Identification of DNA Coding Regions Using Transformers

Gustavo H. F. Cruz, Aurora T. R. Pozo

Paraná Federal University, Brazil
{ghfcruz, aurora}@inf.ufpr.br

Abstract. Identifying coding (exon) and non-coding (intron) regions in DNA sequences is fundamental to understanding gene expression and its implications for biological processes and genetic diseases. In this work, we investigate the application of Transformer-based architectures to the task of intron and exon classification, comparing three distinct models: GPT-2, BERT, and DNABERT. These models were selected to evaluate the impact of context modeling strategies—autoregressive, bidirectional, and k-mer-based—on genomic sequence analysis. Experiments were carried out on a curated dataset comprising 100 000 training sequences and 30 000 test sequences, using mutually exclusive samples to ensure robust evaluation. All models were fine-tuned under uniform conditions, with a fixed batch size of 32 and learning rate constraints, and executed three times with different seeds. The results show that BERT achieved the highest classification accuracy (0.9905), outperforming DNABERT (0.9569) and GPT-2 (0.9867). While DNABERT was the fastest to train due to its k-mer tokenization and lighter computational requirements, its limited capacity to model long-range dependencies impaired its performance. In contrast, GPT-2 demonstrated competitive accuracy but at a higher computational cost, reinforcing the trade-off between generative modeling power and efficiency. This study highlights the importance of context-aware attention mechanisms in genomic sequence modeling and confirms the viability of Transformer architectures—especially bidirectional models like BERT—for high-accuracy classification of intronic and exonic regions. Future work may benefit from exploring larger models, sequence representation alternatives, and training optimization techniques to further enhance performance in genomics applications.

CCS Concepts: • **Computing methodologies** → **Artificial intelligence**.

Keywords: bert, classification, coding regions, deep learning, dnabert, exons, gpt, introns, transformers

1. INTRODUCTION

Deoxyribonucleic acid (DNA) is the foundation of life, storing vital information necessary for the survival of organisms [Watson et al. 2015]. To be utilized, this information must be transcribed into ribonucleic acid (RNA) [Crick 1958; Watson et al. 2015], which then undergoes a process known as splicing. During splicing, coding regions (exons) are preserved while non-coding regions (introns) are removed [Sharp 2005]. Disruptions in this process can lead to serious diseases, highlighting the importance of accurately identifying these regions.

Recent advances in machine learning have significantly improved the identification of coding and non-coding regions. Early methods applied Natural Language Processing (NLP) techniques to leverage the sequential nature of DNA, followed by deep learning approaches such as Convolutional Neural Networks (CNN) [Lecun et al. 1998], Recurrent Neural Networks (RNN) [Rumelhart et al. 1986], and Long Short-Term Memory (LSTM) networks [Hochreiter and Schmidhuber 1997]. More recently, Transformer-based models [Vaswani et al. 2017] have emerged as promising tools for genomic sequence processing.

This work investigates the application of Transformer architectures for the classification of introns and exons, focusing on three models: GPT (Generative Pre-trained Transformer) [Radford et al. 2018], BERT (Bidirectional Encoder Representations from Transformers) [Devlin et al. 2019], and

DNABERT [Ji et al. 2021], a DNA-specific variant of BERT.

2. THEORETICAL FOUNDATION

DNA (deoxyribonucleic acid) is the molecule responsible for storing and transmitting genetic information, playing a crucial role in the formation and function of living organisms [Watson et al. 2015; Crick 1958]. It is composed of genes, which carry instructions for protein synthesis, as well as repetitive nucleotide regions [Watson et al. 2015]. Its double-helix structure is maintained by phosphodiester bonds and complementary base pairing (A-T and C-G), enabling the replication and inheritance of genetic material [Watson et al. 2015].

For protein synthesis to occur, DNA is not directly used; instead, it is replicated[1], producing messenger RNA (mRNA), where thymine (T) is replaced by uracil (U). This mRNA is then transported to the cell's cytoplasm[2], where ribosomes—organelles responsible for protein production—translate the genetic information [Crick 1958; Watson et al. 2015].

However, not all parts of DNA are translated into proteins. The coding regions, called *exons*, generate proteins, while the non-coding regions, called *introns*, must be removed [Sharp 2005].

Eukaryotic organisms contain introns in their genes, which are removed in a process known as *splicing* [Chow et al. 1977], prior to final translation. Figure 1 illustrates this transcription and processing from DNA to mRNA.

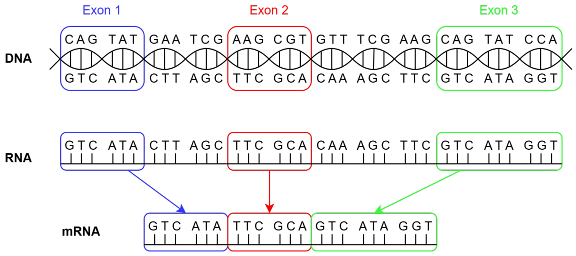
Exon 1
Exon 2
Exon 3
DNA
CAG TAT GAA TCG AAG CGT GTT TCG AAG CAG TAT CCA
GTC ATA CTT AGC TTC GCA CAA AGC TTC GTC ATA GGT
RNA
GTC ATA CTT AGC TTC GCA CAA AGC TTC GTC ATA GGT
mRNA
GTC ATA TTC GCA GTC ATA GGT

Fig. 1. Illustration of mRNA generation process.

2.1 Concepts from Computing and Deep Learning

In computing, Machine Learning (ML) seeks to automate complex tasks through data-driven approaches. With the rise of deep learning, models capable of extracting patterns autonomously have become widespread. One of the most impactful innovations in this area was the introduction of Transformers, which have revolutionized fields such as natural language processing, computer vision, and, more recently, bioinformatics.

2.2 Transformer Architecture and Organization

Transformers, introduced in *Attention Is All You Need* [Vaswani et al. 2017], replaced recurrent architectures like RNNs and LSTMs with fully parallelized processing, enabling more efficient handling of input sequences. Their architecture consists of two main components: the *encoder*, which generates contextual representations of the input, and the *decoder*, which produces outputs based on those representations.

[1] In general, cells do not interact directly with DNA; a copy must be synthesized to enable biological processes [Alberts et al. 2017].

[2] The internal region of the cell encompassing all its structures.

The core mechanism in Transformers is attention, present in both components. It allows each input token to "pay attention" to every other token in the sequence, weighing their importance based on computed similarity [Vaswani et al. 2017], effectively capturing long-range dependencies.

In addition to attention, Transformers include embedding and positional encoding to treat the input text and the output text, residual connections, layer normalization, and fully connected *feedforward* layers, which improve training stability [Vaswani et al. 2017].

Transformers have evolved into three major categories:

—**Encoder-based models:** Use only the encoder. Suitable for classification and feature extraction tasks. The most prominent example is BERT [Devlin et al. 2019], known for its bidirectional understanding of input texts.

—**Decoder-based models:** Use only the decoder. Ideal for text generation, autocompletion, and dialogue systems. GPT [Radford et al. 2018] is the leading model in this category.

—**Encoder-decoder models:** Utilize both components. Best suited for translation and summarization tasks. T5 [Raffel et al. 2020] is a representative of this group.

3. RELATED WORK

To identify the main studies related to the classification of coding regions using machine learning, a systematic search was conducted in scientific databases such as PubMed, IEEE Xplore, and arXiv. Keywords included intron, exon, acceptor, donor, coding regions, neural networks, splicing, and Transformer. The selection prioritized papers published in the last ten years, focusing on deep learning-based approaches such as CNN, RNN, and Transformers.

Several notable works were identified: DeepSS [Du et al. 2018], SpliceFinder [Wang et al. 2019], DeepSplice [Zhang et al. 2018], SpliceRover [Zuallaert et al. 2018], SpliceAI [Jaganathan et al. 2019], Splice2Deep [Albaradei et al. 2020], SpliceJunction [Sarkar et al. 2020], DNABERT [Ji et al. 2021], DNAGPT [Zhang et al. 2023], and SpliceFormer [Jónsson et al. 2024].

CNNs are widely adopted. DeepSS, for instance, converted over 120 000 sequences of acceptor and donor sites from Caenorhabditis elegans into images, achieving an Area Under the Receiver Operating Characteristic (auROC) values up to 0.9956 ± 0.33. SpliceFinder used 30 000 sequences from Homo sapiens, reaching a peak accuracy of 0.965. DeepSplice employed 812 967 sequences from the same species, with accuracies of 0.893 (acceptor) and 0.907 (donor). SpliceRover analyzed approximately 13 000 sequences, achieving accuracies above 0.95. SpliceAI achieved 0.95 accuracy on 10 000 sequences, while Splice2Deep used 500 800 sequences (half real, half synthetic) from five species, with accuracies above 0.96.

Among RNN-based approaches, SpliceJunction reached extremely high accuracies (0.9995 and 1.0) on 3 175 sequences, although the species used were not reported.

Transformer-based models are also emerging. DNABERT and DNAGPT are tailored for DNA sequences but focus on general modeling tasks, without directly targeting intron/exon classification. SpliceFormer, however, proposes a Transformer for identifying acceptor and donor sites, achieving an auPRC of 0.979 and an accuracy of 0.952 on 45 000 sequences. Table I summarizes the main characteristics of the reviewed studies.

Overall, the field remains heavily focused on splice site identification (acceptor and donor), often limited to a small number of species and datasets rarely exceeding millions of sequences. The introduction of Transformers offers a promising opportunity for more robust generalization and large-scale analysis. However, current models do not yet incorporate contextual information such as species of origin, gene structure, or sequence flanks, which may be critical for accurate intron/exon identification.

Table I. Comparative table between related works.

Reference	**Model (Metric)**	**# Data**	**Sequence Length**	**Species**	**Result**
DeepSS	CNN(auROC)	240 104	90(A) 15(D)	*C. elegans*	0.9956 ± 0.33
SpliceFinder	CNN(Acc)	30 000	40-400	*Homo sapiens*	0.9650
DeepSplice	CNN(Acc)	812 967	120	*Homo sapiens*	0.8930(A) 0.9070(D)
SpliceRover	CNN(Acc)	13 123	-	*Homo sapiens*	0.9612(A) 0.9535(D)
SpliceAI	CNN(Acc)	-	up to 5 000	*Homo sapiens*	0.95
Splice2Deep	CNN(Acc)	500 800	602	*H. sapiens*, *A. thaliana*, *O. sativa*, *D. melanogaster* and *C. elegans*	0.9691(A) 0.9738(D)
SpliceJunction	RNN(Acc)	3 175	60	-	0.9995(A) 1.0(D)
DNABERT	BERT	-	-	Various eukaryotes	-
DNAGPT	GPT	200B	-	*H. sapiens*, *M. musculus*, *B. taurus* and *D. melanogaster*	-
SpliceFormer	Trans(auPRC)	45 000	up to 10 000	*Homo sapiens*	0.979

Notes. A: acceptor; D: donor; B: billions (10^9); Acc.: accuracy; auROC: area under ROC curve; auPRC: area under PR curve.

This work aims to fill that gap by combining large-scale Transformer models with relevant metadata such as species and flanking regions. Although experiments described here were limited to 100 000 sequences due to computational constraints, results already point to promising improvements in robustness and generalization, paving the way for accurate, scalable, multi-species applications beyond the scope of previous approaches.

4. MATERIALS AND METHODS

The experiments were conducted using Python 3.12.3. The main libraries used include: PyTorch 3.6[3]: for implementing and training deep learning models; Hugging Face Transformers 4.48.2[4]: for loading and fine-tuning Transformer-based models such as GPT-2, BERT, and DNABERT; Pandas 2.2.3[5]: for tabular data manipulation and analysis; Biopython 1.85[6]: for processing DNA sequences obtained from GenBank[7]. Experiments were run on a machine equipped with an NVIDIA RTX A4500 GPU, using CUDA for acceleration.

4.1 Data Collection and Processing

To build a robust dataset for intron and exon classification, an extensive extraction from GenBank was performed, totaling 868 439 raw DNA sequences from various species without initial filtering. Only fully annotated sequences (without truncations) were retained. Each was processed to extract individual introns and exons, enriched with contextual information such as the organism (always available), gene name (when present), and flanking regions—10 nucleotides on each side of the main sequence, and extended 25-nucleotide flanks for broader context.

Sequences were limited to a maximum length of 512 nucleotides. To remove duplicates, records with identical features were compared, and repeated sequences were discarded. With all of the processed introns and exons sequences dataset, we derived smaller ones, containing 5 000 000, 100 000, 30 000, and 3 000 samples, respectively. Each subset was created with fixed seeds and random shuffling to ensure reproducibility and uniqueness across sets.

[3] https://pytorch.org/

[4] https://huggingface.co

[5] https://pandas.pydata.org/docs/index.html

[6] https://biopython.org/

[7] https://www.ncbi.nlm.nih.gov/genbank/

Each set has two versions: one **full version**, containing up to 512 nucleotides, with extended flanks and a **reduced version** with up to 128 nucleotides, with standard flanks. The reduced version was useful for models with input size restrictions (such as BERT) and for lower-cost experiments. All sets were balanced to include equal amounts of introns and exons.

During the experiments, 100 000 sequences were used for training and 30 000 for evaluation, each from mutually exclusive sets, as previously described. Additionally, the batch size was fixed at 32 sequences for all models. Each model was executed three times with different seeds to ensure reproducibility and result robustness.

4.2 Used Models

Three Transformer-based architectures were explored for the classification task: GPT-2, BERT, and DNABERT. The selection aimed to balance robustness and adaptability.

The first one, GPT, is based exclusively on the decoder architecture of the original Transformer model, employing causal attention - meaning that each token can attend only to the previous tokens seen in the sequence. This is the main feature of GPT: when given a sequence, its task is to predict the next token based on all previous tokens - a unidirectional language modeling approach. In our work, we fine-tuned GPT-2 on the genomic sequences and adapted it to operate in a classification-like setup, where the model predicts the most probable token given a prompt containing both the context and a target sequence - in this case, an exon or intron.

BERT, on the other hand, uses only the encoder of the original Transformer model and is able to capture bidirectional dependencies, processing context from both past and future tokens. This is the main reason why BERT models use Masked Language Modeling (MLM) - hiding some tokens in the input sequence and training the model to predict them based on surrounding context. In our experiments, BERT was fine-tuned to perform a sequence-level classification, learning from both the context and the input sequence to predict whether it corresponds to an intron or exon.

The last model, DNABERT, is an extension of the BERT architecture specialized for genomic sequences. It replaces standard word-piece tokenization with a k-mer tokenization, in our case, using 6-mers. The model was pre-trained from scratch on entire genome sequences. DNABERT can capture patterns specific to genomic syntax, such as promoter motifs or splice sites. In this work, we fine-tuned it with the same objective as BERT, leveraging its prior exposure to large-scale DNA data - which reinforces its potential for intron and exon classification.

The prompt input structure is as follows:

—**GPT-2**: Each input was structured as a prompt containing the sequence and its context. The model was trained to predict a single token indicating the class (intron or exon). A custom vocabulary was created, mapping each nucleotide (A, C, G, T) and the labels INTRON and EXON to unique tokens. A feature masking mechanism was implemented with a probability of 0.4, randomly hiding auxiliary fields. This value was empirically chosen to balance variation and generalization. For GPT-2, the datasets with sequences of 512 nucleotides was used.

—**BERT**: The input structure was similar, but a classification layer was added over the final embeddings using an argmax function. The same feature masking was applied. For this model, the dataset counting with sequences with 128 nucleotides was used.

—**DNABERT**: Specifically adapted for genomic sequences, DNABERT operates on 6-mer tokens, requiring input lengths to be multiples of 6. Padding with the N nucleotide[8] was used, which the tokenizer ignores, functioning as a neutral token. While the architecture is powerful for DNA

[8] The N nucleotide represents ambiguity and can stand for any of the four standard bases (A, C, G, or T). In DNABERT, it is ignored by the tokenizer, effectively functioning as padding.

sequences, it is less flexible for other domains. The impact of padding will be further explored in future work. Due to it's structure, none context was added. In this model, the dataset version with sequences with 128 nucleotides was used.

4.3 Input Structure

For GPT-2 and BERT models, auxiliary fields (other than the nucleotide sequence) were converted into embeddings using the model's internal vocabularies. The prompt structure, before tokenization, was:

```
sequence: [A] [A] [C] [G]...
Gene: FAA
Organism: Homo sapiens
Flank Before: [G] [A] [C] [T]...
Flank After: [C] [T] [C] [T]...
Answer:
```

4.4 Evaluation Metrics and Parameters Configuration

The main metric used was accuracy, since the dataset is balanced. Other metrics, such as F1-score and precision, may be explored in future work, especially if imbalances arise in specific dataset subsets.

Table II summarizes the used hyperparameters. The Section 5 will present the best results obtained within these configurations for each model.

Table II. Parameter configuration for each model.

Model	**Epochs**	**Batch Size**	**Learning Rate**	**Optimizer**
GPT-2	3 and 5	32	0.0005 and 0.00005	AdamW
BERT	3 and 5	32	0.00001, 0.00002 and 0.00005	AdamW
DNABERT	3, 5 and 10	32	0.00002 and 0.00005	AdamW

5. RESULTS AND DISCUSSION

The results indicate that Transformer-based models can accurately classify introns and exons, although they differ in performance and computational efficiency. BERT achieved the highest accuracy (0.9905), likely due to its effective bidirectional context modeling. DNABERT, despite being specialized for DNA sequences, underperformed compared to BERT. This may be attributed to its fixed *k-mer* representation, which hinders its ability to distinguish between ambiguous intron and exon sequences in broader contexts.

In terms of training time, GPT-2 was the slowest, possibly due to the higher computational cost of its generative architecture. Interestingly, GPT-2 yielded better accuracy after 3 epochs than with 5 epochs. DNABERT was the fastest model, highlighting the efficiency gains from its specialization for DNA; however, accuracy did not improve with more training epochs.

Beyond final accuracy, it was observed that BERT and DNABERT required reduced learning rates (at or below 5^{-5}), as higher values led to instability during training, consistent with findings reported by the original authors of these models. The same learning rate also yielded the best results for GPT-2 in this context.

Table III details the experimental configurations and presents the mean performance metrics obtained across runs.

Table III. Configurations and results for each model (average result over three runs).

Model	Learning Rate	Epochs	Overall Accuracy	Loss	Time per Epoch (minutes)
GPT-2 (117M)	5^{-5}	3	0.9867 ± 0.0041	0.0332	24 min
BERT (110M)	5^{-5}	5	0.9905 ± 0.0038	0.0347	15 min
DNABERT (110M)	5^{-5}	5	0.9569 ± 0.0002	0.1331	3 min

Based on these results, we showed that Transformer-based models are promising tools for genomic sequence analysis and, unlike previous studies, which typically focused on a narrow set of organisms and limited context around splice sites, our approach incorporates rich biological context—including organism, gene, and flanking regions—enabling more informed and generalizable predictions. By training on a broad, multi-species dataset and leveraging architectures capable of deep contextual understanding, we show that Transformers, especially BERT and GPT-2, can effectively model coding region boundaries in DNA. This paves the way for future methods that are not restricted to model organisms and highlights the potential of large-scale, context-aware sequence modeling in genomics.

6. CONCLUSION

In this study, we explored the application of Transformer-based models for identifying introns and exons in DNA sequences. Three architectures were evaluated—GPT-2, BERT, and DNABERT—each with distinct strategies for learning and context representation.

Due to hardware limitations, the experiments were conducted using a subset of 100 000 training samples. Training required a dedicated GPU, and batch sizes were limited to a maximum of 32 due to memory constraints. Future work may include the use of optimized models, such as quantized Transformers, and distributed training strategies to support larger datasets.

Among the tested models, BERT achieved the highest accuracy (0.9905), suggesting that its bidirectional attention mechanism is particularly well-suited for genomic sequence classification. Although DNABERT is specialized for DNA, its fixed *k-mer* representation may have limited its contextual flexibility, resulting in slightly lower performance. GPT-2, despite being the slowest model to train, produced competitive results, indicating that generative architectures may also hold promise for this task. In addition to accuracy, we found that all models required reduced learning rates (5^{-5}) to avoid instability and poor convergence, corroborating prior findings in the literature [Devlin et al. 2019].

Future investigations may explore alternative sequence representations, larger model variants like GPT-2 *large* (774M) or *extra large* (1.5B), hyperparameter tuning, hybrid approaches, quantized optimizations, low-rank adaptation (LoRa)[Hu et al. 2021] optimizations, and the use of even larger datasets to further improve performance on genomic tasks.

ACKNOWLEDGMENT

This work was partially supported by the National Council for Scientific and Technological Development (CNPq – 407644/2021-0 and 04515/2022-1) and the Coordination for the Improvement of Higher Education Personnel (CAPES) – Academic Excellence Program (PROEX). The authors also acknowledge the use of artificial intelligence tools, such as the ChatGPT4º model, to support the writing and textual revision of the manuscript.

REFERENCES

Albaradei, S., Magana-Mora, A., Thafar, M., Uludag, M., Bajic, V. B., Gojobori, T., Essack, M., and Jankovic, B. R. Splice2deep: An ensemble of deep convolutional neural networks for improved splice site prediction in genomic dna. *Gene* vol. 763, pp. 100035, 2020. Articles initially published in Gene: X 5, 2020.

Alberts, B., Johnson, A., Lewis, J., Morgan, D., Raff, M., Roberts, K., Walter, P., Wilson, J., and Hunt, T. *Biologia Molecular da Célula*. Vol. 6. Artmed, Porto Alegre, Brasil, 2017.

Chow, L. T., Roberts, J. M., Lewis, J. B., and Broker, T. R. A map of cytoplasmic rna transcripts from lytic adenovirus type 2, determined by electron microscopy of rna:dna hybrids. *Cell* 11 (4): 819–836, 1977.

Crick, F. H. C. On protein synthesis. *Symposia of the Society for Experimental Biology* vol. 12, pp. 138–163, 1958.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran, and T. Solorio (Eds.). Association for Computational Linguistics, Minneapolis, Minnesota, pp. 4171–4186, 2019.

Du, X., Yao, Y., Diao, Y., Zhu, H., Zhang, Y., and Li, S. Deepss: Exploring splice site motif through convolutional neural network directly from dna sequence. *IEEE Access* vol. PP, pp. 1–1, 06, 2018.

Hochreiter, S. and Schmidhuber, J. Long Short-Term Memory. *Neural Computation* 9 (8): 1735–1780, 11, 1997.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. Lora: Low-rank adaptation of large language models, 2021.

Jaganathan, K., Kyriazopoulou Panagiotopoulou, S., McRae, J. F., Darbandi, S. F., Knowles, D., Li, Y. I., Kosmicki, J. A., Arbelaez, J., Cui, W., Schwartz, G. B., Chow, E. D., Kanterakis, E., Gao, H., Kia, A., Batzoglou, S., Sanders, S. J., and Farh, K. K.-H. Predicting splicing from primary sequence with deep learning. *Cell* 176 (3): 535–548.e24, Jan, 2019.

Ji, Y., Zhou, Z., Liu, H., and Davuluri, R. V. Dnabert: Pre-trained bidirectional encoder representations for dna sequences. *Bioinformatics* 37 (15): 2112–2120, 2021.

Jónsson, B. A., Halldórsson, G. H., Árdal, S., Rögnvaldsson, S., Einarsson, E., Sulem, P., Guðbjartsson, D. F., Melsted, P., Stefánsson, K., and Úlfarsson, M. Ö. Transformers significantly improve splice site prediction. *Communications Biology* 7 (1): 1616, December, 2024.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86 (11): 2278–2324, 1998.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. Improving language understanding by generative pre-training. `https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf`, 2018. OpenAI Technical Report.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* 21 (140): 1–67, 2020.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Learning representations by back-propagating errors. *Nature* 323 (6088): 533–536, 1986.

Sarkar, R., Chatterjee, C., Das, S., and Mondal, D. Splice junction prediction in dna sequence using multilayered rnn model. In *Proceedings of the International Conference on Computer Vision and Image Processing (CVIP 2019)*, A. K. Singh, P. Choudhury, and P. P. Chattopadhyay (Eds.). Springer International Publishing, Cham, Switzerland, pp. 39–47, 2020.

Sharp, P. A. The discovery of split genes and rna splicing. *Trends in Biochemical Sciences* vol. 30, pp. 279–281, 2005.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need, 2017.

Wang, R., Wang, Z., Wang, J., and Li, S. Splicefinder: ab initio prediction of splice sites using convolutional neural network. *BMC Bioinformatics* 20 (23): 652, Dec, 2019.

Watson, J. D., Baker, T. A., Bell, S. P., Gann, A., Levine, M., and Losick, R. *Biologia Molecular do Gene*. Vol. 7. Artmed, Porto Alegre, 2015.

Zhang, D., Zhang, W., Zhao, Y., Zhang, J., He, B., Qin, C., and Yao, J. Dnagpt: A generalized pre-trained tool for versatile dna sequence analysis tasks, 2023.

Zhang, Y., Liu, X., MacLeod, J., and Liu, J. Discerning novel splice junctions derived from rna-seq alignment: a deep learning approach. *BMC Genomics* 19 (1): 971, Dec, 2018.

Zuallaert, J., Godin, F., Kim, M., Soete, A., Saeys, Y., and De Neve, W. Splicerover: interpretable convolutional neural networks for improved splice site prediction. *Bioinformatics* 34 (24): 4180–4188, 06, 2018.