scientific reports

Check for updates

OPEN

AptaNet as a deep learning approach for aptamer–protein interaction prediction

Neda Emami[1] & Reza Ferdousi[1,2]✉

Aptamers are short oligonucleotides (DNA/RNA) or peptide molecules that can selectively bind to their specific targets with high specificity and affinity. As a powerful new class of amino acid ligands, aptamers have high potentials in biosensing, therapeutic, and diagnostic fields. Here, we present AptaNet—a new deep neural network—to predict the aptamer–protein interaction pairs by integrating features derived from both aptamers and the target proteins. Aptamers were encoded by using two different strategies, including k-mer and reverse complement k-mer frequency. Amino acid composition (AAC) and pseudo amino acid composition (PseAAC) were applied to represent target information using 24 physicochemical and conformational properties of the proteins. To handle the imbalance problem in the data, we applied a neighborhood cleaning algorithm. The predictor was constructed based on a deep neural network, and optimal features were selected using the random forest algorithm. As a result, 99.79% accuracy was achieved for the training dataset, and 91.38% accuracy was obtained for the testing dataset. AptaNet achieved high performance on our constructed aptamer-protein benchmark dataset. The results indicate that AptaNet can help identify novel aptamer–protein interacting pairs and build more-efficient insights into the relationship between aptamers and proteins. Our benchmark dataset and the source codes for AptaNet are available in: https://github.com/nedaemami/AptaNet.

In 1990, aptamers were first introduced in the last decade of the twentieth[1–3]. They are short single-stranded sequences (RNA/DNA) or peptides. Because of their spatial conformations, they are capable of binding to specific molecular targets with high specificity and affinity[4]. These biological targets can be a broad range of biomolecules (e.g., lipids[5], viruses[6], nucleic acids[7], cytokines[8], ions[9], etc.).

Aptamers are analogous to antibodies, but they have more advantages that make them a better choice. First, the aptamers are more stable at high temperatures. Second, after selection, aptamers amplify easily by polymerase chain reactions to establish large amounts of molecules with high purity. Third, aptamers are screened by the in-vitro process using an artificial library instead of cell lines or animals. And finally, aptamers have a simple structure, so they could be easily modified by adding various functional groups[10–12]. Therefore, as a new class of amino acid ligands, aptamers are expected to have great potential in bio sensing, diagnostics, and therapeutic approaches.

Aptamers are selected and produced by an in-vitro process, called the systematic evolution of ligands by exponential enrichment (SELEX), which contains a repetitive cycle of selection and amplification[3,13]. The process of empirical SELEX is challenging, time-consuming, and often fails to enrich high-affinity aptamers[14,15]. So far, several efforts have been conducted to improve the production and selection of aptamers[16]. However, it is still necessary for to develop other computational methods to accelerate the process, save cost, and design more effective aptamers with high affinity and specificity.

According to the best of our knowledge, four computational methods have been developed so far in terms of predicting aptamer–protein interaction. For the first time, Li et al.[17] proposed a predictor based on a random forest (RF) algorithm. They used nucleotide composition for encoding aptamers, and targets were encoded using AAC and PseAAC. Although their predictor had yielded favorable results, but it had an imbalance problem. Zhang and co-workers[18] developed an ensemble classifier consisting of three RF sub-classifiers to deal with the imbalance problem. They used pseudo K-tuple nucleotide composition (PseKNC) to represent aptamers and discrete cosine transform, bigram position-specific scoring matrix, and disorder information used for target encoding. In another work for imbalanced data handling, in 2019, Yang et al.[19] presented an ensemble classifier

[1]Department of Health Information Technology, School of Management and Medical Informatics, Tabriz University of Medical Sciences, Tabriz, Iran. [2]Research Center for Pharmaceutical Nanotechnology, Biomedicine Institute, Tabriz University of Medical Sciences, Tabriz, Iran. ✉email: ferdousi.r@gmail.com

https://doi.org/10.1038/s41598-021-85629-0

with three support vector machine (SVM)s sub-classifiers. They used nucleic acid composition and PseKNC for aptamer encoding, and a sparse autoencoder was applied to represent the targets. And recently, Li et al.[20] developed a web server to predict protein–aptamer interactions using an integrated framework Adaboost and random forest and features derived from sequences of aptamers and proteins. Aptamers were represented by nucleotide composition, PseKNC, and normalized Moreau-Broto autocorrelation coefficient. However, proteins were characterized by AAC, PseAAC, grouped amino acid composition, C/T/D composition and sequence-order-coupling number.

These four methods have been developed to generate interaction predictions and yielded good results, but there are several opportunities and requirements for enhancing this field. Since deep learning techniques have had a high performance in biological predictions[21–24], and they have not yet been exploited as a predictor to predict aptamer–protein interactions (API). Therefore, in this paper, we present AptaNet (a first deep neural network predictor) to predict the API pairs by integrating features derived from both aptamers and the target proteins. Aptamers were encoded by using two different strategies includes k-mer and revck-mer frequency. AAC and PseAAC were applied to represent target information using 24 physicochemical and conformational properties of proteins. To handle the imbalance problem in our dataset, we applied neighborhood cleaning algorithm.

AptaNet achieved high performance on our constructed aptamer-protein benchmark dataset. The results indicate that AptaNet could help to identify novel aptamer–protein interacting pairs and build more-efficient biological insights into understanding the relationship between aptamers and proteins, which could benefit all aptamer scientists and researchers.

The rest of the paper is organized as follows: presentation of the predictor's performance using cross-validation, discussion, conclusion, dataset description, statistical samples formulation, selection, and development of a deep neural network.

Results

The technical details of AptaNet are provided in the "Materials and Methods" section. In this section, the results of several evaluation experiments are presented.

The results of feature group effects. We created four groups (1, 2, 3, and 4). Each group consists of eight feature groups to describe aptamers encoding by the K-mer (k = 3), K-mer (k = 4), RevcK-mer (k = 3), and RevcK-mer (k = 4) methods, respectively, with proteins that encoded by AAC and PseAAC. We used a total of 24 properties (i.e., physicochemical, conformational, and energetic) of proteins. We added three groups of properties each time to the previous dataset, sequentially. The feature groups and their information have been reported in Table 1. We have performed four sets of experiments to investigate the feature group's effectiveness. In these experiments, we changed the feature groups on different deep neural networks by applying a balancing method to analyze the effect of features and performance of different neural networks. The results of these experiments are reported in Table 2. It is noticeable that the best results are related to the balanced datasets.

Since the number of properties is high, we have briefly named every three groups of properties: A, B, C, D, E, F, G, and H. A: hydrophobicity, hydrophilicity, mass; B: polarity, molecular weight, melting point; C: transfer free energy, buriability, bulkiness; D: solvation free energy, relative mutability, residue volume; E: volume, amino acid distribution, hydration number; F: isoelectric point, compressibility, chromatographic index; G: unfolding entropy change, unfolding enthalpy, unfolding Gibbs free energy change; H: the power to beat the N terminal, C terminal, and middle of alpha-helix; and apt for aptamers. It should be noted that various combinations of these 24 properties were examined (e.g., hydrophobicity, hydrophilicity, mass; hydrophobicity, hydrophilicity, polarity; isoelectric point, compressibility, hydrophilicity, etc.), and finally, the best combinations were selected based on their results.

First, we evaluate the performance using feature group k-mer2 + A, next added A + B, A + B + C, and finally A + B + C + D + E + F + G + H, sequentially. We have generated the average results of each feature group and different combinations of these feature groups based on sequential forward selection.

Table 2 represents the average performance of two different neural networks on the 32 datasets during our experiments. We have generated the results of various combinations of the feature groups by adding them in a forward selection scheme by sorting them based on their spatial performance for different aptamer encoding methods. For k-mer = *4*, the best results were achieved when 21 properties(i.e., hydrophobicity, hydrophilicity, mass, polarity, molecular weight, melting point, transfer-free energy, buriability, bulkiness, solvation free energy, relative mutability, residue volume, volume, amino acid distribution, hydration number, isoelectric point, compressibility, chromatographic index, unfolding entropy change, unfolding enthalpy, and unfolding Gibbs free energy charge) were applied. In the second place, the combination of 15 properties (i.e., hydrophobicity, hydrophilicity, mass, polarity, molecular weight, melting point, transfer-free energy, buriability, bulkiness, solvation free energy, relative mutability, residue volume, volume, amino acid distribution, and hydration number) had the highest performance.

For k-mer = *3,* the best results were achieved when 18 properties (i.e., hydrophobicity, hydrophilicity, mass, polarity, molecular weight, melting point, transfer-free energy, buriability, bulkiness, solvation free energy, relative mutability, residue volume, volume, amino acid distribution, hydration number, isoelectric point, compressibility, and chromatographic index) were applied. In the second place, the combination of 12 properties (i.e., hydrophobicity, hydrophilicity, mass, polarity, molecular weight, melting point, transfer-free energy, buriability, bulkiness, solvation free energy, relative mutability, and residue volume) had the highest performance.

For the Revck-mer = *3*, the best results were achieved when six properties (i.e., hydrophobicity, hydrophilicity, mass, polarity, molecular weight, and melting point) were applied. In the second place, the combination of three properties (i.e., hydrophobicity, hydrophilicity, and mass) had the highest performance.

	Aptamer encoding			**Protein encoding**				
Group	**Method**	**Group name**	**Number of feature**	**Features**	**Group name**	**Number of feature**	**Dataset**	**Total number**
Group 1	Kmer (k = 3)	Apt	84	hydrophobicity, hydrophilicity, mass	A	50	Apt + A	134
				polarity, molecular weight, melting point	B	50	Apt + A + B	184
				transfer free energy, buriability, bulkiness	C	50	Apt + A + B + C	234
				solvation free energy, relative mutability, residue volume	D	50	Apt + A + B + C + D	384
				volume, amino acid distribution, hydration number	E	50	Apt + A + B + C + D + E	334
				isoelectric point, compressibility, chromatographic index	F	50	Apt + A + B + C + D + E + F	384
				unfolding entropy change, unfolding enthalpy, unfolding Gibbs free energy charge	G	50	Apt + A + B + C + D + E + F + G	434
				Power of N terminal of alphahelix, Power of C terminal of alphahelix, Power of best the middle of alphahelix	H	50	Apt + A + B + C + D + E + F + G + H	484
Group 2	Kmer (k = 4)	Apt	339	hydrophobicity, hydrophilicity, mass	A	50	Apt + A	389
				polarity, molecular weight, melting point	B	50	Apt + A + B	439
				transfer free energy, buriability, bulkiness	C	50	Apt + A + B + C	489
				solvation free energy, relative mutability, residue volume	D	50	Apt + A + B + C + D	539
				volume, amino acid distribution, hydration number	E	50	Apt + A + B + C + D + E	589
				isoelectric point, compressibility, chromatographic index	F	50	Apt + A + B + C + D + E + F	639
				unfolding entropy change, unfolding enthalpy, unfolding Gibbs free energy charge	G	50	Apt + A + B + C + D + E + F + G	689
				Power of N terminal of alphahelix, Power of C terminal of alphahelix, Power of best the middle of alphahelix	H	50	Apt + A + B + C + D + E + F + G + H	739
Continued								

	Aptamer encoding			Protein encoding				
Group	Method	Group name	Number of feature	Features	Group name	Number of feature	Dataset	Total number
Group 3	Revckmer (k = 3)	Apt	44	hydrophobicity, hydrophilicity, mass	A	50	Apt + A	94
				polarity, molecular weight, melting point	B	50	Apt + A + B	144
				transfer free energy, buriability, bulkiness	C	50	Apt + A + B + C	194
				solvation free energy, relative mutability, residue volume	D	50	Apt + A + B + C + D	244
				volume, amino acid distribution, hydration number	E	50	Apt + A + B + C + D + E	294
				isoelectric point, compressibility, chromatographic index	F	50	Apt + A + B + C + D + E + F	344
				unfolding entropy change, unfolding enthalpy, unfolding Gibbs free energy charge	G	50	Apt + A + B + C + D + E + F + G	394
				Power of N terminal of alphahelix, Power of C terminal of alphahelix, Power of best the middle of alphahelix	H	50	Apt + A + B + C + D + E + F + G + H	444
Group 4	Revckmer (k = 4)	Apt	179	hydrophobicity, hydrophilicity, mass	A	50	Apt + A	229
				polarity, molecular weight, melting point	B	50	Apt + A + B	279
				transfer free energy, buriability, bulkiness	C	50	Apt + A + B + C	329
				solvation free energy, relative mutability, residue volume	D	50	Apt + A + B + C + D	379
				volume, amino acid distribution, hydration number	E	50	Apt + A + B + C + D + E	429
				isoelectric point, compressibility, chromatographic index	F	50	Apt + A + B + C + D + E +	479
				unfolding entropy change, unfolding enthalpy, unfolding Gibbs free energy charge	G	50	Apt + A + B + C + D + E + F + G	529
				Power of N terminal of alphahelix, Power of C terminal of alphahelix, Power of best the middle of alphahelix	H	50	Apt + A + B + C + D + E + F + G + H	579

Table 1. Feature group description. Where Kmer (k = 3) is 3mer frequency, Kmer (k = 4) is 4mer frequency, Revckmer (k = 3) is reverse complement 3mer, and Revckmer frequency (k = 4) is reverse complement 4mer frequency for apt, which represents aptamer properties. And, Apt indicates aptamer properties.For protein properties: A indicates hydrophobicity, hydrophilicity, and mass; B indicates polarity, molecular weight, and melting point; C indicates transfer free energy, buriability, and bulkiness; D indicates solvation free energy, relative mutability, and residue volume; E indicates volume, amino acid distribution, and hydration number; F indicates isoelectric point, compressibility, and chromatographic index; G indicates unfolding entropy change, unfolding enthalpy and unfolding Gibbs free energy change; H indicates power to beat the N terminal, C terminal, and middle of the alpha helix.

For the Revck-mer = *4*, the best results were achieved when 12 properties (i.e., hydrophobicity, hydrophilicity, mass, polarity, molecular weight, melting point, transfer-free energy, buriability, bulkiness, solvation free energy,

Dataset	Feature combination	Deep neural networks F1-score (imbalanced)		Deep neural networks F1-score (balanced)	
		MLP	CNN	MLP	CNN
Kmer (k = 3) + PseAAC	Apt + A	0.8071	0.8197	0.8417	0.8577
	Apt + A + B	0.7992	0.8624	0.8507	0.8416
	Apt + A + B + C	0.8002	0.8420	0.8510	0.8465
	Apt + A + B + C + D	0.7800	0.8409	0.8555	0.8553
	Apt + A + B + C + D + E	0.7764	0.7406	0.8584	0.7983
	Apt + A + B + C + D + E + F	0.7675	0.7618	0.8539	0.7829
	Apt + A + B + C + D + E + F + G	0.7824	0.8043	0.8598	0.8456
	Apt + A + B + C + D + E + F + G + H	0.7753	0.8455	0.8497	0.8105
Kmer (k = 4) + PseAAC	Apt + A	0.8610	0.8542	0.8739	0.8630
	Apt + A + B	0.8461	0.8551	0.8616	0.8720
	Apt + A + B + C	0.8307	0.8608	0.8521	0.8220
	Apt + A + B + C + D	0.8299	0.8347	0.8660	0.8184
	Apt + A + B + C + D + E	0.7997	0.8086	0.8410	0.8379
	Apt + A + B + C + D + E + F	0.7908	0.8389	0.8761	0.8354
	Apt + A + B + C + D + E + F + G	0.8173	0.8257	0.8493	0.8038
	Apt + A + B + C + D + E + F + G + H	0.8130	0.8228	0.8595	0.7939
Revckmer (k = 3) + PseAAC	Apt + A	0.7689	0.8196	0.8215	0.8564
	Apt + A + B	0.7562	0.8287	0.8183	0.8567
	Apt + A + B + C	0.7596	0.8210	0.8119	0.8513
	Apt + A + B + C + D	0.7411	0.8074	0.8265	0.8240
	Apt + A + B + C + D + E	0.7532	0.8418	0.8203	0.8349
	Apt + A + B + C + D + E + F	0.7302	0.8364	0.8256	0.8517
	Apt + A + B + C + D + E + F + G	0.7321	0.7150	0.8192	0.7990
	Apt + A + B + C + D + E + F + G + H	0.7347	0.8043	0.8246	0.8368
Revckmer (k = 4) + PseAAC	Apt + A	0.7870	0.8524	0.8471	0.8559
	Apt + A + B	0.8192	0.8125	0.8434	0.8416
	Apt + A + B + C	0.8020	0.8566	0.8542	0.8537
	Apt + A + B + C + D	0.7928	0.8560	0.8581	0.8501
	Apt + A + B + C + D + E	0.7846	0.8278	0.8378	0.8317
	Apt + A + B + C + D + E + F	0.7893	0.8325	0.8332	0.8168
	Apt + A + B + C + D + E + F + G	0.7839	0.8288	0.8404	0.7995
	Apt + A + B + C + D + E + F + G + H	0.7725	0.7971	0.8393	0.8195

Table 2. The average performances of two deep neural network classifiers on 32 different datasets, with and without the balancing method. Where Kmer 3 is 3mer frequency, Kmer 4 is 4mer frequency, Revckmer 3 is reverse complement 3mer, and Revckmer frequency 4 is reverse complement 4mer frequency for apt, which represents aptamer properties. For protein properties: A indicates hydrophobicity, hydrophilicity, and mass; B indicates polarity, molecular weight, and melting point; C indicates transfer free energy, buriability, and bulkiness; D indicates solvation free energy, relative mutability, and residue volume; E indicates volume, amino acid distribution, and hydration number; F indicates isoelectric point, compressibility, and chromatographic index; G indicates unfolding entropy change, unfolding enthalpy and unfolding Gibbs free energy change; H indicates power to beat the N terminal, C terminal, and middle of the alpha helix. *MLP* multi-layer perceptron; *CNN* convolutional neural network.

relative mutability, and residue volume) were used. In the second place, the combination of 9 properties (i.e., hydrophobicity, hydrophilicity, mass, polarity, molecular weight, melting point, transfer-free energy, buriability, and bulkiness). had the highest performance.

Therefore, according to the results of *32* different datasets, four datasets were selected which had the best values in each group (i.e., Apt + A + B + C + D + E + F + G in group 1, Apt + A + B + C + D + E + F in group 2, Apt + A + B in group 3, and Apt + A + B + C + D in group 4).

The results of neural networks performances. To test and select the appropriate deep neural network for our problem, we tested two deep neural networks: MLP and CNN. The experiment was performed once on the balance data and once on the imbalance data. For these experiments, we applied random under-sampling as the balancing method. Thirty-two different combinations were used as features that have been mentioned already. To set the neural networks: the number of batch sizes for MLP and CNN were 310 and 16, respectively.

Datasets	Methods	Accuracy	Precision	F1 Score	Matthews's correlation coefficient	Specificity	Sensitivity
Kmer 3 + A + B + C + D + E + F	Shallow neural network	0.625	0.374	0.175	0.019	0.894	0.118
	K nearest neighbor	0.589	0.381	0.199	0.002	0.864	0.137
	Random forest	0.739	0.397	0.242	0.13	0.914	0.181
	Support vector machine	0.546	0.39	0.065	− 0.022	0.954	0.036
	Multilayer perceptron	0.872	0.861	0.854	0.741	0.89	0.849
Kmer 4 + A + B + C + D + E + F	Shallow neural network	0.687	0.382	0.272	0.101	0.868	0.212
	K nearest neighbor	0.601	0.38	0.2	0.011	0.871	0.136
	Random forest	0.795	0.396	0.291	0.196	0.918	0.24
	Support vector machine	0.548	0.39	0.058	− 0.02	0.96	0.032
	Multilayer perceptron	0.886	0.881	0.869	0.77	0.906	0.861
Revckmer_3 + A + B	Shallow neural network	0.561	0.388	0.146	− 0.018	0.892	0.095
	K nearest neighbor	0.578	0.391	0.194	− 0.001	0.868	0.13
	Random forest	0.732	0.398	0.259	0.136	0.901	0.203
	Support vector machine	0.529	0.4	0.043	− 0.022	0.97	0.021
	Multilayer perceptron	0.831	0.821	0.809	0.576	0.948	0.558
Revckmer_4 + A + B + C + D	Shallow neural network	0.674	0.386	0.252	0.085	0.873	0.192
	K nearest neighbor	0.557	0.387	0.203	− 0.019	0.846	0.139
	Random forest	0.77	0.397	0.284	0.175	0.909	0.23
	Support vector machine	0.5	0.4	0.036	− 0.029	0.971	0.019
	Multilayer perceptron	0.86	0.847	0.845	0.718	0.874	0.843

Table 3. The average performances of our model with four machine learning algorithms on four different datasets. Where Kmer 3 is 3mer frequency, Kmer 4 is 4mer frequency, Revckmer 3 is reverse complement 3mer, and Revckmer frequency 4 is reverse complement 4mer frequency for aptamer properties. For protein properties: A indicates hydrophobicity, hydrophilicity, and mass; B indicates polarity, molecular weight, and melting point; C indicates transfer free energy, buriability, and bulkiness; D indicates solvation free energy, relative mutability, and residue volume; E indicates volume, amino acid distribution, and hydration number; F indicates isoelectric point, compressibility, and chromatographic index.

Additionally, the number of epochs was determined 200, rmsprop was also considered as an optimizer with its default values, and the activation function was sigmoid. The results regarding the F1-score were presented in Table 2.

In Table 2, the highest performance values achieved from MLP and CNN are highlighted. It is evident that MLP provides the highest values for 22 different feature group combinations. And CNN has achieved the highest values for ten datasets.

Except for three cases, including Apt + A, Apt + A + B, and Apt + A, MLP have the highest values in the three groups 1, 2, and 4. Also exception of Apt + A + B + C + D and Apt + A + B + C + D + E + F + G, CNN has the highest values in group 3. It can be deduced, that by decreasing the dimensionality of datasets CNN has better performance. In the other words, MLP achieves better performance for datasets with more dimension. Therefore, we select MLP as out classifier.

The results of MLP with machine learning algorithms.

In this section, we compared the performance of AptaNet against some machine learning algorithms (shallow neural networks (SNN), k-nearest neighbor (KNN), RF, and SVM). All of algorithms were implemented in SKlearn library with the following parameters: Knn: n = 5, leaf size = 25, p = 2; RF: max depth = 3, n estimators = 10; SVM: degree = 3, c = 1, kernel = 'linear', probability = True, cache size = 200; SNN: hidden layer = (3, 2); solver = 'lbfgs', alpha = 0.0001. Which, all parameters were determined by different experiments. We performed a fivefold cross-validation strategy to evaluate the results of the test and training set. The average performance results are indicated in Table 3 and Fig. 1.

According to Table 3, the highest accuracy belongs to MLP for all datasets. Which among them, the best performance was obtained when kmer4_apt + A + B + C + D + E + F were applied.

In second place among the machine learning algorithms, RF has the highest accuracy for all four datasets. The best performance was obtained using kmer4_apt + A + B + C + D + E + F, which could be due to RF as an ensemble classifier consisting of multiple decision trees. Since RF has less overfitting to a particular dataset, accuracy was better than other machine learning algorithms. Therefore, according to four different datasets' accuracy values, kmer4_apt + A + B + C + D + E + F was selected as a dataset for our model (See Supplementary Table S1).

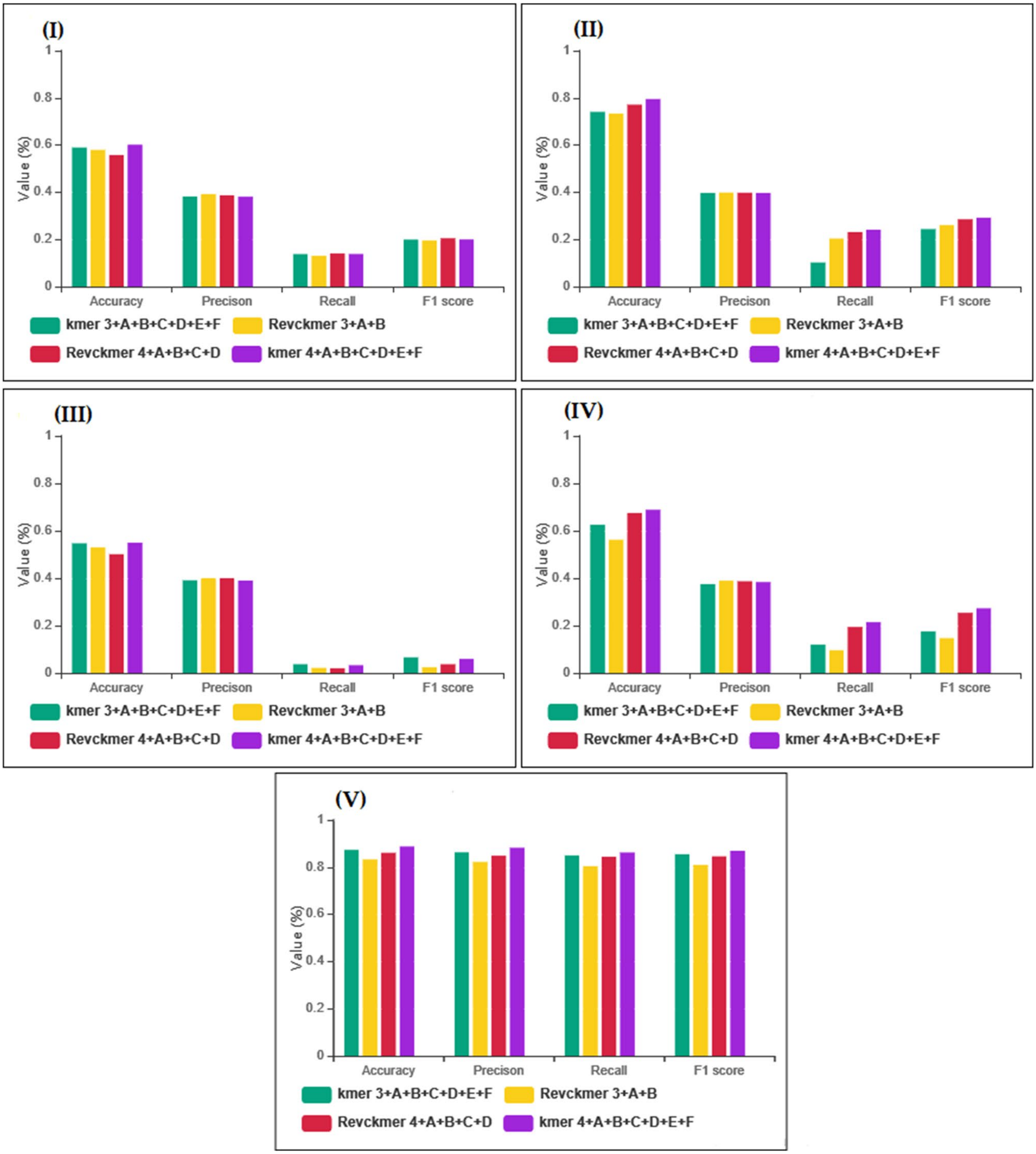


Figure 1. The comparison of the prediction performance of our model and four machine learning algorithms on four different datasets. Where Kmer 3 is 3mer frequency, Kmer 4 is 4mer frequency, Revckmer 3 is reverse complement 3mer, and Revckmer frequency 4 is reverse complement 4mer frequency for aptamer properties. For protein properties: A indicates hydrophobicity, hydrophilicity, and mass; B indicates polarity, molecular weight, and melting point; C indicates transfer free energy, buriability, and bulkiness; D indicates solvation free energy, relative mutability, and residue volume; E indicates volume, amino acid distribution, and hydration number; F indicates isoelectric point, compressibility, and chromatographic index. (**I**) is results of k nearest neighbor algorithm; (**II**) is results of random forest algorithm; (**III**) is results of support vector machine algorithm; (**IV**) is results shallow neural network algorithm and (**V**) is results of our model. The power of predictors is calculated based on four metrics, including accuracy, precision, recall, and F1score.

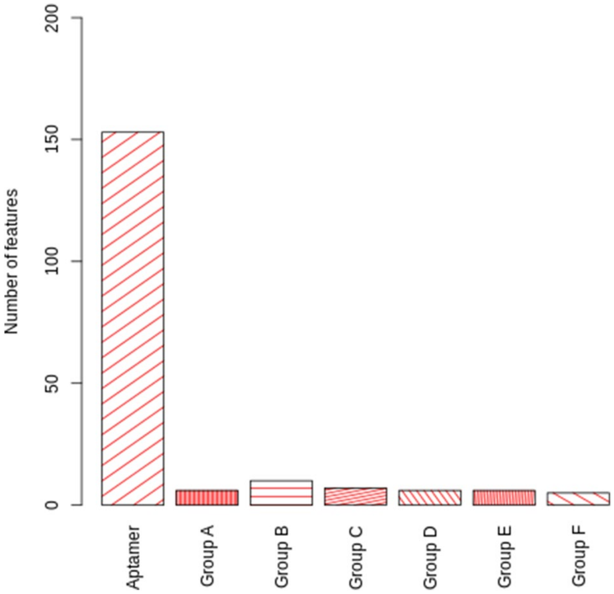


Figure 2. Distributions of the 193 optimal features. Where group A represents hydrophilicity and mass; group B represents polarity, molecular weight, and melting point; group C represents transfer free energy, buriability, and bulkiness; group D represent solvation free energy, relative mutability, and residue volume; group E represent volume, amino acid distribution, and hydration number; group F represent isoelectric point, compressibility and chromatographic index for protein properties.

Among the machine learning algorithms, the lowest performance was achieved for SVM algorithm when aptamer features were combined with 12 properties of protein properties.

The lowest performance among datasets was achieved when aptamer features were combined with six protein properties (i.e., hydrophobicity, hydrophilicity, mass, polarity, molecular weight, melting point).

The results of the MLP optimization and feature selection.

Since MLP had the highest performance among other deep learning and machine learning methods, MLP was selected as our predictor model. To improve our model's performance, we implemented it by selecting the most optimal parameters with different experiments and the final model named AptaNet. We optimized our model with different values for the learning rate, batch size, and epochs. Learning rate = 0.00014, batch size = 5000, and epochs = 260 were the final settings of the presented model.

Also, we applied RF strategy for feature selection and ranking features. The optimal number of features was set to 193 by several experiments on RF parameters and different numbers of features. The 193 optimal features were selected according to the nature of our dataset and parameters of the RF strategy. The parameters were set based on our several feature selection experiments. We set estimators = 300 and max depth = 9 based on our feature selection experiments (See Supplementary Table S2).

Figure 2 describes the feature's importance in ranking values.

As shown in Fig. 2, 193 optimal features obtained from RF strategy could be classified into eight terms: k-mer aptamer frequency, protein composition A, protein composition B, and protein composition F.

In the first place, k-mer aptamer frequency ranks the first; making up approximately 73%. In other words, a considerable part of the optimal features belongs to aptamer features. The k-mer aptamer frequency implies that using k-mer is a key factor for API in this study.

In the second place, there is feature composition B. As follows, in the third place, there is feature composition C. In the fourth place, another effective trait is feature composition A, composition D, and E, in which their counts are nearly equal. This finding means that the impact of using these three feature composition is the same. And in the last place, there is feature composition F.

According to Fig. 3, the ROC value for AptaNet was 0.914 before feature selection and 0.948 after feature selection.

Moreover, the results of testing and training of AptaNet before and after applying feature selection are presented in Tables 4 and 5, respectively. All results of the AptaNet were higher after applying the feature selection technique.

Additionally, Fig. 4 illustrates the model accuracy and loss of AptaNet for epoch = 260 and batch size = 5000.

Discussion

One of the most important challenges in the field of aptamer–target interaction is that the aptamers' special databases(DBs) are scarce. To the best of our knowledge, there are four DBs about aptamer–target interaction, containing Aptamer Database[25], RiboaptDB[26], Aptamer Base[27], and Aptagen (https://www.aptagen.com/). Which

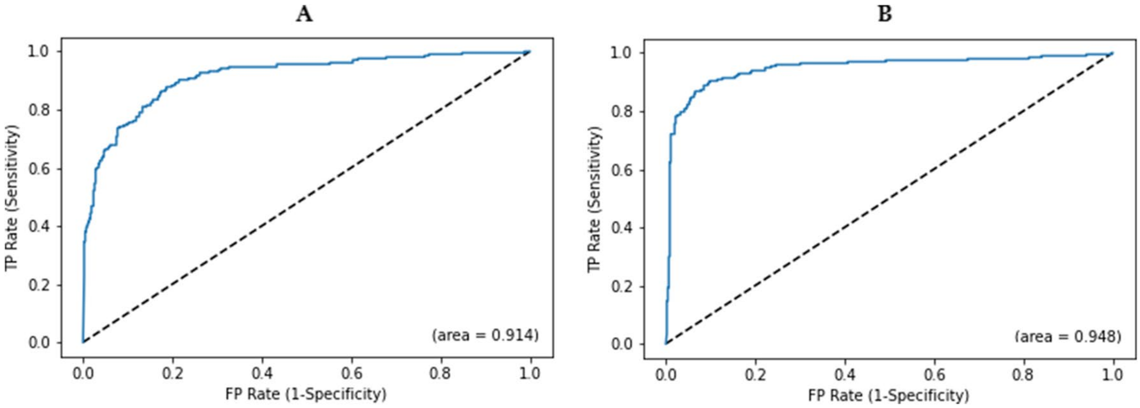


Figure 3. Receiver operating characteristic (ROC) curves of AptaNet before and after the feature selection on our benchmark dataset. Where (**A**) depicts the prediction performance of AptaNet before using feature selection and (**B**) illustrates the prediction performance of AptaNet after using feature selection. *FP* False Positive rate, *TP* True Positive rate.

Dataset	Accuracy	Precision	F1-score	Matthews's correlation coefficient	Specificity	Sensitivity
Kmer 4 + A + B + C + D + E + F (train)	0.998	0.998	0.998	0.997	0.999	0.998
Kmer 4 + A + B + C + D + E + F (test)	0.892	0.883	0.877	0.782	0.908	0.873

Table 4. The overall results of AptaNet before feature selection. Where Kmer 4 is 4mer frequency for aptamer properties. For protein properties: A indicates hydrophobicity, hydrophilicity, and mass; B indicates polarity, molecular weight, and melting point; C indicates transfer free energy, buriability, and bulkiness; D indicates solvation free energy, relative mutability, and residue volume; E indicates volume, amino acid distribution, and hydration number; F indicates isoelectric point, compressibility, and chromatographic index.

Dataset	Accuracy	Precision	F1-score	Matthews's correlation coefficient	Specificity	Sensitivity
Kmer 4 + A + B + C + D + E + F (train)	0.997	0.997	0.997	0.995	0.995	0.999
Kmer 4 + A + B + C + D + E + F (test)	0.913	0.909	0.899	0.824	0.931	0.89

Table 5. The overall results of AptaNet after feature selection. Where Kmer 4 is 4mer frequency for aptamer properties. For protein properties: A indicates hydrophobicity, hydrophilicity, and mass; B indicates polarity, molecular weight, and melting point; C indicates transfer free energy, buriability, and bulkiness; D indicates solvation free energy, relative mutability, and residue volume; E indicates volume, amino acid distribution, and hydration number; F indicates isoelectric point, compressibility and chromatographic index.

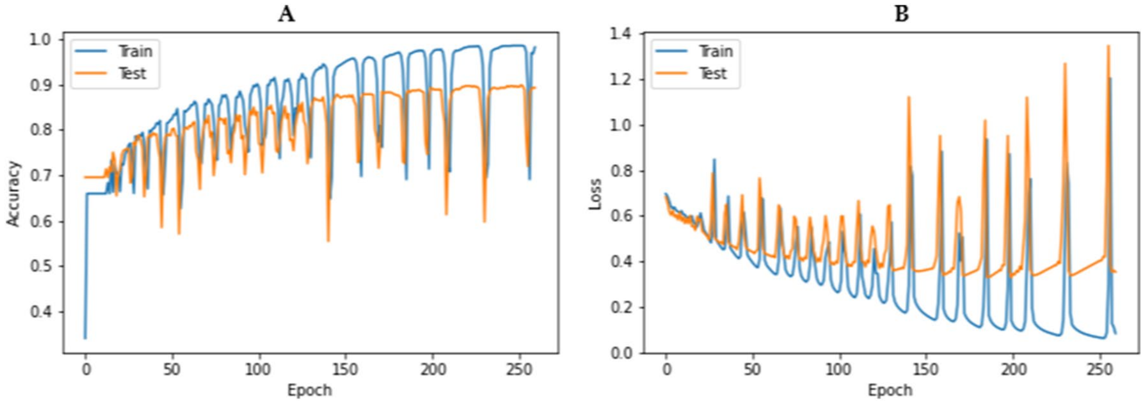


Figure 4. Model accuracy and loss of AptaNet on our benchmark dataset. Where (**A**) is model accuracy, and (**B**) is model loss of AptaNet.

among them, Aptamer Database and RiboaptDB no longer exist. Therefore, in this study, to have a complete dataset of API pairs, for the first time, in addition to Freebase, we also used Aptagen data, which are generated by independent studies. Aptagen provides useful information about aptamer type, target type, and experimental conditions.

Kmer frequency method has been widely used in many bioinformatics studies[28–30] and has had successful results. The basic idea behind this method is that the encoding of each item is based on its interaction with its context. If we consider each sequence as a sentence and each K-mer as a word, we can extend this method to encode aptamer and protein sequences. Thus, Because of the simplicity, considering almost more sequence information compared to other methods available for encoding nucleotide sequences, we have used this method for encoding aptamer sequences.

One of the main problems in AAC strategy is losing the information of protein sequences. To overcome this restriction that can be affected on prediction performances, we have used PseAAC method. In the previous studies related to protein interaction prediction PseAAC method has been widely used and has had successful outcomes[31–37]. Therefore, we applied this strategy to representing protein target sequences in this study.

In several previous studies, it is proved that the physicochemical properties (e.g., hydrophilicity, hydrophobicity, average accessible surface area, and polarity) and biochemical contacts (e.g., residue contacts, atom contacts, salt bridges, and hydrogen bonds) play an essential and constructive role in protein interactions[38–44]. Thus, we have used 32 structural-based and sequence-based properties from protein sequences. However, in the previous studies related to aptamer–target interaction, this large volume of features has not been used.

In this study, we have used the NCL strategy to overcome the imbalanced dataset problem. According to Table 2, in the experimental results obtained from the two neural networks (MLP and CNN), the results on the test dataset were significantly improved after applying the NCL method. Since the NCL technique not only focuses on data reduction but also focuses on cleaning data. NCL reduces the majority class by eliminating low-quality data using Edited Nearest Neighbor (ENN) rules. Moreover, ENN removes classified data that are classified incorrectly. Therefore, the data cleaning process is intended for both majority and minority class samples[45,46]. Consequently, according to[46–50], NCL as an under-sampling method has superior outcomes compared to other common over-sampling methods.

As a subfield of machine learning approaches, deep learning methods have been shown to exhibit unprecedented performance in various areas of biological prediction[51–61]. We described a novel deep neural network model in the present study, termed AptaNet, for predicting API.

We compared the MLP and CNN performances on the 32 different datasets to develop our prediction model. The performance of each network and algorithm was determined by assessing how they could correctly predict whether the aptamers were interacting with a specific target or not.

According to Table 2, MLP had superior outcomes compared to CNN. According to[62–68], MLP networks have been more efficient in text processing problems. In this study, since API data were similar to text data, MLP networks had higher performance. Moreover, CNNs have had better outcomes in classification image data[69–74].

Next, we compared MLP against some machine learning algorithms. Among the applied machine learning algorithms, the SVM algorithm achieved the lowest performance when aptamer features were combined with 12 protein properties. The lowest performance of the SVM algorithm in the prediction of API may be attributed to the following shortcomings.

According to[75–78], First, the SVM algorithm generally has not a convenient performance for large data sets. Second, in cases where the dataset has noises (target classes are overlapping), the SVM classifier will underperform. Third, SVM is not suitable when the number of the training data sample is lower than the number of features for each data point. And finally, since the SVM algorithm works by placing data points, there is no probabilistic explanation for the classification above and below the hyperplane classification.

The previous three studies have compared machine learning approaches (e.g., SVM and RF) to build their predictor. However, in the present study, the performance of two neural networks and four different machine learning algorithms (SNN, SVM, KNN, and RF) was compared and selected based on the best predictor of API.

It is essential to define which properties of the aptamer and protein determine their potential for interaction. The lowest performance among datasets was achieved when aptamer features were combined with six protein properties (i.e., hydrophobicity, hydrophilicity, mass, polarity, molecular weight, melting point). According to the[79–86], energetic and conformational properties have essential effects on protein interactions. Therefore, only the presence of features which are depending on the physicochemical properties and absence of energy and conformational properties could be the reason for low performance.

According to Fig. 2, the k-mer aptamer frequency implies that the k-mer usage is an essential factor for APIs. This finding is justified by the previous studies[87–92] which proved k-mer frequency plays an important role in interaction related to riboswitch, DNA, RNA, ncRNA, lncRNA, etc. This may be due to aptamers in this study were considered as the type of RNA and DNA.

In the second place, there is feature composition B. As follows, in the third place, there is feature composition C. In the fourth place, other effective traits are feature composition A, composition D, and E, in which their counts are nearly remaining equal. This finding means that the impact of using these three feature composition is the same. And in the last place, there is feature composition F. Since our study targets are proteins, according to previous studies on protein interaction and protein complexes[17,93–99], it is shown that physicochemical properties (e.g., hydrophobicity, hydrophilicity, mass, volume, etc.) are the main factors which affected on protein interaction. For example, according to[100], high molecular weight provides strong protein binding affinity. It has also been shown that aptamers are sensitive protein binding based on the local environment polarity at different modification sites[101]. However, the effect of the melting point on protein binding is also indicated[102].

In this study, we applied RF strategy for feature selection and ranking features. The optimal number of features was set to 193 by several experiments on RF parameters and different numbers of optimal features. The

193 optimal features were selected according to the nature of our dataset and the optimized parameters, which we set to RF strategy. The parameters were set based on our several feature selection experiments.

Roc curves have been broadly used in machine learning and deep learning approaches for performance evaluation[86,103–106]. Therefore, as a popular method for performance evaluation, the ROC curve was utilized in our experiments. Which, ROC instead of considering only the numerical AUC values could be a better strategy in this study.

An oscillation in loss and accuracy in our model can be because of the nature of our dataset. It means that the number of API data (which are the result of laboratory results) that are recorded in freebase and aptagen databases is low (only 850). Since deep learning methods require large volumes of data, therefore, in this study, we see oscillation in loss and accuracy. Indeed, in the future, with more laboratory experiments on API, the amount of API data will be increased, and consequently, the results of deep learning models on them will be better.

Conclusion

In this study, we have presented AptaNet, a novel deep learning method for predicting API. AptaNet is unique in its exploitation of sequence-based features for aptamers along with the physicochemical and conformational properties for targets to predict API. It also uses a balancing technique and a deep neural network. We have performed extensive experiments to analyze and test AptaNet performance. Experimental evaluations show that, on our 32 benchmark datasets, AptaNet has superior performance compared to other methods examined in this study in terms of accuracy. Moreover, AptaNet has shown to be able to provide biological insights into understanding API's nature, which can be helpful for all aptamer scientists and researchers.

There is still a lot of room for improvement in this field. The study mentioned above focuses on target proteins, but there are other types of targets, such as compounds. Due to the important role of aptamers in various biological processes, further research is needed to focus on the aptamer's interactions with other types of targets. Additionally, due to the large number of properties of aptamers and proteins that can affect the API, further investigations are required to use other features of aptamers and proteins that have been recommended in the literature. Since the existence of an accessible web-server is required in this field, and given the successful results of AptaNet, other extensive efforts are required to provide a powerful web server based on the predicting method presented in this article in the future. Research on aptamer–target interaction prediction is likely to continue for the next years with deep learning approaches and unprecedented feature extraction strategies for aptamers and targets. Consequently, it leads to new opportunities and challenges in this scope.

Materials and methods

This section provides detailed information of the datasets, feature extraction methods, balancing strategy, two types of examined deep neural networks, feature selection method, and evaluation metrics used in this study. All the methods were implemented in python language using Python 3.6 version. Keras and Scikit-learn library of python was used to implement deep learning methods and the machine learning algorithms, respectively. All experiments were conducted in a Google collaboratory notebook environment. Also, each experiment was carried out five times, and the average of the results was reported. Figure 5 shows the training module of our proposed neural network, AptaNet. The training dataset of AptaNet includes both interacting (positive) and non-interacting (negative) aptamer-protein pairs. For each sample of the aptamer-protein pairs, aptamer sequences were obtained from Aptagen[107] and aptamer free base[108] databases.

Similarly, protein sequences are obtained from Swissprot[109] based on their protein-id. Next, a balancing strategy was applied to prevent the unbalancing problem. Then, feature extraction methods used these data to generate k-mer and Revck-mer for aptamers features and AAC and PseAAC for protein features. After that, by applying a feature selection method, important features were ranked and selected. Finally, obtained features in this step were fed to AptaNet, which learns the model for predicting API.

We perform several kinds of experiments. In particular, we performed four different sets of experiments: First, we investigated the effectiveness of the different feature groups, as mentioned in Table 1. We investigated the dataset effectiveness by considering the performance of multi-layer perceptron (MLP) and convolutional neural network (CNN) neural networks on the 32 different datasets. Secondly, we compared the performance of the two deep neural networks used in our research. Subsequently, we compared our model against some machine learning algorithms. Finally, we investigate the result of the feature selection method and AptaNet. In Fig. 6, the overall workflow for our methodology is graphically demonstrated.

Data collection.

To prepare our dataset, we obtained known API from two different databases: *Aptagen* and *Aptamer Base*. Aptagen contains 554 entries of interactions, which among them a total number of 477 are aptamers of RNA/DNA interacting and 241 target proteins. Freebase consists of 1638 interaction entries which 1381 of them are aptamers of DNA/RNA and 211 target proteins. For the proteins, since the identifiers of target proteins are presented in aptagen and Aptamer Base, like *Aspartame*, *Caffeine*, *Colicin E3*, etc., therefore, we prepared their sequences by searching in UniProtKB/Swiss-Prot based on the best name matches. For Aptagen, in 477 aptamers, there is 269 interaction with the 241 protein targets. Therefore, the 269 pairs of APIs are considered as positive samples. And for Aptamer Base in 1381 aptamers, only 725 interaction with 164 proteins was exist, so the 725 API are assumed as positive samples. To remove duplicate APIs between two databases, we unified the IDs of aptamers and the proteins. Thus, 850 APIs with 452 proteins target were obtained. Since all collected APIs are considered as positive instances and negative APIs are not determined in the databases listed. Therefore, negative instances of APIs were generated by a random pairs of aptamers and proteins, so that they did not any overlap with the positive instances. Finally, the total number of instances were 3404, which contain 850 positive and 2554 negative instances.

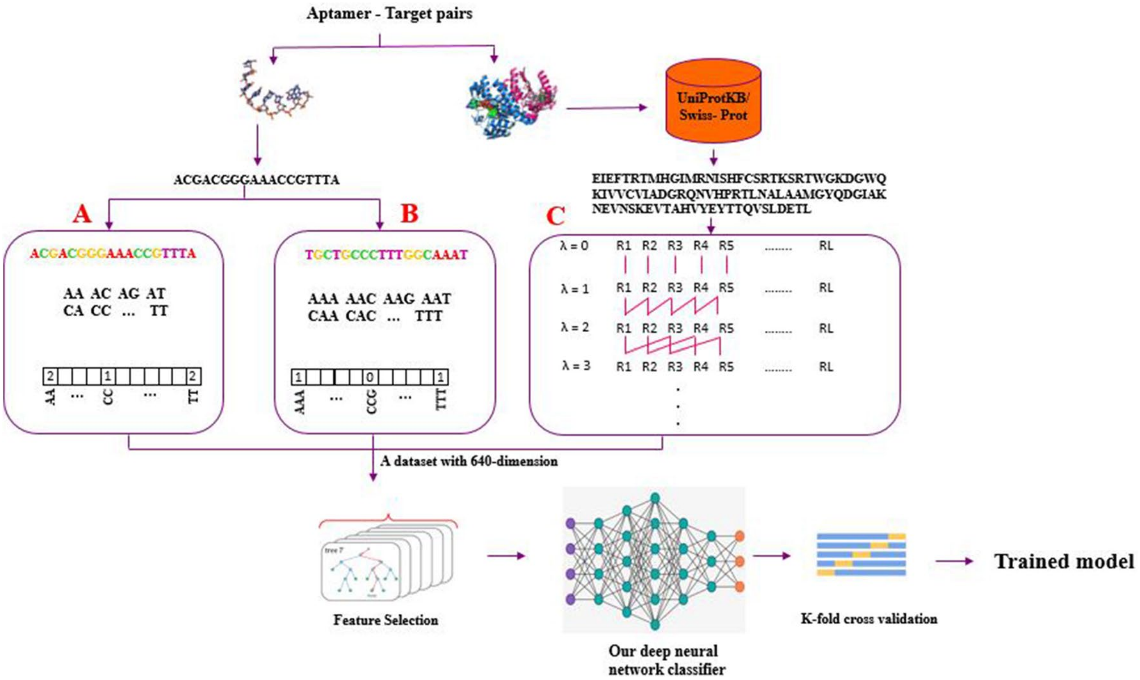

Aptamer - Target pairs
UniProtKB/ Swiss- Prot
ACGACGGGAAACCGTTTA
EIEFTRTMHGIMRNISHFCSRTKSRTWGKDGWQ KIVVCVIADGRQNVHPRTLNALAAMGYQDGIAK NEVNSKEVTAHVYEYTTQVSLDETL
A
B
C
ACGACGGGAAACCGTTTA
AA AC AG AT CA CC ... TT
2 1 2
AA ... CC ... TT
TGCTGCCCTTTGGCAAAT
AAA AAC AAG AAT CAA CAC ... TTT
1 0 1
AAA ... CCG ... TTT
λ = 0 R1 R2 R3 R4 R5 .... RL
λ = 1 R1 R2 R3 R4 R5 .... RL
λ = 2 R1 R2 R3 R4 R5 .... RL
λ = 3 R1 R2 R3 R4 R5 .... RL
A dataset with 640-dimension
Feature Selection
Our deep neural network classifier
K-fold cross validation
Trained model

Figure 5. A schematic overview of the training module of AptaNet. For a given sequence (A) Shows K-mer frequency strategy for aptamer encoding; (B) depicts Reverse compliment k-mer strategy for aptamer encoding and (C) Displays Pseudo-Amino Acid Composition strategy for protein encoding. The figure is drawn by using of Microsoft PowerPoint 2016 that is available in https://www.office.com/.

Balancing the dataset. Generally, in classification problems, learning methods assume that the number of samples in each class is almost equal[110]. However, in most real conditions, the class distribution is not similar because the number of representations of some classes is much more than the others. As a result, it can restrict the learning model's performance since they will be biased towards the majority class. Therefore, in this study, we used neighborhood cleaning to deal with such incompatibility (NCL). NCL is categorized into the groups of under-sampling strategy[111]. NCL was first introduced by Laurikkala[112] for balancing dataset by removing some instances from the majority class randomly to reduce the size of the majority class with/ or without replacement.

NCL procedure. Suppose we have an imbalanced dataset where O is a majority class, and C is a minority class. So, NCL, by using Wilson's edited nearest neighbor rule (ENN)[113], identified noisy data (A_1) and reduced O by removing A_1 in O. In other words, ENN removes instances that differ from at least two of their three nearest neighbors. Furthermore, NCL is intensifying size reduction by removing the three nearest neighbors that misclassify instances of C. Figure 7 describes the NCL algorithm.

Feature construction. In this study, Kmer frequency and Reverse compliment k-mer (k=2, 3) were adopted separately to encode the aptamer sequences. The amino acid composition and pseudo-amino acid composition were employed to encode the protein sequences.

K-mer frequency. Since, T-(Thymine) in DNA is similar to U-(uracil) in RNA therefore, we converted each RNA to DNA sequences by replacing U to T. K-mers are subsequences of length k (A, T, C, and G) to represent the DNAs sequence. Suppose n is the number of possible monomers (A, C, G, and T) so, a total possible k-mers will be n^k. (See Fig. 5A) In this study, k=3, 4 were adopted for each aptamer, and as a result, each aptamer was encoded into an 84-dimension for k=3 and *339*-dimension for k=4 numerical vector.

Reverse compliment k-mer. The reverse complement of a DNA sequence is organized by replacing T and A, exchanging G and C, and reversing the letters. (See Fig. 5B) As an example, if k=2 so there are *16* basic k-mers in total, but by reverse compliment k-mers, there are *ten* different k-mers. Therefore, the total possible number of reverse complement k-mer will be calculated as follows:

$$\begin{cases} 2^{2k-1}(k = 1, 3, 5, \ldots) \\ 2^{2k-1} + 2^{k-1}(k = 2, 4, 6, \ldots) \end{cases} \tag{1}$$

We set (k=3, 4) for each aptamer, and as a result, each aptamer was encoded into a *44*-dimensional vector for k=3 and a *179*-dimensional vector for k=4. In this study, to generate characteristics of aptamers, we used a powerful python package known as RepDNA[114].

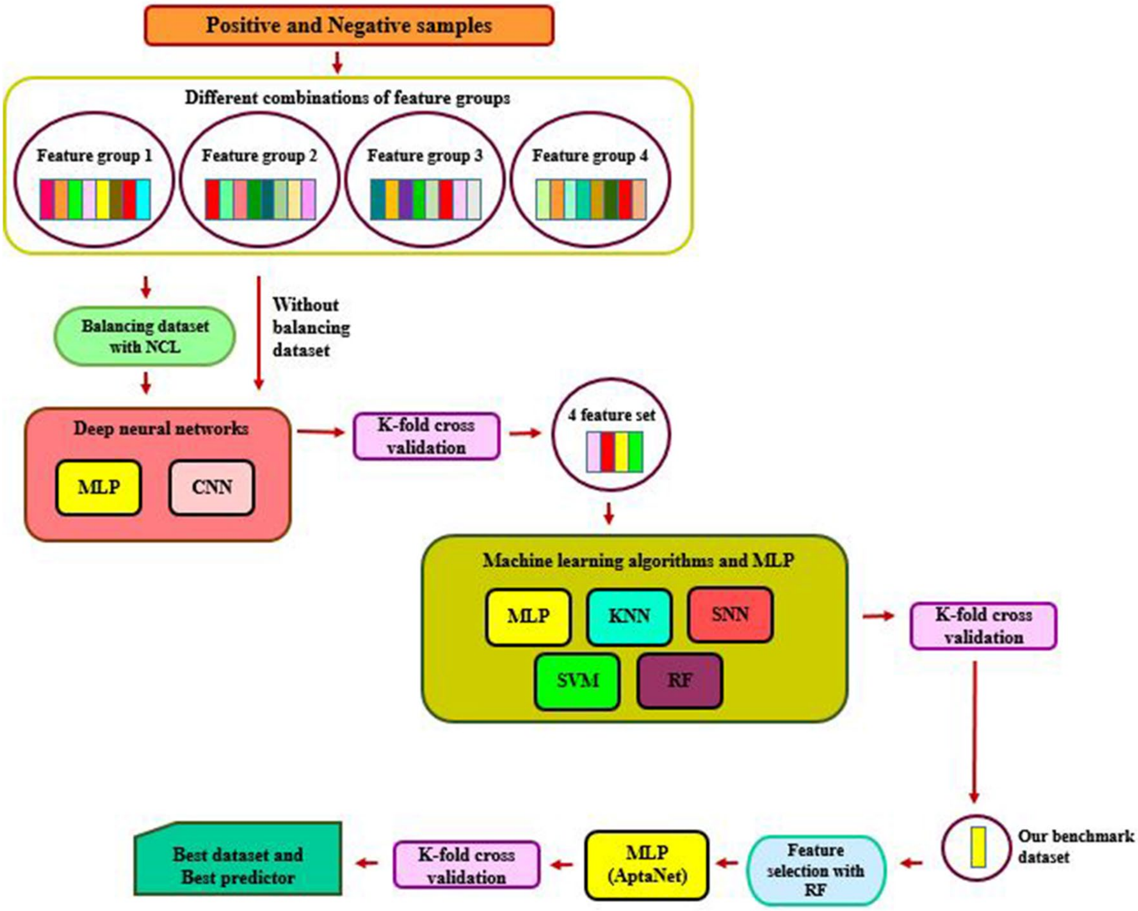
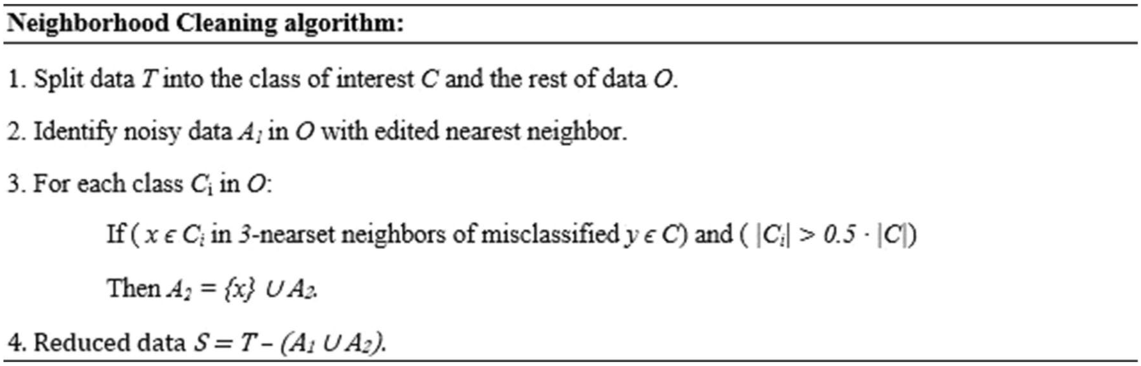


Figure 6. Flowchart of performed methodology to find the best dataset and predictor for potential aptamer-protein prediction. Where feature group 1 contained k-mer (k = 3) frequency for aptamers and 24 physicochemical and conformational properties of proteins sequences; group 2 involved k-mer (k = 4) frequency for aptamers and 24 physicochemical and conformational properties of proteins sequences, group 3 comprised Revckmer (k = 3) frequency for aptamers and 24 physicochemical and conformational properties of proteins sequences and group 4 included Revckmer (k = 4) frequency for aptamers and 24 physicochemical and conformational properties of proteins sequences. *NCL* neighborhood cleaning, *MLP* multi-layer perceptron, *CNN* convolutional neural network, *KNN* k nearest neighbor, *SNN* neural network, *SVM* support vector machine and *RF* random forest. The figure is drawn by using of Microsoft PowerPoint 2016 that is available in https://www.office.com/.



Neighborhood Cleaning algorithm:

```
1. Split data $T$ into the class of interest $C$ and the rest of data $O$.
2. Identify noisy data $A_1$ in $O$ with edited nearest neighbor.
3. For each class $C_i$ in $O$:
        If ( $x \epsilon C_i$ in 3-nearset neighbors of misclassified $y \epsilon C$) and ( $|C_i| > 0.5 \cdot |C|$)
        Then $A_2 = \{x\} \cup A_2$.
4. Reduced data $S = T - (A_1 \cup A_2)$.
```

Figure 7. The Neighborhood Cleaning Algorithm. The figure is drawn by using of Microsoft PowerPoint 2016 that is available in https://www.office.com/.

Amino acid composition (AAC).

AAC is a kind of protein sequence feature based on protein attributes, like folding types, secondary structure, domain, subcellular location, etc. AAC calculates the frequency of each amino acid in a protein sequence. For a protein sequence with *N* amino acid residues, the frequencies of each residue could be considered as follows:

$$f(t) = \frac{N(t)}{N}, t\epsilon \tag{2}$$

N(t) is the number of amino acid type *t*.

Pseudo-amino acid composition (PAAC).

Chou first introduced this group of descriptors for the prediction of protein subcellular properties[115]. (See Fig. 5C) PseAAC has been used as an effective feature extraction method in several biological problems[31–37]. The PseAAC method can be demonstrated as defined below:

Given a protein chain P with N amino acid residues:

$$P = R_1 R_2 R_3 \ldots R_N \tag{3}$$

The protein sequence order effect can be demonstrated by a set of separate correlation factors, as follows:

$$\begin{cases} \theta_1 = \frac{1}{L-1}\sum_{i=1}^{L-1}\Theta(R_i, R_{i+1}) \\ \theta_2 = \frac{1}{L-2}\sum_{i=1}^{L-2}\Theta(R_i, R_{i+2}) \\ \theta_3 = \frac{1}{L-3}\sum_{i=1}^{L-3}\Theta(R_i, R_{i+3}) \\ . \\ . \\ . \\ \theta_\lambda = \frac{1}{L-\lambda}\sum_{i=1}^{L-\lambda}\Theta\left(R_i, R_{i+\lambda}\right), (\lambda < L) \end{cases} \tag{4}$$

where θ_1, θ_2, …, θ_λ are called the 1-tier, 2-tier, and λ-th tier correlation factors, respectively. The correlation function can be shown by:

$$\Theta\left(R_i, R_j\right) = \frac{1}{3}\left\{\left[H_1\left(R_j\right) - H_1(R_i)\right]^2 + \left[H_2\left(R_j\right) - H_2(R_i)\right]^2 + \left[M\left(R_j\right) - M(R_i)\right]^2\right\} \tag{5}$$

where $H_1(R_i)$, $H_2(R_i)$, and $M(R_i)$ are, some properties (e.g., physicochemical, conformational, and energetic) value for the amino acid R_i; and also $H_1(R_j)$, $H_2(R_j)$, and $M(R_j)$ are the corresponding values of the amino acid R_j. Notably, each property values are transformed from the original values based on the following equation:

$$\begin{cases} H_1(i) = \frac{H_1^0(i) - \sum_{i=1}^{20}\frac{H_1^0(i)}{20}}{\sqrt{\frac{\sum_{i=1}^{20}\left[H_1^0(i) - \sum_{i=1}^{20}\frac{H_1^0(i)}{20}\right]}{20}}} \\ H_2(i) = \frac{H_2^0(i) - \sum_{i=1}^{20}\frac{H_2^0(i)}{20}}{\sqrt{\frac{\sum_{i=1}^{20}\left[H_2^0(i) - \sum_{i=1}^{20}\frac{H_2^0(i)}{20}\right]}{20}}} \\ M(i) = \frac{M^0(i) - \sum_{i=1}^{20}\frac{M^0(i)}{20}}{\sqrt{\frac{\sum_{i=1}^{20}\left[M^0(i) - \sum_{i=1}^{20}\frac{M^0(i)}{20}\right]}{20}}} \end{cases} \tag{6}$$

where H_1 (i), H_2 (i), and M (i) are the original property values for the amino acids. Therefore, for a protein sequence P, the PseAAC can be illustrated by a vector with $(20 + \lambda)$—Dimensional as:

$$[V_1, V_2, \ldots, V_{20}, V_{21}, \ldots, V_{20+\lambda}]^T \tag{7}$$

T is the transpose operator.

$$X_u = \begin{cases} \frac{f_u}{\sum_{i=1}^{20} f_i + \omega\sum_{j=1}^{\lambda}\theta_j}, (1 \le u \le 20) \\ \frac{\omega\theta_{u-20}}{\sum_{i=1}^{20} f_i + \omega\sum_{j=1}^{\lambda}\theta_j}, (20 + 1 \le u \le 20 + \lambda) \end{cases} \tag{8}$$

where for the mentioned protein sequence P, f_i shows the occurrence frequencies of the 20 amino acids, and θj shows j-th tier sequence correlation factor that is calculated based on Eq. (2), and ω shows the weight factor of the sequence order effect. We set ω = 0.05. According to the above description, the first 20 components in Eq. (5) shows the amino acid composition effect, and the other remaining components (20 + 1 to 20 + λ) show the effect of sequence order. So, the whole of 20 + λ components will be PseAAC. We set λ = 30.

In this study, we used 24 physicochemical and biochemical (i.e., hydrophobicity, hydrophilicity, mass, polarity, molecular weight, melting point, transfer-free energy, buriability, bulkiness, solvation free energy, relative mutability, residue volume, volume, amino acid distribution, hydration number, isoelectric point, compressibility, chromatographic index, unfolding entropy change, unfolding enthalpy, unfolding Gibbs free energy change, power to beat the N terminal, C terminal and middle of alpha helix) properties of amino acids. The 24 properties were retrieved from[116,117] which could be found in Supplementary Table S3.

Also, in our study, to generate characteristics of proteins, we used a python package known as iFeature[118].

Description of deep neural network model.

Multi-layer perceptron (MLP). We have selected the MLP as our classification model. A seven-layer neuron network was performed in the fully connected layer to generate the final prediction of interaction between aptamer and protein. The number of layers we selected here was depended on the various tests among the seven different layers (i.e., *3, 4, 5, 6, 7, 8*, and *9*) and the comparison of their outcomes. Finally, the best outcomes were obtained when the seven-layer network was applied.

All the neuron units in each layer (layer i) were connected to its previous layer (i – 1), and outcomes produced by using non-linear transformation function f as follows:

$$o_j = \left(\sum_{i=1}^{H} w_i o_i + b_i \right) \tag{9}$$

where H represents the number of hidden neurons, w and b are the weights and bias of neuron j, which summarize all the hidden units. After each fully-connected layer, the network performed an activation function named rectified linear unit (Relu).

$$ReLU(x) = \begin{cases} x, x \geq 0 \\ 0, x < 0 \end{cases} \tag{10}$$

Relu is a non-linear function that can extract hidden patterns in the data and reduce gradient vanishing. The Dropout was applied in order to avoid overfitting behind every fully connected layer. The outcome in the last layer was obtained using the sigmoid function as follows:

$$Sigmoid(x) = \frac{1}{1 + e^{-x}} \tag{11}$$

To train the network, we minimized the objective function for loss minimization. We used a function named binary cross-entropy cost function C, as follows:

$$C = -\frac{1}{n} \sum_{x} \sum_{t} \left[yIna + (1 - y)In(1 - a) \right] \tag{12}$$

where C is the output of the loss function called the binary cross-entropy cost function. Furthermore, x indicates the training sample index, and t represents the index of different labels, y indicates the true value of sample x, which can be *0* or *1*. And *a* is the predicted output of the network for *0* or *1* value given input sample x. When the predicted outputs are close to the true values, the value of C will get less. Since the cross-entropy is a non-negative function, so to get the best prediction, the function must minimize.

Comparison with machine learning algorithms.

To compare our model and some machine learning algorithms, we compared the performance of AptaNet against some machine learning algorithms. We selected five machine learning algorithms, namely, SNN, KNN, RF, and SVM. We performed fivefold cross-validation to evaluate the performance of the test and training set.

Feature selection.

In order to prevent overfitting and select the most important features, we used the RF algorithm. RF is one of the robust machine learning algorithms created by Loe Breiman[119]. RF is an ensemble learning containing multiple decision trees. Each of the decision trees is created based on a random extraction of the instances and the features. Since each tree does not check all features and instances, so, it can be concluded that trees are de-correlated, and therefore the possibility of their over-fitting is less.

RF selects important features by ranking all features based on the improvement of the node purity. The node's probability is the number of instances that reach the node, divided by the whole number of instances. Therefore, the importance of the features has a direct relation with the high value of the probability. We chose forests containing nine trees based on our feature selection experiments.

Performance evaluation.

In this study, we used fivefold cross-validation to evaluate the performance of our model. During this procedure, the whole dataset is evenly and randomly divided into five folds which four folds are used for training and one for testing. This method was repeated five times, and each instance was tested just once. In order of evaluation of the predictor performance, the prediction accuracy, f1_macro, precision, specificity, sensitivity (recall), and matthews's correlation coefficient were computed as below:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

$$Precision = \frac{TP}{TP + FP} \tag{14}$$

$$F1score = \frac{Precision \times Sensitivity}{Precision + Sensitivity} \tag{15}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{16}$$

$$Specificity = \frac{TN}{TN + FP} \tag{17}$$

$$Matthews' sCorrelationCoefficient = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{18}$$

where TP, FP, TN, and FN represent true positive, false positive, true negative, and false negative, respectively.

Received: 26 September 2020; Accepted: 3 March 2021
Published online: 16 March 2021

References

1. Robertson, D. L. & Joyce, G. F. Selection in vitro of an RNA enzyme that specifically cleaves single-stranded DNA. *Nature* **344**, 467–468. https://doi.org/10.1038/344467a0 (1990).
2. Ellington, A. D. & Szostak, J. W. In vitro selection of RNA molecules that bind specific ligands. *Nature* **346**, 818–822. https://doi.org/10.1038/346818a0 (1990).
3. Tuerk, C. & Gold, L. Systematic evolution of ligands by exponential enrichment: RNA ligands to bacteriophage T4 DNA polymerase. *Science* **249**, 505–510 (1990).
4. Iliuk, A. B., Hu, L. & Tao, W. A. Aptamer in bioanalytical applications. *Anal. Chem.* **83**, 4440–4452. https://doi.org/10.1021/ac201057w (2011).
5. Ashrafuzzaman, M. Aptamers as both drugs and drug-carriers. *Biomed. Res. Int.* **2014**, 697923. https://doi.org/10.1155/2014/697923 (2014).
6. Binning, J. M. *et al.* Development of RNA aptamers targeting Ebola virus VP35. *Biochemistry* **52**, 8406–8419. https://doi.org/10.1021/bi400704d (2013).
7. Jaax, M. E. *et al.* Complex formation with nucleic acids and aptamers alters the antigenic properties of platelet factor 4. *Blood* **122**, 272–281. https://doi.org/10.1182/blood-2013-01-478966 (2013).
8. Wang, P. *et al.* Aptamers as therapeutics in cardiovascular diseases. *Curr. Med. Chem.* **18**, 4169–4174 (2011).
9. Tõpala, T. *et al.* New sulfonamide complexes with essential metal ions [Cu(II), Co(II), Ni(II) and Zn(II)]. Effect of the geometry and the metal ion on DNA binding and nuclease activity. BSA protein interaction. *J. Inorg. Biochem.* **202**, 110823 (2020).
10. Zhu, Q., Liu, G. & Kai, M. DNA aptamers in the diagnosis and treatment of human diseases. *Molecules* **20**, 20979–20997. https://doi.org/10.3390/molecules201219739 (2015).
11. Gonzalez, V. M., Martin, M. E., Fernandez, G. & Garcia-Sacristan, A. Use of aptamers as diagnostics tools and antiviral agents for human viruses. *Pharmaceuticals* https://doi.org/10.3390/ph9040078 (2016).
12. Passariello, M., Camorani, S., Vetrei, C., Cerchia, L. & De Lorenzo, C. Novel human bispecific aptamer-antibody conjugates for efficient cancer cell killing. *Cancers* https://doi.org/10.3390/cancers11091268 (2019).
13. Tian, H., Duan, N., Wu, S. & Wang, Z. Selection and application of ssDNA aptamers against spermine based on Capture-SELEX. *Anal. Chim. Acta* **1081**, 168–175. https://doi.org/10.1016/j.aca.2019.07.031 (2019).
14. Flamme, M., McKenzie, L. K., Sarac, I. & Hollenstein, M. Chemical methods for the modification of RNA. *Methods* **161**, 64–82. https://doi.org/10.1016/j.ymeth.2019.03.018 (2019).
15. Zhu, C., Yang, G., Ghulam, M., Li, L. & Qu, F. Evolution of multi-functional capillary electrophoresis for high-efficiency selection of aptamers. *Biotechnol. Adv.* **107**, 432. https://doi.org/10.1016/j.biotechadv.2019.107432 (2019).
16. Emami, N., Pakchin, P. S. & Ferdousi, R. Computational predictive approaches for interaction and structure of aptamers. *J. Theor. Biol.* **497**, 110268 (2020).
17. Li, B. Q. *et al.* Prediction of aptamer-target interacting pairs with pseudo-amino acid composition. *PLoS ONE* **9**, e86729. https://doi.org/10.1371/journal.pone.0086729 (2014).
18. Zhang, L., Zhang, C., Gao, R., Yang, R. & Song, Q. Prediction of aptamer-protein interacting pairs using an ensemble classifier in combination with various protein sequence attributes. *BMC Bioinform.* **17**, 225. https://doi.org/10.1186/s12859-016-1087-5 (2016).
19. Yang, Q., Jia, C. & Li, T. Prediction of aptamer-protein interacting pairs based on sparse autoencoder feature extraction and an ensemble classifier. *Math. Biosci.* **311**, 103–108. https://doi.org/10.1016/j.mbs.2019.01.009 (2019).
20. Li, J., Ma, X., Li, X. & Gu, J. PPAI: A web server for predicting protein-aptamer interactions. *BMC Bioinform.* **21**, 1–15 (2020).
21. Wang, Y., Cao, Z., Zeng, D., Wang, X. & Wang, Q. Using deep learning to predict the hand-foot-and-mouth disease of enterovirus A71 subtype in Beijing from 2011 to 2018. *Sci. Rep.* **10**, 1–10 (2020).
22. Beknazarov, N., Jin, S. & Poptsova, M. Deep learning approach for predicting functional Z-DNA regions using omics data. *Sci. Rep.* **10**, 1–15 (2020).
23. Gao, M., Zhou, H. & Skolnick, J. DESTINI: A deep-learning approach to contact-driven protein structure prediction. *Sci. Rep.* **9**, 1–13 (2019).
24. El-Attar, N. E., Hassan, M. K., Alghamdi, O. A. & Awad, W. A. Deep learning model for classification and bioactivity prediction of essential oil-producing plants from Egypt. *Sci. Rep.* **10**, 1–10 (2020).
25. Lee, J. F., Hesselberth, J. R., Meyers, L. A. & Ellington, A. D. Aptamer database. *Nucleic Acids Res.* **32**, D95–D100 (2004).
26. Thodima, V., Pirooznia, M. & Deng, Y. *BMC Bioinformatics* 1–6 (BioMed Central, 2020).
27. Cruz-Toledo, J. *et al.* Aptamer base: A collaborative knowledge base to describe aptamers and SELEX experiments. *Database* 2012 (2012).
28. Khatun, M. S., Hasan, M. M., Shoombuatong, W. & Kurata, H. ProIn-Fuse: Improved and robust prediction of proinflammatory peptides by fusing of multiple feature representations. *J. Comput. Aided Mol. Des.* **34**, 1229–1236 (2020).
29. Hasan, M. M. *et al.* Meta-i6mA: An interspecies predictor for identifying DNA N6-methyladenine sites of plant genomes by exploiting informative features in an integrative machine-learning framework. *Brief. Bioinform.* (2020).

30. Hasan, M. M., Manavalan, B., Khatun, M. S. & Kurata, H. i4mC-ROSE, a bioinformatics tool for the identification of DNA N4-methylcytosine sites in the Rosaceae genome. *Int. J. Biol. Macromol.* **157**, 752–758 (2020).
31. Bakhtiarizadeh, M. R., Rahimi, M., Mohammadi-Sangcheshmeh, A., Shariati, J. V. & Salami, S. A. PrESOgenesis: A two-layer multi-label predictor for identifying fertility-related proteins using support vector machine and pseudo amino acid composition approach. *Sci. Rep.* **8**, 9025. https://doi.org/10.1038/s41598-018-27338-9 (2018).
32. Mei, J. & Zhao, J. Prediction of HIV-1 and HIV-2 proteins by using Chou's pseudo amino acid compositions and different classifiers. *Sci. Rep.* **8**, 2359. https://doi.org/10.1038/s41598-018-20819-x (2018).
33. Ariaeenejad, S. *et al.* A computational method for prediction of xylanase enzymes activity in strains of *Bacillus subtilis* based on pseudo amino acid composition features. *PLoS ONE* **13**, e0205796. https://doi.org/10.1371/journal.pone.0205796 (2018).
34. Xiao, X., Cheng, X., Chen, G., Mao, Q. & Chou, K. C. pLoc_bal-mVirus: predict subcellular localization of multi-label virus proteins by Chou's general PseAAC and IHTS treatment to balance training dataset. *Med. Chem.* **15**, 496–509. https://doi.org/10.2174/1573406415666181217114710 (2019).
35. Jia, J., Li, X., Qiu, W., Xiao, X. & Chou, K. C. iPPI-PseAAC(CGR): Identify protein-protein interactions by incorporating chaos game representation into PseAAC. *J. Theor. Biol.* **460**, 195–203. https://doi.org/10.1016/j.jtbi.2018.10.021 (2019).
36. Ju, Z. & Wang, S. Y. Prediction of citrullination sites by incorporating k-spaced amino acid pairs into Chou's general pseudo amino acid composition. *Gene* **664**, 78–83. https://doi.org/10.1016/j.gene.2018.04.055 (2018).
37. Yu, B. *et al.* Prediction of protein structural class for low-similarity sequences using Chou's pseudo amino acid composition and wavelet denoising. *J. Mol. Graph. Model.* **76**, 260–273. https://doi.org/10.1016/j.jmgm.2017.07.012 (2017).
38. Saghapour, E. & Sehhati, M. Physicochemical position-dependent properties in the protein secondary structures. *Iran. Biomed. J.* **23**, 253 (2019).
39. Ma, X., Guo, J. & Sun, X. DNABP: Identification of DNA-binding proteins based on feature selection using a random forest and predicting binding residues. *PLoS ONE* **11**, e0167354 (2016).
40. Gleeson, M. P., Hersey, A., Montanari, D. & Overington, J. Probing the links between in vitro potency, ADMET and physicochemical parameters. *Nat. Rev. Drug Discov.* **10**, 197–208 (2011).
41. Macalino, S. J. Y. *et al.* Evolution of in silico strategies for protein-protein interaction drug discovery. *Molecules* **23**, 1963 (2018).
42. Ding, Y., Tang, J. & Guo, F. Predicting protein-protein interactions via multivariate mutual information of protein sequences. *BMC Bioinform.* **17**, 398 (2016).
43. Ding, Y., Tang, J. & Guo, F. Identification of protein–protein interactions via a novel matrix-based sequence representation model with amino acid contact information. *Int. J. Mol. Sci.* **17**, 1623 (2016).
44. Guo, F. *et al.* Identifying protein-protein interface via a novel multi-scale local sequence and structural representation. *BMC Bioinform.* **20**, 1–11 (2019).
45. Agustianto, K. & Destarianto, P. in *2019 International Conference on Computer Science, Information Technology, and Electrical Engineering (ICOMITEE).* 86–89 (IEEE).
46. Faris, H. Neighborhood cleaning rules and particle swarm optimization for predicting customer churn behavior in telecom industry. *Int. J. Adv. Sci. Technol.* **68**, 11–22 (2014).
47. Suman, S., Laddhad, K. & Deshmukh, U. Methods for handling highly skewed datasets. *Part I-October* 3 (2005).
48. Bach, M., Werner, A. & Palt, M. The proposal of undersampling method for learning from imbalanced datasets. *Procedia Comput. Sci.* **159**, 125–134 (2019).
49. Sun, Y. & Liu, F. in *2016 2nd IEEE International Conference on Computer and Communications (ICCC).* 1157–1161 (IEEE).
50. Rekha, G., Reddy, V. K. & Tyagi, A. K. CIRUS: Critical instances removal based under-sampling: A solution for class imbalance problem. *Int. J. Hybrid Intell. Syst.* **16**, 55–66 (2020).
51. Choi, J., Park, S. & Ahn, J. RefDNN: A reference drug based neural network for more accurate prediction of anticancer drug resistance. *Sci. Rep.* **10**, 1–11 (2020).
52. Jha, D. *et al.* Elemnet: Deep learning the chemistry of materials from only elemental composition. *Sci. Rep.* **8**, 1–13 (2018).
53. Rifaioglu, A. S., Doğan, T., Martin, M. J., Cetin-Atalay, R. & Atalay, V. DEEPred: Automated protein function prediction with multi-task feed-forward deep neural networks. *Sci. Rep.* **9**, 1–16 (2019).
54. Pan, X. & Shen, H.-B. Predicting RNA–protein binding sites and motifs through combining local and global deep convolutional neural networks. *Bioinform.* **34**, 3427–3436 (2018).
55. Lo, C. & Marculescu, R. MetaNN: Accurate classification of host phenotypes from metagenomic data using neural networks. *BMC Bioinform.* **20**, 314 (2019).
56. Peng, C., Han, S., Zhang, H. & Li, Y. RPITER: A hierarchical deep learning framework for ncRNA–protein interaction prediction. *Int. J. Mol. Sci.* **20**, 1070 (2019).
57. Lam, J. H. *et al.* A deep learning framework to predict binding preference of RNA constituents on protein surface. *Nat. Commun.* **10**, 1–13 (2019).
58. Tian, K., Shao, M., Wang, Y., Guan, J. & Zhou, S. Boosting compound-protein interaction prediction by deep learning. *Methods* **110**, 64–72 (2016).
59. Hashemifar, S., Neyshabur, B., Khan, A. A. & Xu, J. Predicting protein–protein interactions through sequence-based deep learning. *Bioinformatics* **34**, i802–i810 (2018).
60. Guo, Y. & Chen, X. A deep learning framework for improving protein interaction prediction using sequence properties. *bioRxiv* **13**, 843755 (2019).
61. Xie, Z., Deng, X. & Shu, K. Prediction of protein–protein interaction sites using convolutional neural network and improved data sets. *Int. J. Mol. Sci.* **21**, 467 (2020).
62. Hirwani, A. & Gonnade, S. Character recognition using multi-layer perceptron. *Int. J. Comput. Sci. Inf. Technol.* **5**, 558–661 (2014).
63. He, H., Zhao, J. & Sun, G. Prediction of MoRFs in protein sequences with MLPs based on sequence properties and evolution information. *Entropy* **21**, 635 (2019).
64. Feng, S., Zhao, C. & Fu, P. A deep neural network based hierarchical multi-label classification method. *Rev. Sci. Instrum.* **91**, 024103 (2020).
65. Lin, Z., Lanchantin, J. & Qi, Y. MUST-CNN: A multi-layer shift-and-stitch deep convolutional architecture for sequence-based protein structure prediction. http://arxiv.org/abs/1605.03004 (2016).
66. Kushwaha, S. K. & Shakya, M. in *2009 International Conference on Advances in Recent Technologies in Communication and Computing.* 465–467 (IEEE).
67. Xie, Y., Jin, P., Gong, M., Zhang, C. & Yu, B. Multi-task network representation learning. *Front. Neurosci.* **14**, 1–1. https://doi.org/10.3389/fnins.2020.00001 (2020).
68. Wang, Z. *et al.* Optimized multi-layer perceptrons for molecular classification and diagnosis using genomic data. *Bioinformatics* **22**, 755–761 (2006).
69. Rastegari, M., Ordonez, V., Redmon, J. & Farhadi, A. *European Conference on Computer Vision* 525–542 (Springer, 2019).
70. Zhang, Y.-D. *et al.* Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation. *Multimed. Tools Appl.* **78**, 3613–3632 (2019).
71. Rutter, E. M., Lagergren, J. H. & Flores, K. B. *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data* 190–198 (Springer, 2019).

72. Kang, M.-S. *et al.* Accuracy improvement of quantification information using super-resolution with convolutional neural network for microscopy images. *Biomed. Signal Process. Control* **58**, 101846 (2020).
73. Ghose, S., Singh, N. & Singh, P. in *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence).* 511–517 (IEEE).
74. Tan, C. *et al.* DeepBrainSeg: Automated brain region segmentation for micro-optical images with a convolutional neural network. *Front. Neurosci.* **14**, 1 (2020).
75. Karamizadeh, S., Abdullah, S. M., Halimi, M., Shayan, J. & javad Rajabi, M. in *2014 international conference on computer, communications, and control technology (I4CT).* 63–65 (IEEE).
76. Hou, Q., Lv, M., Zhen, L. & Jing, L. Support vector machine with hypergraph-based pairwise constraints. *Springerplus* **5**, 1651–1651. https://doi.org/10.1186/s40064-016-3315-x (2016).
77. Chou, J.-S., Cheng, M.-Y., Wu, Y.-W. & Pham, A.-D. Optimizing parameters of support vector machine using fast messy genetic algorithm for dispute classification. *Expert Syst. Appl.* **41**, 3955–3964 (2014).
78. Deka, P. C. Support vector machine applications in the field of hydrology: A review. *Appl. Soft Comput.* **19**, 372–386 (2014).
79. Kortemme, T. & Baker, D. Computational design of protein–protein interactions. *Curr. Opin. Chem. Biol.* **8**, 91–97 (2004).
80. Verkhivker, G., Appelt, K., Freer, S. & Villafranca, J. Empirical free energy calculations of ligand-protein crystallographic complexes. I. Knowledge-based ligand-protein interaction potentials applied to the prediction of human immunodeficiency virus 1 protease binding affinity. *Protein Eng. Des. Select.* **8**, 677–691 (1995).
81. Lockless, S. W. & Ranganathan, R. Evolutionarily conserved pathways of energetic connectivity in protein families. *Science* **286**, 295–299 (1999).
82. Darnell, S. J., Page, D. & Mitchell, J. C. An automated decision-tree approach to predicting protein interaction hot spots. *Proteins Struct. Funct. Bioinform.* **68**, 813–823 (2007).
83. Mattice, W. L., Riser, J. M. & Clark, D. S. Conformational properties of the complexes formed by proteins and sodium dodecyl sulfate. *Biochemistry* **15**, 4264–4272 (1976).
84. Das, K. P., Petrash, J. M. & Surewicz, W. K. Conformational properties of substrate proteins bound to a molecular chaperone-crystallin. *J. Biol. Chem.* **271**, 10449–10452 (1996).
85. Vaccaro, A. M. *et al.* pH-dependent conformational properties of saposins and their interactions with phospholipid membranes. *J. Biol. Chem.* **270**, 30576–30580 (1995).
86. Tsai, C.-J., Ma, B. & Nussinov, R. Protein–protein interaction networks: how can a hub protein bind so many different partners?. *Trends Biochem. Sci.* **34**, 594–600 (2009).
87. Guillen-Ramirez, H. A. & Martinez-Perez, I. M. Classification of riboswitch sequences using k-mer frequencies. *Biosystems* **174**, 63–76. https://doi.org/10.1016/j.biosystems.2018.09.001 (2018).
88. Zeng, C. & Hamada, M. Identifying sequence features that drive ribosomal association for lncRNA. *BMC Genom.* **19**, 906. https://doi.org/10.1186/s12864-018-5275-8 (2018).
89. Wen, J. *et al.* A classification model for lncRNA and mRNA based on k-mers and a convolutional neural network. *BMC Bioinform.* **20**, 469. https://doi.org/10.1186/s12859-019-3039-3 (2019).
90. Cheng, S. *et al.* DM-RPIs: Predicting ncRNA-protein interactions using stacked ensembling strategy. *Comput. Biol. Chem.* **83**, 107088. https://doi.org/10.1016/j.compbiolchem.2019.107088 (2019).
91. Wekesa, J. S., Luan, Y., Chen, M. & Meng, J. A Hybrid prediction method for plant lncRNA–protein interaction. *Cells* https://doi.org/10.3390/cells8060521 (2019).
92. Kirk, J. M. *et al.* Functional classification of long non-coding RNAs by k-mer content. *Nat. Genet.* **50**, 1474–1482. https://doi.org/10.1038/s41588-018-0207-8 (2018).
93. Yousef, A. & Charkari, N. M. A novel method based on physicochemical properties of amino acids and one class classification algorithm for disease gene identification. *J. Biomed. Inform.* **56**, 300–306. https://doi.org/10.1016/j.jbi.2015.06.018 (2015).
94. Sęczyk, Ł, Świeca, M., Kapusta, I. & Gawlik-Dziki, U. Protein–phenolic interactions as a factor affecting the physicochemical properties of white bean proteins. *Molecules* **24**, 408. https://doi.org/10.3390/molecules24030408 (2019).
95. Tran, K. T. *et al.* A comparative assessment study of known small-molecule Keap1-Nrf2 protein–protein interaction inhibitors: Chemical synthesis, binding properties, and cellular activity. *J. Med. Chem.* **62**, 8028–8052. https://doi.org/10.1021/acs.jmedchem.9b00723 (2019).
96. Lazar, T., Guharoy, M., Schad, E. & Tompa, P. Unique physicochemical patterns of residues in protein–protein interfaces. *J. Chem. Inf. Model.* **58**, 2164–2173. https://doi.org/10.1021/acs.jcim.8b00270 (2018).
97. Li, G. & Zhu, F. Physicochemical properties of quinoa flour as affected by starch interactions. *Food Chem.* **221**, 1560–1568. https://doi.org/10.1016/j.foodchem.2016.10.137 (2017).
98. Xiang, N., Lyu, Y., Zhu, X., Bhunia, A. K. & Narsimhan, G. Effect of physicochemical properties of peptides from soy protein on their antimicrobial activity. *Peptides* **94**, 10–18. https://doi.org/10.1016/j.peptides.2017.05.010 (2017).
99. Guo, F., Li, S. C., Du, P. & Wang, L. Probabilistic models for capturing more physicochemical properties on protein-protein interface. *J. Chem. Inf. Model.* **54**, 1798–1809. https://doi.org/10.1021/ci5002372 (2014).
100. Paengkoum, P. *et al.* Molecular weight, protein binding affinity and methane mitigation of condensed tannins from mangosteen-peel (*Garcinia mangostana* L). *Asian-Austral. J. Anim. Sci.* **28**, 1442–1448. https://doi.org/10.5713/ajas.13.0834 (2015).
101. Seelam Prabhakar, P. *et al.* Impact of the position of the chemically modified 5-furyl-2'-deoxyuridine nucleoside on the thrombin DNA aptamer-protein complex: structural insights into aptamer response from MD simulations. *Molecules* https://doi.org/10.3390/molecules24162908 (2019).
102. Rupesh, K. R., Smith, A. & Boehmer, P. E. Ligand induced stabilization of the melting temperature of the HSV-1 single-strand DNA binding protein using the thermal shift assay. *Biochem. Biophys. Res. Commun.* **454**, 604–608. https://doi.org/10.1016/j.bbrc.2014.10.145 (2014).
103. Zhu, R., Li, G., Liu, J.-X., Dai, L.-Y. & Guo, Y. ACCBN: Ant-colony-clustering-based bipartite network method for predicting long non-coding RNA–protein interactions. *BMC Bioinform.* **20**, 16 (2019).
104. Zhan, Z.-H., Jia, L.-N., Zhou, Y., Li, L.-P. & Yi, H.-C. BGFE: A deep learning model for ncRNA-protein interaction predictions based on improved sequence information. *Int. J. Mol. Sci.* **20**, 978 (2019).
105. Sumonja, N., Gemovic, B., Veljkovic, N. & Perovic, V. Automated feature engineering improves prediction of protein–protein interactions. *Amino Acids* **51**, 1187–1200 (2019).
106. Xie, G., Wu, C., Sun, Y., Fan, Z. & Liu, J. Lpi-ibnra: Long non-coding rna-protein interaction prediction based on improved bipartite network recommender algorithm. *Front. Genet.* **10**, 343 (2019).
107. https://www.aptagen.com/.
108. Cruz-Toledo, J. *et al.* Aptamer Base: A collaborative knowledge base to describe aptamers and SELEX experiments. *Database* https://doi.org/10.1093/database/bas006 (2012).
109. https://www.uniprot.org/uniprot/.
110. Ali, A., Shamsuddin, S. M. & Ralescu, A. L. Classification with class imbalance problem: A review. *Int. J. Adv. Soft CompuT. Appl* **7**, 176–204 (2015).
111. Liu, X.-Y., Wu, J. & Zhou, Z.-H. Exploratory undersampling for class-imbalance learning. *IEEE Trans. Syst. Man Cybern. B* **39**, 539–550 (2008).
112. Laurikkala, J. *Conference on Artificial Intelligence in Medicine in Europe* 63–66 (Springer, 2019).

113. Wilson, D. L. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Trans. Syst. Man Cybern.* **3**, 408–421 (1972).
114. Liu, B., Liu, F., Fang, L., Wang, X. & Chou, K. C. repDNA: A Python package to generate various modes of feature vectors for DNA sequences by incorporating user-defined physicochemical properties and sequence-order effects. *Bioinformatics* **31**, 1307–1309. https://doi.org/10.1093/bioinformatics/btu820 (2015).
115. Ding, Y. S., Zhang, T. L. & Chou, K. C. Prediction of protein structure classes with pseudo amino acid composition and fuzzy support vector machine network. *Protein Pept. Lett.* **14**, 811–815. https://doi.org/10.2174/092986607781483778 (2007).
116. Kawashima, S. *et al.* AAindex: amino acid index database, progress report 2008. *Nucleic Acids Res.* **36**, D202-205. https://doi.org/10.1093/nar/gkm998 (2008).
117. Gromiha, M. M. A statistical model for predicting protein folding rates from amino acid sequence with structural class information. *J. Chem. Inf. Model.* **45**, 494–501. https://doi.org/10.1021/ci049757q (2005).
118. Chen, Z. *et al.* iFeature: A Python package and web server for features extraction and selection from protein and peptide sequences. *Bioinformatics* **34**, 2499–2502. https://doi.org/10.1093/bioinformatics/bty140 (2018).
119. Breiman, L. Random forests. *Mach. Learn.* **45**, 5–32 (2001).

Acknowledgements

This research was a part of an MSc thesis approved and funded by Tabriz University of Medical Sciences.

Author contributions

All authors contributed to writing manuscript text, building datasets, implementing neural networks and machine learning algorithms, analyzing data and predictors, and reviewing the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-021-85629-0.

Correspondence and requests for materials should be addressed to R.F.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.