2023 11th International Conference on Bioinformatics and Computational Biology (ICBCB) | 979-8-3503-9787-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICBCB57893.2023.10246468

Exploring the Transcriptional and Translational Features Using Deep Neural Networks for mRNAs Classification

Amira Kefi*
Department of BioMedical Engineering
College of Engineering,
University of Illinois at Chicago
Chicago, IL, USA
akefi2@uic.edu

Morris Chukhman*
Department of Bioengineering
College of Engineering,
University of Illinois at Chicago
Chicago, IL, USA
chukhman@uic.edu

Vinayakumar Karintha
Machine Learning Practice, UST,
Aliso Viejo CA
Vinayakumar.karintha@ust.com

Sadok Bouamama
National Engineering School of Manouba (ENSI), Manouba University,
LaManouba, Tunisia
sadok.bouamama@ensi.rnu.tn

Jie Yang
Department of Mathematics, Statistics, and Computer Science.
University of Illinois at Chicago
Chicago, USA
jyang06@uic.edu

Chunyu Liu
Department of Psychiatry, Department of Neuroscience and physiology, *SUNY Upstate Medical University*
Syracuse, NY, USA
luich@upstate.edu

***Abstract*—Recent advent of the second and third generation of sequencing has uncovered many novel transcripts. These novel transcripts could have crucial functions in different biological processes and might be related to challenging diseases and pathogenesis. However, whether these genes should be classified as protein coding RNAs (pcRNAs) or long non-coding RNAs (lncRNAs) is still debated and unclear. In this study we propose a coding potential classification framework based on deep neural networks and novel features from RNA-seq and Ribo-seq data to classify RNAs transcripts into protein coding and long non coding. As far as we know, this is the first method that uses RNA-seq and Ribo-seq as predictors to classify RNAs using a deep neural network model. Compared to other methods, the prediction of our method reached 97.4% accuracy.**

Keywords—deep neural networks, protein coding RNAs, long non-coding RNAs, RNA-seq, Ribo-seq, Machine learning Models

I. INTRODUCTION

In the last decade, the increasing number of transcripts from the second and third generations of sequencing technologies has detected many novel transcripts [1], [2], [3]. A lot of the produced transcripts are either from protein coding or non-coding genes. A protein coding gene has RNAs that could be translated into a functional protein while the non-coding genes do not encode a protein but their RNAs still can be functional and have important roles in the regulation of gene expression and many diseases progression [4]. Differentiating coding RNAs from non-coding RNAs (ncRNAs), especially the lncRNAs, is crucial for downstream analysis and determination of biological processes.

Machine learning (ML) is a section of artificial intelligence that represents a set of intelligent algorithms by imitating human behavior to resolve complex problems. ML algorithms have many applications in bioinformatics [5], [6], [7], [8], [9]. Many ML models have been widely used to predict the coding state of RNA transcripts [10-12], and a diversity of features and classifiers are used to construct ML frameworks for the prediction.

The commonly used ML classifiers to predict protein coding RNAs are support vector machines (SVM) such as in CPC, CNCI, and PLEK [13-18], then random forest (RF) was used in COME and FEElnc [19-20], logistic regression (LR) was used for CPAT [21] and finally deep neural networks (DNN) [22-27]. In recent years, DNNs have been outperforming other classifiers and were used to develop coding prediction models, such as in mRNN [22], RNAsamba [23], and LncADeep [27]. These DNN models have shown superior results to other machine learning methods [28-29].

Different features used to predict protein coding RNAs could include Open Reading Frame (ORF) characteristics such as ORF integrity, ORF coverage, ORF length, or sequence internal composition features such as Fickett score, Hexamer score and, physical and chemical characteristics [13-15], [21] etc. The currently used features appear to be groupable into different categories such as sequence features, homology features, and physicochemical features. However, as far as we know, no work so far has used Omics features such as gene expression quantification using RNA-seq (RNA-sequencing) or ribosome-bound transcripts signals known as Ribo-seq. RNA-seq method from second generation of sequencing is used to

*Authors contributed equally.

quantify the RNA abundance in a sample. It analyzes the transcriptome to reveal whether genes are active and quantify their expression. Ribosome profiling (Ribo-seq) is a cellular snapshot of protein. Compared with conventional RNA-Seq studies, ribosome profiling was generated to evaluate the mRNA present during an active translation.

In this work, we investigate the coding potential of RNAs from genes of the GENCODE [30] database. We explore novel features, called here Omics features representing the transcription and translation signals of genes to enhance the discrimination of RNA sequences. We also propose a classification method (OmicsRNADNN) using deep learning as a meta learner (Figure 1), which constructs a predictive model by combining scores from various feature categories.

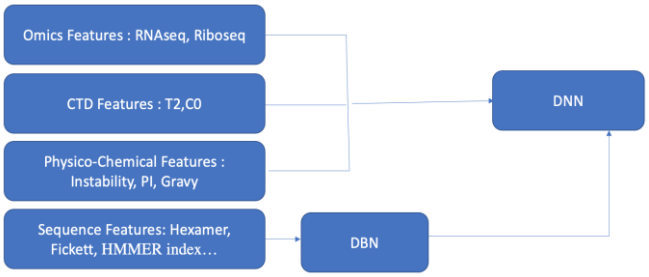

Fig. 1. Proposed OmicsRNADNN Model

II. Materials and Methods

A. Dataset of Transcripts

We mainly gathered two classes of transcripts from GENCODE [30] to run the experiment: protein-coding RNAs (pcRNA) and long non coding RNAs (lncRNAs). GENCODE is the most comprehensive database describing the human genome annotation that uses computational and manual annotation in addition to experimental validation. We downloaded the raw data of lncRNAs and pcRNAs from Release 41 of GENCODE. For each gene, we use the longest transcript which ensures non-redundant sequence. The human pcRNAs are considered as positive samples, while lncRNAs are the negative samples. Approximately two-thirds of the transcripts are used for the training set while the rest was assigned for the validation set. Moreover, only the RNA sequences with the best ORF starting with ATG codon and ending with conventional stop codons are selected. We obtained 10,010 pcRNAs and 14,275 lncRNAs transcripts.

B. RNAseq and Riboseq Data

Data were obtained from the PsychENCODE repository [31]. FastQC [32] was used to perform quality control analysis. Samples adapters were trimmed using Cutadapt [33] and selected samples were aligned using the STAR tool [34] to the same reference mentioned above. We had 288 samples from RNA-seq and 133 from Ribo-seq. The samples were from normal, bipolar disorder, and schizophrenia human brains. Quantifying the reads was performed using the feature counts [35] then normalized counts were obtained using the DESeq2 package [36]. Next the RNA-seq and Ribo-seq values were rescaled to the interval [0,1] to obtain final scores.

III. RNAs Classification

The pipeline of the classification method (section B) is explained in the following. First, the training set transcripts are represented by 8 scores as described in section A. Then, we use the randomly selected pcRNAs samples with the lncRNAs to construct the deep learning classification model.

A. Feature Preparation

Different features have been used to classify RNAs into coding or lncRNAs. We derive two scores from the RNA-seq and Ribo-seq data. We derive from CPPRED [37] five scores, T2 and C0 as CTD (Composition, Transition, and Distribution of nucleotides) features, and instability, PI and Gravy as physicochemical features. Finally, we derive from LncADeep [27] a sequence score (SeqScore) based on sequence and homology features including Fickett, hexamer, ORF length etc.

The RNA-seq score is used to ensure the transcription of the genes, while Ribo-seq score helps discriminate between what is being translated and what is not. The physico-chemical features encompass the instability score which is an assessment of the stability of a predicted peptide while the PI score represents its isoelectric point (PI). The Gravy feature of a predicted peptide is defined as the grand average of its hydropathicity. Finally, the global descriptor (CTD) features: T2, C0 describe the transition and composition calculations between nucleotides as described in [37].

The Fickett feature and calculation are well described in [38-39] and mainly helps discriminate the coding state of RNAs based on the nucleotide composition and codon usage bias. The most discriminating feature is the Hexamer score because it is based on the relationship between consecutive amino acids in peptide sequence [21] [38] [39]. ORF length is also commonly used in addition to ORF coverage, which is calculated as the longest ORF divided by the gene transcript length. ORF coverage feature is considered complementary to the ORF length and has shown high classification power [17-21].

B. Model Design

To classify the genes, the model was designed as a fully connected feedforward DNN. A DNN has multiple hidden layers, an input layer and an output layer. Due to their excellent classification performance, DNNs have been widely used in different bioinformatics applications [40], such as protein prediction and long non-coding RNAs identification. The model proposed here is built to have one input node per predictor, totaling eight input nodes and one output node for each gene. The input data is fed to two hidden layers. The hidden layer functions and parameters include the Rectifier linear unit (RELU) activation function on all nodes and across all layers. The Adam algorithm [41] is also used to minimize the mean squared error with a learning rate of 0.0005. The output layer has one node and uses the sigmoid activation function for binary classification and outputs either protein coding or long noncoding. For implementation, we use Keras and scikit.learn packages from Python 3.10.

IV. Results and Discussion

A. Feature Evaluation

- **Removing Redundant Features**

Data can include redundancies represented by features that are highly correlated with each other. Therefore, eliminating highly correlated attributes could improve the performance. Generally, attributes with an absolute correlation of 0.75 or higher are highly recommended to be removed. Our data features do not have any correlation beyond the 0.75 cutoff (Figure 2-a), all features were used**.**

- **Ranking Features by Importance**

The importance estimation of features is performed to understand the contribution magnitude of each feature. In our case, we estimated the importance of data by building a model constructed based on a Learning Vector Quantization (LVQ). Figure 2-b shows that the SeqScore, Riboseq and RNAseq are the most important attributes in the dataset, while Gravy and T2 attributes are the least important.

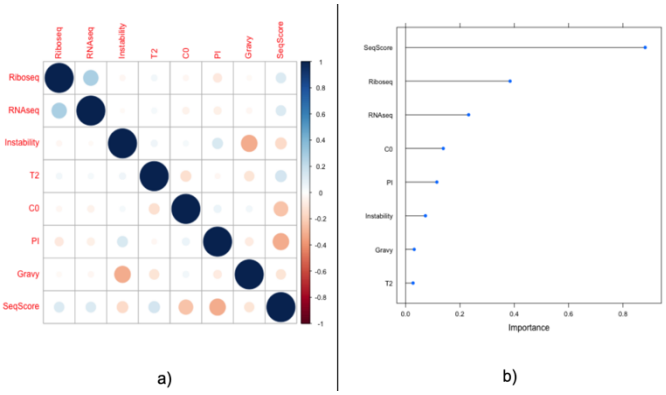

Fig. 2. Feature Evaluation a) Correlation matrix, b) Importance ranking.

B. Model Evaluation

The assessment of models used for the prediction is ensured with the 5-fold cross-validation (5-CV) process. The choice of 5-CV was based on our training size and to avoid longer training time. In each fold of 5-CV, the dataset is divided into 5 roughly equal sets. The 20% of the dataset is used for the testing, while 80% is used for the training/validation of the DNN. The models used for prediction are first trained on the training set and optimized using a validation set then later the accuracy is assessed using the testing set. The performance evaluation of the prediction models uses different metrics such as Recall, also called Sensitivity. Precision, Accuracy and the AUC (the area under the ROC curve) are also used to assess the model in addition to the Harmonic Mean (HM) and Mathew Correlation Coefficient (MCC). The metrics formulas are as follows:

$$Recall = \frac{TP}{TP + FN}$$

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

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

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

$$HM = \frac{2 \times SN \times SP}{SN + SP}$$

where TP is the true positive, FP is false positive; TN is true negative and FN is false negative; SN is sensitivity and SP is specificity.

C. Hyperparameters Discussion

We used Keras Tuner [42] for hyperparameter tuning. The process of selecting the right set of parameters for our OmicsRNADNN model is called hyperparameter tuning or hypertuning. Hyperparameter variables affect the training process and can directly impact the performance of the model. Our hyperparameter tuning runs determine that the best performance is accomplished using two hidden layers with 64 and 32 nodes, no dropout and a learning rate equal to 0.0005. The produced model shows acceptable fitting based on the training accuracy/loss and validation accuracy/loss curves (Figure 3).

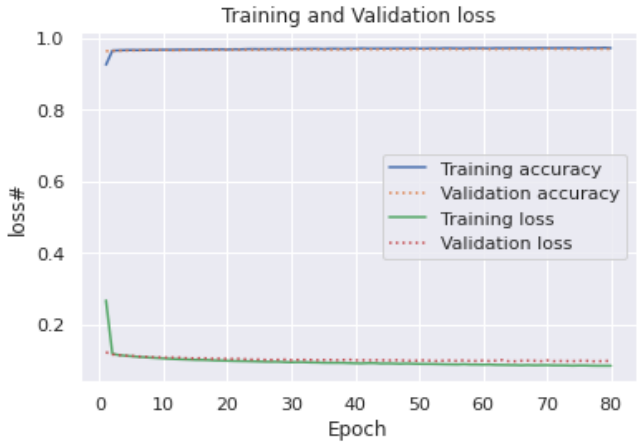

Fig. 3. Assessing the accuracy and loss of the training/validation of the OmicRNADNN model.

D. Classifier Discussion

We considered different classifiers including SVM, RF, LR, DNN, which are among the most popular algorithms that have been adopted by most of the RNAs classification. We also include other algorithms such as the gradient boosting (GB) from ensemble learning, decision trees (DT), naïve Bayes (NB), and the k-nearest neighbor (KNN). The performances of the classification model using different classifiers are shown in Figure 4. Among all classifiers, the 5-fold cross-validation shows that OmicRNADNN performs the best on the dataset (Table I).

TABLE I. PERFORMANCE OF DIFFERENT CLASSIFIER

	Accuracy	*Precision*	*Recall*	*AUC*	*HM*	*MCC*
GB	0.973	0.974	0.979	0.991	0.972	0.924
LR	0.967	0.969	0.975	0.988	0.967	0.910
SVM	0.966	0.966	0.966	0.988	0.966	0.966
DT	0.954	0.962	0.960	0.959	0.953	0.871
RF	0.973	0.976	0.978	0.991	0.973	0.924
NB	0.964	0.959	0.979	0.980	0.965	0.907
KNN	0.965	0.964	0.976	0.979	0.965	0.907
OmicRNADNN	0.974	0.976	0.980	0.994	0.974	0.929

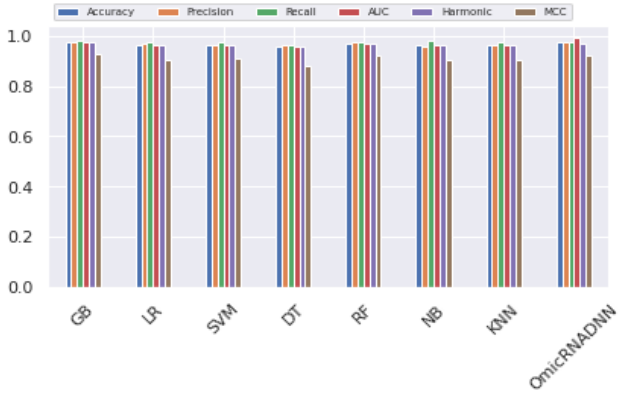

Fig. 4. Metric evaluation of different classifiers. Gradient Boosting (GB), Linear Regression (LR), Support Vector Machine (SVM), Decision Trees (DT), Random Forest (RF), Naïve Bayes (NB), K-Nearest Neighbors (KNN).

E. Comparison with State-of-the-Art Methods

Since the best performers in pcRNAs and lncRNAs classification are deep learning tools [28-29] we run the data using the top three deep learning tools as mentioned by the literature [34]: mRNN [22], RNAsamba [23], and LncAdeep [27]. mRNN uses a Recurrent Neural Network (RNN) which is based on a gated recurrent unit architecture and a one-hot encoding scheme for the input sequences. RNAsamba is built on a convolutional neural network model based on an IGLOO architecture that receives sequences as input divided into two branches, one for the complete sequence and the other for the longest ORF. LncADeep uses a deep belief neural network (DBN) and three restricted Boltzmann machines stacked between the input and output layers. LncADeep integrates sequence features and homology features. We run the three tools on our testing set and produce the different metrics to compare them to our model which shows better performance (Table II).

TABLE II. COMPARISON WITH STATE-OF-THE-ART METHODS

	mRNN	**RNAsamba**	**LncADeep**	**OmicRNADNN**
Recall	0.911	0.963	0.966	0.982
Precision	0.960	0.951	0.978	0.982
Accuracy	0.920	0.949	0.967	0.979
AUC	0.971	0.986	0.987	0.994

The false positive rate (1-SP) compared against the true positive rate (SN) for various cutoff levels allows us to additionally plot the ROC curve and calculate the AUC measure. The AUC metric, which is frequently employed as a key statistic, assesses the effectiveness of prediction models regardless of any threshold.

V. CONCLUSIONS

Classification of genes into protein coding and long non-coding is a very crucial issue. This work outlines a novel classification method based on deep neural networks. One of the highlights of this new method is that it adopts omics-derived features based on RNAseq and Riboseq scores, in addition to the commonly used sequence-specified, homology, physico-chemical and CTD features. Another highlight of this work is that it utilizes and compares different machine learning and deep learning schemes as meta learners to combine the different characteristics defining the features. Experiments indicate that deep learning methods can differentiate positive and negative instances better than other classification algorithms.

Compared to other three state-of-art deep learning models, the framework proposed here produces better metrics. We had come to this result thanks to our proposal of a newer coding potential classification framework based on using deep neural networks and novel features from RNAseq and Riboseq data. Based on our literature review, it is tempting to note that this is the first method that uses RNAseq and Riboseq as predictors to classify RNAs in addition to other features.

In conclusion, this work is providing a useful method and features to improve RNAs prediction, complementary to experiments and traditional techniques.

REFERENCES

[1] J. A. Reuter, D. V. Spacek, and M. P. Snyder, “High-throughput sequencing technologies,” Mol. Cell, vol. 58, no. 4, pp. 586–597, May 21, 2015.

[2] Chen, Zhiao and He, Xianghuo. "Application of third-generation sequencing in cancer research" *Medical Review*, vol. 1, no. 2, 2021, pp. 150-171.

[3] Y. Luo, X. Liao, F.-X. Wu, and J. Wang, “Computational approaches for transcriptome assembly based on sequencing technologies,” Curr. Bioinf., vol. 15, no. 1, pp. 2–16, 2020.

[4] X. Shi, M. Sun, H. Liu, Y. Yao, and Y. Song, “Long non-coding RNAs: A new frontier in the study of human diseases,” Cancer Lett., vol. 339, no. 2, pp. 159–166, Oct. 10, 2013.

[5] F. Huang, X. Yue, Z. Xiong, Z. Yu, S. Liu, and W. Zhang, “Tensor decomposition with relational constraints for predicting multiple types of microRNA-disease associations,” Brief. Bioinf., to be published, doi: 10.1093/bib/bbaa140.

[6] Y. Deng, X. Xu, Y. Qiu, J. Xia, W. Zhang, and S. Liu, “A multimodal deep learning framework for predicting drug-drug interaction events,” Bioinformatics, to be published, doi: 10.1093/ bioinformatics/btaa501.

[7] Q. Zou, “Latest machine learning techniques for biomedicine and bioinformatics,” Curr. Bioinf., vol. 14, no. 3, pp. 176–177, 2019.

[8] K. Patil and U. Chouhan, “Relevance of machine learning techniques and various protein features in protein fold classification: A review,” Curr. Bioinf., vol. 14, no. 8, pp. 688–697, 2019.

[9] Q. Zou, K. Qu, Y. Luo, D. Yin, Y. Ju, and H. Tang, “Predicting diabetes mellitus with machine learning techniques,” Front. Genet., vol. 9, Nov. 6, 2018, Art. no. 515.

[10] Abbas Q, Raza SM, Biyabani AA, Jaffar MA. A Review of Computational Methods for Finding Non-Coding RNA Genes. Genes (Basel). 2016 Dec 3;7(12):113.

[11] T. D. C. Negri, W. A. L. Alves, P. H. Bugatti, P. T. M. Saito, D. S. Domingues, and A. R. Paschoal, "Pattern recognition analysis on long noncoding RNAs: A tool for prediction in plants," Brief. Bioinf., vol. 20, no. 2, pp. 682–689, Mar. 25, 2019.

[12] E. A. Ito, I. Katahira, F. Vicente, L. F. P. Pereira, and F. M. Lopes, "BASiNET-BiologicAl sequences NETwork: A case study on coding and non-coding RNAs identification," Nucleic Acids Res., vol. 46, no. 16, Sep. 19, 2018, Art. no. e96.

[13] L. Kong et al., "CPC: Assess the protein-coding potential of transcripts using sequence features and support vector machine," Nucleic Acids Res, vol. 35, no. Web Server issue, pp. W345–W349, Jul. 2007.

[14] L. Sun et al., "Utilizing sequence intrinsic composition to classify protein-coding and long non-coding transcripts," Nucleic Acids Res, vol. 41, no. 17, Sep. 2013, Art. no. e166.

[15] A. M. Li, J. Y. Zhang, and Z. Y. Zhou, "PLEK: A tool for predicting long non-coding RNAs and messenger RNAs based on an improved k-mer scheme," BMC Bioinf., vol. 15, Sep. 19, 2014, Art. no. 311.

[16] L. Sun, H. Liu, L. Zhang, and J. Meng, "lncRScan-SVM: A tool for predicting long non-coding RNAs using support vector machine," PLoS One, vol. 10, no. 10, Oct. 5, 2015, Art. no. e0139654.

[17] Y. J. Kang et al., "CPC2: A fast and accurate coding potential calculator based on sequence intrinsic features," Nucleic Acids Res., vol. 45, no. W1, pp. W12–W16, Jul. 3, 2017.

[18] H. W. Schneider, T. Raiol, M. M. Brigido, M. Walter, and P. F. Stadler, "A support vector machine based method to distinguish long noncoding RNAs from protein coding transcripts," BMC Genomics, vol. 18, no. 1, Oct. 18, 2017, Art. no. 804.

[19] L. Hu, Z. Xu, B. Hu, and Z. J. Lu, "COME: A robust coding potential calculation tool for lncRNA identification and characterization based on multiple features," Nucleic Acids Res., vol. 45, no. 1, Jan. 9, 2017, Art. no. e2.

[20] V. Wucher et al., "FEELnc: A tool for long non-coding RNA annotation and its application to the dog transcriptome," Nucleic Acids Res., vol. 45, no. 8, May 5, 2017, Art. no. e57.

[21] L. Wang, H. J. Park, S. Dasari, S. Q. Wang, J. P. Kocher, and W. Li, "CPAT: Coding-potential assessment tool using an alignment-free logistic regression model," Nucleic Acids Res., vol. 41, no. 6, Apr. 2013, Art. no. e74.

[22] Hill ST, Kuintzle R, Teegarden A, Merrill E 3rd, Danaee P, Hendrix DA. A deep recurrent neural network discovers complex biological rules to decipher RNA protein-coding potential. Nucleic Acids Res. 2018 Sep 19, vol 46, no. 16, pp. 8105-8113.

[23] Antonio P Camargo, Vsevolod Sourkov, Gonçalo A G Pereira, Marcelo F Carazzolle, RNAsamba: neural network-based assessment of the protein-coding potential of RNA sequences, *NAR Genomics and Bioinformatics*, Volume 2, Issue 1, March 2020

[24] Y. Z. Xu et al., "LncPred-IEL: A long non-coding RNA prediction method using iterative ensemble learning," in Proc. IEEE Int. Conf. Bioinf. Biomed., 2019, pp. 555–562

[25] X. N. Fan and S. W. Zhang, "lncRNA-MFDL: Identification of human long non-coding RNAs by fusing multiple features and using deep learning," Mol. Biosyst., vol. 11, no. 3, pp. 892–897, 2015.

[26] J. Baek, B. Lee, S. Kwon, and S. Yoon, "LncRNAnet: Long noncoding RNA identification using deep learning," Bioinformatics, vol. 34, no. 22, pp. 3889–3897, Nov. 15, 2018.

[27] C. Yang et al., "LncADeep: An ab initio lncRNA identification and functional annotation tool based on deep learning," Bioinformatics, vol. 34, no. 22, pp. 3825–3834, Nov. 15, 2018.

[28] Tea Ammunét, Ning Wang, Sofia Khan, Laura L Elo, Deep learning tools are top performers in long non-coding RNA prediction, *Briefings in* Functional *Genomics*, Volume 21, Issue 3, May 2022, Pages 230–241

[29] N. Amin, A. McGrath, and Y. P. P. Chen, "Evaluation of deep learning in non-coding RNA classification," Nat. Mach. Intell., vol. 1, pp. 246–256, May, 2019.

[30] Frankish A, et al., GENCODE 2021. Nucleic Acids Res. 2021 Jan 8;49(D1): D916-D923.

[31] https://psychencode.synapse.org/

[32] Babraham bioinformatics - FastQC a quality control tool for high throughput sequence data. http://www.bioinformatics.babraham.ac.uk/projects/fastqc.

[33] Martin, M.: Cutadapt removes adapter sequences from high-throughput sequencing reads. EMBnet J., 17(1):10, May 2011.

[34] Dobin, A., Davis, C. A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M., and Gingeras, T. R.: STAR: ultrafast universal RNA-seq aligner. Bioinformatics, 29(1):15–21, January 2013.

[35] Liao, Y., Smyth, G. K., and Shi, W.: featurecounts: an efficient general purpose program for assigning sequence reads to genomic features. Bioinformatics, 30(7):923–930, April 2014.

[36] Love, M.I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol* 15, 550 (2014).

[37] Xiaoxue Tong, Shiyong Liu, CPPred: coding potential prediction based on the global description of RNA sequence, *Nucleic Acids Research*, Volume 47, Issue 8, 07 May 2019, Page e43

[38] Fickett, J.W. (1982) Recognition of protein coding regions in DNA sequences. Nucleic Acids Res., 10, 5303–5318.

[39] Fickett, J.W. and Tung, C.-S. (1992) Assessment of protein coding measures. Nucleic Acids Res., 20, 6441–6450.

[40] Seonwoo Min, Byunghan Lee, Sungroh Yoon, Deep learning in bioinformatics, *Briefings in Bioinformatics*, Volume 18, Issue 5, September 2017, Pages 851–869

[41] Kingma DP, Ba J, "Adam: a method for stochastic optimization". In: International conference on learning representations. (2015)

[42] O'Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L., & others. (2019). KerasTuner.