Computational and Structural Biotechnology Journal 23 (2024) 2267–2276

Contents lists available at ScienceDirect

Computational and Structural Biotechnology Journal

journal homepage: www.elsevier.com/locate/csbj

Research Article

BioPrediction-RPI: Democratizing the prediction of interaction between non-coding RNA and protein with end-to-end machine learning

Bruno Rafael Florentino [a], Robson Parmezan Bonidia [a,b,*], Natan Henrique Sanches [a], Ulisses N. da Rocha [c], André C.P.L.F. de Carvalho [a]

[a] Institute of Mathematics and Computer Sciences, University of São Paulo, São Carlos, 13566-590, São Paulo, Brazil
[b] Department of Computer Science, Federal University of Technology-Paraná (UTFPR), Cornélio Procópio, 86300-000, Paraná, Brazil
[c] Department of Environmental Microbiology, Helmholtz Centre for Environmental Research-UFZ GmbH, Leipzig, Saxony, Germany

ARTICLE INFO

Keywords:
End-to-end ML
Democratizing ML
RNA-protein interaction
Interaction prediction

ABSTRACT

Machine Learning (ML) algorithms have been important tools for the extraction of useful knowledge from biological sequences, particularly in healthcare, agriculture, and the environment. However, the categorical and unstructured nature of these sequences requiring usually additional feature engineering steps, before an ML algorithm can be efficiently applied. The addition of these steps to the ML algorithm creates a processing pipeline, known as end-to-end ML. Despite the excellent results obtained by applying end-to-end ML to biotechnology problems, the performance obtained depends on the expertise of the user in the components of the pipeline. In this work, we propose an end-to-end ML-based framework called BioPrediction-RPI, which can identify implicit interactions between sequences, such as pairs of non-coding RNA and proteins, without the need for specialized expertise in end-to-end ML. This framework applies feature engineering to represent each sequence by structural and topological features. These features are divided into feature groups and used to train partial models, whose partial decisions are combined into a final decision, which, provides insights to the user by giving an interpretability report. In our experiments, the developed framework was competitive when compared with various expert-created models. We assessed BioPrediction-RPI with 12 datasets when it presented equal or better performance than all tools in 40% to 100% of cases, depending on the experiment. Finally, BioPrediction-RPI can fine-tune models based on new data and perform at the same level as ML experts, democratizing end-to-end ML and increasing its access to those working in biological sciences.

1. Introduction

With the advent of modern genetic sequencing techniques, there has been a large increase in the volume of biological sequences stored in databases [1,2]. Consequently, a diverse range of species information is cataloged within these repositories [3,4]. This accumulation of data requires developing advanced computational tools, designed for high performance, to efficiently process and extract valuable information [5]. Within the problems involving the analysis of biological sequences is the interaction between sequences, e.g., non-coding RNA (ncRNA) and protein interactions, collectively called RPIs. ncRNAs are a class of genetic material that cannot simply be categorized as part of the non-essential DNA in the genome [6], as they play a complex role with numerous functions in the organism [7].

Different structures are present in ncRNAs. Among them, are Long Non-Coding RNAs (lncRNA), which are biological structures with at least 200 nucleotides [8]. LncRNAs play a crucial role in regulating genetic expressions and chromatin, influencing not only regions close to their transcription site but also more distant regions [7]. Furthermore, it is important to note that the expression levels of some lncRNAs are directly related to initial regulation pathways of solid cancers, conferring them a significant role as biomarkers [9]. These observations underscore the prognostic importance of understanding this molecular class.

* Corresponding author at: Institute of Mathematics and Computer Sciences, University of São Paulo, São Carlos, 13566-590, São Paulo, Brazil.
E-mail addresses: bonidia@utfpr.edu.br, rpbonidia@gmail.com (R. Parmezan Bonidia), andre@icmc.usp.br (A.C.P.L.F. de Carvalho).

https://doi.org/10.1016/j.csbj.2024.05.031
Received 14 March 2024; Received in revised form 16 May 2024; Accepted 16 May 2024
Available online 22 May 2024

A computational approach that has been gaining ground in this field is Machine Learning (ML). However, their efficient use faces several challenges. One of the major complications is the nature of the data: categorical and non-structured [10]. For example, we have A1L167, a human enzyme, which serves as an example of primary protein structure. In this sequence, denoted by the letters MKELQDIARLSD..., each letter represents a unique amino acid. This sequence exemplifies the linear arrangement of amino acids, forming the backbone of the protein. In this context, addressing RNA-protein interaction problems with an ML approach requires preprocessing this data to extract relevant information to develop a predictive model. This process is known as feature engineering, typically carried out by experts, and is characterized as the most time-consuming step in ML [11].

Another significant challenge in applying ML models to biological data is the frequent lack of interpretability for biology professionals to effectively utilize these tools [12]. A model is considered a 'black box' when its complexity is so high that humans cannot interpret it, as is often the case with Deep Learning (DL) techniques [12]. The conception of the model as a "black box" raises concerns in the medical/biological context, as these models will influence decision-making, which can impact other individuals or entities [13,12].

Recently, there has been a significant increase in studies focusing on enhancing the interpretability of ML algorithm results. This has led to the development of new methodologies, such as SHAP (SHapley Additive exPlanations) [14] and LIME (Local Interpretable Model-agnostic Explanations) [15]. Applying these approaches to trained models enables a more detailed analysis by users, allowing them to understand which features influenced more the classification process and identify the global decision pattern, thereby leading to a deeper understanding and reliability regarding the predictions. Although many ML libraries and platforms are open to all users, not everyone knows how to start their studies on creating projects using ML [16]. In this context, a discussion about the democratization of Artificial Intelligence (AI) arises, involving many aspects such as democratization in model development [17], which can accelerate technological innovations [18]. This can be done by making models available in open-source, as well as using automated pipelines [17,19,18].

Considering this, we propose BioPrediction-RPI, a framework that can automatically extract and select the best features, identify the ideal ML model for each input, and adjust the best hyperparameters of this model to predict new RPIs. BioPrediction-RPI represents an end-to-end ML pipeline that encompasses all necessary steps for this type of task without human intervention. This means that even professionals who do not specialize in ML can use BioPrediction-RPI to develop models to predict new RPIs computationally. BioPrediction-RPI exclusively adopts classification models based on decision trees, intending to preserve the interpretability of the final model. Additionally, BioPrediction-RPI incorporates an interpretability module based on SHAP, providing the user with graphics that facilitate the understanding and interpretation of model decisions. This approach significantly contributes to a deeper understanding of the reasons why the model makes specific predictions, making the analysis more accessible and strengthening confidence in the obtained results. This article is guided by the following Research Question (RQ):

RQ: Is it possible to develop an end-to-end ML framework that operates without expert intervention, aiming to generate a classification and detection model for implicit interactions between pairs of sequences, for example, ncRNA-protein, exhibiting competitive performance compared to expert models?

Finally, BioPrediction-RPI can play a crucial role in democratizing ML for non-experts, aiding in the advancement of studies related to metabolism and providing a deeper understanding of pathways involved in diseases. Our proposal is available on GitHub.[1] The main contributions of this study are:

- To the best of our knowledge, the first study to propose an automated pipeline to classify interactions between biological sequences, competitive with models developed by experts;
- BioPrediction-RPI does not require specialist human assistance;
- BioPrediction-RPI can accelerate new studies, democratizing the use of ML techniques by non-experts.

2. State-of-the-art

Several models are being developed to predict RPIs in numerous datasets that leverage the most modern approaches in ML. A representative example of the state-of-the-art is the RPITER model [20], based on DL. In this study, various approaches are explored to structure and extract features from the data, encompassing features such as amino acid frequency and extending to those associated with DL, such as word2vec and doc2vec. Subsequently, the model's performance is evaluated comparatively against classical models (Random Forest and Support Vector Machine) and DL (Convolutional Neural Network (CNN) and Stacked Auto-Encoder (SAE)).

Another model is IPMiner [21], which employs a hybrid white and black box approach to predict new RPIs. Specifically, IPMiner utilizes a stacked autoencoder to extract features, followed by stacked models based on decision trees to classify molecular pairs as interactive or non-interactive. Another study, called EDLMFC [22], also employs DL approaches to classify interactions between RNA and proteins. EDLMFC uses feature derivatives for the three structural levels of the biological molecules, as input to a DL ensemble model. This ensemble combines with a bidirectional Long Short-Term Memory (BLSTM) network, enabling more accurate and comprehensive predictions in this specific context [22].

The LPI-deepGBDT, as described by Zhou et al. [23], utilizes decision trees with gradient boosting and an ML algorithm that combines multiple weak models to create a strong and robust model. Additionally, the approach complements the prediction process with a deep mapping architecture to identify implicit interactions. EnANNDeep [24] proposes an approach based on neural networks and deep decision trees, complemented by the application of an adaptive classifier based on K-nearest neighbors. LPI-BLS [25] is a tool for predicting RPIs that does not employ DL techniques. Instead, it extracts frequency features from sequences and makes predictions through a stacked ensemble of linear regression models.

To clarify this, we present Table 1, which compares the studies from the literature with our proposal using some variables. The first column (labeled "End-to-End") refers to models capable of directly inputting biological sequences in FASTA format and producing a model tailored to user data. The second column indicates the availability of interpretability reports to assist the user. Next, the third column describes whether the studies utilize DL approaches or not. Finally, the last column (Experimental Setting) denotes the number of tools compared in the validation step.

After reviewing these studies available in the literature, it is evident that most tools do not allow end-to-end ML, often requiring manual feature extraction or operating with models on servers that make predictions. However, these models are usually already trained for specific situations, meaning they do not customize a model according to user data. Additionally, we observe that not all models use black box techniques, but none of them, except for BioPrediction-RPI, have a dedicated interpretability module. Finally, we note that BioPrediction-RPI was the proposal with the largest and most diverse validation, ensuring extensive validation of its performance. Following this evaluation,

[1] https://github.com/0nurB/BioPredictionRPI-1.0.

Table 1
Categories evaluated in related works, such as end-to-end models, the presence of interpretability, the use of DL techniques, and the number of tools compared during the validation.

Tool	End-to-end	Interp. Module	DL	Validation
RPITER [20]	no	no	yes	4
IPMiner [21]	no	no	yes	4
RPISeq-RF [26]	no	no	no	1
lncPro [27]	no	no	no	1
EDLMFC [22]	no	no	yes	3
CFRP [28]	no	no	no	2
LPI-BLS [25]	no	no	no	3
LPI-CatBoost [29]	no	no	no	3
PLIPCOM [30]	no	no	no	4
LPI-SKF [31]	no	no	no	6
LPI-HNM [32]	no	no	no	1
LPI-deepGBDT [23]	no	no	yes	6
BioPrediction-RPI	yes	yes	no	12

we realize that BioPrediction-RPI fills a gap in the literature by being an inclusive framework that assists users in building a personalized model according to their data. It achieves this without requiring technical knowledge in steps such as feature engineering, model selection, hyperparameter tuning, interpretability, and more.

3. Methodology

3.1. Validation experiments

To validate BioPrediction-RPI, we compared its performance with other state-of-the-art tools that predict RPIs. There are a total of 12 different datasets, split into three different experiments. The first experiment utilize five datasets (RPI369, RPI488, RPI1807, RPI2241, and NPInter), which are available in the RPITER article [20]. The objective is to compare the BioPrediction-RPI performance with the RPITER model and other tools reported in the original article. The size of each dataset is provided in Table 2. This comparison helps us understand how the new framework stands out in terms of effectiveness in predicting interactions between RNA and proteins compared to several studies.

The datasets RPI369, RPI488, RPI1807, and RPI224, from RPITER [20], consist of a wide range of species in each dataset, ensuring a diverse set of genomes for validation in BioPrediction-RPI. For example, proteins and RNAs from various organisms such as the *Hepatitis delta virus*, *Homo sapiens*, *Aquifex aeolicus*, *Escherichia coli*, *Thermotoga maritima*, *Methanocaldococcus jannaschii*, *Bacillus subtilis*, and *Thermus thermophilus* were found in these datasets, representing mammals, bacteria, and archaea.

In the second experiment, we used alternative versions of the RPI1807 and NPInter datasets, provided by EDLMFC [22]. These alternative versions consider the same positive cases but adopt a different methodology to determine negative cases, in addition to filtering some positive interactions. The summary of the datasets for this experiment is presented in Table 3. This variation in the methodology for selecting negative cases provides a more comprehensive and robust analysis of BioPrediction-RPI's performance.

In the last experiment, the focus is to test interactions between long non-coding RNA (lncRNA) and proteins. This experiment uses five datasets, three from human RNA-protein interactions (RPIs) and the other two correspond to plant RPIs [23], as referenced in Table 4. The intention is to enable the comparison of BioPrediction-RPI's performance with six previously validated tools using the same datasets.

It is important to emphasize that our model does not aim to be the most powerful in the literature; instead, it aims to be the most accessible with a performance close to models developed by experts and/or those that make use of DL techniques. Based on this, in the first experiment, we assessed the competitiveness of BioPrediction-RPI concerning other studies, using 5-fold cross-validation, following the approach outlined in the article [20]. Since the author only reported the mean, it was used for performance comparison in a non-parametric Mann-Whitney U test, with a significance level (alpha) of 0.05, comparing the means of all metrics at once. The null hypothesis was that the mean of a given study was not statistically superior to that of BioPrediction-RPI (i.e., BioPrediction-RPI was competitive), and the alternative hypothesis was that the mean of a certain study was superior to that of BioPrediction-RPI.

Furthermore, in experiments two and three, where mean, and standard deviations were reported, a one-tailed hypothesis t-test was conducted. The null hypothesis posits that the mean of the first sample is equal to the mean of the second sample, while the alternative hypothesis suggests that the mean of the first sample is greater than the mean of the second sample, in which BioPrediction-RPI is the second sample in all cases. If the mean of the compared tool is greater than that of BioPrediction-RPI in more than half of the metrics, our proposal will be considered non-competitive; otherwise, it will be considered competitive. It is important to highlight that in all experiments, the mean and standard deviation resulted from 20 executions of our proposal.

Table 2
Summary of datasets in the first experiment.

Dataset	Interaction pairs	Non-interaction pairs	RNAs	Proteins
RPI369	369	369	332	338
RPI488	243	245	25	247
RPI1807	1807	1436	1078	3131
RPI2241	2241	2241	841	2042
NPInter	10412	10412	4636	449

Table 3
Summary of datasets in the second experiment.

Dataset	Interaction pairs	Non-interaction pairs	RNAs	Proteins
NPInter	1943	1943	513	448
RPI1807	652	221	646	868

Table 4
Summary of datasets in the third experiment.

Dataset	Interaction pairs	Non-interaction pairs	RNAs	Proteins
1	3480	51686	935	59
2	3265	71075	885	84
3	4158	22572	990	27
4	948	2867	109	35
5	22133	49435	1704	42

3.2. Workflow: BioPrediction-RPI

BioPrediction-RPI has an automated workflow for building an end-to-end ML pipeline to predict interactions, along with a report designed to explore some characteristics of the model. To initiate the model construction, it is essential to input the path to the data, which consists of three main files: the list of known interactions and dictionaries containing the sequences for all proteins and RNAs. The list of interactions is shown in a table with three columns, where the first column contains protein names, the second column contains RNA names, and the third column indicates whether there is an interaction or not. The dictionaries are files in FASTA format containing all the primary sequences of the biological sequences involved in the problem.

Afterward, the feature extraction module starts, where features are obtained to characterize each biological sequence. More specifically, there are two main types of features: structural and topological. Structural features refer to those extracted directly from the primary sequence of each molecule, including examples such as amino acid fre-

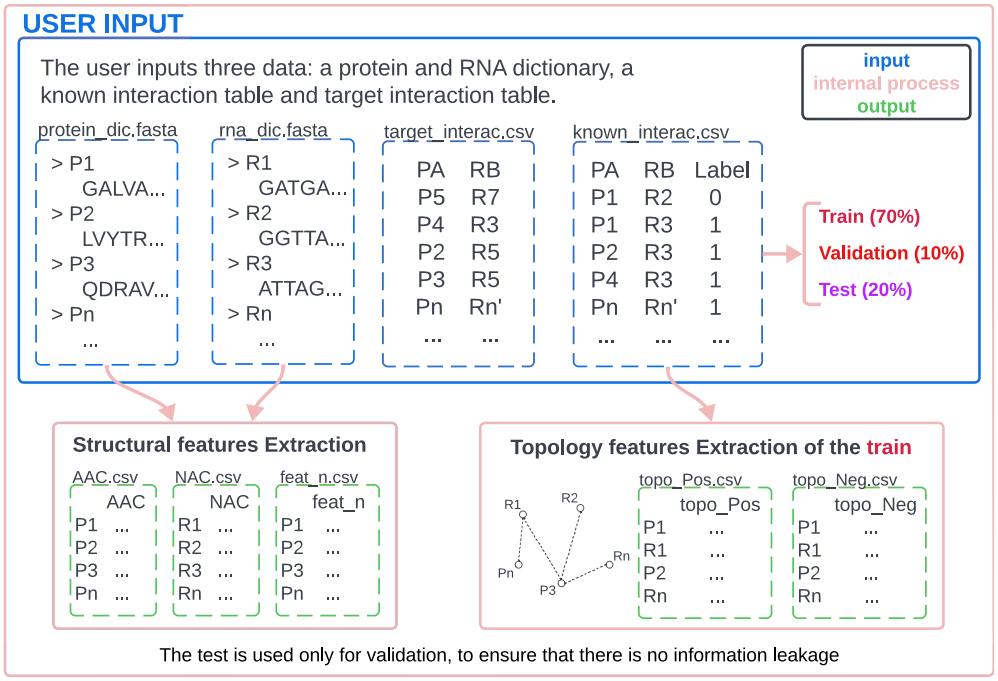

Fig. 1. The initial part of the workflow, in blue, represents the inputs, while the pink indicates the internal processes. We observe the interactions being divided into their subsets, and the primary sequences are used to extract structural features, while the train set is used to derive topological features.

Table 5
Feature subsets to train the partial models.

Protein Subset 1	AAC, DPC, FFT_mean
RNA Subset 1	NAC, DNC, TNC
Protein Subset 2	FFT_H1, FFT_P1, FFT_V
RNA Subset 2	Rev_kmer, Pse_dnc, Pse_tnc
Protein Subset 3	FFT_H2, FFT_P2, FFT_NCI, FFT_SASA
RNA Subset 3	SCPseDNC, SCPseTNC
Protein Subset 4	Red_alphabet
RNA Subset 4	QNC
Protein and RNA Subset 5	Hub score, authority score, spectrum, degree centrality, etc.

quencies, Shannon entropy, and those utilizing physicochemical properties of each amino acid, such as hydrophobicity (H1), used to construct a numeric signal and treated with Fast Fourier Transform (FFT) for sequence characterization over a series of frequencies. On the other hand, topological features are derived from the interaction network present exclusively in the training set. These features include the number of interactions and other graph measures, such as centrality and betweenness. Thus, each RNA and protein mentioned in the problem has a set of numerical columns that characterize their various properties. These initial steps can be seen in Fig. 1.

Next, datasets are constructed for the modeling stage by concatenating the features of proteins and RNAs with the interaction table, creating a table where each row contains the features of the sequences and the label associated with that pair. In total, 5 subsets of features were created, four exclusively with structural features and one exclusive for topological features, with these sets presented in Table 5.

More specifically, we have several structural features based on frequency, such as AAC, DPC, NAC, DNC, TNC, Pseudo-DNC, Pseudo-KNC [33], SCPseDNC, SCPseTNC [34], Reduced Alphabet, QNC [20]. Additionally, there are some features for proteins based on physicochemical properties and Fourier transform, such as those evaluating hydrophobicity (H1), hydrophilicity (H2), side-chain length (V), polarity (P1), polarizability (P2), solvent-accessible surface area (SASA), and net charge index (NCI) [35]. Measures of complex networks are also included, both with only positive interactions and with positive and negative interactions. At this point, if the data has less than 20% positive samples, BioPrediction-RPI uses the training data concatenated with the first set of features to evaluate the best undersampling technique to balance the data. Three implemented techniques are available: random undersampling, cluster centroids, and near miss.

The subsequent step involves training partial models for each feature set, aiming to reduce the dimensionality of the problem and, consequently, improve efficiency in the final execution. In this process, the training set is used for model construction, and the validation set is utilized for performance evaluation. After developing these partial models, the probability of an instance belonging to the interaction class is employed as the new compressed feature. For example, consider using amino acid composition (AAC) to build a partial model, which initially includes 20 columns. After processing, the predicted probability derived from this partial model serves as the newly compressed AAC feature. However, in this case, it is used in conjunction with more than one descriptor, as can be seen in Table 5.

This procedure is repeated for all feature sets, leading to the compilation of a final dataset that contains all compressed features. This dataset is then used in the final training phase to integrate partial decisions into a conclusive decision. At this stage, the validation set is utilized for training purposes, and the test set is employed to assess the performance of the final model. This approach ensures that there is no information leakage, as the test set remains unseen until the final evaluation. In Fig. 2, the workflow of the partial models is also schematically outlined.

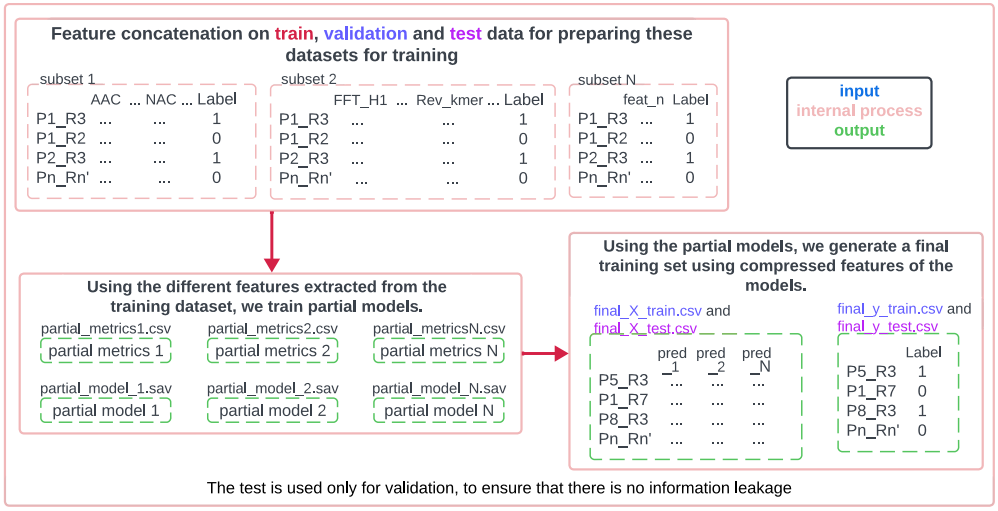

Fig. 2. In pink, the internal processes, while in green, the outputs of BioPrediction-RPI. We can observe the formation of feature groups for training the partial models and how the partial predictions are utilized to compile the final training and testing sets.

Table 6
Hyperparameter Tuning in BioPrediction-RPI.

Hyperparameters	Decision Tree	Random Forest	CatBoost	XGBoost
Criterion	gini, entropy	gini, entropy	-	-
Depth/Max_Depth	None, 1, 2, 4, 6, 8	None, 1, 2, 4, 6, 8	None, 1, 2, 4, 6, 8	None, 1, 2, 4, 6, 8
Min_Samples_Split	5, 10, 20	5, 10, 20	-	-
Min_Samples_Leaf	5, 10, 20	5, 10, 20	-	-
Learning_Rate	-	-	0.01, 0.05, 0.1	0.01, 0.05, 0.1
Scale_Pos_Weight	-	-	1, 2, 3	1, 2, 3
Class_Weight	None, balanced	-	-	-
Max_Features	None, sqrt, log2	-	-	-
L2_Leaf_Reg	-	-	1, 3, 5	-

Finally, the model is constructed using the training sets to make the definitive decision regarding the classification of each interaction pair into its respective class. Both models are based on decision trees, such as Random Forest, Catboost, and XGBoost. However, only the final model involves tuning the hyperparameters. For this, a random search approach is employed with 100 iterations. In each iteration, cross-validation with 5 folds is conducted, using the F1-score as the evaluation metric. The hyperparameter ranges are shown in Table 6. Once the model is ready, an interpretability report based on the SHAP Values library [14] is generated to elucidate the decision-making process, and a usability report is created to clarify the metrics and properties of the model to the user.

3.3. Interpretability module

BioPrediction-RPI is tasked with generating an explanatory report for the end user, which succinctly outlines the key considerations of the trained model in the classification process. Additionally, it proposes an analysis of the most influential features based on a specific sample of inputs for each class. Thereby, each class is individually analyzed, providing a deep understanding of the underlying reasons behind the model's predictions. Additionally, the interpretability module proposes to analyze how the magnitude of the values of each feature influences the model's decision. These results are summarized through illustrative graphs in the report along with the model, offering a clear and informative visual representation of the relationships between the features and the model's decisions. This feature enhances the understanding and interpretation of the model by the end user.

To conduct this analysis, the SHAP method [14] was adopted as an interpretation methodology, responsible for consolidating various other methods existing in the literature, such as LIME [15]. This method presents an exclusive module aimed at tree-based algorithms [36], which produces consistent results in conjunction with the models trained by BioPrediction-RPI. SHAP (SHapley Additive exPlanations) is an interpretation methodology for ML models that is grounded in Game Theory. It uses the Shapley model to assign a contribution metric to each feature analyzed in classification tasks.

This model facilitates the extraction of Shapley values, which are numerical coefficients that quantify the individual contribution of each feature when it collaborates with two or more other features in the model. In the context of ML, the features used in prediction models are analogous to players in the Shapley model. The Shapley values calculated therefore quantify the individual contributions of each feature, taking into account various combinations or coalitions of features. This allows for a detailed understanding of how each feature influences the model's classification decisions, providing insights into the relative importance and impact of each feature within the model.

4. Results

Next, a compilation of experimental results is presented, elucidating the comparative performance of BioPrediction-RPI with literature studies. In the first experiment, we assessed the performances of the datasets RPI369, RPI488, RPI1807, RPI2241, and NPInter, as shown in Table 7. Initially, in all datasets, BioPrediction-RPI demonstrated competitiveness with all studies, both white-box and black-box models, according to the Mann-Whitney U one-sided test, with a significance level (alpha) of 0.05. In other words, even though there might be individual metrics with higher values when compared individually, there is no significant difference between the models as a whole. Furthermore, in

Table 7
Performance of the models in Experiment 1, measuring accuracy (ACC), precision (Pre), recall (Rec), specificity (Spec), Matthews correlation coefficient (MCC), and Area Under the Curve (AUC). All comparisons use the same data set.

Dataset	Study	ACC	Pre	Rec	Spec	MCC	AUC
RPI369	RPITER	72.8	70.1	79.7	65.9	46.1	82.1
	IPMiner	70.0	**84.0**	78.4	56.0	42.8	70.0
	RPISeq-RF	71.3	72.4	71.6	70.2	42.6	71.3
	lncPro	50.2	51.2	23.7	**77.1**	00.9	46.8
	BioPrediction-RPI	**79.1 ± 2.0**	75.8 ± 3.0	**88.7 ± 3.1**	69.2 ± 6.0	**60.4 ± 4.1**	**89.5 ± 1.9**
RPI488	RPITER	**89.3**	94.3	83.9	**94.7**	**79.3**	91.1
	IPMiner	**89.3**	**95.1**	**94.6**	83.5	**79.3**	89.3
	RPISeq-RF	88.3	93.5	92.8	83.1	77.1	88.3
	lncPro	85.6	94.0	77.0	**94.7**	72.5	**92.9**
	BioPrediction-RPI	88.7 ± 1.4	92.2 ± 2.3	84.8 ± 0.8	92.5 ± 2.7	78.0 ± 2.6	90.1 ± 1.5
RPI1807	RPITER	96.8	95.9	**98.6**	94.6	93.6	**99.0**
	IPMiner	96.8	95.5	96.5	**97.8**	93.5	96.6
	RPISeq-RF	**97.0**	96.2	97.0	97.6	**93.9**	96.9
	lncPro	47.2	53.2	44.5	50.6	-4.9	50.6
	BioPrediction-RPI	95.3 ± 0.2	**96.3 ± 0.5**	95.3 ± 0.4	95.3 ± 0.7	90.5 ± 0.4	98.3 ± 0.3
RPI2241	RPITER	**89.0**	87.1	**91.7**	86.3	**78.1**	**95.7**
	IPMiner	86.1	**88.2**	87.7	84.1	72.4	86.1
	RPISeq-RF	85.1	86.3	86.1	83.8	70.2	85.1
	lncPro	60.6	63.2	51.8	69.5	21.6	64.4
	BioPrediction-RPI	84.8 ± 0.3	86.3 ± 1.0	82.9 ± 0.8	**86.7 ± 1.3**	69.8 ± 0.7	92.4 ± 0.2
NPInter	RPITER	95.5	93.9	**97.3**	93.7	91.0	98.5
	IPMiner	**95.7**	**95.6**	95.6	**95.8**	**91.4**	95.7
	RPISeq-RF	94.3	93.6	93.7	94.9	88.5	94.3
	lncPro	50.8	50.5	73.9	27.6	1.7	51.7
	BioPrediction-RPI	95.3 ± 0.1	94.8 ± 0.1	95.8 ± 0.1	94.7 ± 0.1	90.5 ± 0.1	**98.5 ± 0.1**

Table 8
Performance of the models in Experiment 2. All comparisons use the same data set.

Dataset	Tool	ACC	Pre	Rec	Spec	F1	MCC	AUC
RPI1807	EDLMC	93.8 ± 0.3	94.9 ± 0.3	96.9 ± 0.3	84.5 ± 0.9	95.9 ± 0.2	83.3 ± 0.8	96.7 ± 0.3
	RPITER	93.5 ± 0.4	94.3 ± 0.3	97.1 ± 0.4	82.7 ± 1.1	95.7 ± 0.2	82.4 ± 1.0	97.7 ± 0.3
	IPMiner	93.5 ± 0.3	92.7 ± 0.7	**99.2 ± 0.4**	76.8 ± 2.4	95.8 ± 0.2	82.6 ± 0.9	88.0 ± 0.3
	CFRP	92.8 ± 0.4	77.4 ± 0.6	97.6 ± 0.4	77.4 ± 0.6	95.2 ± 0.2	79.7 ± 0.9	96.4 ± 0.1
	BioPrediction-RPI	**94.9 ± 1.2**	**96.8 ± 0.9**	96.3 ± 0.9	**90.5 ± 2.5**	**96.5 ± 0.8**	**86.8 ± 2.9**	**97.2 ± 1.3**
NPInter v2.0	EDLMC	**89.7 ± 0.2**	88.2 ± 0.3	**91.7 ± 0.4**	87.7 ± 0.4	**89.9 ± 0.2**	**79.5 ± 0.4**	**95.9 ± 0.2**
	RPITER	89.0 ± 0.6	87.0 ± 0.8	91.6 ± 0.6	86.2 ± 0.1	89.3 ± 0.6	78.1 ± 1.2	95.7 ± 0.4
	IPMiner	82.8 ± 1.0	81.3 ± 1.3	84.3 ± 0.9	81.3 ± 1.3	83.2 ± 0.9	65.6 ± 2.0	82.7 ± 1.0
	CFRP	82.1 ± 1.0	81.1 ± 0.3	77.2 ± 0.5	86.9 ± 0.3	81.1 ± 0.3	64.4 ± 0.5	88.4 ± 0.2
	BioPrediction-RPI	88.5 ± 0.2	**90.1 ± 0.4**	86.4 ± 0.2	**90.7 ± 0.4**	88.1 ± 0.2	77.1 ± 0.4	93.7 ± 0.1

total, distributed among the 4 studies and 5 datasets, 120 metrics were tested, and only 29% of these metrics showed performance superior to BioPrediction-RPI by more than 1%. This finding represents the first evidence that BioPrediction-RPI is capable of competing with models developed by experts.

In the second experiment (see Table 8), the versions of EDLMC in the datasets RPI1807 and NPInter were evaluated. In the RPI1807 dataset, BioPrediction-RPI demonstrated competitiveness with all four evaluated models. However, in the NPInter dataset, our proposal was competitive with 2 of the four models, losing only to EDLMC and RPITER. In total, distributed among the 4 tools and 2 datasets, 48 metrics were evaluated. Only 11 of these metrics, when tested using the hypothesis t-test, showed, on average, a superior performance to BioPrediction-RPI, representing only 20% of the total metrics evaluated. In other words, once again, BioPrediction-RPI is competitive in certain situations compared to models built by experts.

In the third experiment (see Table 9), we evaluated five datasets of lncRNA-protein interactions. In the first three datasets, BioPrediction-RPI did not demonstrate competitiveness with any of the 6 tools used in the comparison. These datasets showed the highest class imbalance, with 6%, 4%, and 16% of positive samples, respectively. On the other hand, BioPrediction-RPI demonstrated competitiveness with all tools in datasets 4 and 5, which had the highest proportion of positive samples at 25% and 31%, respectively. That is, BioPrediction-RPI performs better on datasets without significant class imbalances, which already suggests a potential improvement in future versions of the tool. In total, among the 6 studies and 5 datasets, 180 metrics were assessed. However, only 40% of them were superior to BioPrediction-RPI, as observed by the hypothesis test. This represents further evidence that BioPrediction-RPI is capable of competing with manually developed models by experts in certain scenarios.

4.1. Imbalanced data and negative sample

Analyzing datasets, especially from experiment 3, focusing on how negative samples are generated, can provide crucial insights, especially in the context of imbalanced data. In these datasets, there are R RNAs and P proteins, resulting in $R \times P = A$ possible combinations. Additionally, each dataset includes a validated set Cp with positive interactions. In Zhou et al. [23], all possible interactions that are not in Cp are con-

Table 9
Performance of the models in Experiment 3, measuring accuracy (ACC), precision (Pre), recall (Rec), F1-Score, Area Under the Curve (AUC), and Area Under the Curve precision-recall (AUPR). All comparisons use the same data set.

Datasets	Metrics	LPI-BLS	LPI-CatBoost	PLIPCOM	LPI-SKF	LPI-HNM	LPI-deepGBDT	BioPrediction-RPI
dataset 1	Precision	84.58 ± 0.14	83.17 ± 1.32	84.28 ± 0.60	**87.57 ± 0.86**	70.06 ± 1.71	84.57 ± 0.46	53.94 ± 2.93
	Recall	65.50 ± 0.09	83.31 ± 1.40	**96.32 ± 0.28**	59.32 ± 1.56	71.34 ± 1.52	94.56 ± 0.70	55.95 ± 5.26
	ACC	75.12 ± 0.05	83.10 ± 0.71	89.17 ± 0.39	72.54 ± 0.32	65.71 ± 1.12	89.64 ± 0.32	**93.75 ± 0.38**
	F1	73.81 ± 0.12	83.14 ± 0.67	**89.89 ± 0.33**	62.98 ± 0.70	70.69 ± 1.48	89.27 ± 0.31	52.73 ± 1.62
	AUC	91.92 ± 0.05	88.60 ± 0.48	93.13 ± 0.30	93.44 ± 0.73	77.74 ± 1.47	93.46 ± 0.40	**95.04 ± 0.51**
	AUPR	88.51 ± 0.22	89.36 ± 0.49	**92.24 ± 0.37**	91.96 ± 0.92	82.60 ± 1.80	88.89 ± 0.91	56.35 ± 1.13
dataset 2	Precision	85.47 ± 0.31	82.20 ± 1.39	85.37 ± 0.65	**86.27 ± 2.23**	70.09 ± 1.69	85.67 ± 0.38	48.34 ± 4.35
	Recall	67.38 ± 0.13	83.99 ± 2.01	**96.28 ± 0.43**	52.12 ± 1.07	68.93 ± 1.46	94.95 ± 0.63	54.78 ± 6.85
	ACC	76.20 ± 0.18	82.58 ± 0.64	89.87 ± 0.34	70.65 ± 0.81	64.74 ± 0.88	89.52 ± 0.24	**95.10 ± 0.28**
	F1	75.33 ± 0.20	82.82 ± 0.67	90.48 ± 0.27	58.28 ± 1.17	69.49 ± 1.40	**91.05 ± 0.24**	48.62 ± 2.15
	AUC	93.01 ± 0.17	89.09 ± 0.44	93.89 ± 0.34	91.99 ± 1.49	76.77 ± 1.33	93.98 ± 0.28	**96.01 ± 0.05**
	AUPR	89.75 ± 0.32	89.29 ± 0.50	92.66 ± 0.44	87.87 ± 2.60	80.39 ± 1.87	**89.91 ± 0.68**	52.56 ± 1.22
dataset 3	Precision	71.10 ± 0.11	68.71 ± 0.60	71.73 ± 0.84	**72.98 ± 1.53**	70.54 ± 1.69	70.89 ± 1.15	39.64 ± 3.47
	Recall	62.70 ± 0.06	61.54 ± 2.41	76.18 ± 1.41	62.26 ± 0.58	69.30 ± 1.13	**76.49 ± 2.49**	45.59 ± 8.42
	ACC	66.05 ± 0.12	66.77 ± 0.91	72.98 ± 0.34	65.44 ± 0.92	65.85 ± 0.97	72.36 ± 0.43	**91.76 ± 1.60**
	F1	66.63 ± 0.08	64.80 ± 1.48	**73.77 ± 0.34**	59.50 ± 0.86	69.91 ± 1.19	73.37 ± 0.68	39.32 ± 2.11
	AUC	78.49 ± 0.20	71.51 ± 1.21	82.23 ± 0.29	81.17 ± 1.59	77.94 ± 1.26	80.83 ± 0.42	**88.78 ± 1.00**
	AUPR	74.69 ± 0.06	70.24 ± 1.09	**80.60 ± 0.44**	77.72 ± 1.98	80.39 ± 1.61	77.92 ± 0.70	44.26 ± 2.77
dataset 4	Precision	56.53 ± 0.88	46.13 ± 3.69	48.94 ± 5.08	61.08 ± 2.49	66.24 ± 5.01	58.70 ± 2.89	**70.94 ± 2.53**
	Recall	53.28 ± 0.74	35.39 ± 7.00	31.90 ± 6.68	60.56 ± 2.80	63.42 ± 3.96	36.13 ± 4.53	**76.65 ± 3.40**
	ACC	54.24 ± 0.48	48.01 ± 2.01	49.72 ± 3.06	57.27 ± 1.96	61.00 ± 2.74	55.06 ± 1.67	**86.08 ± 0.64**
	F1	54.83 ± 0.81	38.12 ± 5.73	37.83 ± 5.97	54.01 ± 2.32	64.80 ± 4.45	43.97 ± 3.62	**73.31 ± 0.60**
	AUC	58.43 ± 0.94	47.26 ± 2.70	48.91 ± 3.26	64.79 ± 3.79	70.38 ± 4.38	57.90 ± 2.07	**91.44 ± 0.36**
	AUPR	**85.79 ± 0.36**	82.74 ± 0.79	49.87 ± 2.72	63.48 ± 3.40	74.35 ± 6.89	59.65 ± 1.76	76.70 ± 0.52
dataset 5	Precision	79.01 ± 0.21	77.13 ± 0.40	77.21 ± 0.21	75.17 ± 0.98	79.59 ± 1.57	80.18 ± 1.89	**82.68 ± 0.70**
	Recall	70.63 ± 0.38	79.21 ± 1.35	85.69 ± 0.37	67.27 ± 0.37	66.82 ± 0.77	84.25 ± 2.61	**86.97 ± 0.69**
	ACC	73.37 ± 0.25	77.85 ± 0.67	80.18 ± 0.18	67.26 ± 0.36	71.17 ± 0.53	81.29 ± 1.32	**90.30 ± 0.12**
	F1	54.67 ± 2.50	79.70 ± 1.84	79.20 ± 0.71	59.08 ± 7.34	75.37 ± 2.90	81.15 ± 0.84	**84.70 ± 0.09**
	AUC	50.13 ± 0.25	87.17 ± 1.33	85.44 ± 0.63	80.00 ± 11.36	89.59 ± 2.12	88.02 ± 1.72	**96.31 ± 0.04**
	AUPR	73.08 ± 0.46	84.71 ± 1.64	81.87 ± 1.19	76.00 ± 16.57	88.36 ± 5.63	86.43 ± 2.53	**86.84 ± 0.13**

Table 10
Experiment 3 with new negative samples.

Dataset	Precision (%)	Recall (%)	ACC (%)	F1 (%)	AUC (%)	AUPR (%)
dataset 1	70.10 ± 0.05	98.71 ± 0.18	78.30 ± 0.02	81.96 ± 0.03	81.11 ± 0.10	84.73 ± 0.02
dataset 2	77.59 ± 0.10	96.56 ± 0.24	84.33 ± 0.05	85.60 ± 0.05	88.25 ± 0.19	87.92 ± 0.03
dataset 3	61.74 ± 1.71	82.53 ± 2.94	93.28 ± 0.27	69.97 ± 0.99	96.31 ± 0.41	73.00 ± 0.97
dataset 4	83.45 ± 0.68	84.17 ± 0.70	83.63 ± 0.31	83.57 ± 0.36	90.72 ± 0.37	87.78 ± 0.30
dataset 5	87.45 ± 0.62	93.02 ± 0.58	89.80 ± 0.14	90.13 ± 0.07	96.43 ± 0.24	91.98 ± 0.17

sidered negative samples, forming the negative sample set, Cn, so that $Cp + Cn = A$. Since the corresponding study requires an input matrix of size (R, P), which includes all labeled interactions, the author must incorporate all unknown interactions into the negative sample set.

However, this approach diverges with other methods of generating the negative set, as typically, in state-of-the-art practices, negative sets are created primarily to balance the datasets [37,26], avoiding adding possibly positive interactions without any criteria that hinder the model's learning. In this context, exclusion criteria are often applied to prevent the inclusion of false-negative interactions in the dataset. The main criteria involve adding only negative pairs that have different subcellular localization information [38–43] or only add negative pair with minimum structural dissimilarity with positive pairs (reducing the homologous sequences bias) [21,20,26].

Therefore, the datasets from Zhou et al. [23] may not fully represent biological sequence networks. The author includes numerous negative interactions that have not been experimentally verified, without applying specific exclusion criteria. This might introduce some potential false negatives, leading to deviations from an accurate representation. Consequently, it raises questions about what the models trained on these data are learning and their potential performance in real-world applications.

In this context, we conducted an additional experiment by reformulating the negative samples in the datasets from experiment 3. We first etracted positive interactions from each dataset. Then, for identifying negative interactions, a pair of lncRNA-proteins is randomly selected. This pair, R1-P1, is discarded if there exists another pair, R2-P2, where the lncRNA R1 shares more than 80% sequence identity with R2, and the protein P1 shares more than 40% sequence identity with P2, following a method similar to the RPITER approach [20]. This process helps ensure a minimal dissimilarity between the assumed negative sequences and the positive ones, thereby reducing the likelihood of false negatives. The procedure is repeated until an equal number of positive and negative interactions are compiled, after which the data is used as input for BioPrediction-RPI. The results of this additional experiment are presented in Table 10.

We observe that in this new selection of negative data (essentially representing undersampling in the data), the performance across all datasets significantly improves compared to its previous performance. However, internally, BioPrediction-RPI applies subsampling techniques to resolve the imbalance in the data. This observation highlights how the method of selecting interactions to form datasets can influence model performance and its potential real-world applicability.

Therefore, our performance was validated on 12 datasets, proving to be competitive with all other studies on at least one dataset. This suggests that BioPrediction-RPI is capable of predicting interactions across various contexts and can be utilized by non-experts to create predictive models with satisfactory performance. This reduces the necessity for deep technical knowledge in ML to carry out this task.

4.2. Illustrative example

To illustrate the applicability of BioPrediction-RPI, we conducted a complementary experiment using dataset 2 from experiment 3, which consists of 3265 positive samples and the negative samples generated in the previous section. This dataset comprises 885 RNAs and 84 proteins, totaling 74340 possible combinations. After generating all possible combinations, 3,265 known positive interactions were removed, along with 3,265 generated negative interactions, leaving over sixty thousand unlabeled interactions to be used as candidate interacting pairs. First, we observe the estimated performances during the model validation and for the first of the five folds generated in the cross-validation process (refer to Table 11).

In the subsequent step, the model from the first fold was utilized to predict candidate interactions. At this stage, 13,069 possible positive interactions between RNAs and proteins were identified, with this fold demonstrating a sensitivity of 99.58% and a precision of 76.42%. In other words, given the estimated sensitivity of the model, it is expected that approximately 99% of the positive interactions within the candidate set will be among the thirteen thousand interactions predicted as positive. Additionally, based on the estimated precision of the model, it is expected that approximately three out of every four interactions, among the thirteen thousand predicted as positive, will be confirmed as positive when tested experimentally. This implies the potential discovery of approximately ten thousand new interactions upon experimental testing of this entire group predicted as positive. Finally, this experiment shows that by knowing only 5% of the combinations, it is possible to build a model to accelerate the discovery of the remaining interactions.

4.3. Running time

To evaluate the running time of BioPrediction-RPI, the execution times were measured across three datasets. In this experiment, an IdeaPad C340 (Intel Core i7-8565U, 20 GB RAM, and integrated Intel UHD Graphics 620) was used to execute the framework. The datasets used were RPI488, RPI2241, and NPInter from the initial experiments, and the size of each dataset along with its respective running time is presented in Table 12.

4.4. Interpretability

In the interpretability report, several graphs aim to emphasize which features contributed most to a particular classification, and how the distribution of possible values for each feature influences the classification of each class. In Fig. 3 and Fig. 4, we observe two examples of graphs that illustrate how each feature influences the classification, one for the final classification and another graph for a partial model. These graphs are designed to uncover patterns and analyze the relationship between the magnitude of a particular feature and its corresponding class. This visual representation helps in understanding the impact of individual features on the decision-making process of the model.

In the graph, each sample point is marked with a color within a red-blue spectrum. Red points represent the high magnitude of the feature in question, while blue points represent the low magnitude. The distance on the horizontal axis of the point concerning the center of the distribution (0.0) indicates how intensely this feature contributed, positively (positive SHAP values) or negatively (negative SHAP values), to the final classification in inferences of a particular class. On the left side, are the nine most influential features, accompanied by the particular analysis of each one.

Additionally, Fig. 5 displays an example of another type of chart generated by the interpretability module. This chart illustrates how each feature contributed to a specific classification. The chart title shows the inferred class and the identification number representing the sample in question. The features and their corresponding values are displayed on the left side of the chart. In the chart, the directed bars represent each feature's contribution to the classification of the sample: a positive contribution (supporting the classification, shown in red) or a negative contribution (countering the classification, shown in blue). Similarly, longer bars are those that had the most influence on the model's decision.

When examining partial model 2 as shown in Fig. 4, it is evident that the most significant features primarily relate to physicochemical properties and pseudo-frequencies. Notably, characteristics such as the volume, hydrophobicity, and polarity of proteins stand out as crucial for interaction. Upon closer inspection of a random sample in Fig. 5, features derived from the hydrophobicity (H1) and polarity (P1) of the protein are highlighted. These attributes are particularly valuable for biologists to explore further, as they may play critical roles in molecular interaction.

An important detail is that the user has the option to rely solely on a single partial model, using its decision to guide their experiments, even though this might result in slightly lower performance. This simplifies the decision-making process and makes it easier to interpret, which can be seen as an advantage. However, it's important to acknowledge that not all features possess a direct biological meaning. Therefore, a crucial consideration for future versions is to focus on the development of features with enhanced biological significance, which could replace those that are currently less interpretable.

5. Conclusion

In conclusion, the experimental results affirm that BioPrediction-RPI is on par with and in some cases exceeds the performance of

Table 11
Performance of the model in the hypothetical example.

Dataset	Precision (%)	Recall (%)	ACC (%)	F1 (%)	AUC (%)	AUPR (%)
dataset 2	76.54 ± 2.25	98.80 ± 1.58	87.79 ± 1.19	86.23 ± 1.38	87.79 ± 1.19	87.97 ± 1.06
dataset 2 fold 1	76.42	99.58	84.68	86.47	88.83	88.10

Table 12
Execution Times for Datasets.

Dataset	Interaction pairs	Running time
RPI488	243	945.46 s
RPI2241	2241	5302.42 s
NPInter	10412	22967.61 s

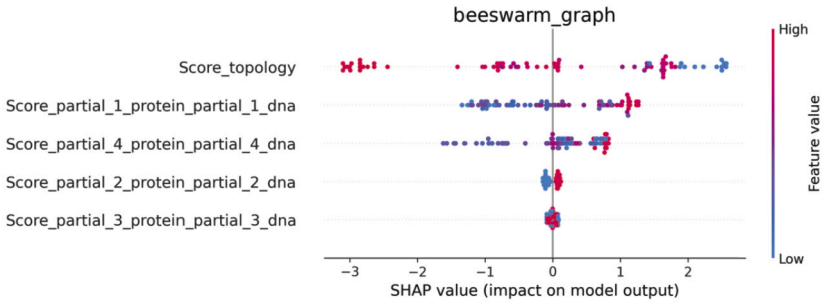

Fig. 3. This graph reveals how the partial predictions behave in each partial model, which is useful for evaluating if there are any unusual patterns in the decision-making process.

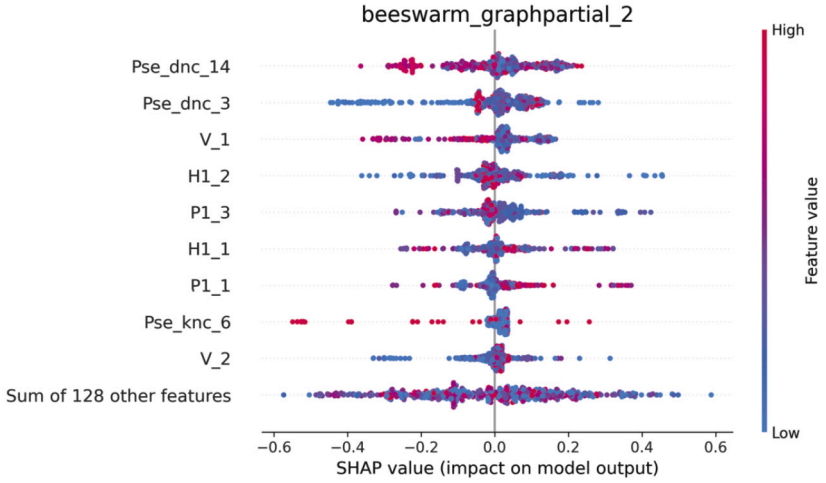

Fig. 4. This graph reveals that the most relevant features are those associated with RNA (described as DNA, as the information is encoded in A, T, C, and G).

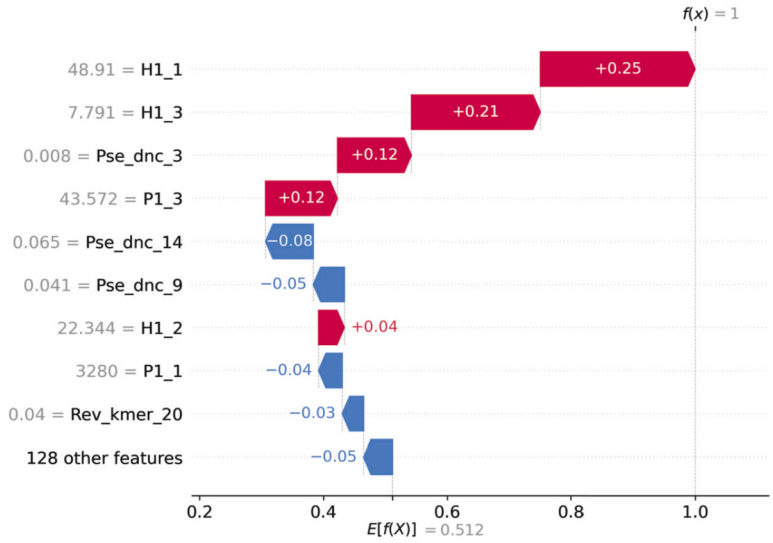

Fig. 5. An individual analysis of sample 212 from the training set, which belongs to the positive class, reveals that features related to SCPseTNC are the most important. The top 9 features, when summed, represent half of the importance of the other 128 features.

existing studies, marking it as a robust competitor in the field. While BioPrediction-RPI exhibits some limitations in handling unbalanced datasets, this gap presents a clear avenue for enhancement, particularly by adopting advanced data balancing techniques in future iterations. Despite these areas for improvement, comprehensive evidence suggests that BioPrediction-RPI not only competes but also challenges the dominance of expert-developed models, affirming its position as a viable alternative for biological prediction tasks. Finally, BioPrediction-RPI paves the way for the democratization of RPI prediction model development, making sophisticated ML applications accessible to those without deep ML expertise.

CRediT authorship contribution statement

Bruno Rafael Florentino: Conceptualization, Data curation, Formal analysis, Methodology, Software, Validation, Visualization, Writing –

original draft, Writing – review & editing. **Robson Parmezan Bonidia:** Conceptualization, Formal analysis, Funding acquisition, Methodology, Project administration, Supervision, Validation, Writing – original draft, Writing – review & editing. **Natan Henrique Sanches:** Software, Writing – original draft. **Ulisses N. da Rocha:** Formal analysis, Supervision, Validation, Writing – review & editing. **André C.P.L.F. de Carvalho:** Conceptualization, Formal analysis, Funding acquisition, Project administration, Supervision, Writing – original draft, Writing – review & editing.

Declaration of competing interest

We have no conflicts of interest to declare.

Acknowledgements

This research is funded by Canada's International Development Research Centre (IDRC) (Grant No. 109981).

References

[1] Jiang Pengfei, Sinha Sanjay, Aldape Kenneth, et al. Big data in basic and translational cancer research. Nat Rev Cancer 2022;22:625–39.
[2] Sadat Golestan Hashemi Farahnaz, Razi Ismail Mohd, Rafii Yusop Mohd, Sadat Golestan Hashemi Mahboobe, Hossein Nadimi Shahraki Mohammad, Rastegari Hamid, et al. Intelligent mining of large-scale bio-data: bioinformatics applications. Biotechnol Biotechnol Equip 2018;32(1):10–29.
[3] Mingyue Cheng, Le Cao, Kang Ning. Microbiome big-data mining and applications using single-cell technologies and metagenomics approaches toward precision medicine. Front Genet 2019;10.
[4] Behzadi P, Gajdács M. Worldwide protein data bank (wwpdb): a virtual treasure for research in biotechnology. Eur J Microbiol Immunol (Bp) 2021;11(4):77–86.
[5] Chicco D. Ten quick tips for machine learning in computational biology. BioData Min 2017;10(35).
[6] Zhang Wenzhen, Wang Jianfang, Li Bingzhi, Sun Bing, Yu Shengchen, Wang Xiaoyu, et al. Long non-coding rna bnip3 inhibited the proliferation of bovine intramuscular preadipocytes via cell cycle. Int J Mol Sci 2023;24(4).
[7] Kopp Florian, Mendell Joshua T. Functional classification and experimental dissection of long noncoding rnas. Cell 2018;172(3):393–407.
[8] Xu Jinyang, Xu Jing, Liu Xinyu, et al. The role of lncrna-mediated cerna regulatory networks in pancreatic cancer. Cell Death Discov 2022;8:287.
[9] Cantile Monica, Di Bonito Maurizio, De Bellis Maura Tracey, Botti Gerardo. Functional interaction among lncrna hotair and micrornas in cancer and other human diseases. Cancers 2021;13(3).
[10] Bonidia Robson P, Avila Santos Anderson P, de Almeida Breno LS, Stadler Peter F, da Rocha Ulisses N, Sanches Danilo S, de Carvalho André CPLF. BioAutoML: automated feature engineering and metalearning to predict noncoding RNAs in bacteria. Brief Bioinform 2022;23(4).
[11] Waring Jonathan, Lindvall Charlotta, Umeton Renato. Automated machine learning: review of the state-of-the-art and opportunities for healthcare. Artif Intell Med 2020;104:101822.
[12] Petch Jeremy, Di Shuang, Nelson Walter. Opening the black box: the promise and limitations of explainable machine learning in cardiology. Can J Cardiol 2022;38(2):204–13. Focus Issue: New Digital Technologies in Cardiology.
[13] Rudin Cynthia. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat Mach Intell 2019;1(5):206–15.
[14] Lundberg Su-In, e Lee Scott M. A unified approach to interpreting model predictions. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, et al., editors, Advances in neural information processing systems, vol. 30. Curran Associates, Inc.; 2017. p. 4765–74.
[15] Ribeiro Marco, Singh Sameer, Guestrin Carlos. "Why should I trust you?": explaining the predictions of any classifier; 2016. p. 97–101.
[16] Dwivedi Yogesh K, Kshetri Nir, Hughes Lois, Slade Emma L, Jeyaraj Anand, Kar Anirban K, et al. "So what if chatgpt wrote it?" multidisciplinary perspectives on opportunities, challenges and implications of generative conversational ai for research, practice and policy. Int J Inf Manag 2023;71:102642.
[17] Seger Elizabeth, Ovadya Aviv, Siddarth Divya, Garfinkel Ben, Dafoe Allan. Democratising ai: multiple meanings, goals, and methods. In: Proceedings of the 2023 AAAI/ACM conference on AI, ethics, and society, AIES '23. New York, NY, USA: Association for Computing Machinery; 2023. p. 715–22.
[18] Vanschoren Joaquin. Democratising artificial intelligence to accelerate scientific discovery. In: Artificial intelligence in science: challenges, opportunities and the future of research. Paris: OECD Publishing; 2023.
[19] Thirunavukarasu Anand, Elangovan Karthik, Gutierrez Luis, Li Ying, Tan Ivan, Keane Patrick, et al. Democratizing artificial intelligence imaging analysis with automated machine learning: tutorial. J Med Internet Res 2023;25:e49949.
[20] Peng C, Han S, Zhang H, Li Y. Rpiter: a hierarchical deep learning framework for ncrna-protein interaction prediction. Int J Mol Sci March 2019;20(5):1070.
[21] Pan Xiaoyong, Fan Yong-Xian, Yan Junchi, Shen Hong-Bin. Ipminer: hidden ncrna-protein interaction sequential pattern mining with stacked autoencoder for accurate computational prediction. BMC Genomics August 2016;17(1):582.
[22] Wang Jingjing, Zhao Yanpeng, Gong Weikang, Liu Yang, Wang Mei, Huang Xiaoqian, et al. Edlmfc: an ensemble deep learning framework with multi-scale features combination for ncrna–protein interaction prediction. BMC Bioinform March 2021;22(1):133.
[23] Zhou Lin, Wang Zexuan, Tian Xiaojing, et al. LPI-deepGBDT: a multiple-layer deep framework based on gradient boosting decision trees for lncrna–protein interaction identification. BMC Bioinform 2021;22:479.
[24] Peng Lihong, Tan Jingwei, Tian Xiongfei, Zhou Liqian. EnANNDeep: an ensemble-based lncRNA-protein interaction prediction framework with adaptive k-nearest neighbor classifier and deep models. Interdiscip Sci March 2022;14(1):209–32.
[25] Fan Xiao-Nan, Zhang Shao-Wu. Lpi-bls: predicting lncrna–protein interactions with a broad learning system-based stacked ensemble classifier. Neurocomputing 2019;370:88–93.
[26] Muppirala Uday K, Honavar Vasant G, Dobbs Drena. Predicting rna-protein interactions using only sequence information. BMC Bioinform 2011;12:489.
[27] Lu Qian, Ren Su, Lu Mingyuan, et al. Computational prediction of associations between long non-coding rnas and proteins. BMC Genomics 2013;14:651.
[28] Dai Qian, Guo Mengqi, Duan Xiangfeng, Teng Zhen, Fu Yiran. Construction of complex features for computational predicting ncrna–protein interaction. Front Genet 2019;10(18).
[29] Wekesa Julius S, Meng Jia, Luan Yushi. Multi-feature fusion for deep learning to predict plant lncrna-protein interaction. Genomics 2020;112(5):2928–36.
[30] Deng L, Wang J, Xiao Y, Wang Z, Liu H. Accurate prediction of protein-lncrna interactions by diffusion and hetesim features across heterogeneous network. BMC Bioinform 2018;19(1):1–11.
[31] Zhou Yu-Kun, Hu Jie, Shen Zhi-An, Zhang Wei-Yun, Du Peng-Fei. Lpi-skf: predicting lncrna-protein interactions using similarity kernel fusions. Front Genet 2020;11:1554.
[32] Zhou Yu-Kun, Shen Zhi-An, Yu Han, Luo Ting, Gao Yu, Du Peng-Fei. Predicting lncrna-protein interactions with mirnas as mediators in a heterogeneous network model. Front Genet 2020;10:1341.
[33] Bonidia Robson P, Domingues Douglas S, Sanches Danilo S, de Carvalho André CPLF. Mathfeature: feature extraction package for dna, rna and protein sequences based on mathematical descriptors. Brief Bioinform 2021:bbab434.
[34] Zhang Wen, Shi Jingwen, Tang Guifeng, Wu Wenjian, Yue Xiang, Li Dingfang. Predicting small rnas in bacteria via sequence learning ensemble method. In: 2017 IEEE international conference on bioinformatics and biomedicine (BIBM); 2017. p. 643–7.
[35] Arrigo Patrizio, Yang Lei, Han Yukun, Zhang Huixue, Li Wenlong, Dai Yu. Prediction of protein-protein interactions with local weight-sharing mechanism in deep learning. BioMed Res Int June 2020;2020:5072520.
[36] Lundberg Scott M, Erion Gabriel, Chen Hugh, DeGrave Alex, Prutkin Jordan M, Nair Bala, et al. From local explanations to global understanding with explainable ai for trees. Nat Mach Intell 2020;2(1):2522–5839.
[37] Wekesa Jael Sanyanda, Meng Jun, Luan Yushi. Multi-feature fusion for deep learning to predict plant lncrna-protein interaction. Genomics 2020;112(5):2928–36.
[38] Chen Muhao, Ju Chelsea J-T, Zhou Guangyu, Chen Xuelu, Zhang Tianran, Chang Kai-Wei, et al. Multifaceted protein–protein interaction prediction based on Siamese residual rcnn. Bioinformatics July 2019;35(14):i305–14.
[39] Yu Bin, Chen Cheng, Zhou Hongyan, Liu Bingqiang, Ma Qin. Gtb-ppi: predict protein–protein interactions based on l1-regularized logistic regression and gradient tree boosting. Genomics Proteomics Bioinform 2020;18(5):582–92.
[40] Li H, et al. Deep neural network based predictions of protein interactions using primary sequences. Molecules 2018;23:1923.
[41] Sun T, Zhou B, Lai L, et al. Sequence-based prediction of protein-protein interaction using a deep-learning algorithm. BMC Bioinform 2017;18:277.
[42] Guo Yanzhi, Yu Lezheng, Wen Zhining, Li Menglong. Using support vector machine combined with auto covariance to predict protein–protein interactions from protein sequences. Nucleic Acids Res 2008;36:3025–30.
[43] Yang Lei, Xia Jun-Feng, Gui Jie. Prediction of protein-protein interactions from protein sequence using local descriptors. Prot Peptide Lett 2010;17(9).