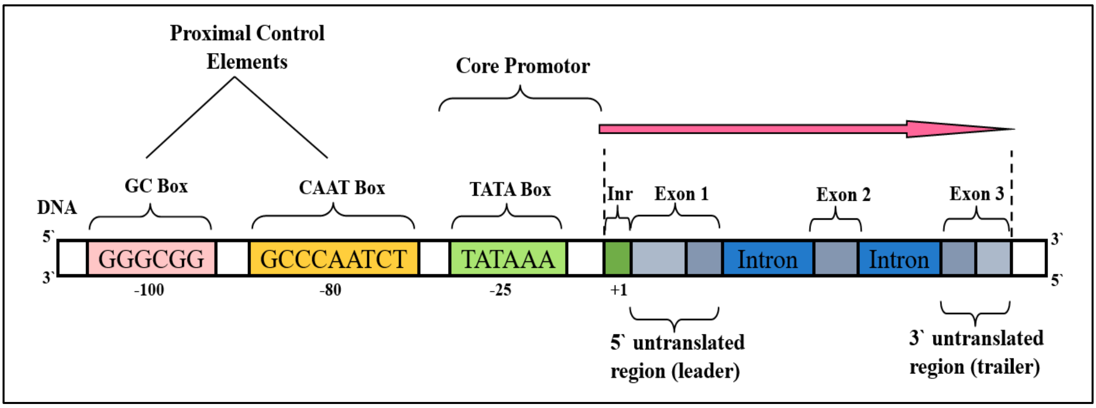


Figure 2. Core promotor region

i) The TATA Box

DNA has a base sequence called the TATA box, also recognized as the Goldberg-Hogness box, which instructs an RNA polymerase on where to begin transcription. Within the central promoter region of genes in eukaryotes, it is naturally found. Transcription commences at this area, which is critical because it serves as a binding site for transcription factors and RNA polymerase II. To recognize the TATA box sequence, the base sequence TATAAA is frequently utilised. But there is also a ration of variations within this series.

The TATA box can be found in many genes, while its exact location varies, usually 25–30 base pairs upstream from the transcription start site (TSS). The TATA box provides a binding site for transcription factors such as the TATA-binding protein (TBP), which is required for the beginning of transcription. TATA box-binding transcription factor IID (TFIID) complex includes TBP, which attracts RNA polymerase II to the promoter region. Not all genes have a TATA box in the promoter region, but when one does, transcription start effectiveness is usually increased [7]. Transcription initiation can occur without TATA in certain genes with alternative promoter sequences.

The TATA box is often associated with other regulatory elements and transcription factor binding sites in the promoter region. Together, these elements regulate the onset rate of transcription and ensure that genes are produced appropriately in response to various environmental and cellular inputs. The first step in determining the TATA box in a DNA sequence is typically to find the consensus TATAAA sequence at the appropriate distance upstream from the transcription start site.

ii) The CAAT Box

The conserved DNA sequence element known as the CAAT box, also referred to as the CCAAT box or Hogness box, is found in the promoter regions of eukaryotic genes. It is usually found upstream of the transcription start site. The nucleotide sequence CCAAT normally makes up the CAAT box sequence, though there may be very minor differences in the precise sequence. Transcription factors are proteins that bind to specific DNA sequences and control the transcription of neighbouring genes. The CAAT box acts as a recognition site for these proteins. Depending on the particular regulatory proteins involved, binding of transcription factors to the CAAT box can either enhance or repress gene transcription.

The CAAT box is an enhancer element in many genes, which means that it increases the rate at which transcription initiation occurs. Increased gene expression can result from transcription factors binding to the CAAT box, which in turn recruit’s other proteins involved in transcription initiation. Depending on the particular gene and cell type, the CAAT box's presence and significance can change. In certain cases, the CAAT box may be dispensable or its function may be modulated by other regulatory elements, but in other genes, a strong CAAT box is required for their expression[8]. The general motif of the CAAT box is conserved across a broad range of eukaryotic species, despite minor variations in the exact sequence between genes and organisms. Its significance in gene regulation is highlighted by this conservation, which also implies that it is a key player in regulating gene expression.

iii) The GC Box

The promoter regions of numerous eukaryotic genes contain a common DNA sequence element called the GC box, which is also referred to as the Sp1 binding site. GGGCGG or similar variations are the nucleotide sequence that normally makes up the GC box. Several guanine (G) and cytosine (C) bases are present in it, which is why it is called a "GC box." The GC box's principal role is to act as a binding site for transcription factors, especially those in the Sp1 family. The GC-rich sequence found in the target gene's promoter region is specifically recognised and bound to by the DNA-binding domains of these transcription factors.

Activating or repressive effects on gene expression can result from transcription factors binding to the GC box; this depends on the particular context and regulatory proteins involved. Transcription factors that bind to the GC box frequently function as transcriptional activators, encouraging the recruitment of RNA polymerase and other transcriptional machinery members to start the transcription of genes[9]. Many different genes have promoter regions that contain the GC box, and the presence of this region is frequently linked to constitutive or housekeeping gene expression. Zinc finger DNA-binding domains found in Sp1 proteins are specifically designed to recognise and bind to the GC-rich sequences found in gene promoters[10]. Sp1 proteins are widely expressed and involved in the control of gene expression, cell division, and proliferation. The presence of neighbouring DNA sequences, interactions with other transcription factors, and epigenetic modifications like DNA methylation can all affect the function of the GC box, despite it being a common regulatory element.

iv) The E-Boxes

E boxes, are especially DNA sequences that frequently appear in the gene promoter regions. A particular DNA sequence motif referred to as CANNTG, a six-nucleotide palindromic pattern where N may contain any nucleotide (A, T, C, or G), is frequently employed for recognizing E boxes. Basic helix-loop-helix (bHLH) proteins have transcription factors that notice and bind to this consensus sequence. The transcription factors bHLH gather into dimers and connect one another to the E-box sequences identified in the area of interest gene promoter regions[11]. These transcription factors possess two functional domains: a helix-loop-helix domain involved in protein splitting, and a basic location that binds to DNA, which is located at the E-box sequence.

Depending on the particular transcription factors involved and the target gene's context, the binding of bHLH transcription factors to E boxes can have either activating or repressive effects on gene expression. Certain bHLH proteins function as transcriptional activators, encouraging the recruitment of RNA polymerase and other elements of the transcriptional machinery to start the transcription of genes. In other situations, they might function as repressors, preventing the expression of a gene by obstructing the binding of transcription factors that activate it or by enlisting the help of corepressors. Numerous genes involved in basic biological processes, such as cell cycle regulation, metabolism, neurogenesis, myogenesis, and immune response, have E boxes in their promoter regions. A number of variables can affect the activity of E-box-mediated transcriptional regulation, such as the presence of bHLH transcription factors, changes made to these factors after they are translated, interactions with other transcriptional regulators, and epigenetic changes made to the chromatin environment around the E-box sequences[12].

Though not all promotor regions should have these motifs, the aforementioned boxes, and their motifs are present in the promotor region. The computational model used to identify promotor areas is described in depth in the following sections.

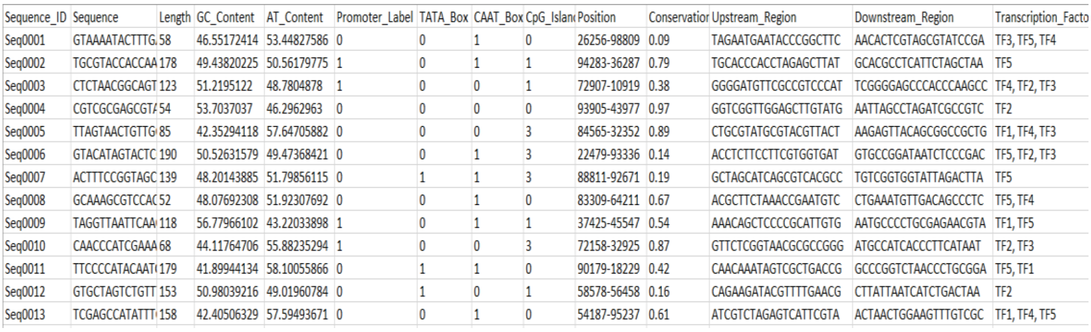
3. Material and Methods

Work flow of identifying DNA Sequence Promoter Region:

a) Benchmark Dataset:

Our goal in this study was to categorise the promoters (promoters or non-promoters) as well as their activity (strong or weak). The Eukaryotic Promoter Database-EPD (https://epd.expasy.org/epd/EPDnew_select.php) provided the training benchmark dataset. Each sequencing fragment in this dataset was split into 100 bp segments by the biological characteristics of DNA strands There were ~3000 promoters (among these 1629 strong promoter samples and 1371 weak promoter samples) and ~3000 non-promoters after duplicated sequences were removed and imbalance subgroups were chosen at random. The next stages of extracting features using SHAP and applying neural network models to the DNA sequence samples[13].

Table1. Promoter Region Identification Dataset



Sequence_ID	Sequence	Length	GC_Content	AT_Content	Promoter_Label	TATA_Box	CAAT_Box	CpG_Islan	Position	Conservatior	Upstream_Region	Downstream_Region	Transcription_Facto
Seq0001	GTAAAATACTTTG	58	46.55172414	53.44827586	0	0	1	0	26256-98809	0.09	TAGAATGAATACCCGGCTTC	AACACTCGTAGCGTATCCGA	TF3, TF5, TF4
Seq0002	TGCGTACCACCAA	178	49.43820225	50.56179775	1	0	1	1	94283-36287	0.79	TGCACCCACCTAGAGCTTAT	GCACGCCTCATTCTAGCTAA	TF5
Seq0003	CTCTAACGGCAGT	123	51.2195122	48.7804878	1	0	0	1	72907-10919	0.38	GGGGATGTTCGCCGTCCCAT	TCGGGGAGCCCACCCAAGCC	TF4, TF2, TF3
Seq0004	CGTCGCGAGCGT	54	53.7037037	46.2962963	0	0	0	0	93905-43977	0.97	GGTCGGTTGGAGCTTGTATG	AATTAGCCTAGATCGCCGTC	TF2
Seq0005	TTAGTAACTGTTG	85	42.35294118	57.64705882	0	0	0	3	84565-32352	0.89	CTGCGTATGCGTACGTTACT	AAGAGTTACAGCGGCCGCTG	TF1, TF4, TF3
Seq0006	GTACATAGTACTC	190	50.52631579	49.47368421	0	0	1	3	22479-93336	0.14	ACCTCTTCCTTCGTGGTGAT	GTGCCGGATAATCTCCCGAC	TF5, TF2, TF3
Seq0007	ACTTTCCGGTAGC	139	48.20143885	51.79856115	0	1	1	3	88811-92671	0.19	GCTAGCATCAGCGTCACGCC	TGTCGGTGGTATTAGACTTA	TF5
Seq0008	GCAAAGCGTCCAC	52	48.07692308	51.92307692	0	0	1	0	83309-64211	0.67	ACGCTTCTAAACCGAATGTC	CTGAAATGTTGACAGCCCTC	TF5, TF4
Seq0009	TAGGTTAATTCAA	118	56.77966102	43.22033898	1	0	1	1	37425-45547	0.54	AAACAGCTCCCCGCATTGTG	AATGCCCCTGCGAGAACGTA	TF1, TF5
Seq0010	CAACCCATCGAAA	68	44.11764706	55.88235294	1	0	0	3	72158-32925	0.87	GTTCTCGGTAACGCGCCGGG	ATGCCATCACCCTTCATAAT	TF2, TF3
Seq0011	TTCCCCATACAAT	179	41.89944134	58.10055866	0	1	1	0	90179-18229	0.42	CAACAAATAGTCGCTGACCG	GCCCGGTCTAACCCTGCGGA	TF5, TF1
Seq0012	GTGCTAGTCTGTT	153	50.98039216	49.01960784	0	1	0	1	58578-56458	0.16	CAGAAGATACGTTTTGAACG	CTTATTAATCATCTGACTAA	TF2
Seq0013	TCGAGCCATATTT	158	42.40506329	57.59493671	0	0	1	0	54187-95237	0.61	ATCGTCTAGAGTCATTCGTA	ACTAACTGGAAGTTTGTCGC	TF1, TF4, TF5

b) Feature Representation:

K-Mer counting technique is used to transform DNA sequences into numerical formats appropriate for use with machine learning algorithm[14]. In a high-dimensional space, represent each nucleotide (A, C, G, and T) in the promoter region as a vector or binary feature. K-Mer counting is used to convert a string of DNA sequences into an ordinal vector[15]. First, we divide the lengthy biological sequence into overlapping "words" of k mer length. Using "words" of length 3 (hexamers), for instance, “TATAATTAT” becomes “TATAAT”, “ATAATT”, “TAATTA”, “AATTAT”. Thus, there are 6 hexamer words with vector representation in our example sequence as shown below.

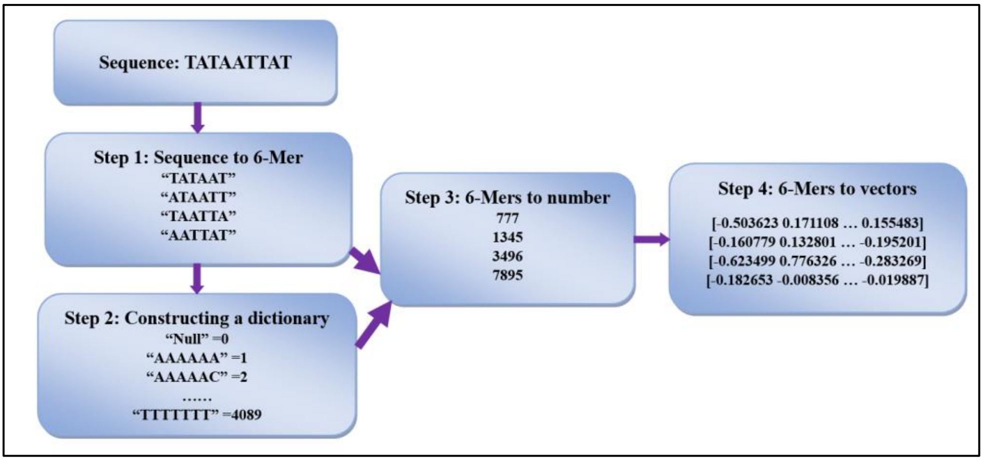


Figure 3. Example of sequence encoding - Generation of frequency-based vector representation

The following is a discussion of how to locate the promoter region of a DNA sequence using SHAP and RNN:

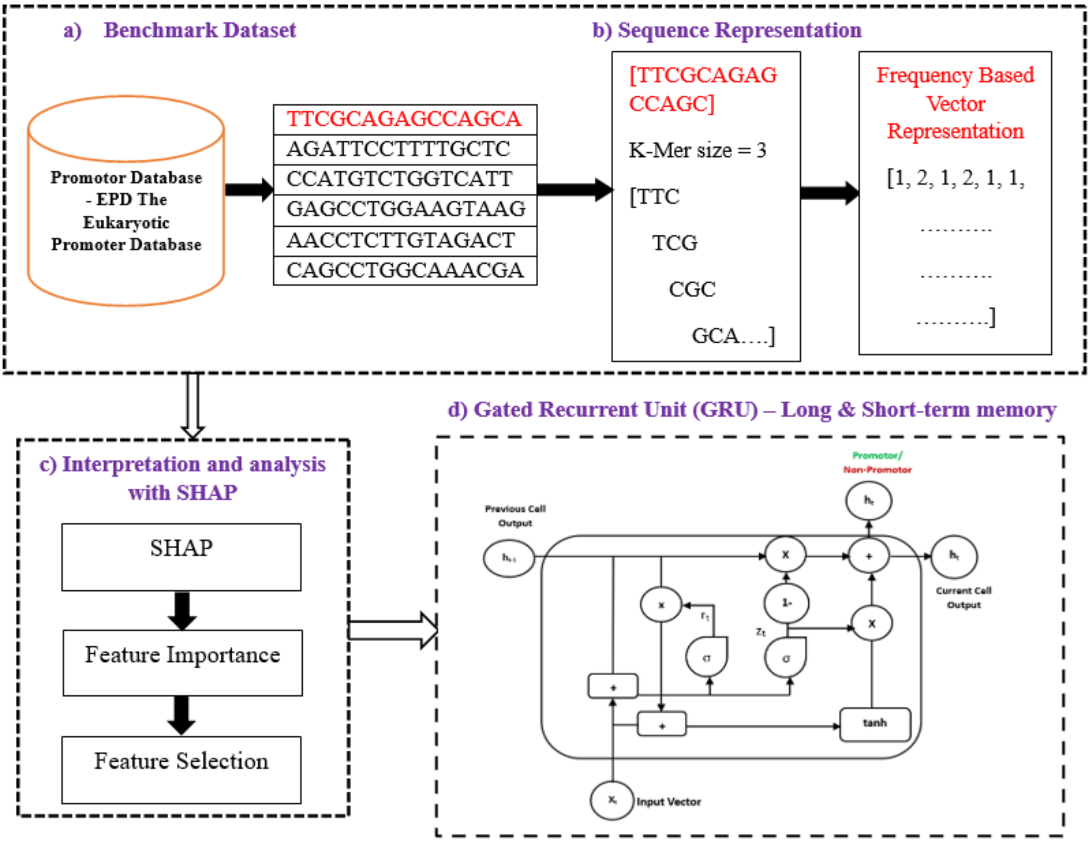


Figure 4. The pipeline for the entire framework.

c) SHAP Values Computation:

One effective technique for explaining machine learning model output is the SHAP (SHapley Additive exPlanations) algorithm, which assigns a contribution to each feature in the model's prediction. Although SHAP was first created for tabular data, it can be modified to work with DNA sequences and identify promoter regions[16]. Determine the SHAP values for every nucleotide position (feature) in the promoter region sequences. The contribution of each nucleotide position to the model's determination of whether to classify a sequence as a promoter or non-promoter is represented by the SHAP values[17].

Interpretation:

1. Nucleotide positions that help the model predict a sequence as a promoter region are indicated by positive SHAP values.

2. Nucleotide positions that help the model predict a sequence as a non-promoter region are indicated by negative SHAP values.

3. Nucleotide positions with SHAP values close to zero are thought to have minimal impact on the model's classification decision.

SHAP Formula for DNA Sequence Promoter Region Identification:

The SHAP algorithm, which takes into account all potential subsets of features, can be used to calculate the SHAP value for a particular feature (nucleotide position) in the promoter region sequence. The following formula is applied to identify the promoter region:

$$\mathrm{SHAP}(\mathrm{f_i}) = \phi_\mathrm{i} = \sum \mathrm{S} \subseteq \{1,2 \ldots \mathrm{N}\} \setminus \{\mathrm{i}\} \frac{|\mathrm{S}|!\,(\mathrm{N} - |\mathrm{S}| - 1)!}{\mathrm{N}!} [\mathrm{f}(\mathrm{x_S} \cup \{\mathrm{f_i}\}) - \mathrm{f}(\mathrm{x_S})]$$

Where:

- $\mathrm{f_i}$ represents the feature of interest (nucleotide position) in the promoter region sequence.
- ϕ_i represents the SHAP value for feature $\mathrm{f_i}$.
- N represents the total number of features (nucleotide positions) in the promoter region sequence.
- S represents a subset of features excluding feature $\mathrm{f_i}$.
- $\mathrm{x_S}$ represents the input data with features in subset S.
- $\mathrm{f}_{(\mathrm{x_s})}$ represents the model's output when the input data contains features in subset S.
- $\mathrm{f}_{(\mathrm{x_s} \cup \{\mathrm{f_i}\})}$ represents the model's output when the input data contains features in subset S plus feature $\mathrm{f_i}$.

>>>#Algorithm to load sequences and labels:

```
BEGIN
   INITIALIZE sequences as an empty list
   INITIALIZE labels as an empty list

   FOR each record in "promoter_seq.fasta" DO
      SET sequence_string to the string representation of the sequence in record
      APPEND sequence_string to sequences
      APPEND 1 to labels  // 1 for promoter
   END FOR

   FOR each record in "non_promoter_seq.fasta" DO
      SET sequence_string to the string representation of the sequence in record
      APPEND sequence_string to sequences
      APPEND 0 to labels  // 0 for non-promoter
   END FOR
END
```

>>># Algorithm to create a function to predict and get SHAP values

```
BEGIN
   FUNCTION predict_and_get_shap(model, X_test):
      FUNCTION f(X):
         RETURN model.predict(X)

      INITIALIZE explainer as shap.DeepExplainer(f, X_train[:100])
      COMPUTE shap_values using explainer.shap_values(X_test)
      RETURN shap_values
   END FUNCTION

   CALL predict_and_get_shap with model and X_test
END
```

d) Model Training: GRU for Promoter Region Identification

Teach the recurrent neural network (RNN) to identify encoded features in sequences and classify them as either promoter or non-promoter regions[18]. Now the model gains the ability to distinguish between sequences with and without promoter motifs.

One kind of recurrent neural network (RNN) architecture that is frequently used for sequence modelling tasks, such as DNA sequence analysis like promoter region identification, is the Gated Recurrent Unit (GRU) architecture. By placing gating mechanisms to regulate information flow throughout the network, GRUs improve long-term dependency modelling and reduce the vanishing gradient issue, which is some of the drawbacks of conventional RNNs.

The update and reset gates that make up the GRU architecture control the information flow within the network. GRUs function as follows:

1. Update Gate z_t:

1. The update gate determines how much of the past information h_{t-1} to keep and how much of the new information $\tilde{h}_t$ to incorporate at each time step t.

2. It takes as input the current input x_t and the previous hidden state h_{t-1}, passes them through a sigmoid function, and outputs values between 0 and 1

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z)$$

2. Reset Gate x_t:

1. The reset gate determines how much of the past information h_{t-1} to forget at the current time step t.

2. It takes as input the current input x_t and the previous hidden state h_{t-1}, passes them through a sigmoid function, and outputs values between 0 and 1.

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r)$$

3. Candidate Hidden State $\tilde{h}_t$:

1. A candidate hidden state $\tilde{h}_t$ is computed based on the current input x_t and the reset gate r_t.

2. It captures the new information that could be added to the memory cell.

$$\tilde{h}_t = \tanh(W_h \cdot [r_t \odot h_{t-1}, x_t] + b_h)$$

4. Hidden State Update:

1. The hidden state h_t is updated based on the update gate z_t and the previous hidden state h_{t-1}, as well as the candidate hidden state $\tilde{h}_t$.

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

In the context of promoter region identification, the input x_t at each time step t represents a nucleotide at position t in the DNA sequence. The output h_t of the GRU at each time step can be interpreted as the hidden representation or features extracted from the sequence up to that point. The final hidden state h_T (where T is the sequence length) can be used for classification or further analysis to identify promoter regions.

The parameters W_z, W_r, W_h, b_z, b_r, and b_h represent weight matrices and bias terms that are learned during training. The symbol ⊙ denotes element-wise multiplication, and σ represents the sigmoid activation function. By training a GRU model on labelled DNA sequences (promoter / non-promoter regions) and examining the learned parameters and activations[19], we can identify important sequence motifs and regulatory patterns associated with promoter regions.

>>># Algorithm to build GRU model

```
BEGIN
   INITIALIZE model as a Sequential model


   ADD Embedding layer to model with parameters:
      input_dim = 4
      output_dim = 16
      input_length = number of columns in sequences_padded
   ADD GRU layer to model with parameters:
      units = 64
      dropout = 0.2
      recurrent_dropout = 0.2
   ADD Dense layer to model with parameters:
      units = 1
      activation = 'sigmoid'
   COMPILE model with parameters:
      loss = 'binary_crossentropy'
      optimizer = Adam with learning_rate = 0.001
      metrics = ['accuracy']
END
```

```
>>># Algorithm to train the model
>>>model.fit(X_train, y_train, epochs=10, batch_size=32, >>>validation_split=0.1)
```

4. Results

From the EDP website, we took about ~3,000 promotor region human sapiens datasets. The same number of non-promotor sequences are also taken into account when training. With an assigned value of 90% and 10%, respectively, the data was divided into two sets: the train set and the test set. 10% of the data utilized for training is used as well for parameter adjustment and validation. Training models and feature encoding methods were compared using the resultant datasets. 800 random sequences from the human genome were used as a dataset to assess the models' performance.

We conducted experiments to evaluate the effectiveness and computing demands of frequency-based tokenization (FBT) and K-Mer encoding methods[20]. To do this, we investigated the feature encoding techniques on k-mer sizes 3, 4, 5, and 6 using datasets that were both promoter and non-promoter (Figure 5).

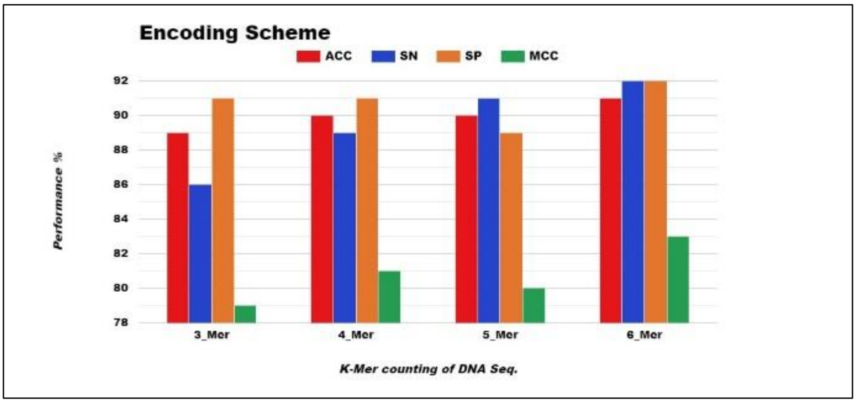


Figure 5. Comparing several encoding strategies to identify promoters.

At first, it takes longer for the K-Mer to train with more than 2 k-Mer. Based on the accuracy found at higher k-mer values, we chose k-mer sizes of 3, 4, 5, and 6. We discovered that the 6 k-mer performed better than the 3, 4, and 5 k-mer, with ACC scores of 91.86%, SN scores of 92.74%, SP scores of 91%, and MCC scores of 83%. In contrast, the 3-Mer ACC, SN, SP, and MCC are 89.65%, 86.81%, and 79%, respectively. ACC is 90.76%, SN is 90.26%, SP is 91.21%, and MCC is 81% for 4-Mer. According to the following data, for 5-Mer, ACC is 90.09%, SN is 90.81%, SP is 89.38%, and MCC is 0.80%.

The first series of experiments involved binary DNA sequence classification into promotor as well as non-promotor classes. The human genome dataset's Table 2 displays the explainable AI SHAP[21] and RNN performances. On the test set of sequences, the GRU model's average performance was as follows: precision 91%, accuracy 93%, recall 100%, f1-score92 %, and cross-validation score 96%. With a k-Mer size of 2, RF, Adaboost, and Decision tree all functioned nicely. When examining overall performance, SHAP and GRU score highly across all measures. While SVM linear and Naive Bayse have good performance metrics in precision and recall. Nevertheless, GRU outperformed both with k-mer sizes of 5 and 6. GRU requires more computing power as the size of the k-mer rises.

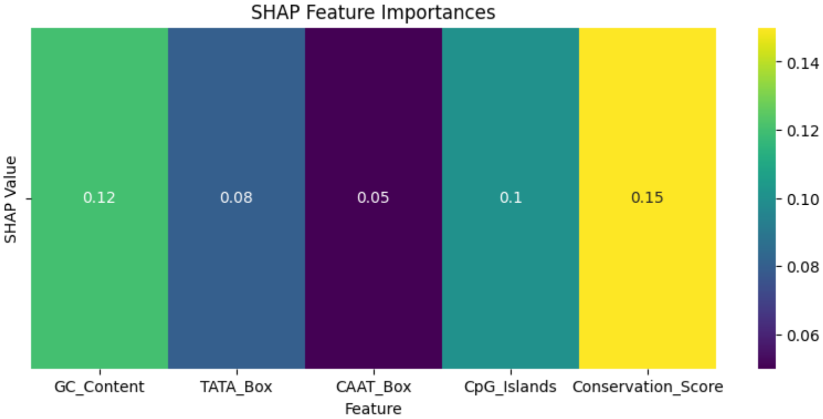


Figure 6: SHAP feature extraction

Table 2. Assessment of different machine learning algorithms in predicting DNA promoter.

Classifier	Precision	Accuracy	Recall	F1-score	Cross Validation score
SHAP +RNN	0.91	0.93	1.00	0.91	0.9625
SVM Linear	0.85	0.89	0.92	0.86	0.8750
Naive Bayes	0.90	0.87	0.89	0.91	0.9215
Random Forest	0.68	0.64	0.62	0.62	0.6428
Ada Boost	0.79	0.85	0.76	0.87	0.8857
Decision Tree	0.81	0.78	0.71	0.80	0.7589

Lastly, we used 800 random sequences from the human genome as testing data to assess how well these ML/DL models performed[12]. Compared to RF, DT, SVM, Naive Bayse and other models[22], SHAP and GRU performed better in predictions. Together with 91% precision, 93% accuracy, 100% recall, 91% f1-score and 96% for cross validation, the SHAP and GRU models yield an average accuracy of 94.4% (Table 2). The following figure shows the accuracy obtained on the 100% of the data.

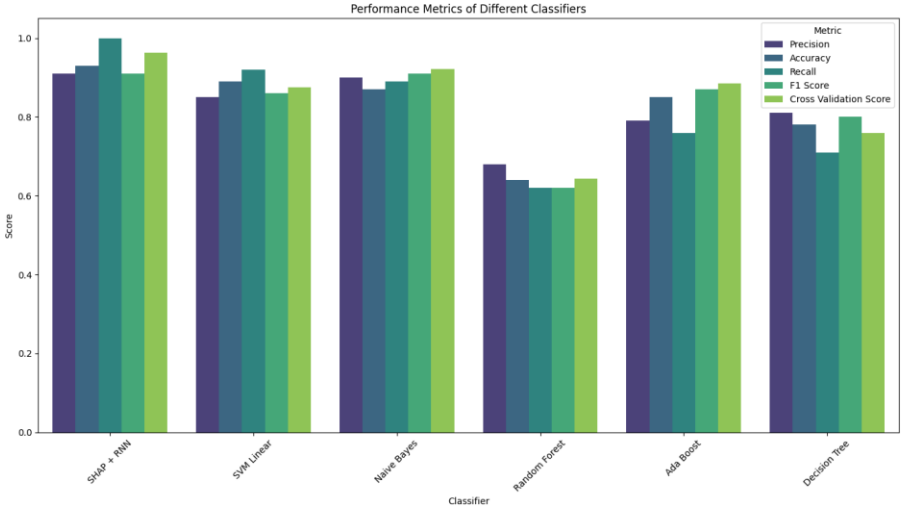


Figure 7: Performance comparison of different classifiers

Table 3. List of gene promotors with its location

S. No	Gene Promotor	Location	Promotor Sequence
1.	TP53	Chromosome17: 7,565,161-7,565,641 (hg38)	5'-GAGCGGGAGCAGGGAGGCGCAGAGGAGGAAGGAGGAAGGAGGAGCAGCAGCGGAGGCGCAGCAGCAGCAGCAG-3'
2.	MYC	Chromosome 8: 128,748,315-128,748,745 (hg38)	5'-GCTGCGGGGCGGGGCGGGGCGGGGCGGGGCGGCGGGGCGGAGGAGGAGGAGGAGGAGGAGGAGGAGGA-3'
3.	HBB	Chromosome 11: 5,225,464-5,225,564 (hg38)	5'-AGCTTCTGAGTCCACACACACCCACCCATGCAGTCCAGCCTCCTCTTCCCTCTCCTCCTCTCCTCCCCTCCCCCC-3'

4.	GAPDH	Chromosome 12: 6,533,682-6,534,682 (hg38)	5'-GCAGTGGTGGAGGTGTGGGCGGTTGAGCGTGAGTGGGCTCCTGGTGTGGCAGTGGTCGGTGTGGAAGGAGCAGC-3'
5.	BRCA1	Chromosome 17: 43,044,294-43,044,794 (hg38)	5'-GGGGTGGGAGGCGGCGGCGGCGGGGCGGGCGGGGCGGGCGGCGGGGCGGGGCGGGGCGGGGCGGGCGGGGCGG-3'
6.	ACTB (Beta-Actin)	Chromosome 7: 5,552,999-5,553,999 (hg38)	5'-TCGAGCAGGCCGGAGGAGCGCGTCCCTGAGGACAGGAGAGGGAGCCGGGCGGGGCTGACGGCCGGCGGGGCG-3'
7.	EGFR (Epidermal Growth Factor Receptor)	Chromosome 7: 55,019,031-55,019,531 (hg38)	5'-GAGCGGCCGCGGCGGGCGGGGCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGGCGGGCGGCGGCGGGGCG-3'
8.	TNF (Tumor Necrosis Factor)	Chromosome 6: 31,501,633-31,501,733 (hg38)	5'-GGGGCGGCGGGGCGGGCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGC-3'
9.	FOXO3 (Forkhead Box O3)	Chromosome 6: 108,780,723-108,781,223 (hg38)	5'-GCCGCGCGGCGGCGCGCGGCGGCGCGCGGCGGCGCGCGGCGGCGCGGCGGCGCGCGGCGGCGCGCGCGGCGG-3'
10.	VEGFA (Vascular Endothelial Growth Factor A)	Chromosome 6: 43,673,985-43,674,485 (hg38)	5'-GCGGGCGCGGGCGCGGGCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGC-3'
11.	IL6 (Interleukin 6)	Chromosome 7: 22,720,595-22,721,095 (hg38)	5'-CGAGCAGGCCGAGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGG-3'
12.	APOE (Apolipoprotein E)	Chromosome 19: 44,909,584-44,910,084 (hg38)	5'-AGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGC-3'
13.	ESR1 (Estrogen Receptor 1)	Chromosome 6: 151,656,690-151,657,190 (hg38)	5'-GAGGCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGGCGGGCGGGGC-3'
14.	INS (Insulin)	Chromosome 11: 2,151,250-2,151,750 (hg38)	5'-GCGGCGGGGCGGGGCGGGGCGGGGCGGGCGGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGC-3'
15.	CFTR (Cystic Fibrosis Transmembrane Conductance Regulator)	Chromosome 7: 117,120,016-117,120,516 (hg38)	5'-CTTGGAGAGGCCGCGGAGGCGGCGGCGGGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCGGCG-3'
16.	BRCA2 (Breast Cancer 2)	Chromosome 13: 32,889,616-32,890,116 (hg38)	5'-GCGGGCGGGGCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGGC-3'
17.	MTHFR (Methylenetetrahydrofolate Reductase)	Chromosome 1: 11,824,520-11,825,020 (hg38)	5'-AGGGCGGGGCGGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGC-3'
18.	SOD1 (Superoxide Dismutase 1)	Chromosome 21: 31,580,546-31,581,046 (hg38)	5'-GGGGCGGGGCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGGCG-3'

19.	BCL2 (B-cell CLL/Lymphoma 2)	Chromosome 18: 63,529,950-63,530,450 (hg38)	5'-AGGGGCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGC-3'
20.	CYP2D6 (Cytochrome P450 2D6)	Chromosome 22: 42,561,822-42,562,322 (hg38)	5'-GCGGCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGGCGGGGC-3'
21.	IGF1 (Insulin-Like Growth Factor 1)	Chromosome 12: 102,853,470-102,853,970 (hg38)	5'-GCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGG-3'
22.	HLA-DRA (Major Histocompatibility Complex, Class II, DR Alpha)	Chromosome 6: 32,609,790-32,610,290 (hg38)	5'-CGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGGC-3'
23.	PTEN (Phosphatase and Tensin Homolog)	Chromosome 10: 87,875,136-87,875,636 (hg38)	5'-GCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGGC-3'
24.	CDKN1A (Cyclin Dependent Kinase Inhibitor 1A)	Chromosome 6: 36,678,887-36,679,387 (hg38)	5'-GGGCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCG-3'
25.	RB1 (Retinoblastoma 1)	Chromosome 13: 48,806,407-48,806,907 (hg38)	5'-GGCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGG-3'
26.	MTOR (Mechanistic Target of Rapamycin)	Chromosome 1: 11,176,383-11,176,883 (hg38)	5'-GGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGC-3'
27.	NF1 (Neurofibromin 1)	Chromosome 17: 29,556,025-29,556,525 (hg38)	5'-GCGGGGCGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGG-3'
28.	STAT1 (Signal Transducer and Activator of Transcription 1)	Chromosome 2: 191,396,496-191,396,996 (hg38)	5'-GGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGC-3'
29.	ATM (ATM Serine/Threonine Kinase)	Chromosome 11: 108,093,645-108,094,145 (hg38)	5'-GCGGGCGGGGCGGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGCGGGGCGGGG-3'

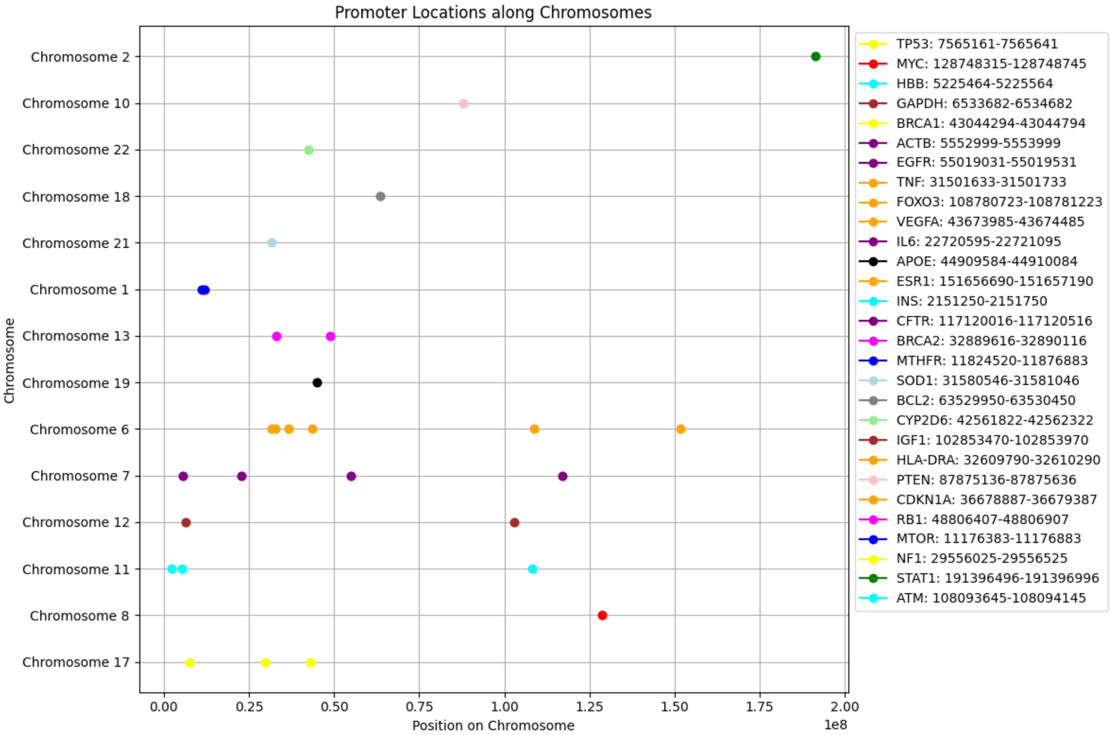


Figure 8: Transcription factor binding sites in DNA sequences.

Evaluation of the performance metrics:

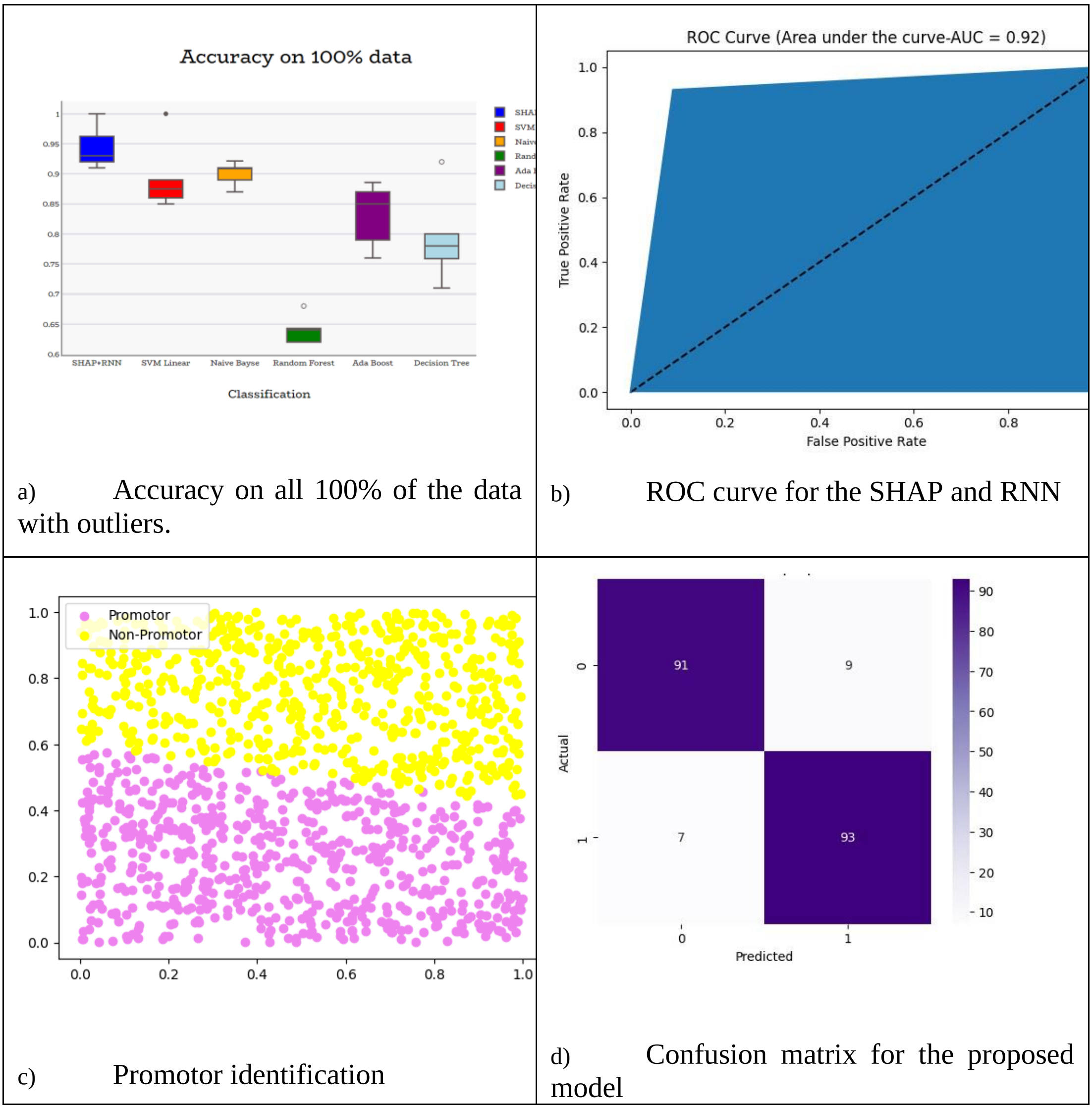
After the prediction model was applied to the dataset, the number of actually identified promoter as well as non-promoter sequences was calculated as true positive (TP) and true negative (TN). In addition, we gathered the number of detected promoter and non-promoter sequences that were falsely negative (FN) and erroneously positive (FP) [23]. We used the following formulas to calculate accuracy (Acc), sensitivity (Sn), specificity (Sp), and Matthew's correlation coefficient (MCC) in order to assess the efficacy of classification models:

$$\text{Accuracy (Acc)} = 1 - \frac{\text{FN} + \text{FP}}{(\text{TP} + \text{FN}) + (\text{TN} + \text{FP})}, 0 \leq \text{Acc} \ \leq 1$$

$$\text{Specificity (Sp)} = 1 - \frac{\text{FP}}{\text{TN} + \text{FP}}, 0 \ \leq \text{Spec} \ \leq 1$$

$$\text{Sensitvity (Sn)} = 1 - \frac{\text{FN}}{(\text{TP} + \text{FN})}, 0 \ \leq \text{Sens} \ \leq 1$$

$$\text{Matthew's correlation coefficient (MCC)} = \frac{1 - (\frac{\text{FN}}{\text{TP} + \text{FN})} + \frac{\text{FP}}{(\text{TN} + \text{FP})})}{\sqrt{((1 + \frac{\text{FP} - \text{FN}}{\text{TP} + \text{FN}})(1 + \frac{\text{FN} - \text{FP}}{\text{TN} + \text{FP}})}}$$



a) Accuracy on all 100% of the data with outliers.

b) ROC curve for the SHAP and RNN

c) Promotor identification

d) Confusion matrix for the proposed model

5. Discussion

When analysing DNA sequences, it is standard practice to use a set of "background" sequences to assess the false-positive rates of gene recognition techniques for identifying cis-regulatory regions. Another method for creating a background dataset involves rearranging some of the nucleotide sequences in the positive dataset. We opted for frequency-based tokenization of k-mers as opposed to one-hot encoding for feature encoding. A 4k × L matrix of 0s and 1s, where k represents the sequence length and L signifies the k-mer length, represents each input sequence in one-hot encoding. The input to the embedding layer is relatively sparse due to this representation. As the k-mer length grows, the input dimension also increases, resulting in longer computational processing times.

In addition, the matrix represented in a one-hot encoded format for a sequence does not capture the importance of the frequency of occurrence of a subsequence or motif. In comparison to one-hot encoding, tokenization based on frequency offers the benefit of reducing the input dimension for the AI model, which could potentially lead to significant savings in training time [24]. Consequently, one-hot encoding might yield similarly accurate results. Nevertheless, many researchers may lack the necessary computational resources to perform one-hot encoding for larger k-mer sizes.

Optimising parameters is an essential step in developing a sequence predictor. Along with the network architecture design, it was shown that K-mer size is an empirically tuneable parameter during the categorisation of sequences into promoter and non-promoter groups. The model's predictive power was impacted by differences in the sizes of the generated feature vectors and k-mers. We looked at how different 3-, 4-, 5-, and 6-mer fingerprints affected SHAP and RNN's capacity for prediction. As was already established, the LSTM model's performance increased as the k-mer got bigger. Additionally, we evaluated each model with k-mer sizes of 12 and 16. But because of the enormous memory usage, this led to an exponential increase in the quantity of training parameters and a "resource exhaustion problem"[25].

The findings demonstrate that for all unique organisms, RNN and SHAP perform better than other designated classification models. In binary and multiclass scenarios, the proposed network models have achieved outstanding classification accuracy while lowering the rate of false-positive and false-negative predictions. Nevertheless, the observed variations in accuracy, sensitivity, specificity, and MCC could not necessarily be a reflection of the constructed model due to the variations in data kinds and sizes [23].

The current limitations of the accessibility dataset repository are known to us. Since we are working with a dataset of human species, we need to be more concise. While we aim to offer more variability, most benchmark datasets have a set number of classes and sequence lengths [26]. Our dataset, which comes from the EDP website, contains every regulatory feature. To find out how sensitive the model is to these kinds of characteristics, we intend to further diversify our datasets in the future. We intend to expand our benchmark collection with additional multi-class and imbalanced datasets.

6. Conclusion

The main goal of this work is to effectively discriminate between promoter and non-promoter sequences with enhanced accuracy, true positive rate, and true negative rate. We used a promotor region and randomly selected sequence to evaluate the validity of the model and construct a robust and general framework for classification problems in the genomic domain, obtaining the required heterogeneity and robustness for our research. The effectiveness of the models was evaluated using a collection of randomly chosen sequences from the human genome. We have employed frequency-based tokenization of sequences for vector representation and feature extraction, and k-mer-based subsampling for data preprocessing to decrease training time.

To efficiently discriminate between promoter as well as non-promoter sequences with increased accuracy, true positive rate, and true negative rate is the major objective of this work.

In order to obtain the necessary heterogeneity and robustness for our research, we employed a promotor region and randomly picked sequence to test the model's correctness and build a robust and universal agenda for classification problems in the genomic domain. A set of randomly selected sequences from the human genome was used to assess the models' performance. To minimise training time, we have used k-mer-based subsampling for data preprocessing and frequency-based tokenisation of sequences for vector representation and feature extraction, respectively. Finding promoters in DNA sequences is an essential initial step in understanding the regulation of gene transcription, even though promoters often start the transcription of a gene. In order to precisely forecast promoters and their strength in DNA sequences, a computational model called SHAP and GRU are employed in this work. Because GRU, a long-short-term memory neural network, and SHAP feature extraction allow the model to account for the state of each promoter identification modification feature for each state, the model's performance has increased. We were able to increase accuracy on such a large dataset with good precision and recall by using SHAP and GRU. In the end, our model performed better than the state-of-the-art model, proving its usefulness as a tool for prediction identification and its notable advancement over earlier techniques.

References

[1] Z. W. Ma, J. P. Zhao, J. Tian, and C. H. Zheng, “DeeProPre: A promoter predictor based on deep learning,” Comput. Biol. Chem., vol. 101, no. March, p. 107770, 2022, doi: 10.1016/j.compbiolchem.2022.107770.

[2] L. Zheng, L. Liu, W. Zhu, Y. Ding, and F. Wu, “Predicting enhancer-promoter interaction based on epigenomic signals,” Front. Genet., vol. 14, no. April, pp. 1–8, 2023, doi: 10.3389/fgene.2023.1133775.

[3] Z. H. Fu, S. Z. He, Y. Wu, and G. R. Zhao, “Design and deep learning of synthetic B-cell-specific promoters,” Nucleic Acids Res., vol. 51, no. 21, pp. 11967–11979, 2023, doi: 10.1093/nar/gkad930.

[4] Y. Wang, S. Tai, S. Zhang, N. Sheng, and X. Xie, “PromGER: Promoter Prediction Based on Graph Embedding and Ensemble Learning for Eukaryotic Sequence,” Genes (Basel)., vol. 14, no. 7, 2023, doi: 10.3390/genes14071441.

[5] H. Y. Lai et al., “iProEP: A Computational Predictor for Predicting Promoter,” Mol. Ther. Nucleic Acids, vol. 17, no. September, pp. 337–346, 2019, doi: 10.1016/j.omtn.2019.05.028.

[6] H. Zulfiqar, Z. Ahmed, B. Kissanga Grace-Mercure, F. Hassan, Z. Y. Zhang, and F. Liu, “Computational prediction of promotors in Agrobacterium tumefaciens strain C58 by using the machine learning technique,” Front. Microbiol., vol. 14, 2023, doi: 10.3389/fmicb.2023.1170785.

[7] J. T. Guo and F. Malik, “Single-Stranded DNA Binding Proteins and Their Identification Using Machine Learning-Based Approaches,” Biomolecules, vol. 12, no. 9, 2022, doi: 10.3390/biom12091187.

[8] A. Agarwal and L. Chen, “DeepPHiC: predicting promoter-centered chromatin interactions using a novel deep learning approach,” Bioinformatics, vol. 39, no. 1, 2023, doi: 10.1093/bioinformatics/btac801.

[9] A. Osman et al., “Identification of genomic binding sites and direct target genes for the transcription factor DDIT3/CHOP,” Exp. Cell Res., vol. 422, no. 1, 2023, doi: 10.1016/j.yexcr.2022.113418.

[10] X. Wang, K. Xu, Y. Tan, S. Yu, X. Zhao, and J. Zhou, “ Deep Learning-Assisted Design of

Novel Promoters in Escherichia coli ," Adv. Genet., vol. 4, no. 4, pp. 1–11, 2023, doi: 10.1002/ggn2.202300184.

[11] Z. Hong, X. Zeng, L. Wei, and X. Liu, "Identifying enhancer-promoter interactions with neural network based on pre-trained DNA vectors and attention mechanism," Bioinformatics, vol. 36, no. 4, pp. 1037–1043, 2020, doi: 10.1093/bioinformatics/btz694.

[12] H. Gong et al., "Computational methods for identifying enhancer-promoter interactions," Quant. Biol., vol. 11, no. 2, pp. 122–142, 2023, doi: 10.15302/J-QB-022-0322.

[13] K. Grešová, V. Martinek, D. Čechák, P. Šimeček, and P. Alexiou, "Genomic benchmarks: a collection of datasets for genomic sequence classification," BMC Genomic Data, vol. 24, no. 1, pp. 1–9, 2023, doi: 10.1186/s12863-023-01123-8.

[14] N. Q. K. Le, E. K. Y. Yapp, N. Nagasundaram, and H. Y. Yeh, "Classifying Promoters by Interpreting the Hidden Information of DNA Sequences via Deep Learning and Combination of Continuous FastText N-Grams," Front. Bioeng. Biotechnol., vol. 7, no. November, pp. 1–9, 2019, doi: 10.3389/fbioe.2019.00305.

[15] Q. Geng, R. Yang, and L. Zhang, "A deep learning framework for enhancer prediction using word embedding and sequence generation," Biophys. Chem., vol. 286, no. January, p. 106822, 2022, doi: 10.1016/j.bpc.2022.106822.

[16] N. Q. K. Le, Q. T. Ho, V. N. Nguyen, and J. S. Chang, "BERT-Promoter: An improved sequence-based predictor of DNA promoter using BERT pre-trained model and SHAP feature selection," Comput. Biol. Chem., vol. 99, no. June, p. 107732, 2022, doi: 10.1016/j.compbiolchem.2022.107732.

[17] Q. Dickinson and J. G. Meyer, "Positional SHAP (PoSHAP) for Interpretation of machine learning models trained from biological sequences," PLoS Comput. Biol., vol. 18, no. 1, pp. 1–24, 2022, doi: 10.1371/journal.pcbi.1009736.

[18] A. Raza, W. Alam, S. Khan, M. Tahir, and K. T. Chong, "iPro-TCN: Prediction of DNA Promoters Recognition and Their Strength Using Temporal Convolutional Network," IEEE Access, vol. 11, no. July, pp. 66113–66121, 2023, doi: 10.1109/ACCESS.2023.3285197.

[19] M. Uddin, M. K. Islam, M. R. Hassan, F. Jahan, and J. H. Baek, "A fast and efficient algorithm for DNA sequence similarity identification," Complex Intell. Syst., vol. 9, no. 2, pp. 1265–1280, 2022, doi: 10.1007/s40747-022-00846-y.

[20] H. Kamran, M. Tahir, H. Tayara, and K. T. Chong, "iEnhancer-Deep: A Computational Predictor for Enhancer Sites and Their Strength Using Deep Learning," Appl. Sci., vol. 12, no. 4, 2022, doi: 10.3390/app12042120.

[21] G. Sganzerla Martinez, E. Perez-Rueda, A. Kumar, S. Sarkar, and S. de Avila e Silva, "Explainable artificial intelligence as a reliable annotator of archaeal promoter regions," Sci. Rep., vol. 13, no. 1, pp. 1–12, 2023, doi: 10.1038/s41598-023-28571-7.

[22] J. A. Barbero-Aparicio, A. Olivares-Gil, J. F. Díez-Pastor, and C. García-Osorio, "Deep learning and support vector machines for transcription start site identification," PeerJ Comput. Sci., vol. 9, 2023, doi: 10.7717/PEERJ-CS.1340.

[23] R. Umarov, H. Kuwahara, Y. Li, X. Gao, and V. Solovyev, "Promoter analysis and prediction in the human genome using sequence-based deep learning models," Bioinformatics, vol. 35, no. 16, pp. 2730–2737, 2019, doi: 10.1093/bioinformatics/bty1068.

[24] N. Bhandari, S. Khare, R. Walambe, and K. Kotecha, "Comparison of machine learning and deep learning techniques in promoter prediction across diverse species," PeerJ Comput. Sci., vol. 7, pp. 1–17, 2021, doi: 10.7717/PEERJ-CS.365.

[25] Y. Zhu and F. Li, "Computational Identification of Eukaryotic," vol. 22, no. September, pp. 1–11, 2005.

[26] B. Liu and K. Li, "iPromoter-2L2.0: Identifying Promoters and Their Types by Combining Smoothing Cutting Window Algorithm and Sequence-Based Features," Mol. Ther. Nucleic Acids, vol. 18, no. 5, pp. 80–87, 2019, doi: 10.1016/j.omtn.2019.08.008.