

RESEARCH PAPER



## BioDeepfuse: a hybrid deep learning approach with integrated feature extraction techniques for enhanced non-coding RNA classification

Anderson P. Avila Santos<sup>a,b\*</sup>, Breno L. S. de Almeida<sup>a\*</sup>, Robson P. Bonidia<sup>a,c</sup>, Peter F. Stadler<sup>d</sup>, Polonca Stefanic<sup>e</sup>, Ines Mandic-Mulec<sup>e</sup>, Ulisses Rocha<sup>b</sup>, Danilo S. Sanches<sup>c</sup>, and André C.P.L.F. de Carvalho<sup>a</sup>

<sup>a</sup>Institute of Mathematics and Computer Sciences, University of São Paulo, São Carlos, Brazil; <sup>b</sup>Department of Applied Microbial Ecology, Helmholtz Centre for Environmental Research – UFZ GmbH, Leipzig, Saxony, Germany; <sup>c</sup>Department of Computer Science, Federal University of Technology – Paraná, UTFPR, Cornélio Procopio, Brazil; <sup>d</sup>Department of Computer Science and Interdisciplinary Center of Bioinformatics, University of Leipzig, Leipzig, Saxony, Germany; <sup>e</sup>Department of Food Science and Technology, Biotechnical Faculty, University of Ljubljana, Ljubljana, Slovenia

### ABSTRACT

The accurate classification of non-coding RNA (ncRNA) sequences is pivotal for advanced non-coding genome annotation and analysis, a fundamental aspect of genomics that facilitates understanding of ncRNA functions and regulatory mechanisms in various biological processes. While traditional machine learning approaches have been employed for distinguishing ncRNA, these often necessitate extensive feature engineering. Recently, deep learning algorithms have provided advancements in ncRNA classification. This study presents BioDeepFuse, a hybrid deep learning framework integrating convolutional neural networks (CNN) or bidirectional long short-term memory (BiLSTM) networks with handcrafted features for enhanced accuracy. This framework employs a combination of *k*-mer one-hot, *k*-mer dictionary, and feature extraction techniques for input representation. Extracted features, when embedded into the deep network, enable optimal utilization of spatial and sequential nuances of ncRNA sequences. Using benchmark datasets and real-world RNA samples from bacterial organisms, we evaluated the performance of BioDeepFuse. Results exhibited high accuracy in ncRNA classification, underscoring the robustness of our tool in addressing complex ncRNA sequence data challenges. The effective melding of CNN or BiLSTM with external features heralds promising directions for future research, particularly in refining ncRNA classifiers and deepening insights into ncRNAs in cellular processes and disease manifestations. In addition to its original application in the context of bacterial organisms, the methodologies and techniques integrated into our framework can potentially render BioDeepFuse effective in various and broader domains.

### ARTICLE HISTORY

Revised 31 October 2023  
Accepted 23 January 2024



### KEYWORDS

Non-coding RNA; deep learning; neural networks; RNA identification; feature extraction; model performance; gene regulation; biological processes

## 1. Background

In contemporary times, the ubiquity of digital applications has led to the accumulation of a vast volume of biological data, including RNA sequences. RNA is a vital component in the biology of life, requiring sophisticated and efficient approaches for comprehensive analysis and interpretation [1]. These RNA molecules can be broadly classified into two primary groups: protein-coding and non-coding RNAs. Protein-coding RNAs contribute to the translation process, facilitating the synthesis of proteins. In contrast, non-coding RNAs (ncRNAs) predominantly do not participate in protein generation. While some ncRNAs have had their functional roles identified, the biological importance of most is still not fully understood [2]. Recent advances in high-throughput sequencing technologies have led to the discovery of numerous ncRNA molecules, posing new challenges for the identification and classification of ncRNA families [3–6].

ncRNAs play crucial roles in numerous biological processes, and their dysregulation has been linked to the development and progression of various diseases and disorders. Aberrant expression of ncRNAs, such as miRNAs and lncRNAs, is implicated in the initiation, progression, and metastasis of several cancer types, including breast, lung, colorectal, and prostate cancer [7–10]. Dysregulation of ncRNAs has also been associated with neurodegenerative diseases, such as Alzheimer's disease [11], Parkinson's disease [12,13], and Huntington's disease [14], in addition to psychiatric disorders such as schizophrenia [15], bipolar disorder [16], and autism spectrum disorder [17]. Altered expression of ncRNAs is connected to cardiovascular diseases [18], including heart failure [19], myocardial infarction [20], atherosclerosis [21], and hypertension [22]. Moreover, ncRNA dysregulation has been implicated in the development of metabolic disorders, such as diabetes [23], obesity [24], and non-alcoholic fatty liver disease [25]. Additionally, ncRNAs

**CONTACT** Ulisses Rocha  [ulisses.rocha@ufz.de](mailto:ulisses.rocha@ufz.de)  Department of Environmental Microbiology, Helmholtz Centre for Environmental Research – UFZ GmbH, Permoserstraße 15, Leipzig, Saxony 04318, Germany

\*Anderson P. Avila Santos and Breno L. S. de Almeida have contributed equally to the manuscript.

© 2024 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

regulate various biological processes in plants, which can indirectly influence their response to climate change [26,27].

The identification of ncRNAs is a challenging task due to various factors, including their diverse structures, functions, and sequences [28]. Additionally, ncRNAs have lower conservation levels than protein-coding genes, making it difficult to detect homologous ncRNA sequences across different species [29]. High-throughput sequencing technologies have facilitated the discovery of novel ncRNAs [30]. However, these technologies can generate a very high amount of data, making it troublesome to accurately distinguish between functional ncRNAs and transcriptional noise [31]. One of the significant issues in identifying ncRNAs is the use of alignment-based tools, which have been primarily designed for protein-coding sequences. These tools may not perform optimally for ncRNAs, as their structures and sequence conservation patterns differ from those of protein-coding genes [32]. Consequently, developing specialized tools tailored for ncRNA identification is necessary to address these challenges. Furthermore, the lack of a clear, universally accepted definition for ncRNAs and the absence of a comprehensive database complicate their identification and classification [33]. Computational tools and algorithms for ncRNA prediction and annotation are being continually developed and refined. However, there is still a need for improvement in sensitivity, specificity, and computational efficiency [34].

Hence, these issues render biological sequence classification a complex task, leading to an increasing demand for innovative techniques and methods that can effectively and efficiently analyse sequences [35]. Machine learning (ML) algorithms have been successfully used for identifying ncRNAs due to their ability to model complex patterns and relationships within large datasets [36]. By leveraging advanced algorithms and statistical techniques, ML can capture the unique characteristics of ncRNAs, such as sequence features and structural motifs, which can be used to distinguish them from protein-coding sequences [37]. Deep learning, a subset of ML, has been recently used with good results in this area. Convolutional neural networks (CNNs) and recurrent neural networks (RNNs), variations of deep learning, enable the automatic extraction of relevant features from raw sequence data, thereby enhancing the predictive accuracy and efficiency of ncRNA identification [38,39].

The classification and analysis of ncRNA sequences have become increasingly important in understanding biological processes and disease mechanisms. Fiannaca, La Rosa, La Paglia, Rizzo, and Urso [40] developed nRC, a non-coding RNA classifier that uses features from ncRNA secondary structure and CNNs trained with deep learning algorithms.

The nRC tool classified with high predictive accuracy data from 13 different ncRNA classes, outperforming existing classification tools. On a related note, Liu et al. [39] constructed a deep learning-based model to effectively distinguish long non-coding RNAs (lncRNAs) from messenger RNAs (mRNAs), which can aid in studying diseases at the molecular level. Their model used *k*-mer embedding vectors, a bidirectional long short-term memory (BiLSTM) layer, and a CNN layer with three hidden layers to achieve classification performance superior to those obtained by traditional

methods, such as PLEK, CNCI, and CPC. Our model's design, in contrast, supports various forms of sequence encoding, providing versatility and effectiveness in analysing different types of biological sequences. This offers an advantage over models restricted to specific encodings, thus, enhancing our model's adaptability to varying biological data types and boosting its classification performance.

Chantsalnym, Lim, Tayara, and Chong [41] introduced ncRDeep, a method for predicting the class of ncRNAs using only RNA sequence information as input and employing a CNN for classification. This approach improved the average accuracy by 8.32% compared to state-of-the-art methods. Furthermore, Chantsalnym et al. [2] proposed ncRDense, a densely connected CNN architecture that extracts high-level features from RNA sequences and uses additional features, such as secondary structure and nucleotide chemical properties, to improve classification accuracy. This model outperformed existing methods when evaluated on benchmark datasets. Similarly, Liu et al. [39] presented a novel deep learning model that distinguished lncRNAs from mRNAs, achieving high F1-score, accuracy, and AUROC values. This model has the potential to detect and understand diseases associated with lncRNAs, opening up new possibilities for diagnostic and therapeutic approaches.

While deep learning methods have successfully automatically extracted features from data [42], combining external features with deep learning still needs to be adequately explored, as there is significant room for improvement in this area. Notably, Dong, Feng, Zhai, Chang, and Mai [43] achieved an accuracy exceeding 99% in classifying white blood cell images by incorporating handcrafted and deep learning features. Similarly, Chantsalnym et al. [2] demonstrated a gain of over 1% in accuracy by integrating external features in the classification of ncRNA families. These findings underscore the importance of fusing deep semantic features obtained from deep learning methods with traditional artificial features to enhance the performance of deep learning models. Nevertheless, further research and investigation into combining diverse and additional features in deep learning models are imperative to exploit their potential fully.

To decipher the complex landscape of biological data, we present a hybrid deep learning framework, BioDeepFuse. This framework synergistically integrates either CNN and/or BiLSTM networks with external features to achieve enhanced classification accuracy. Including CNN and BiLSTM networks is a strategic choice based on their proven strengths. CNN recognized for its ability to discern both local and global patterns in data, is well-suited for sequence analysis [44,45]. In contrast, BiLSTM, with its unique ability to capture long-term dependencies in sequence data [46], suits the temporal nature of RNA sequences.

Our approach employs a combination of one-hot encoding, dictionary encoding, and advanced feature extraction techniques to express the input sequences. One-hot encoding captures the compositional information of the sequences by representing nucleotides or amino acids as distinct binary vectors [47,48]. In contrast, dictionary encoding may offer compactness and simplicity by assigning a unique ordinal number to each base or residue, and the encoded sequences

can be represented using integers rather than high-dimensional vectors [49,50]. The resultant features from these encoding methods are then incorporated into the deep learning architecture. This enables the model to capitalize patterns within ncRNA sequences. Further, we employ advanced training strategies, such as dropout and batch normalization to enhance the model's learning capability and generalizability. These strategies mitigate overfitting, one of the primary challenges in training deep learning models.

In this study, we have explored various aspects to substantiate our hypothesis, as outlined below:

- **Hypothesis:** The integration of advanced deep neural networks, such as CNN and BiLSTM, with feature extraction techniques, can significantly enhance the accuracy of ncRNA classification compared to models induced by traditional machine learning algorithms. This hybrid approach may also be effective in analysing complex and diverse biological sequence data, facilitating an understanding of the biological roles and functions of ncRNA.

Our methodology will enable us to address our Research Question (RQ), and as a result, validate or refute the hypothesis, as delineated below:

- **RQ:** How effectively can the integration of advanced deep learning techniques, such as CNN and BiLSTM, with feature extraction techniques enhance the accuracy and efficiency of ncRNA classification, thereby aiding in the comprehensive analysis and understanding of diverse biological sequence data?

To evaluate our framework and to support our hypothesis and research question, we utilized both benchmark datasets and real-life laboratory RNA samples extracted using the Infernal tool. The results reflected a remarkable level of accuracy in ncRNA classification, surpassing existing methodologies and solidifying the effectiveness of our hybrid deep learning framework in handling complex ncRNA sequence data. The successful integration of either CNN or BiLSTM with advanced encoding techniques opens promising avenues for future research in ncRNA classification. It enhances our understanding of the biological roles and functions of ncRNA.

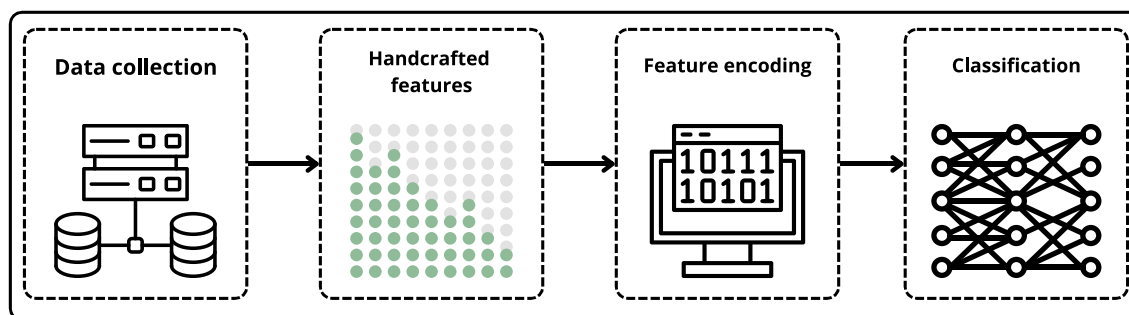
## 2. Materials and methods

In the ever-evolving field of genomics, accurate sequence analysis and characterization are essential for understanding the complexities inherent in biological systems. Our research methodology, illustrated in [Figure 1](#), provides a comprehensive framework for the classification of non-coding RNA (ncRNA). It begins with the 'Data Collection', detailing the rigorous processes and strategies we employed to curate a diverse and representative dataset. This is followed by the 'Handcrafted Features', which elucidates the domain-specific features we manually curated, harnessing our deep understanding of the dataset. Subsequently, the 'Feature Encoding' elaborates on the mechanisms we utilized to transform and represent our data optimally for machine-learning models. Finally, the 'Proposed Architectures' section presents the advanced computational models we adopted, combining both convolutional and recurrent neural network structures to achieve precise ncRNA classification. Together, these steps offer a holistic view of our approach.

### 2.1. Data collection

The methodology employed in training the algorithm involved using an extensive and diversified dataset derived from Bonidia et al. [51]. This dataset consisted of 14 different types of bacteria belonging to seven distinct phyla. This diversity ensured a rich and representative training base, allowing the algorithm to learn and generalize from various bacterial examples. Additionally, the inclusion of various phyla also helped to enhance the robustness of the model, as it exposed the algorithm to different bacterial structures and features, thus contributing to the accuracy and effectiveness of the trained model.

We elected to investigate four well-recognized ncRNA classes: sRNA, tRNA, rRNA, and cis-regulatory element. This decision was driven by their prevalent occurrence within bacterial genomes and their marked significance in analytical scenarios. For instance, tRNA and rRNA transcripts' presence can cloud the analysis of sRNA samples derived from cytoplasmic total RNA extractions, highlighting the need for precise sRNA prediction [52]. Including the cis-regulatory element not only serves as a counterpoint against other gene classes in Rfam's hierarchy [40] but also enables baseline



**Figure 1.** Process for classifying non-coding RNA (ncRNA). First, we gather a diverse dataset in the “data collection” step. Next, in “handcrafted features”, we pull out specific details from the data. Then, in “feature encoding”, we adjust these details to prepare them for the final step. Lastly, in “classification”, we use modern computer models that combine different techniques to classify RNA.

benchmarking, validating our model's efficacy. By embracing this diversity of ncRNA classes, we ensure robustness in our model, catering to both well-known and lesser-known ncRNA types, streamlining classifications in real-world applications, and enabling meaningful comparisons with prior research.

We applied the same genomic pipeline adopted in Bonidia et al. [51] to extract the ncRNA classes using the Infernal tool [53]. We accessed the Rfam Public MySQL Database (version 14.9) to obtain RNA type-associated family lists [54]. Four CM files were generated using the complete Rfam covariance model (CM) and these lists using `cmfetch`, with one file dedicated to each RNA type. We utilized `cmsearch` with the Rfam curators' selected gathering cut-off (GA) value [55] for sequence extraction of RNA types from a given genome. The extraction pipeline is illustrated in Figure 2.

To test our model, we introduced a set of 48 new bacterial genomes that had been freshly sequenced. These genomes come from real-world laboratory RNA and refer to newly extracted RNA sequences from specific experimental conditions in our partner laboratory. These sequences were not part of standard benchmark datasets and had not been classified previously, and their inclusion in our study was to validate the robustness and applicability of our hybrid deep learning framework in real-world scenarios. Using these samples provided a unique challenge and opportunity. Since these sequences were not part of standard benchmark datasets and had not been classified previously, we could evaluate our model's robustness and generalization capability. The experimental validation of our predictions added an extra layer of confidence in our model's performance.

This additional data pool was strategically chosen to provide an unbiased and rigorous evaluation of the model's capabilities. Each genome underwent the identical ncRNA extraction for the sRNA, tRNA, rRNA, and cis-regulatory elements, ensuring consistency in methodology between the training and testing phases. Introducing these new genomes

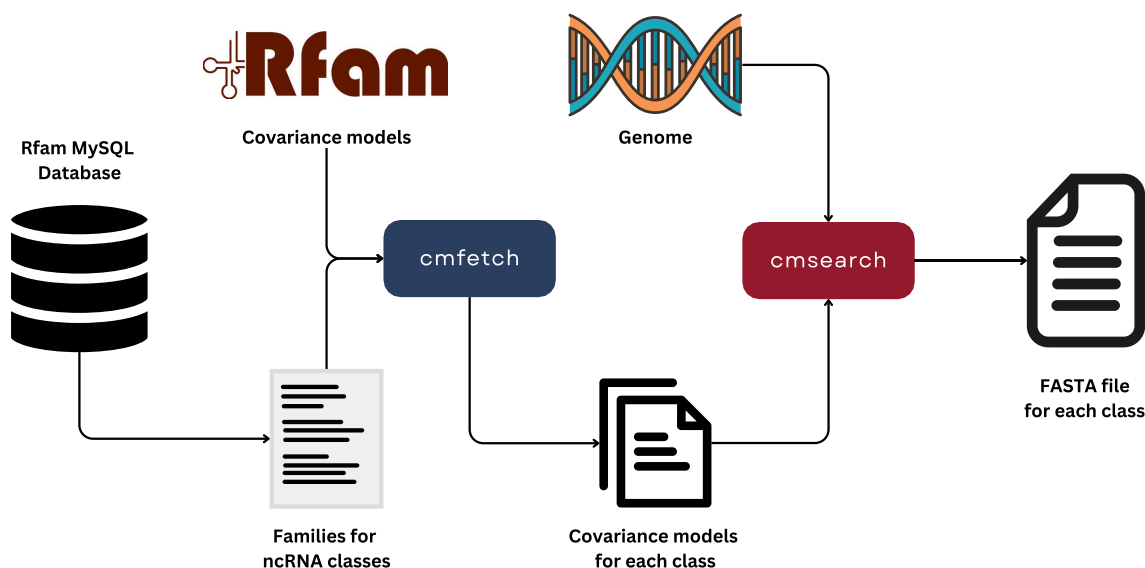
into the evaluation framework facilitated a comprehensive assessment of the model's aptitude for dealing with diverse and previously unseen biological data. All sequences extracted from these sequenced genomes and the dataset with seven phyla are discriminated in Table 1.

## 2.2. Handcrafted features and feature encoding

The principles of handcrafted features and feature encoding are fundamental to the data preparation phase in machine learning applications. The pre-processing stage allows data to be encoded through several methods, including  $k$ -mer one-hot,  $k$ -mer dictionary, or, in some cases, no encoding is applied, with the emphasis placed solely on feature extraction. Notably, our model is engineered to facilitate the extraction of a broad spectrum of features from the dataset.

Handcrafted features, often referred to as engineered features, are variables deliberately extracted from raw data based on domain-specific knowledge. Their design requires a profound understanding of the dataset and the unique problem at hand [56]. In biological sequence classification, these features tap into the intricate nuances of biological sequences, capturing patterns or characteristics potentially missed by automated models. They emerge from rigorous data analysis and iterative experimentation tailored to enhance the efficacy of a specific ML task. Especially when automated feature learning falls short, handcrafted features become invaluable, allowing domain experts to distil essential information for the task [57,58]

Our work used biological features to make a handcraft feature representation. We used descriptors often used in the literature, among them: Nucleic acid composition (NAC), dinucleotide composition (DNC), trinucleotide composition (TNC), Fickett score, Xmer  $k$ -Spaced Ymer composition frequency (kGap), and Open Reading Frames (ORF).



**Figure 2.** Illustration of the data collection pipeline that uses infernal to extract the desired ncRNA sequences from genomes for training and testing. The 'cmfetch' function retrieves covariance models from the complete Rfam database, and 'cmsearch' matches these models against a genome. The output consists of multiple FASTA files, each corresponding to a specific ncRNA class. This method is based on the genomic pipeline in Bonidia et al. [51].



**Table 1.** Number of sequences used for training and testing.

RNA type	Samples	Training	Testing
sRNA	616	497	119
tRNA	631	581	50
rRNA	567	242	325
cis-regulatory	415	246	169

- **Nucleic Acid Composition (NAC):** This feature descriptor calculates the frequency of each nucleotide (A, C, G, T/U) in a DNA or RNA sequence. It provides a simple yet effective sequence characterization, often serving as a baseline for further analysis [59].
- **Dinucleotide Composition (DNC):** The DNC measures the frequency of each possible pair of nucleotides (AA, AC, AG, ..., TT) in a sequence. The DNC represents RNA sequences based on the frequency of adjacent dinucleotides, essentially pairs of nucleotides. This method retains important information embedded in the RNA sequences or fragments [60].
- **Trinucleotide Composition (TNC):** Similar to DNC, TNC calculates the frequency of each possible triplet of nucleotides (AAA, AAC, AAG, ..., TTT) in a sequence. TNC provides even more detailed information about local sequence features, and it is particularly relevant in protein-coding regions, where each triplet (codon) corresponds to a specific amino acid or a stop signal [61].
- **Fickett score:** The Fickett score is used to identify coding regions (exons) in a DNA sequence. It is based on the observation that nucleotide usage is typically different between coding and non-coding regions. The score is calculated using position-specific base preferences and the relative frequencies of the four nucleotides (L 62).
- **Xmer *k*-Spaced Ymer Composition Frequency (kGap):** This method calculates the frequency of nucleotide patterns with *k* nucleotides intervening between Xmer and Ymer (fixed-length nucleotide patterns). It provides a way to analyse dependencies between nonadjacent nucleotides, capturing longer-range sequence features [63].
- **Open Reading Frames (ORF):** ORFs are sequences of DNA or RNA that have the potential to be translated into proteins. ORFs start with a start codon (ATG in DNA or AUG in RNA) and end with one of the three stop codons. Features derived from ORFs, such as their count, length, and position, can provide important clues about the protein-coding potential of a sequence [64].

In contrast, feature encoding is a process that converts raw data into a format more conducive to processing by ML algorithms. This could encompass relatively simple transformations, such as transforming categorical variables into numerical values through one-hot encoding. Alternatively, more complex techniques, such as *k*-mer embedding, can map subsequences of length '*k*' into a continuous vector space [65]. The central aim of feature encoding is to represent the data in a way that streamlines pattern recognition and prediction generation for machine learning models [66].

Within the realm of genomic data, one-hot encoding is a widely used approach for representing nucleotide sequences in ML models [47,48]. This encoding method establishes

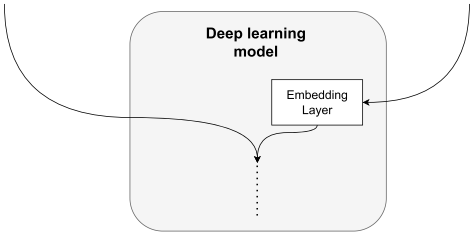
a direct correspondence between nucleotides and vectors. Each nucleotide is represented by a four-dimensional vector, where only one element is nonzero, indicating the specific nucleotide at a given position. In contrast, all other elements are set to zero. For example, Adenine (A), Cytosine (C), Guanine (G), and Thymine/Uracil (T/U) are encoded as [1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], and [0, 0, 0, 1], respectively. On the other hand, dictionary encoding assigns ordinal numbers (ranging from 0 to 3 for bases) to represent each base, providing an alternative mapping approach [49,50]. Additionally, the *k*-mer representation can be combined with these encoding methods. The representation per se involves breaking down sequences into subsequences of length *k*. For example, the sequence 'ATGC' can be segmented into 2-mers, resulting in 'AT', 'TG', and 'GC'. Each unique *k*-mer is then treated as a feature. In the case of one-hot encoding with *k*-mer (*k*-mer one-hot), binary vectors are created for each *k*-mer. In contrast, for dictionary encoding with *k*-mer (*k*-mer dictionary), each *k*-mer is mapped to a numerical value. Combining these representations with both encoding methods has been shown to be effective in deep learning models, as demonstrated in studies by Deng et al. [65]; Jing et al. [49]. Figure 3 provides an illustration of the encoding process for a sequence.

### 2.3. Proposed architectures

Our approach to classifying biological sequences is based on the innovative combination of CNNs and LSTM neural networks, leveraging their respective abilities to identify spatial and temporal patterns in complex data. This ML pipeline is designed to work with various forms of sequence encoding, such as *k*-mer one-hot and *k*-mer dictionary.

In Figure 4, we can see the developed pipeline. This begins with the data loading phase, where the training and test sets are prepared. At point A of Figure 4, the sequences are appropriately encoded and then undergo the feature extraction process. The selection of encoding methods is flexible, allowing researchers to choose the most suitable approach for the specific type of data they are analysing. After this encoding step, features that are most relevant to the learning task are extracted.

Once the data is preprocessed, it is inputted into a deep learning model comprising a series of convolutional and LSTM blocks. Notably, to facilitate *k*-mer dictionary encoding, the data undergoes an initial step of being fed into an embedding layer with the dimension of the dense embedding set to 128. The convolutional layers, which may vary in number based on the input parameters, are specifically designed to capture spatial patterns within the sequence. Each convolutional block consists of a 1D convolutional layer equipped with 128 filters and a kernel size of 3. Additionally, optional batch normalization, an activation function (such as ReLU or Leaky ReLU), and a max pooling layer are included in each block. Subsequently, the extracted information is forwarded to the LSTM layers, composed of 128 nodes, and tasked with identifying and learning long-term dependencies within the sequence. Finally, dropout layers can be applied at the end of each block to mitigate overfitting.



After the LSTM stage, the data can either be routed to point C of Figure 4 if utilizing only the CNN architecture or to point B of the same figure if applying the CNN-BiLSTM approach. To enhance both the flexibility and efficacy of the pipeline, each LSTM block can incorporate bidirectional LSTM layers and additional dropout blocks as required. The data are then flattened into a one-dimensional representation, which is fed into fully connected layers to produce the final output. This final step allows the model to generate a compact and insightful representation of the input data, simplifying the subsequent classification task. With such a robust and adaptable pipeline, the efficient classification of biological sequences becomes a practical and potent endeavour.

The model provides a valuable capability for concatenating (Figure 1 at point D) handcrafted features and integrating multiple encodings to infer effectively from diverse representations of biological sequences. This approach leads to the formation of multiple branches, each exploring a unique combination of inputs. The concatenation process can be executed immediately after the data is flattened (Concat I) or after incorporating a dense layer at the end of each branch (Concat II). The selection between direct concatenation and concatenation after a dense layer bears considerable consequences in terms of the model's performance and interpretability. In practice, in this order, we have at least two dense layers with 128 and 64 nodes before the dense layer with a softmax activation function.

We apply the softmax activation function in the final layer for prediction purposes. The Adam optimizer is used with a learning rate of  $1 \times 10^{-4}$ , while the loss function employed was categorical cross-entropy. We chose a batch size of 32 and trained the model of 100 epochs. To address overfitting, we employed early stopping during training. This technique involved monitoring the test loss over the previous 20 epochs and halting the training process if no reduction in the test loss was observed. As a result, the model's weights were restored to the best-performing configuration in such cases. The training samples were partitioned into training and validation sets using a ratio of 9:1, preserving the percentage of samples for each class. We used the Keras library to implement the model, and we set the layers and parameters empirically and based on works such as Chantsalnym et al. [2]; Jing et al. [49]; Noviello, Ceccarelli, Ceccarelli, and Cerulo [67].

### 3. Results and discussion

To ensure a comprehensive evaluation and meaningful comparison of our proposed model, we conducted experiments using classification models induced by conventional machine learning algorithms: Support Vector Machines (SVMs) and Extreme Gradient Boosting (XGBoost) [68] using only the handcrafted features. We used only handcrafted features for these algorithms, chosen for their robustness when dealing with biological data, which is known to be high-dimensional [69,70], and XGBoost being known to outperform deep models on tabular data [71].

We maintained a consistent training and validation ratio with the deep learning models by employing a 10-fold cross-

validation approach. For a fair comparison, we fine-tuned the hyperparameters of these classifiers using the Optuna library [72]. Our optimization targeted the best-weighted precision average across the folds. We performed 100 trials with Optuna, matching the number of epochs used for the deep-learning models.

For SVM, we utilized the radial basis function (RBF) kernel while varying the regularization parameter (C) and the kernel coefficient ( $\gamma$ ). As for XGBoost, we explored various hyperparameters, including the maximum depth of a tree, learning rate, number of gradient-boosted trees, minimum sum of instance weight needed in a child, minimum loss reduction required to make a further partition on a leaf node of the tree, subsample ratio of training instances, subsample ratio of columns when constructing each tree, L1 regularization, and L2 regularization. Finally, we evaluated these models using the separate test set and obtained weighted precision scores of 0.8953 and 0.9189 for SVM and XGBoost, respectively.

We thoroughly evaluated our proposed model through a series of experiments. We began by categorizing our experiments into two neural architectures: CNN (without the LSTM block) and CNN-BiLSTM (with a Bidirectional LSTM block). Then, to assess the impact of convolutional layers on classification performance, we varied the number of convolutional blocks used. Regarding the concatenation process, we explored two approaches. The first approach involved direct concatenation (Concat I), while the second employed a dense layer for each branch before concatenation (Concat II). We carefully considered these concatenation methods with handcrafted features during the experiments.

Additionally, we investigated the influence of the encoding method on performance. Specifically, we compared  $k$ -mer one-hot (Enc I) and  $k$ -mer dictionary (Enc II). Finally, we combined these encoding methods through concatenation to explore potential synergistic effects and examined whether significant improvements could be achieved. Table 2 presents the outcomes of utilizing the CNN architecture.

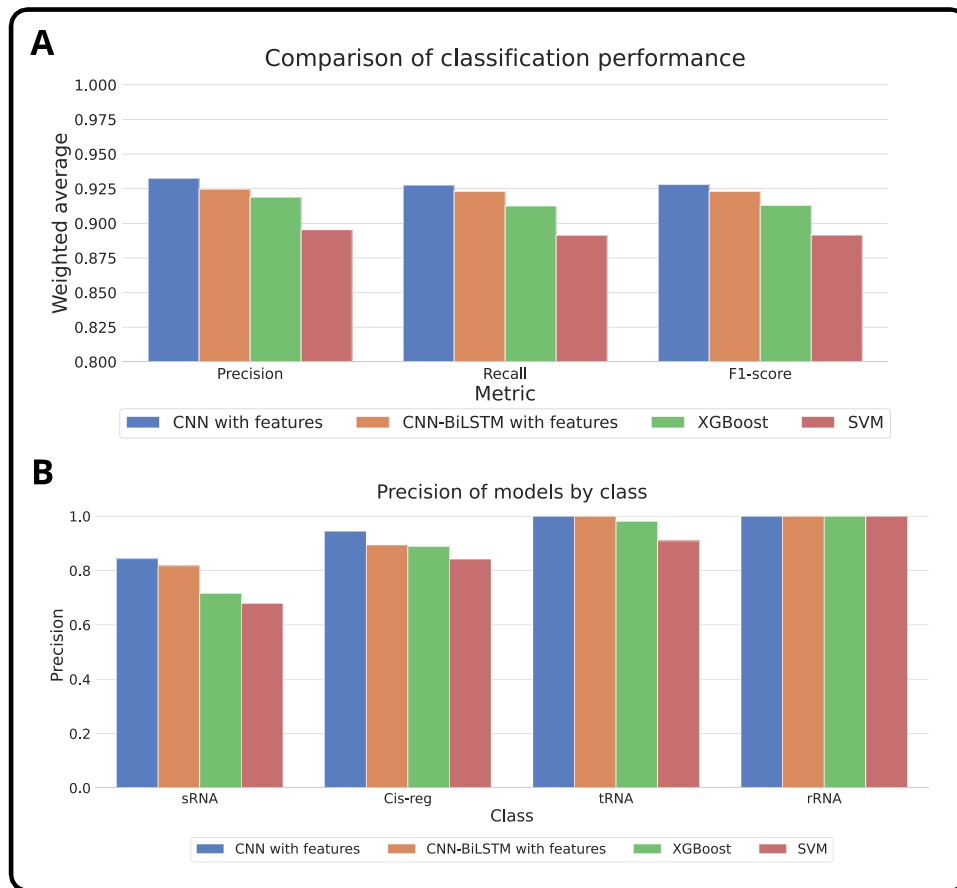
Table 2 demonstrates a crucial observation: incrementing the number of convolutional blocks enhances the model's performance. This improvement is evident in both CNN models, with and without concatenation, using handcrafted features. However, fixed convolutional block numbers across varying  $k$ -mer lengths do not necessarily yield improvements.

Furthermore, we observe the effectiveness of adopting  $k$ -mer dictionary, which consistently produces superior results across multiple experiments. When a CNN model is used without concatenation,  $k$ -mer dictionary emerges as the optimal encoding method. On the other hand, concatenation with different encoding methods did not significantly improve the performance and even worsened the performance by a considerable amount in some cases.

Figure 5(a) demonstrates the efficacy of concatenation with handcrafted features across various scenarios, irrespective of the specific concatenation approach employed. Notably, using a  $k$ -mer dictionary with direct concatenation yielded the highest weighted precision of 0.9326, outperforming the optimized XGBoost algorithm by 1.37%. Even though certain classes, such as rRNA, exhibit a reduced level of complexity in terms of classification, as depicted in Figure 5(b), the illustration

**Table 2.** Weighted precision results obtained with different variants of the CNN architecture. The models are categorized based on the encoding method ( $k$ -mer one-hot (enc I),  $k$ -mer dictionary (enc II), and their combination (enc I + enc II)), the number of convolution blocks, and the concatenation approach (direct concatenation (concat I) and concatenation with a dense layer for each branch (concat II)). Results are compared for 1-mer, 2-mer, and 3-mer configurations. Results surpassing those of traditional classifiers are highlighted in bold.

	Conv Blocks	Enc I			Enc II			Enc I + Enc II		
		CNN	Concat I	Concat II	CNN	Concat I	Concat II	CNN	Concat I	Concat II
1-mer	1	0.8602	0.8770	0.8584	0.8746	0.8826	0.8647	0.8682	0.8708	0.8938
	2	0.8738	0.8899	0.8909	0.8775	0.8615	0.8938	0.5009	0.8137	0.9051
	3	0.8822	0.9031	0.8955	0.8850	0.8684	0.9176	0.8872	0.8514	0.9052
	4	0.9077	0.8736	0.9179	<b>0.9261</b>	<b>0.9326</b>	0.8967	0.8822	0.8814	0.9051
2-mer	1	0.8505	0.8285	0.8968	0.8672	0.8079	0.8159	0.8186	0.8471	0.8390
	2	0.8304	0.8382	0.8510	0.8093	0.8390	0.8773	0.8558	0.6307	0.8523
	3	0.8729	0.8331	0.8966	0.8884	0.8801	0.9184	0.8206	0.8471	0.9123
	4	0.8504	0.8716	0.8473	0.8921	0.8736	0.8841	0.8778	0.9027	0.8824
3-mer	1	0.8585	0.8542	0.8743	0.8806	0.8766	0.9032	0.8921	0.8389	0.8894
	2	0.8835	0.8747	0.8934	0.6398	0.8317	0.8658	0.6115	0.6343	0.8897
	3	0.8902	0.8382	0.8817	0.8715	0.8462	0.8592	0.8604	0.8937	0.8912
	4	0.8837	0.8528	0.8970	<b>0.9224</b>	0.8873	0.8676	0.8550	0.8846	0.8776



**Figure 5.** Comparative performance analysis of our proposed architectures against traditional machine learning classification models (SVM and XGBoost), considering the best configurations obtained. (a) Performance comparison of the algorithms using the weighted average of metrics such as precision, recall, and F1-score. The results show consistency throughout the metrics and better performance for CNN with external features. (b) Performance comparison by class using the precision obtained with the algorithms. Even though classes such as rRNA seem to be classified effortlessly, there are clear improvements in precision in classes such as sRNA and cis-regulatory elements.

reveals a substantial improvement in the prediction of sRNA and cis-regulatory element sequences when employing deep-learning techniques in conjunction with handcrafted features.

Although direct concatenation yielded the highest weighted precision, several outcomes indicate the potential for more consistent results by incorporating a dense layer before concatenation. This shows how this approach may enable the model to actively learn complex relationships

among the features before merging them, thus enhancing the overall representation power. This approach was adopted by works such as Dong et al. [43], while others such as Chantsalnym et al. [2] used direct concatenation. Considering how these approaches may work, researchers should carefully consider the trade-offs and select the most appropriate concatenation method based on the task and dataset.



**Table 3.** Weighted precision results obtained with different variants of the CNN-BiLSTM architecture. The models are categorized based on the encoding method ( $k$ -mer one-hot (enc I),  $k$ -mer dictionary (enc II), and their combination (enc I + enc II)), the number of convolution blocks, and the concatenation approach (direct concatenation (concat I) and concatenation with a dense layer for each branch (concat II)). Results are compared for 1-mer, 2-mer, and 3-mer configurations. Results surpassing those of traditional classifiers are highlighted in bold.

	Conv Blocks	Enc I			Enc II			Enc I + Enc II		
		CNN-BiLSTM	Concat I	Concat II	CNN-BiLSTM	Concat I	Concat II	CNN-BiLSTM	Concat I	Concat II
1-mer	1	0.8797	0.8700	0.8864	0.8759	0.8700	0.8700	0.8407	0.8986	0.8847
	2	0.9020	0.8732	0.8639	0.8791	0.8929	0.8726	0.8833	0.8733	0.9062
	3	0.8375	0.9025	0.8875	0.8661	0.9157	<b>0.9247</b>	0.8430	0.9037	0.8924
	4	0.8856	0.9071	0.9109	0.8905	0.9093	0.9110	0.9061	0.9079	0.9006
2-mer	1	0.8839	0.8764	0.8896	0.8916	0.8880	0.8560	0.8906	0.8790	0.8935
	2	0.8791	0.8843	0.8179	0.8638	0.8732	0.8722	0.8669	0.8948	0.9047
	3	0.9018	0.8649	0.8912	0.8670	0.8828	0.9042	0.8966	0.8852	0.8440
	4	0.8528	0.8924	0.9046	0.8591	0.8733	0.8982	0.8210	0.8999	0.9010
3-mer	1	0.8799	0.8544	0.8695	0.8596	0.8713	0.8403	0.8993	0.8583	0.8640
	2	0.8849	0.8810	0.8587	0.8849	0.8571	0.8730	0.8904	0.8568	0.9097
	3	0.8781	0.8762	0.8643	0.8469	0.8756	0.8860	0.8359	0.8868	0.8712
	4	0.7749	0.8559	0.8624	0.8315	0.8906	0.8804	0.7811	0.8901	0.8865

Afterwards, applying the same experiments for the CNN-BiLSTM architecture, we obtained the results shown in Table 3.

Several notable similarities emerge when comparing CNN with CNN-BiLSTM. First, it becomes apparent that increasing the  $k$ -mer size does not necessarily result in improved performance, and incorporating external handcrafted features demonstrates significant gains in multiple scenarios. Specifically, the highest weighted precision achieved was 0.9247, using a  $k$ -mer dictionary with concatenation by applying dense layers at the end of each branch. While this value falls below the precision achieved with the CNN architecture, it underscores the crucial role of integrating external features to attain optimal performance.

In contrast to CNN, the CNN-BiLSTM architecture has fewer convolutional blocks and was shown to improve predictive performance in several scenarios. CNNs excel at capturing local patterns and hierarchical representations through filter convolutions applied to the input. Nevertheless, the network's complexity escalates as the CNN's depth increases, rendering it susceptible to overfitting. When combined with the BiLSTM block, this increased complexity may explain the observed decline in performance.

Furthermore, it is important to note that the CNN-BiLSTM architecture introduces an additional computational cost due to including the BiLSTM layer. The bidirectional nature necessitates processing the input sequence in both forward and backward directions, resulting in a higher number of computations than the unidirectional LSTM or the CNN alone. This increased computational complexity can pose challenges regarding training time and resource requirements.

Considering the results obtained in our study, where the highest weighted precision was achieved using the CNN architecture with external handcrafted features, it becomes evident that the simpler CNN model may be a more practical and viable choice. Moreover, the computational efficiency of the CNN allows for faster training and inference times, making it a more efficient option for similar applications.

Regarding computational efficiency, the different space complexity for working with the encoding methods proposed are notable.  $k$ -mer one-hot needs  $4^k \times L$  numbers to properly

represent a biological sequence, while  $k$ -mer dictionary needs  $L - k + 1$ , where  $L$  is the sequence length and  $k$  is the  $k$ -mer size. Despite its more concise representation, the  $k$ -mer dictionary encoding consistently yielded the highest weighted precision for both architectures. This shows how researchers can accurately represent biological sequences by employing the  $k$ -mer dictionary approach while minimizing memory requirements.

Our research employed Infernal as a fundamental tool for collecting ncRNA sequences derived from bacterial genomes. Infernal is known for its reliability and capacity to discern RNA sequences based on secondary structure features [53]. However, we chose to focus exclusively on primary structure features in our classification approach. The results of our approach demonstrated a consistently robust performance in classifying the obtained sequences, highlighting the effectiveness of our primary structure-based classification methodology.

Although BioDeepFuse was initially conceived with a primary focus on bacterial data, it is essential to recognize that our framework's foundational methodologies and techniques can extend to other domains, contingent upon the availability of pertinent training data. One salient limitation deserving explicit acknowledgement pertains to BioDeepFuse's applicability to eukaryotic organisms. This limitation arises from the distinctive complexities inherent in eukaryotic genome annotation [73].

## 4. Conclusion

In this study, we developed into the exploration of two architectures: the simpler CNN (without LSTM block) and the more intricate CNN-BiLSTM (with a Bidirectional LSTM block). We revealed how variations in the number of convolutional blocks and the incorporation of handcrafted features substantially impact model performance. This insight uncovers valuable directions for future optimizations and research.

Furthermore, we investigated two encoding methods,  $k$ -mer one-hot and  $k$ -mer dictionary, and assessed their combination using various concatenation strategies. The  $k$ -mer

dictionary encoding method consistently outshone its counterpart, establishing itself as an efficient tool for representing biological sequences with fewer memory demands. This critical finding emphasizes the substantial role of encoding methods in model design and performance, indicating an area ripe for further exploration.

Our exploration extended to integrating a dense layer before concatenation, aimed at enhancing the model's capability to discern intricate relationships among features. Although the impact on performance wasn't striking in this study, this approach still showed potential, underscoring that minute adjustments in model design can yield significant learning benefits.

Moreover, our results indicated that the simpler CNN model surpassed the more complex CNN-BiLSTM regarding weighted precision. This outcome underlines the importance of balancing model complexity and computational efficiency, and it points to the need for careful selection and fine-tuning of model architectures in accordance with task-specific requirements and resource constraints.

Moving forward, our next steps involve automating the presented methodologies, aiming to establish an end-to-end pipeline by leveraging AutoML. This transition would further increase efficiency and ensure the entire process can be executed with minimal manual intervention. Moreover, we plan to extend our research scope into multi-omics approaches, exploring metagenomic and transcriptomic data to better comprehend the myriad intricacies of biological information.

This paper offers insights that can significantly contribute to the handling of biological data by ML algorithms. The results obtained by rigorous experimentation underscore the importance of each component within a predictive model and lay a strong foundation for future advancements in this area.

## Acknowledgments

The authors would like to thank USP, CAPES, CNPq, FAPESP, AI4PEP, IDRC, FEMS, ARIS, and HIDA for the financial support for this research. We also thank Denny Popp for his assistance in acquiring the data used in the study.

## Disclosure statement

No potential conflict of interest was reported by the author(s).

## Funding

This project was supported by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior [CAPES] - Universidade de São Paulo [USP]; grant [#2023/00264-0], São Paulo Research Foundation [FAPESP]; Canada's International Development Research Centre [IDRC] - Grant No.109981; and HIDA - Helmholtz Information and Data Science Academy. The work performed by AVS was supported by a FEMS (Federation of European Microbiological Societies) research and training grant and the Helmholtz Information & Data Science Academy visiting research grant. The work performed by PS and IMM was supported by ARIS project J1-4411.

## Availability of data and materials

The documentation, pipeline, images, and results from the models are available in the GitHub repository: <https://github.com/brenoslivio/BioDeepFuse>. The complete set of 48 new bacterial genomes (and their sequences in FASTA format) are available on the long-term data archive at the Helmholtz Center for Environmental Research - UFZ data centre using the link (<https://www.ufz.de/record/dmp/archive/14024>).

## References

- [1] Chen X, Huang L. Computational model for ncRNA research. Oxford University Press; 2022. (Vol. 23) (No. 6). doi: [10.1093/bib/bbac472](https://doi.org/10.1093/bib/bbac472).
- [2] Chantsalanyam T, Siraj A, Tayara H, et al. Ncrdense: a novel computational approach for classification of non-coding RNA family by deep learning. *Genomics*. 2021;113(5):3030–3038. doi: [10.1016/j.ygeno.2021.07.004](https://doi.org/10.1016/j.ygeno.2021.07.004)
- [3] Guan S, Zhang Z, Wu J. Non-coding RNA delivery for bone tissue engineering: progress, challenges, and potential solutions. *Iscience*. 2022;25(8):104807. doi: [10.1016/j.isci.2022.104807](https://doi.org/10.1016/j.isci.2022.104807)
- [4] Panni S, Lovering RC, Porras P, et al. Non-coding RNA regulatory networks. *Biophys Acta Gene Regul Mech*. 2020;6(6):194417. doi: [10.1016/j.bbagr.2019.194417](https://doi.org/10.1016/j.bbagr.2019.194417)
- [5] Vilaça A, de Windt LJ, Fernandes H, et al. Strategies and challenges for non-viral delivery of non-coding RNAs to the heart. *Trends Mol Med*. 2023;29(1):70–91. doi: [10.1016/j.molmed.2022.10.002](https://doi.org/10.1016/j.molmed.2022.10.002)
- [6] Xu D, Yuan W, Fan C, et al. Opportunities and challenges of predictive approaches for the non-coding RNA in plants. *Front Plant Sci*. 2022;13:890663. doi: [10.3389/fpls.2022.890663](https://doi.org/10.3389/fpls.2022.890663)
- [7] Adnane S, Marino A, Leucci E. LncRNAs in human cancers: signal from noise. *Trends Cell Biol*. 2022;32(7):565–573. doi: [10.1016/j.tcb.2022.01.006](https://doi.org/10.1016/j.tcb.2022.01.006)
- [8] Anastasiadou E, Jacob LS, Slack FJ. Non-coding RNA networks in cancer. *Nat Rev Cancer*. 2018;18(1):5–18. doi: [10.1038/nrc.2017.99](https://doi.org/10.1038/nrc.2017.99)
- [9] Aprile M, Costa V, Cimmino A, et al. Emerging role of oncogenic long noncoding RNA as cancer biomarkers. *Int J Cancer*. 2023;152(5):822–834. doi: [10.1002/ijc.34282](https://doi.org/10.1002/ijc.34282)
- [10] Esteller M. Non-coding RNAs in human disease. *Nat Rev Genet*. 2011;12(12):861–874. doi: [10.1038/nrg3074](https://doi.org/10.1038/nrg3074)
- [11] Lauretti E, Dabrowski K, Praticò D. The neurobiology of non-coding RNAs and Alzheimer's disease pathogenesis: pathways, mechanisms and translational opportunities. *Ageing Res Rev*. 2021;71:101425. doi: [10.1016/j.arr.2021.101425](https://doi.org/10.1016/j.arr.2021.101425)
- [12] Rezaei O, Nateghinia S, Estiar MA, et al. Assessment of the role of non-coding RNAs in the pathophysiology of Parkinson's disease. *Eur J Pharmacol*. 2021;896:173914. doi: [10.1016/j.ejphar.2021.173914](https://doi.org/10.1016/j.ejphar.2021.173914)
- [13] Zhang H, Liu X, Liu Y, et al. Crosstalk between regulatory non-coding RNAs and oxidative stress in Parkinson's disease. *Front Aging Neurosci*. 2022;14. doi: [10.3389/fnagi.2022.975248](https://doi.org/10.3389/fnagi.2022.975248)
- [14] Tan X, Liu Y, Liu Y, et al. Dysregulation of long non-coding RNAs and their mechanisms in Huntington's disease. *J Neurosci Res*. 2021;99(9):2074–2090. doi: [10.1002/jnr.24825](https://doi.org/10.1002/jnr.24825)
- [15] Sabaie H, Moghaddam MM, Moghaddam MM, et al. Bioinformatics analysis of long non-coding RNA-associated competing endogenous RNA network in schizophrenia. *Sci Rep*. 2021;11(1):24413. doi: [10.1038/s41598-021-03993-3](https://doi.org/10.1038/s41598-021-03993-3)
- [16] Bella F, Campo S. Long non-coding RNAs and their involvement in bipolar disorders. *Gene*. 2021;796–797:145803. doi: [10.1016/j.gene.2021.145803](https://doi.org/10.1016/j.gene.2021.145803)
- [17] Tong Z, Zhou Y, Wang J. Identification and functional analysis of long non-coding RNAs in autism spectrum disorders. *Front Genet*. 2020;11:849. doi: [10.3389/fgene.2020.00849](https://doi.org/10.3389/fgene.2020.00849)
- [18] Correia CCM, Rodrigues LF, de Avila Pelozin BR, et al. Long non-coding RNAs in cardiovascular diseases: potential function as biomarkers and therapeutic targets of exercise training. *Noncoding RNA*. 2021;7(4):65. doi: [10.3390/ncrna7040065](https://doi.org/10.3390/ncrna7040065)
- [19] Poller W, Dimmeler S, Heymans S, et al. Non-coding RNAs in cardiovascular diseases: diagnostic and therapeutic perspectives.

- Eur Heart J. 2018;39(29):2704–2716. doi: [10.1093/eurheartj/ehx165](https://doi.org/10.1093/eurheartj/ehx165)
- [20] Hermann DM, Xin W, Bähr M, et al. Emerging roles of extra-cellular vesicle-associated non-coding RNAs in hypoxia: insights from cancer, myocardial infarction and ischemic stroke. *Theranostics*. 2022;12(13):5776. doi: [10.7150/thno.73931](https://doi.org/10.7150/thno.73931)
- [21] Li X, Qi H, Cui W, et al. Recent advances in targeted delivery of non-coding RNA-based therapeutics for atherosclerosis. *Molecular Therapy*; 2022.
- [22] Zahid KR, Raza U, Chen J, et al. Pathobiology of pulmonary artery hypertension: role of long non-coding rnas. *Cardiovasc Res*. 2020;116(12):1937–1947. doi: [10.1093/cvr/cvaa050](https://doi.org/10.1093/cvr/cvaa050)
- [23] Tanwar VS, Reddy MA, Natarajan R. Emerging role of long non-coding RNAs in diabetic vascular complications. *Front Endocrinol (Lausanne)*. 2021;12:665811. doi: [10.3389/fendo.2021.665811](https://doi.org/10.3389/fendo.2021.665811)
- [24] Ghafouri-Fard S, Taheri M. The expression profile and role of non-coding RNAs in obesity. *Eur J Pharmacol*. 2021;892:173809. doi: [10.1016/j.ejphar.2020.173809](https://doi.org/10.1016/j.ejphar.2020.173809)
- [25] Shabgah AG, Norouzi F, Hedayati-Moghadam M, et al. A comprehensive review of long non-coding RNAs in the pathogenesis and development of non-alcoholic fatty liver disease. *Nutri Metabol*. 2021;18(1):1–15. doi: [10.1186/s12986-021-00552-5](https://doi.org/10.1186/s12986-021-00552-5)
- [26] Sunkar R, Li Y-F, Jagadeeswaran G. Functions of microRNAs in plant stress responses. *Trends Plant Sci*. 2012;17(4):196–203. doi: [10.1016/j.tplants.2012.01.010](https://doi.org/10.1016/j.tplants.2012.01.010)
- [27] Yang H, Cui Y, Feng Y, et al. Long non-coding RNAs of plants in response to abiotic stresses and their regulating roles in promoting environmental adaption. *Cells*. 2023;12(5):729. doi: [10.3390/cells12050729](https://doi.org/10.3390/cells12050729)
- [28] Chillón I, Marcia M. The molecular structure of long non-coding RNAs: emerging patterns and functional implications. *Crit Rev Biochem Mol Biol*. 2020;55(6):662–690. doi: [10.1080/10409238.2020.1828259](https://doi.org/10.1080/10409238.2020.1828259)
- [29] Qu S, Zhong Y, Shang R, et al. The emerging landscape of circular RNA in life processes. *RNA Biol*. 2017;14(8):992–999. doi: [10.1080/15476286.2016.1220473](https://doi.org/10.1080/15476286.2016.1220473)
- [30] Micheel J, Safrastyan A, Wollny D. Advances in non-coding RNA sequencing. *Noncoding RNA*. 2021;7(4):70. doi: [10.3390/ncrna7040070](https://doi.org/10.3390/ncrna7040070)
- [31] Wang H-LV, Chekanova JA. An overview of methodologies in studying lncrnas in the high-throughput era: when acronyms attack! *Plant Long Non-Coding RNAs: Methods Protoc*. 2019;1–30.
- [32] Legeai F, Derrien T. Identification of long non-coding RNAs in insects genomes. *Curr Opin Insect Sci*. 2015;7:37–44. doi: [10.1016/j.cois.2015.01.003](https://doi.org/10.1016/j.cois.2015.01.003)
- [33] Ulitsky I. Evolution to the rescue: using comparative genomics to understand long non-coding rnas. *Nat Rev Genet*. 2016;17(10):601–614. doi: [10.1038/nrg.2016.85](https://doi.org/10.1038/nrg.2016.85)
- [34] Fang S, Zhang L, Guo J, et al. Noncodev5: a comprehensive annotation database for long non-coding rnas. *Nucleic Acids Res*. 2018;46(D1):D308–D314. doi: [10.1093/nar/gkx1107](https://doi.org/10.1093/nar/gkx1107)
- [35] Bonidia RP, Sampaio LD, Domingues DS, et al. Feature extraction approaches for biological sequences: a comparative study of mathematical features. *Brief Bioinform*. 2021;22(5):bbab011. doi: [10.1093/bib/bbab011](https://doi.org/10.1093/bib/bbab011)
- [36] Min S, Lee B, Yoon S. Deep learning in bioinformatics. *Brief Bioinform*. 2017;18(5):851–869. doi: [10.1093/bib/bbw068](https://doi.org/10.1093/bib/bbw068)
- [37] Pan X, Shen H-B. RNA-protein binding motifs mining with a new hybrid deep learning based cross-domain knowledge integration approach. *BMC Bioinf*. 2017;18(1):1–14. doi: [10.1186/s12859-017-1561-8](https://doi.org/10.1186/s12859-017-1561-8)
- [38] Chaabane M, Williams RM, Stephens AT, et al. Circdeep: deep learning approach for circular RNA classification from other long non-coding rna. *Bioinformatics*. 2020;36(1):73–80. doi: [10.1093/bioinformatics/btz537](https://doi.org/10.1093/bioinformatics/btz537)
- [39] Liu X-Q, Li B-X, Zeng G-R, et al. Prediction of long non-coding RNAs based on deep learning. *Genes (Basel)*. 2019;10(4):273. doi: [10.3390/genes10040273](https://doi.org/10.3390/genes10040273)
- [40] Fiannaca A, La Rosa M, La Paglia L, et al. Nrc: non-coding RNA classifier based on structural features. *BioData Min*. 2017;10(1):1–18. doi: [10.1186/s13040-017-0148-2](https://doi.org/10.1186/s13040-017-0148-2)
- [41] Chantsalnym T, Lim DY, Tayara H, et al. Ncrdeep: non-coding RNA classification with convolutional neural network. *Comput Biol Chem*. 2020;88:107364. doi: [10.1016/j.compbiolchem.2020.107364](https://doi.org/10.1016/j.compbiolchem.2020.107364)
- [42] Dara S, Tumma P. Feature extraction by using deep learning: a survey. In 2018 second international conference on electronics, communication and aerospace technology (iceca); 2018. p. 1795–1801).
- [43] Dong N, Feng Q, Zhai M, et al. A novel feature fusion based deep learning framework for white blood cell classification. *J Ambient Intell Humaniz Comput*. 2022;14(8):9839–9851. doi: [10.1007/s12652-021-03642-7](https://doi.org/10.1007/s12652-021-03642-7)
- [44] Kusumoto D, Yuasa S. The application of convolutional neural network to stem cell biology. *Inflamm Regen*. 2019;39(1):1–7. doi: [10.1186/s41232-019-0103-3](https://doi.org/10.1186/s41232-019-0103-3)
- [45] Li Z, Liu F, Yang W, et al. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*; 2021.
- [46] Van Houdt G, Mosquera C, Nápoles G. A review on the long short-term memory model. *Artif Intell Rev*. 2020;53(8):5929–5955. doi: [10.1007/s10462-020-09838-1](https://doi.org/10.1007/s10462-020-09838-1)
- [47] ELabd H, Bromberg Y, Hoarfrost A, et al. Amino acid encoding for deep learning applications. *BMC Bioinf*. 2020;21(1):1–14. doi: [10.1186/s12859-020-03546-x](https://doi.org/10.1186/s12859-020-03546-x)
- [48] Morales JA, Saldaña R, Santana-Castolo MH, et al. Deep learning for the classification of genomic signals. *Math Prob Eng*. 2020;2020:1–9. doi: [10.1155/2020/7698590](https://doi.org/10.1155/2020/7698590)
- [49] Jing R, Li Y, Xue L, et al. Autobiocseqpy: a deep learning tool for the classification of biological sequences. *J Chem Inf Model*. 2020;60(8):3755–3764. doi: [10.1021/acs.jcim.0c00409](https://doi.org/10.1021/acs.jcim.0c00409)
- [50] Zhang Y, Patel K, Endrawis T, et al. A fastq compressor based on integer-mapped k-mer indexing for biologist. *Gene*. 2016;579(1):75–81. doi: [10.1016/j.gene.2015.12.053](https://doi.org/10.1016/j.gene.2015.12.053)
- [51] Bonidia RP, Santos APA, de Almeida BL, et al. Bioautoml: automated feature engineering and metalearning to predict noncoding RNAs in bacteria. *Brief Bioinform*. 2022;23(4):bbac218. doi: [10.1093/bib/bbac218](https://doi.org/10.1093/bib/bbac218)
- [52] Barik A, Das S. A comparative study of sequence-and structure-based features of small RNAs and other RNAs of bacteria. *RNA Biol*. 2018;15(1):95–103. doi: [10.1080/15476286.2017.1387709](https://doi.org/10.1080/15476286.2017.1387709)
- [53] Nawrocki EP, Eddy SR. Infernal 1.1: 100-fold faster RNA homology searches. *Bioinformatics*. 2013;29(22):2933–2935. doi: [10.1093/bioinformatics/btt509](https://doi.org/10.1093/bioinformatics/btt509)
- [54] Kalvari I, Nawrocki EP, Ontiveros-Palacios N, et al. Rfam 14: expanded coverage of metagenomic, viral and microRNA families. *Nucleic Acids Res*. 2021;49(D1):D192–D200. doi: [10.1093/nar/gkaa1047](https://doi.org/10.1093/nar/gkaa1047)
- [55] Kalvari I, Nawrocki EP, Argasinska J, et al. Non-coding RNA analysis using the rfam database. *Curr Protoc Bioinf*. 2018;62(1):e51. doi: [10.1002/cpbi.51](https://doi.org/10.1002/cpbi.51)
- [56] Georgescu M-I, Ionescu RT, Popescu M. Local learning with deep and hand-crafted features for facial expression recognition. *IEEE Access*. 2019;7:64827–64836. doi: [10.1109/ACCESS.2019.2917266](https://doi.org/10.1109/ACCESS.2019.2917266)
- [57] Nanni L, Ghidoni S, Brahnam S. Handcrafted vs. non-handcrafted features for computer vision classification. *Pattern Recognition*. 2017;71:158–172. doi: [10.1016/j.patcog.2017.05.025](https://doi.org/10.1016/j.patcog.2017.05.025)
- [58] Saba T. Computer vision for microscopic skin cancer diagnosis using handcrafted and non-handcrafted features. *Microsc Res Tech*. 2021;84(6):1272–1283. doi: [10.1002/jemt.23686](https://doi.org/10.1002/jemt.23686)
- [59] Chen Z, Zhao P, Li F, et al. iLearn: an integrated platform and meta-learner for feature engineering, machine-learning analysis and modeling of DNA, RNA and protein sequence data. *Brief Bioinform*. 2020;21(3):1047–1057. doi: [10.1093/bib/bbz041](https://doi.org/10.1093/bib/bbz041)
- [60] Jia C-Z, Zhang J-J, Gu W-Z. RNA-methylpred: a high-accuracy predictor to identify n6-methyladenosine in rna. *Anal Biochem*. 2016;510:72–75. doi: [10.1016/j.ab.2016.06.012](https://doi.org/10.1016/j.ab.2016.06.012)

- [61] Tahir M, Hayat M, Kabir M. Sequence based predictor for discrimination of enhancer and their types by applying general form of chou's trinucleotide composition. *Comput Methods Programs Biomed.* **2017**;146:69–75. doi: [10.1016/j.cmpb.2017.05.008](https://doi.org/10.1016/j.cmpb.2017.05.008)
- [62] Wang L, Park HJ, Dasari S, et al. Cpat: coding-potential assessment tool using an alignment-free logistic regression model. *Nucleic Acids Res.* **2013**;41(6):e74–e74. doi: [10.1093/nar/gkt006](https://doi.org/10.1093/nar/gkt006)
- [63] Muhammod R, Ahmed S, Md Farid D. PyFeat: a python-based effective feature generation tool for DNA, RNA and protein sequences. *Bioinformatics.* **2019**;35(19):3831–3833. doi: [10.1093/bioinformatics/btz165](https://doi.org/10.1093/bioinformatics/btz165)
- [64] Sieber P, Platzer M, Schuster S. The definition of open reading frame revisited. *Trends Genet.* **2018**;34(3):167–170. doi: [10.1016/j.tig.2017.12.009](https://doi.org/10.1016/j.tig.2017.12.009)
- [65] Deng L, Wu H, Liu X, et al. Deepd2v: a novel deep learning-based framework for predicting transcription factor binding sites from combined DNA sequence. *Int J Mol Sci.* **2021**;22(11):5521. doi: [10.3390/ijms22115521](https://doi.org/10.3390/ijms22115521)
- [66] Dahouda MK, Joe I. A deep-learned embedding technique for categorical features encoding. *IEEE Access.* **2021**;9:114381–114391. doi: [10.1109/ACCESS.2021.3104357](https://doi.org/10.1109/ACCESS.2021.3104357)
- [67] Noviello TMR, Ceccarelli F, Ceccarelli M, et al. Deep learning predicts short non-coding RNA functions from only raw sequence data. *PLoS comput Biol.* **2020**;16(11):e1008415. doi: [10.1371/journal.pcbi.1008415](https://doi.org/10.1371/journal.pcbi.1008415)
- [68] ... others, Chen T, He T, et al. Xgboost: extreme gradient boosting. *R Package Version 0 4-2.* **2015**;1(4):1–4.
- [69] Ben-Hur A, Ong CS, Sonnenburg S, et al. Support vector machines and kernels for computational biology. *PLoS comput Biol.* **2008**;4(10):e1000173. doi: [10.1371/journal.pcbi.1000173](https://doi.org/10.1371/journal.pcbi.1000173)
- [70] Moon H, Ahn H, Kodell RL, et al. Ensemble methods for classification of patients for personalized medicine with high-dimensional data. *Artif Intell Med.* **2007**;41(3):197–207. doi: [10.1016/j.artmed.2007.07.003](https://doi.org/10.1016/j.artmed.2007.07.003)
- [71] Shwartz-Ziv R, Armon A. Tabular data: deep learning is not all you need. *Inf Fusion.* **2022**;81:84–90. doi: [10.1016/j.inffus.2021.11.011](https://doi.org/10.1016/j.inffus.2021.11.011)
- [72] Akiba T, Sano S, Yanase T, et al. Optuna: a next-generation hyperparameter optimization framework. In: *Proceedings of the 25th acm sigkdd international conference on knowledge discovery & data mining*; **2019**. p. 2623–2631.
- [73] Nawrocki EP. Annotating functional RNAs in genomes using infernal. *RNA Sequence, Structure, And Function: Comput Bioinf Methods.* **2014**;163–197.