Briefings in Bioinformatics, 00(00), 2020, 1–12

doi: 10.1093/bib/bbaa039
Problem Solving Protocol

DeepCPP: a deep neural network based on nucleotide bias information and minimum distribution similarity feature selection for RNA coding potential prediction

Yu Zhang, Cangzhi Jia, Melissa Jane Fullwood and Chee Keong Kwoh

Corresponding authors: Chee Keong Kwoh, School of Computer Science and Engineering, Nanyang Techonological University, 50 Nanyang Avenue, Singapore. Tel.: +65 6790 6057; E-mail: asckkwoh@ntu.edu.sg; Fax: +65 6792 6559; Cangzhi Jia, School of Mathematical Sciences, Dalian University of Technology, No.2 Linggong Road, Dalian, China. Tel.: +86 41184727242; E-mail: cangzhijia@dlmu.edu.cn

Abstract

The development of deep sequencing technologies has led to the discovery of novel transcripts. Many *in silico* methods have been developed to assess the coding potential of these transcripts to further investigate their functions. Existing methods perform well on distinguishing majority long noncoding RNAs (lncRNAs) and coding RNAs (mRNAs) but poorly on RNAs with small open reading frames (sORFs). Here, we present DeepCPP (deep neural network for coding potential prediction), a deep learning method for RNA coding potential prediction. Extensive evaluations on four previous datasets and six new datasets constructed in different species show that DeepCPP outperforms other state-of-the-art methods, especially on sORF type data, which overcomes the bottleneck of sORF mRNA identification by improving more than 4.31, 37.24 and 5.89% on its accuracy for newly discovered human, vertebrate and insect data, respectively. Additionally, we also revealed that discontinuous k-mer, and our newly proposed nucleotide bias and minimal distribution similarity feature selection method play crucial roles in this classification problem. Taken together, DeepCPP is an effective method for RNA coding potential prediction.

Key words: long noncoding RNAs; RNA coding potential; deep learning; sORF RNA

Introduction

Long noncoding RNAs (LncRNAs), defined as nonprotein coding transcripts with lengths larger than 200 nucleotides, making up the majority of the transcriptome [1]. They were previously thought to be 'dark matter,' but now are regarded as treasures as their functions are now being explored following the

Yu Zhang received the BEng Degree from Shandong University, China, and the MSc degree (distinction degree) from Imperial College London, UK, in 2017 and in 2018, respectively. She is currently a PhD candidate in Nanyang Technological University, Singapore. Her research interests include bioinformatics and deep learning.
Cangzhi Jia received the PhD Degree from the School of Mathematical Sciences, Dalian University of Technology, in 2007. She is an Associate Professor with the School of Science, Dalian Maritime University, China. Her research interests include mathematical modeling in bioinformatics and machine learning.
Melissa Jane Fullwood received the B.Sc. Degree (Hons) from Stanford University, USA, and the Ph.D. Degree from National University of Singapore, Singapore, in 2005 and in 2009, respectively. She is Principal Investigator in Cancer Science Institute of Singapore, National University of Singapore, Assistant Professor in School of Biological Sciences, Nanyang Technological University, and adjunct principal investigator in Institute of Molecular and Cell Biology, A*STAR Singapore. Her research interests include using ChIP sequencing, RNA sequencing and ChIA-PET on a gastric cancer cell model, as well as other cancer cell models, to elucidate the detailed epigenomic profiles, allowing for new insights into possible cancer-associated biomarkers and cancer therapies, and develop and refine new genomic technologies to understand chromatin and transcription.
Chee Keong Kwoh received the Bachelor's Degree in electrical engineering (first class) and the Master's Degree in industrial system engineering from the National University of Singapore, Singapore, in 1987 and 1991, respectively. He received the Ph.D. Degree from Imperial College of Science, Technology and Medicine, University of London, in 1995. He has been with the School of Computer Engineering, Nanyang Technological University (NTU), since 1993.He is the Program Director of the MSc in Bioinformatics program at NTU. His research interests include data mining, soft computing and graph-based inference; applications areas include bioinformatics and biomedical engineering.
Submitted: 20 December 2019; **Received (in revised form):** 24 February 2020

development of deep sequencing technologies and the completion of multiple genome projects [2]. lncRNAs play many roles in biological processes, including chromatin modification, chromatin remodeling, transcriptional regulation and gene expression [3–6]. Additionally, they have been implicated in diseases such as cancer and diabetes [7, 8].

The prerequisite to explore new RNAs is to identify them correctly. Some sequences were initially considered to be lncRNA but have proved to have coding potential, especially for those with small open reading frames (sORF) [9, 10]. sORFs are deemed noncoding on the basis of: (i) the short length, (ii) the lack of experimental corroboration of their functions, and (iii) annotation problem caused by their huge amount [11]. The wrong initial assumptions for sORF lead to the difficulty in distinguishing messenger RNAs (mRNAs) with sORFs from lncRNAs. Therefore, accurate and rapid methods are urgently needed to predict the coding potential for newly found genes, especially for those with sORFs, to help in the analyses and understanding of new transcripts and their functions.

A variety of *in silico* methods have been developed to distinguish lncRNAs and mRNAs, they can be divided into two categories: alignment-based methods and alignment-free methods. Alignment-based methods mainly rely on the already known databases. Based on the support vector machine (SVM), Kong *et al.* [12] proposed the coding potential calculator (CPC), which used six features with three of them calculated from an existing protein sequence database. Lin *et al.* [13] developed PhyloCSF, which relies on the likelihood ratio from the comparison between two well-established phylogenetic models representing coding and noncoding genes separately. However, the heavy dependence on existing databases is an obvious drawback for these alignment-based methods because the wide variations between newly discovered data and previous data will lead to poor prediction results.

By contrast, alignment-free methods only depend on the intrinsic information of the given sequences, which make them more flexible than alignment-based methods. For these kinds of programs, the longest open reading frame and k-mer are the most commonly used features, as a long ORF region in the transcript is more likely to be translated to protein and k-mers can reflect some internal features of the sequence. Wang *et al.* [14] introduced CPAT, a logistic regression-based model using ORF related features as well as Fickett and hexamer feature, which is reported to be more efficient than those alignment-based methods. And methods that use k-mer-related features, such as PLEK, CNCI, Hugo's SVM method and DeepLNC, also are reported to result in good performances on distinguishing lncRNAs from mRNAs [15–18]. In addition, unlike the works stated above which adopted small k-mer (generally $k \leq 6$), FEELnc, a random forest-related method, took a large k value, i.e. $k = 12$, as the authors believe that a long k-mer can provide more information about lncRNA-specific repeats or spatial information [19]. Furthermore, one-hot vector, secondary structure, and network learned features also have been applied in the classification between lncRNA and mRNA [20, 24, 25]. Besides, in works like mRNN [20] and CPPred [21], the authors also evaluated different programs on the RNA data with sORFs, although the performances of their proposed methods had been improved, the results were still not desirable, for example, the accuracies of mRNAs with sORFs predicted by CPPred are only 63.34% for its human and 46.81% for its mouse data. Therefore, the coding potential prediction for sORF type data still has room for improvement.

Here, we aimed to develop an effective model to predict RNA coding potential, especially for the RNAs with sORFs. We first proposed a new feature representation method, named nucleotide bias, which reflects the information around start codon and was proved to be useful in sORF type RNA distinction. Additionally, we showed that discontinuous k-mers, i.e. g-bigap (with gap between dinucleotides), contribute more to this classification problem than that of continuous k-mer, which have much better ROC and PRC curves. Next, we presented a novel feature selection method, named minimum distribution similarity (mDS). The comparisons between mDS and other commonly used methods demonstrate its effectiveness. Finally, with the strategies mentioned above and other useful feature representation methods like hexamer score, ORF coverage and ORF length, we built DeepCPP, a deep neural network for RNA coding potential prediction. We evaluated DeepCPP and compared it with seven recently developed state-of-the-art methods, including Hugo's SVM method [17], mRNN [20], lncRNAnet [22], LncADeep [23], LncFinder [24], RNAsamba [25] and CPPred [21], on data from different species, normal and sORF type. DeepCPP showed 98.20, 95.78 and 96.16% predictive power on normal data for our human, vertebrate and insect data, respectively, and at least 1.08, 17.09 and 3.14% improvement on accuracies of sORF type data compared to other methods, which highlights its value in RNA coding potential prediction.

Material and methods

Data description

We aim to construct models of coding potential prediction for different species, including human, vertebrate and insect. We adopted the same human training datasets and four test datasets (human testing, human-sORF-testing, mouse-testing, and mouse-sORF-testing) from CPPred, and constructed new human test datasets (normal and sORF type), training and test datasets (normal and sORF type) for vertebrate and insect from NCBI RefSeq and Ensembl databases [21, 26–31]. The detailed data collection and processing are implemented in Supplementary Section S1, the construction process for all new datasets are illustrated in Supplementary Figure S1, and the name, annotation and size for all datasets used in this work are listed in Table 1.

Sequence encoding

To represent the raw sequence into vectors, we mined sequence information from different perspectives, such as ORF, sequence composition and codon preference around start codon, which are represented by approaches like maximum ORF length and ORF coverage [14], mean hexamer score [14, 21, 23], Fickett score [32], k-mer [18] and g-gap and nucleotide bias. The descriptions for all above features except Nucleotide bias are implemented in Supplementary Section S2.

Nucleotide bias is our newly proposed feature representation method. Although the start codon indicates the translation initiation, translation may not necessarily occur from it. Previous studies show that nucleotides around start codon have influences on the regulation of translation initiation to some degree [33–36]. Nakagawa evaluated nucleotide appearance around the start codon (nucleotide position: +1, +2 and +3) on 10 012 human genes using G-statistic [33], and the result showed that the first codon before (nucleotide position: −3, −2 and −1) and

Table 1. The name, annotation and data amount for all datasets used in this work

Dataset name	Annotation	Number of mRNAs	Number of ncRNAs
Human training dataset from CPPred	D_{train_H}	33 359	24 162
Human test dataset from CPPred	D_{test_H}	8557	8240
Human sORF RNA test dataset from CPPred	$D_{test\_H\_sORF}$	641	641
Mouse test dataset from CPPred	D_{test_M}	31 100	19 930
Mouse-sORF RNA test dataset from CPPred	$D_{test\_M\_sORF}$	846	1000
New human test dataset	$D_{test\_H\_new}$	3373	23 915
New human sORF RNA test dataset	$D_{test\_H\_newsORF}$	232	232
Vertebrate training dataset	D_{train_V}	31 706	18 635
Vertebrate test dataset	D_{test_V}	16 045	12 055
Vertebrate sORF RNA test dataset	$D_{test\_V\_sORF}$	588	588
Insect training dataset	D_{train_I}	25 181	12 465
Insect test dataset	D_{test_I}	12 734	5821
Insect sORF RNA test dataset	$D_{test\_I\_sORF}$	1529	1529

after (nucleotide position: +4, +5 and + 6) the initiation codon in coding sequence (CDS) has obvious nucleotide preference. For example, position −3 prefers A/G and position +5 prefers C. The strong bias of nucleotides around initiation codon are also supported by other studies [34–36]: certain nucleic acids at positions near start codon, e.g. position −3 to −1 and + 4 to +6, may contribute to enhance translation initiation. Besides, results in [33] reported that species in each taxonomic group of eukaryotes shares high similarities between patterns of nucleotides, and the G-values in position from −9 to +6 are relatively high, especially in position from −3 to +6.

Based on the above biology findings, we proposed a novel feature encoding approach to distinguish mRNAs and ncRNAs, named nucleotide bias. We use the nucleotide bias feature to make the overall measurement of nucleotide bias at position −3, −2, −1, 4, 5 and 6 between mRNAs and ncRNAs, as they are reported to have potential ability to enhance translation initiation [33–36]. To do this, firstly, we count the frequency of nucleotide x at position i around the start codon of maximum ORF in mRNA and ncRNA training datasets, respectively (we only consider the complete sequence from position −3 to +6), got p_{mRNA} and p_{ncRNA}, then for each sequence, the nucleotide bias is calculated as follows:

$$\text{nucleotide bias} = \sum_{i \in C} \log \frac{p_{\mathrm{mRNA}}(x_i)}{p_{\mathrm{ncRNA}}(x_i)}, x \in \{A, C, G, T\},$$
$$C = \{-3, -2, -1, 4, 5, 6\} \quad (1)$$

where C denotes the position according to the initial codon of maximum ORF, and $p(x_i)$ is the corresponding value of x_i in p_{mRNA} and p_{ncRNA}. In a word, the nucleotide bias feature reflects the overall information around the start codon.

mDS feature ranking

From the section above, we have five feature encoding schemes and total of 589 features. However, some features may contain noise information and some of them may even be detrimental to the model performance, therefore, feature selection is needed to increase accuracy (Acc) as well as reduce time complexity [37]. Here, we proposed a new feature selection method, which aims to sort features according to their potential contribution to separate the two classes, and we named it as mDS.

The motivation to propose mDS is that it is much easier to classify objects whose features distributions are apart from each other, and this can be roughly measured by relative entropy, also known as Kullback–Leibler Divergence [38], whose original formula is follows:

$$D(\boldsymbol{p} \| \boldsymbol{q}) = \sum_{x \in X} p(x) \log \left(\frac{p(x)}{q(x)} \right). \quad (2)$$

It represents the 'distance' between the probability mass functions p and q, and it also can be regarded as the measurement of information loss of q from p. Here, we regard x as the event representing the instances in one type of feature whose values are in a certain range, this range is denoted as a small bin, X is the event set, and p and q are the distributions of x for two classes separately. We choose to use small bins rather than using the exact value of feature, because when $q(x) = 0, p(x) \neq 0, p(x) \log\left(\frac{p(x)}{q(x)}\right) \to \infty$; when $p(x) = 0, q(x) \neq 0, p(x) \log\left(\frac{p(x)}{q(x)}\right) = 0$. If there are no overlapped values in one feature in two classes, $p(x) \log\left(\frac{p(x)}{q(x)}\right)$ would either tend to ∞ or 0, under this circumstance, $D(p\|q)$ would tend to ∞ or 0 too.

If a large number of features do not have overlapped values in different classes, all these features would be assigned the same weight and we cannot know which one is more important. Introducing small bins can solve this problem, however, if the bin number is too small, the differences for two classes cannot be captured accurately; if it is too large, the relative entropy value would tend to be the same as when taking exact value stated above. Based on such concerns, the value of small bin number here is determined via the validation on the corresponding test datasets by considering the feature subset size that realizing the maximum Acc, but the solution can be not unique with the histogram idea. Furthermore, since $D(\boldsymbol{p}\|\boldsymbol{q}) \neq D(\boldsymbol{q}\|\boldsymbol{p})$, we use $D = D(\boldsymbol{p}\|\boldsymbol{q}) + D(\boldsymbol{q}\|\boldsymbol{p})$ as the criteria to sort the features according to their importance.

Another issue is how to deal with $q(x) = 0$ in $D(\boldsymbol{p}\|\boldsymbol{q})$ in practical. Here, we use another small value, $q(x) = \frac{1}{\text{number of class II}}$ (q is the feature distribution of class II) in $D(\boldsymbol{p}\|\boldsymbol{q})$ to replace $q(x) = 0$, which can reflect the trend of D although may not so precise. When the number of total instances in a certain bin is large, the measure of distribution similarity when there is only one sample of class II in this bin is almost the same as that when there is no sample of class II, both of the two circumstances will lead to an extremely large value of D; when the number of total instances in a certain bin is small, $p(x)$ tends to be 0,

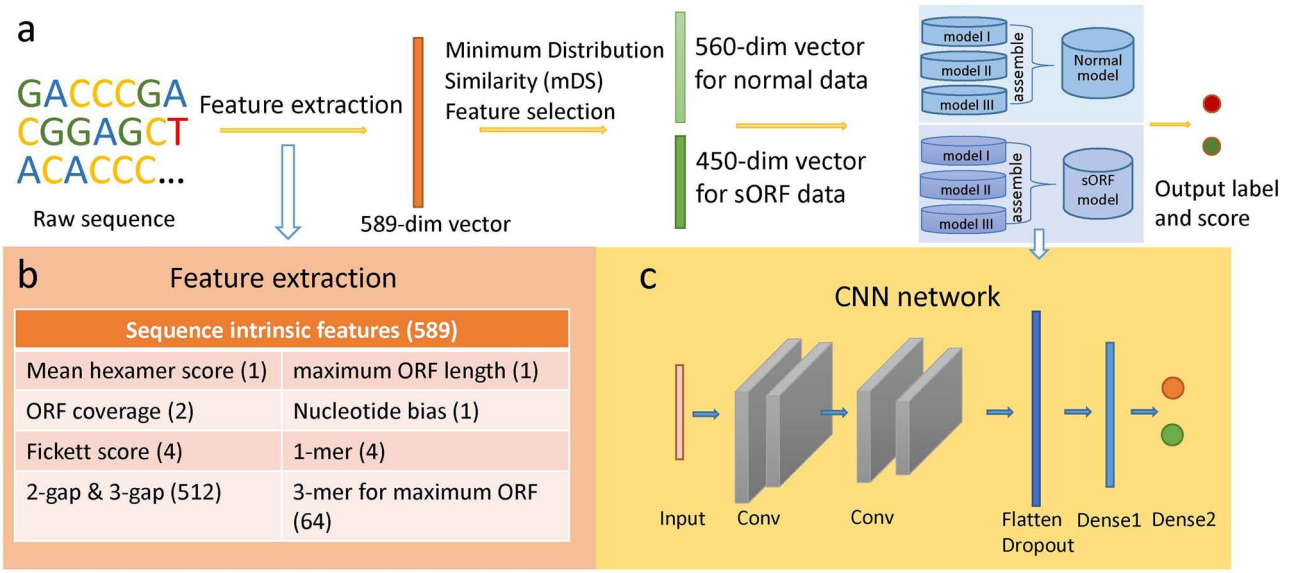

Figure 1. An overview of the mechanism of DeepCPP. (**A**) Overall flow chat. (**B**) Overview of feature extraction strategies and the corresponding feature amount. (**C**) Deep learning model construction process. Input vectors were passed through two CNN layers, and then were flatten and dropped 20% data to avoid overfitting.

when $q(x) = 0$, $p(x)\log\left(\frac{p(x)}{q(x)}\right)$ would be an extremely small value and the difference caused by replacing $q(x) = 0$ with another extremely small value almost can be omitted. Additionally, with the selection of the bin number, the circumstances of $q(x) = 0$ or $p(x) = 0$ can rare happen. The mDS method is summarized in Supplementary Algorithm S1. As mDS provides a feature ranking rather than the explicit best feature subset, the size of the feature subset should be determined via the cross validation. But in this work, because we will construct normal and sORF model with the same training data, the feature subset size will be decided via the validation on the corresponding test datasets.

Model construction

To address the effectiveness of the new features and new feature selection method, we use convolutional neural network (CNN) with two layers (adding further CNN layers would not contribute to the prediction performances on both normal and sORF type data here, as shown in Supplementary Table S1) to construct the model. CNN and deep learning were employed widely in bioinformatics recently [39, 40]. However, because of the differences between normal data and sORF type data, for human and two integrated species, we built two models for each, one for predicting normal data and the other for predicting sORF type data. The two models are trained with the same training data but different feature subsets. Besides, due to the fact that the parameters are initialized randomly in a deep network, we assemble three models into one by taking average scoring and use threshold 0.5 (the output score for each model is normalized from 0 to1) to make the decision in DeepCPP.

The overview of the mechanism for DeepCPP is illustrated in Figure 1. Firstly, features are extracted from the raw sequence, then mDS feature selection method is applied to the complete feature set to reduce the complexity and potential overfitting caused by the huge amounts of parameters in CNN, 560 and 450 features for normal and sORF data are put into the deep network for evaluation. Finally, the assemble results will be output and the coding potential prediction label will be given.

Evaluation criteria

We evaluated the performance of models with criteria of Acc, sensitivity (Sn), specificity (Sp), harmonic mean (Hm) of Sn and Sp, F-score and Matthew's correlation coefficient (Mcc). Additionally, McNemar test is adopted for the comparison between two models from a statistic respective, the descriptions of the evaluation criteria can be referred to Supplementary Section S3.

Results

Feature exploration

In this part, we want to investigate the effectiveness of the new features used in this classification problem: nucleotide bias and discontinuous k-mer. To evaluate the nucleotide bias feature representation method, firstly, we made data analysis to test whether there is an obvious preference of nucleotides in codons before and after start codon in our experimental data. As stated above, nucleotides at position −3, −2, −1, 4, 5 and 6 around start codon are said to play crucial roles in enhancing translation initiation and are adopted in our nucleotide bias feature [33–36]. Therefore, we counted the nucleotide occurrence frequency for these positions and illustrated the results as sequence logo in the human training dataset (Figure 2A), vertebrate training dataset (Figure 2B) and insect training dataset (Figure 2C). Results showed that the favored nucleotides at six positions are A, C, C, G, C and G for human mRNAs, A, C, C, G, C and G for vertebrate mRNAs, and A, A, A, G, C and G for insect mRNAs, respectively, which are consistent with the G-statistic values and the results in the work of Nakagawa *et al.* [33]. As for the nucleotide distributions in ncRNA, the tendencies shown are more uniform and do not follow the same rules as that of mRNA.

Next, we tested the performance using only nucleotide feature with a CNN network on the previous human normal test data (D_{test_H}) and previous human sORF test data ($D_{test\_H\_sORF}$). To see its effectiveness, we also tested the performance of maximum ORF length and coverage, mean hexamer score,

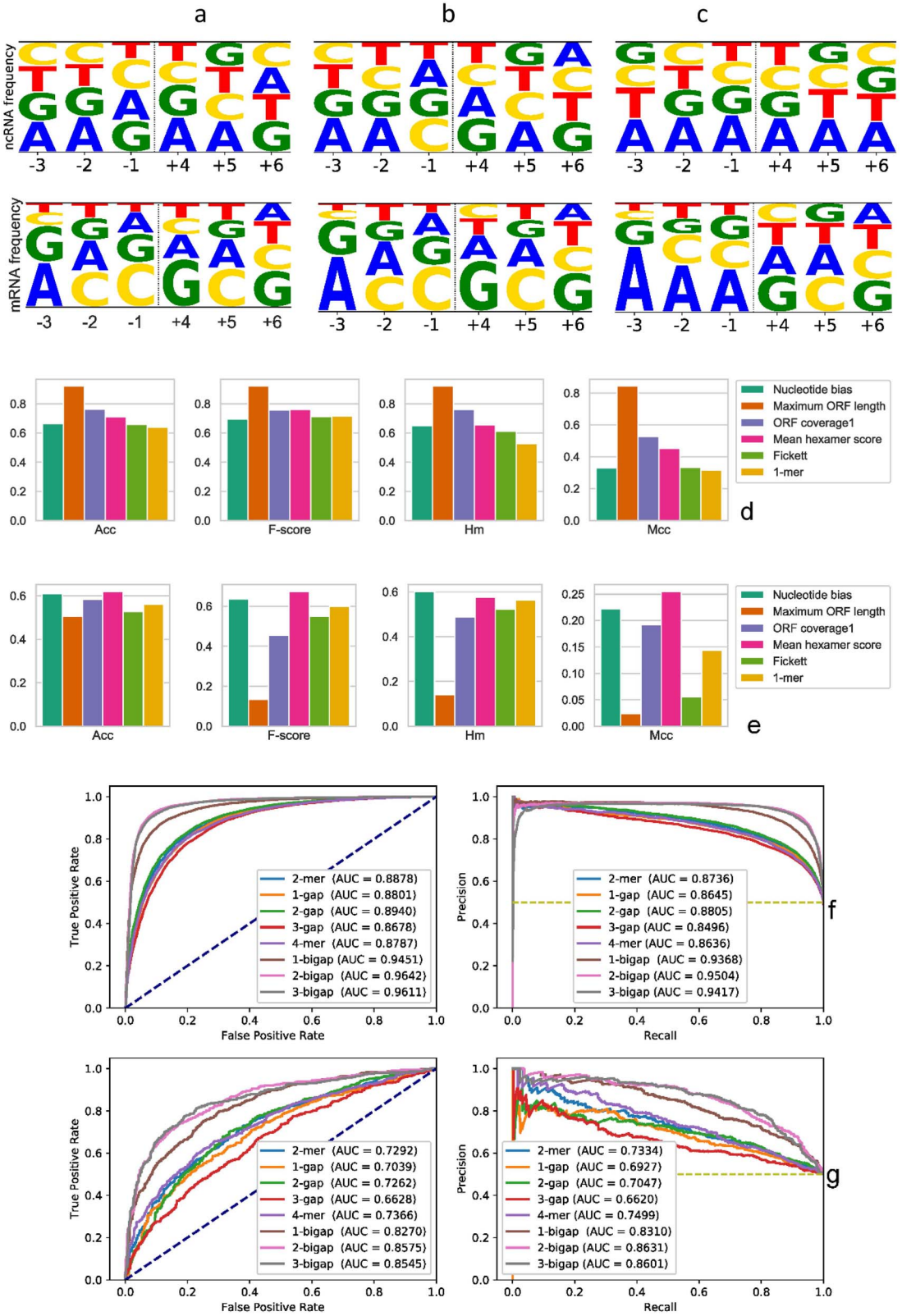

Figure 2. Illustration of the effectiveness of new feature representation methods. (**A**) Sequence logo representation in human training dataset. (**B**) Sequence logo representation in vertebrate training dataset. (**C**) Sequence logo representation in insect training dataset. Upper panel: lncRNA, lower panel: mRNA. (**D**) Comparison of Acc, F-score, Mcc and Hm among different features on CPPred human test D_{test_H}. (**E**) Comparison of Acc, F-score, Mcc and Hm among different features on CPPred human sORF test $D_{test\_H\_sORF}$. (**F**) ROC curve (left) and PRC curve (right) for D_{test_H}. (**G**) ROC curve (left) and PRC curve (right) for $D_{test\_H\_sORF}$.

Fickett score and 1-mer, which are said to be useful in finding coding potential [14, 21–23] on the same datasets to make the comparisons, the average Acc, F-score, Hm and Mcc with five times running are recorded in Figures 2D (D_{test_H}) and E ($D_{test\_H\_sORF}$). Our results show that the nucleotide bias feature does have some effects on distinguishing mRNA and ncRNA, but its impacts on normal test datasets (D_{test_H}) are not notable and the performances are not very desirable, as three features, maximum ORF length, ORF coverage and hexamer score achieve higher Acc value than it. However, nucleotide bias is an important feature for distinguishing sORF type RNA sequences. The top two largest values in all indicators in metric of $D_{test\_H\_sORF}$ belong to nucleotide bias and hexamer score, where the Acc of all other features are below 60%, especially the maximum ORF length feature, which is outstanding in evaluating normal test datasets but performs poorly on sORF datasets, with only 50.62% Acc and extremely small F-score, Mcc and Hm values. This is because that the ORF of mRNA is usually longer than that of noncoding RNA, as it would be difficult to find a long putative ORF in noncoding sequences in probability, hence this feature is rather useless when the ORF lengths of mRNAs are comparable with ncRNAs' lengths. Hexamer score and nucleotide bias are both relatively good features with at least 2.58% higher value in Acc compared to other features on distinguishing sORF mRNA and ncRNA, although the Acc, F-score and Mcc values of hexamer score are slightly larger than that of nucleotide bias, its Hm value is lower, this can be explained from the larger gap between its Sn and Sp values. The Sp and Sn for nucleotide bias are 53.82 and 68.17%, respectively, while for hexamer they are 45.40% and 78.63%. When the accuracies are nearly equal, the nucleotide bias feature leads to a more balanced result between mRNA and ncRNA compared to the hexamer feature.

Short k-mer is reported to be a popular feature in distinguishing mRNA and lncRNA [15–18], here, we use 1-mer and in-frame 3-mer to represent sequences. 1-mer, which contains the nucleotide composition information, together with the fickett score defined in this work, comprise the complete fickett information [32], which is said to be useful in RNA coding potential prediction [14, 21]. And we believe that the in-frame 3-mer can reflect protein-coding information as one amino acid is coded from three-nucleotide acids. Then, we want to further explore the circumstances of 2-mer and 4-mer, as well as the performances of discontinuous k-mer, i.e. g-gap and g-bigap, to see their impactions on mRNA and ncRNA classification, as we believe that the discontinuous k-mer may reflect more sequence information. The ROC and PRC curves for k-mer, g-gap and g-bigap feature representation on D_{test_H} and $D_{test\_H\_sORF}$ were plotted in Figure 2F and G.

The g-bigap feature sets perform superior to other feature sets on both test datasets with much higher AUC values. Particularly, when compared with continuous k-mer, the advantages for 2-bigap and 3-bigap feature sets are extremely obvious, e.g. over 7.64 and 7.68% improvement on AUC of ROC and PRC for D_{test_H}, and over 12.09 and 11.32% improvement on AUC of ROC and PRC for $D_{test\_H\_sORF}$, respectively. The findings here indicate that g-bigap can be an important feature of mRNA and ncRNA classification. However, previous works only addressed the importance of continuous k-mer [15–19] while ignored the impacts of discontinuous k-mer on this classification problem.

Overall, two new kinds of features, nucleotide bias and discontinuous k-mer, both contribute to the coding and noncoding RNA classification especially for RNA data with sORF.

Feature selection

Our whole feature set contains 589 features (hexamer score(1) + ORF coverage(2) + max ORF length(1) + nucleotide bias(1) + Fickett(4) + 1-mer(4) + 2-bigap(256) + 3-bigap(256) + in-frame 3-mer(64)). However, using all features may not only cause the classifier training process to be computationally intensive and slow, but also may lead to the overfitting problem when the parameters in deep neural network are in a huge amount. Therefore, it is necessary to conduct feature selection to reduce the dimensionality of feature vectors by eliminating features that would not contribute too much to the prediction.

We applied mDS on human training data and $D_{test\_H\_sORF}$ to sort features for normal model and sORF model separately, we use $D_{test\_H\_sORF}$ because we want to extract the features that are favored by sORF type data. The bin number is determined via the validation on D_{test_H} and $D_{test\_H\_sORF}$, the selections of bin number and the feature subset size that achieving the maximum Acc are recorded in Supplementary Table S2. We chose bin number as 1000 and 100 for normal and sORF model separately which achieving the best Acc with the least feature subset size, but other choices may also lead to the desired results and the solution is not unique due to the high feature subset coincidence within a range of bin number choices (Supplementary Table S3).

The optimal feature subset sizes for normal and sORF type data ranked by mDS are selected as 560 and 450 separately, where the highest average Acc is achieved as 97.07 and 83.54%. We noted that the feature subset size of normal data is larger than that of sORF type data, this can be explained by the large data amount of normal sequences and the consistency between training and test data. For sORF type data, due to the relatively large difference between training and test data, the features extracted may not contain so much favored information and hence the size of feature subset is relatively smaller. In addition, the feature subset sorted by mDS achieved better performances than using all features, especially for sORF type data, whose average Acc is improved about 2% compared to using all features (whose Acc is 81.53%), which indicates that mDS method can identify favorable features for specific data. Furthermore, the features ranked by mDS are consistent with the results we reported in section above: the maximum ORF length feature, which realizes more than 90% Acc itself in classifying normal data, was ranked at the 1st position in normal model, and two effective features for sORF data, hexamer and nucleotide bias feature, were both ranked in top 10 positions in sORF model.

To illustrate the effectiveness of mDS, we also investigated the performance of other commonly used filter feature selection methods, such as mRMR [41], F-score, mutual information, Pearson correlation and Max-Relevance-Max-Distance (MRMD) [37] on the same test datasets to make the comparison. The cut-off points for the optimal feature subsets for these methods are selected via the validation on D_{test_H} and $D_{test\_H\_sORF}$. The average Acc with five times running on these two datasets using the first n features sorted by different feature selection methods as well as the choice of n are illustrated in Supplementary Figures S2 and S3 separately.

For normal data, the Acc of feature subsets determined by F-score, Pearson correlation and MRMD increase quickly at initial, and are higher than that of mDS when feature subset size is less than 510, but the feature subset size that achieving the maximum Acc for these methods are 570, 555 and 525, respectively, which are comparable with mDS's 560. For sORF type data, except for mRMR whose performance is far worse than others, the tendencies and performances for all other methods

Table 2. (a) Comparison among CPPred, DeepCPP and PLEK on previous test datasets. (b) Comparison of DeepCPP (human normal model) with other predictors on new human test dataset. (c) Comparison of DeepCPP (human sORF model) with other predictors on new human sORF test dataset

Datasets (#lncRNA: #mRNA)	Methods	Sp (%)	Sn (%)	Acc (%)	F-score	Mcc	Hm
(a) Comparison among CPPred, DeepCPP and PLEK on previous test datasets[a]							
D_{test_H} (7378:8555)	DeepCPP	98.47	95.53	96.89	0.971	0.938	0.970
	CPPred	96.72	95.37	96.00	0.962	0.920	0.960
	PLEK	97.89	95.44	96.57	0.968	0.932	0.966
$D_{test\_H\_sORF}$ (639:639)	DeepCPP	93.11	83.41	88.26	0.877	0.769	0.880
	Cppred	97.97	62.60	80.28	0.760	0.647	0.764
	PLEK	97.19	78.09	87.64	0.864	0.757	0.866
D_{test_M} (13,954:31097)	DeepCPP	97.33	96.27	96.60	0.975	0.922	0.968
	Cppred	96.79	95.52	95.91	0.970	0.907	0.962
	PLEK	90.61	87.62	88.55	0.914	0.751	0.891
$D_{test\_M\_sORF}$ (987:843)	DeepCPP	92.80	72.72	83.55	0.803	0.675	0.815
	Cppred	96.96	46.50	73.72	0.620	0.514	0.619
	PLEK	90.68	44.37	69.34	0.571	0.401	0.596
(b) Comparison of DeepCPP (human normal model) with other predictors on $D_{test\_H\_new}$[b]							
		Sp (%)	Sn (%)	ACC (%)	F-score	MCC	Hm
DeepCPP		98.44	96.47	98.20	0.930	0.920	0.974
Hugo's SVM		99.54	8.40	88.28	0.150	0.219	0.155
mRNN		95.12	96.65	95.31	0.836	0.819	0.959
lncRNAnet		97.83	95.79	97.58	0.907	0.895	0.968
LncADEEP		98.53	97.92	98.46	0.940	0.932	0.982
LncFinder		96.03	96.21	96.05	0.858	0.842	0.961
RNAsamba		94.06	98.16	94.57	0.817	0.802	0.961
CPPred		96.02	95.46	95.95	0.853	0.836	0.957
(c) Comparison of DeepCPP (human sORF model) with other predictors on $D_{test\_H\_newsORF}$[b]							
DeepCPP		92.67	84.48	88.58	0.881	0.774	0.884
Hugo's SVM		99.57	6.03	52.80	0.113	0.158	0.114
mRNN		94.83	69.83	82.33	0.798	0.668	0.804
lncRNAnet		97.84	58.62	78.23	0.729	0.614	0.733
LncADEEP		97.84	74.14	85.99	0.841	0.741	0.844
LncFinder		98.28	50.86	74.57	0.667	0.558	0.670
RNAsamba		94.83	80.17	87.50	0.865	0.758	0.869
CPPred		99.14	48.28	73.71	0.647	0.551	0.649

[a]Sp is the Acc of lncRNA data and Sn is the Acc of mRNA data. The datasets used here are the filtered version of four previous test datasets from CPPred.
[b]In Hugo's SVM method, GRCh38_firstOrf model was used; in mRNN, w14u3 model was used; in LncADeep, human full-length model was used; in CPPred, human model was used.

are comparable, but mDS realizes its maximum Acc with the least number of features, i.e. 450, while other methods realize their maximum Acc with 485 to 565 features.

Evaluation of DeepCPP human model

To evaluate the DeepCPP, firstly, we compared it with CPPred and PLEK on four previous datasets in [21]. CPPred [21] is a newly developed method and are proved to outperform previous methods like CPAT [14], CPC2 [42], PLEK [15] and sORF finder [43] on distinguishing coding and noncoding RNAs as well as the sORF type coding and noncoding RNAs. But its performances on two human datasets are worse than that of PLEK, the author ascribed the relatively high performance of PLEK to the high level of redundancy between the training dataset in PLEK and the human test dataset in CPPred. However, CPPred is trained with ncRNA while PLEK is trained with lncRNA, and the datasets used for comparisons in [21] are ncRNAs; hence, the comparisons are not fair. We made an objective comparison among DeepCPP, CPPred and PLEK on the filtered version of four previous test datasets by only keeping lncRNA and mRNA with length longer than 200 (because PLEK does not deal with sequence whose length smaller than 201), and the results are recorded in Table 2(a).

Obviously, DeepCPP works much better than CPPred and PLEK with higher performances on all four test datasets, it improves the overall performance from all kinds of measurements, especially for sORF RNA datasets, whose advantages are quite noticeable. The Acc for two sORF RNA datasets are improved by DeepCPP in different ranges, from 14.21 to 0.62% compared to CPPred and PLEK, which are mainly contributed by the mRNA's prediction performance (i.e. Sn). The improvements on Sn of DeepCPP are 20.81 and 5.32% on $D_{test\_H\_sORF}$, and 26.22% and 28.35% on $D_{test\_M\_sORF}$ from that of CPPred and PLEK separately. Yet, the prediction performances of lncRNA (Sp) for DeepCPP in these two datasets are 93.11 and 92.80%. Even though, the significant improvements of Sn further lead to much higher F-score, MCC and Hm values of DeepCPP for sORF type test data. Besides, it should be noticed that DeepCPP outputs better performances than PLEK on human datasets even under the truth that the data similarity between training and test data in DeepCPP is lower than that of PLEK.

The McNemar test is implemented to further compare the models. The statistic values, χ^2 values and *P*-values (annotated as α) between model of DeepCPP and CPPred or PLEK are illustrated in Figure 3. Since DeepCPP and CPPred are both trained with ncRNA, their McNemar test results are evaluated on the

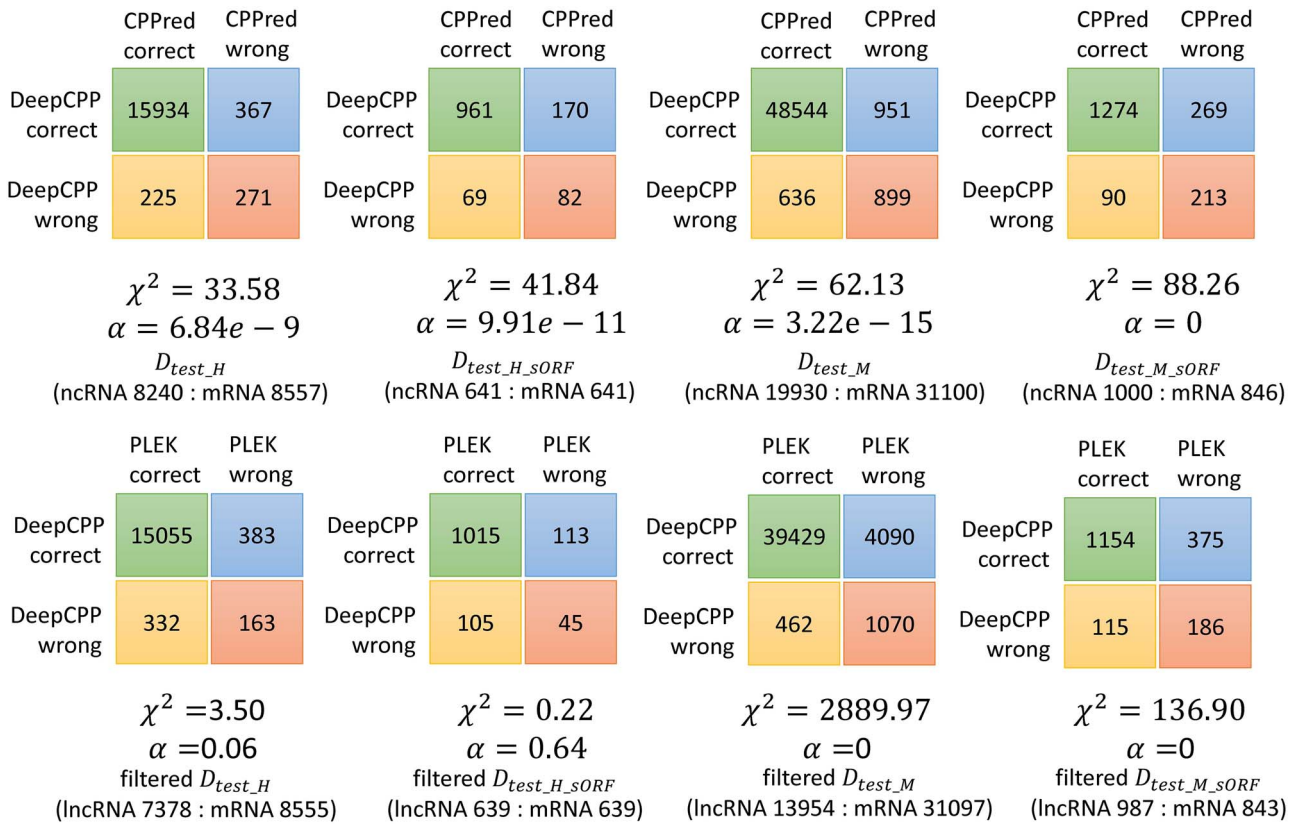

Figure 3. Model comparisons with McNemar test. Upper panel: McNemar test results between DeepCPP and CPPred on four previous test datasets in CPPred. Lower panel: McNemar test results between DeepCPP and PLEK on the filtered version of four previous test datasets in CPPred (only lncRNA and mRNA with length larger than 200).

original test datasets in [21], while the statistic tests between DeepCPP and PLEK are evaluated on the filtered version of these test datasets. When α is below a significant threshold, e.g. 0.05, the null hypothesis we made (proportions are same) will be rejected. Hence, the small *P*-values which are near to 0 in the McNemar test demonstrate that DeepCPP is superior to CPPred, and superior to PLEK on mouse data.

Then, we further evaluated DeepCPP and compared it with seven state-of-the-art methods on our new test datasets, $D_{test\_H\_new}$ and $D_{test\_H\_newsORF}$. The methods used for comparison are all newest published from 2017 onward, including Hugo's SVM method [17], mRNN [20], lncRNAnet [22], LncADeep [23], LncFinder [24], RNAsamba [25] and CPPred [21]. The above seven methods are all machine learning-based, three of them using SVM models and four of them using deep learning models, and they adopt a large range of feature representation methods, such as longest ORF related features, k-mer, one-hot vector, network learned features and so on. A summary of these seven methods can be referred to Supplementary Table S4. These methods are proved to outperform early programs like CPC [12], CPAT [14], FEELnc [19], PLEK [15] and so on. And the data in new test datasets are all released in recent 2 years, where the noncoding RNAs all belong to lncRNA type. The performances for different methods on two new test datasets (normal and sORF) are listed in Table 2(b) and (c).

DeepCPP outperforms most methods on distinguishing normal mRNA and lncRNA, but a little bit worse, i.e. 0.26% lower in Acc, than LncADeep. However, the performance of DeepCPP on sORF type data is quite outstanding. DeepCPP achieves as high as 84.48% of Sn, compared with the second-highest value, RNAsamba's 80.17%, the improvement is 4.31% and the Sn values for all other methods are below 74.14%, which means the improvement of DeepCPP on Sn exceeding 10.34% when compared to these methods. Even evaluated on different test dataset and compared with many other newly published methods trained with different data, DeepCPP still achieves much better Sn value in sORF type RNA data, which is used to be a bottleneck. Although the Sp of DeepCPP is lowest among all methods, its Acc, F-score, Mcc and Hm are the largest due to the significant improvement on Sn.

In addition, we also paid attention to the ability of these methods to identify mRNA with ORF length no longer than 100 codons, which is even more challenging. There are 48 such mRNAs, DeepCPP predicted 36 out of 48 correctly, while Hugo's SVM method, mRNN, lncRNAnet, LncADeep, LncFinder, RNAsamba and CPPred only correctly identified 2, 29, 23, 26, 1, 34 and 16, respectively.

Besides, growing evidences show that some ncRNAs also can encode small peptide [44], we collected eight such RNAs related to cancers or diseases: HOXB-AS3 [45], SHPRH [46], FBXW7 [47], SPAAR [48], STRIT1 [49], MRLN [50], ZNF609 [51] and NPBY [52], to further evaluate the programs. We downloaded the sequences of the above RNAs from NCBI nucleotide database, as some of them have several transcript variants, we obtained 8, 4, 5, 1, 1, 2, 1 and 1 sequences for 8 RNAs, respectively. DeepCPP predicts SHPRH, FBXW7, ZNF609 and two variants of HOXB-AS3 as mRNA (coding) type, while predicts the other four RNAs as noncoding RNA. When evaluating other seven state-of-the-

art methods on the same data, except for hugo's SVM method which predicts all RNAs as noncoding, all other methods predict SHPRH, FBXW7 and ZNF609 as mRNA, and mRNN, RNAsamba, and CPPred also predict part of transcript variants of HOXB-AS3 as mRNA, while lncRNAnet, LncADeep and LncFinder predict all variants of HOXB-AS3 as noncoding RNA. Based on the observations with limited data amount above, DeepCPP shows superiority to hugo's SVM method, lncRNAnet, LncADeep and LncFinder, and has comparative performance with mRNN, RNAsamba and CPPred.

Another advantage of DeepCPP is that it leads to a more balanced model. The smaller gap between Sn and Sp of DeepCPP than other methods on all datasets indicates that the disparity between predicting positive and negative is smaller in DeepCPP, and this can be further proved by the large F-score and Hm values. Besides, a larger MCC value implies that DeepCPP is more agreement between actuals and predictions than other methods.

The performances of DeepCPP on four previous and two new test datasets in this part illustrates the effectiveness of DeepCPP on making classification between lncRNA and mRNA, especially for RNAs with sORFs.

Evaluation of DeepCPP integrate models

Similar to the human model, we also built normal and sORF models for vertebrate and insect species. Mouse and zebrafish data were used to train the vertebrate models, and fruit fly and mosquito data were used to train the insect models. The feature subsets used for integrated models remain the same as that of human models. Besides the new data in training species, the test data also comprise data from new species like pig, dog and honeybee. We compared DeepCPP vertebrate models with Hugo's SVM method [17] and CPPred [21] (Figure 4A and B), which have the integrated models trained by mouse and zebrafish samples, and compared the insect models with CPPred [21] (Figure 4C and D) whose integrate model is trained with insect species data like fruit fly and nematode.

For vertebrate models, Hugo's SVM is still poor in predicting mRNA with extremely low Sn. As to the other two methods, DeepCPP performs better than CPPred, especially on sORF RNA datasets, whose Sn and Acc are 37.24 and 17.09% higher than that of CPPred, and the F-score, MCC and Hm values are also improved by more than 25%. However, during the experiments, we found that the Acc of zebrafish ncRNA test samples are low: 75.92% for DeepCPP and 58.25% for CPPred. We applied cd-hit [28] to test the data similarity between zebrafish ncRNA test and training data, and noticed that only 11 out of 951 zebrafish ncRNA test samples show over 0.8 similarities to its training data. The low similarity here can explain the poor performance of zebrafish ncRNA test samples, and it also reveals that the newly added ncRNA samples in zebrafish are quite different from the samples in early releases in intrinsic features. Besides, for insect models, the overall performances of DeepCPP are higher than that of CPPred in both D_{test_I} and $D_{test\_I\_sORF}$ with 0.36% and 3.14% improvement on Acc.

We further explored the performances of DeepCPP integrate models on specific species and compared it to CPPred (Figure 4E). We first use the DeepCPP vertebrate model to test the mouse dataset built in [21], the Sp, Sn and Acc for D_{test_M} and $D_{test\_M\_sORF}$ are 99.51, 99.10 and 99.26%, and 97.30, 93.74 and 95.67% separately. When compared to CPPred, the improvements are substantial, especially the sORF type data, whose Sn and Acc are improved by 46.93 and 21.67% separately. Besides, when testing sequences of seen species (fruit fly and mosquito) with insect model, DeepCPP is better than CPPred with 2.4% improvement on overall Acc for normal data and about 27.8 and 10.31% improvements on Sn and Acc for sORF data. But for the mRNA identification of unseen species (honeybee), DeepCPP is not as good as CPPred. We ascribed the relatively lower predicting Acc of DeepCPP in honeybee mRNA sequences as the differences of protein between fruit fly & mosquito and honeybee. Research showed that the function and number of proteins of honeybee are much different from fruit fly and mosquito [53], since mRNA guides protein-coding from DNA, it is reasonable to believe that the mRNA sequences of honeybee are much different from that of fruit fly and mosquito either, as well as their intrinsic feature extracted from the sequences.

Furthermore, the relatively small gaps between the value of Sn and Sp as well as the larger F-score and Hm values in both vertebrate and insect models again illustrate that DeepCPP is a balance model, together with its good performance on integrate normal and sORF type data, especially the significant improvement on Sn of sORF type data, DeepCPP is indicated as an effective method to predict RNA coding potential.

Discussion

In this work, firstly, we introduced a novel feature representation method, named nucleotide bias. Nucleotide bias feature extracts information around start codon, as it is reported that the frequencies of nucleotide acids around start codon in CDS are different from those in non-CDS. It is proved to be effective in distinguishing sORF RNAs with at least 2.6% higher value in Hm than other methods. Additionally, we found that g-bigap feature representation is crucial in protein-coding potential prediction especially for sORF type data, but previous works only addressed the effectiveness of continuous k-mer while ignoring the impact of discontinuous k-mer. Then, we proposed a new feature ranking method, named mDS, its successful application on the datasets in this work and the superiority than mRMR highlight its value on selecting features. Finally, based on the new features and other features that are reported to be useful in previous work, together with the mDS method selecting feature subset and ensembled CNN neural network, we developed DeepCPP to predict the coding potential of RNA sequences.

We applied DeepCPP on human data, vertebrate data and insect data to evaluate its performance. For each kind of data, two models, the normal model and the sORF model, were constructed with the corresponding optimal feature subset found by mDS in human data. DeepCPP was firstly evaluated on the same test datasets with CPPred, the results showed that DeepCPP outperforms CPPred and PLEK on both human and mouse species, normal and sORF type data. Particularly, DeepCPP improves more than 5.46% and more than 25.77% on the sORF mRNA identification of human and mouse species. We also compared DeepCPP with seven state-of-the-art programs that are published recently and reported to be better than the previous methods on our new human test datasets. Results showed that DeepCPP outperforms most programs on normal data and is the best model on sORF type data, with at least 4.31% improvement on Sn than that of other methods. Besides, as some ncRNAs are later proved can be coded to small peptide, different methods are also evaluated on such kind of data, but their performances are just satisfying. In the future, such RNAs can be further collected to investigate their features.

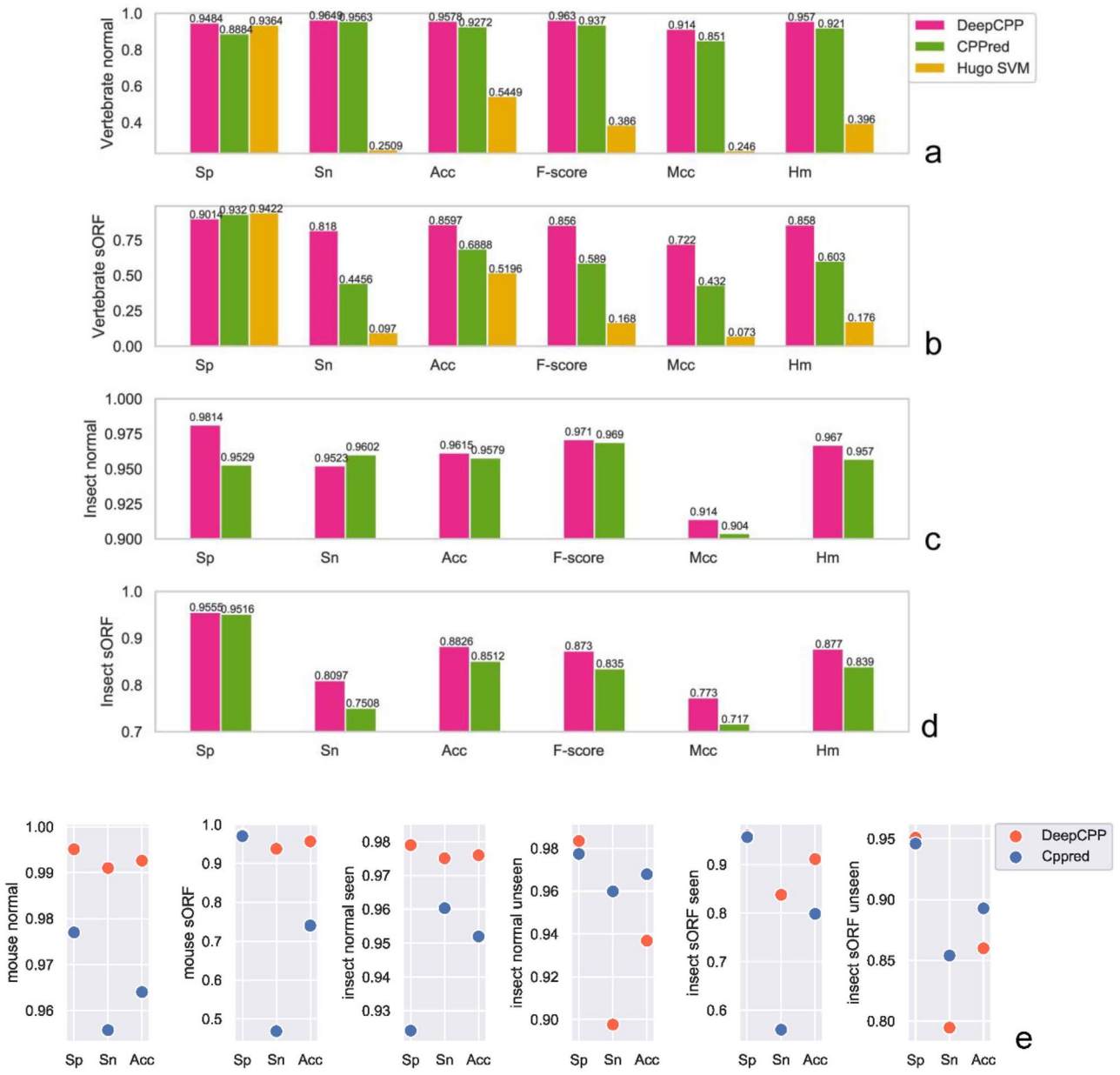

Figure 4. Performance comparisons on integrate datasets. **(A)** Comparison on vertebrate normal dataset D_{test_V}. **(B)** Comparison on vertebrate sORF dataset $D_{test\_V\_sORF}$. **(C)** Comparison on Insect normal dataset D_{test_I}. **(D)** Comparison on Insect sORF dataset $D_{test\_I\_sORF}$. **(E)** Comparisons on (from left to right) CPPred mouse normal dataset D_{test_M}, CPPred mouse-sORF dataset $D_{test\_M\_sORF}$, insect normal seen species data, insect normal unseen species data, insect sORF seen species data and insect sORF unseen species data. Seen species data represent the fruit fly and mosquito samples in D_{test_I}, and unseen species data represents the honeybee samples in D_{test_I}. For vertebrate test, the integrated model for CPPred and *GRCm38_GRCz10* model for hugo's SVM are used; for insect test, the Integrated model for CPPred is used.

We also validated DeepCPP integrate models on data from both seen and unseen species, and compared DeepCPP with other methods that also have integrated models trained by mouse, zebrafish or fruit fly samples. DeepCPP performs excellent on vertebrate datasets, especially, its Sn and Acc are 37.24 and 17.09% higher than that of CPPred in vertebrate sORF test dataset. Additionally, DeepCPP leads to almost 100% prediction rate on mouse normal data and a significant improvement on mouse-sORF type data, i.e. 46.93 and 21.67% improvement than CPPred on Sn and Acc. As to the insect datasets, due to the large differences between the protein of honeybee and fruit fly & mosquito, DeepCPP performs relatively poor on honeybee mRNA identification, but it performs well on fruit fly and mosquito data.

In a nutshell, DeepCPP is an effective method on RNA coding potential prediction, specifically, its notable improvement on sORF mRNA prediction Acc illustrate its significance on identifying RNAs with sORFs.

Data Availability

DeepCPP was implemented in Python3 and is freely available at https://github.com/yuuuuzhang/DeepCPP.

Key Points

- We proposed a novel feature representation method, nucleotide bias, which is useful in sORF RNA coding potential prediction.
- We developed a new feature selection method, which is proved to be effective in this work.
- The comprehensive comparisons between DeepCPP and other state-of-the-art methods on human, vertebrate and insect datasets highlight the superiority of DeepCPP in RNA coding potential prediction, especially on sORF RNA data.

Supplementary Data

Supplementary data are available online at https://academic.oup.com/bib.

Funding

This work was supported by the Fundamental Research Funds for the Central Universities [3132019175, 3132019323, 3132018230]; the National Natural Science Foundation of Liaoning Province [20180550307]; the National Research Foundation (NRF) Singapore through an NRF Fellowship awarded to M.J.F [NRF-NRFF2012–054]; the RNA Biology Center at the Cancer Science Institute of Singapore, NUS, as part of funding under the Singapore Ministry of Education Academic Research Fund Tier 3 awarded to D.G.T [MOE2014-T3–1-006]; and the National Research Foundation Singapore and the Singapore Ministry of Education under its Research Centres of Excellence initiative.

References

1. Mercer TR, Dinger ME, Mattick JS. Long non-coding RNAs: insights into functions. *Nat Rev Genet* 2009;**10**(3):155.
2. Djebali S, Davis CA, Merkel A, *et al*. Landscape of transcription in human cells. *Nature* 2012;**489**(7414):101.
3. Creamer KM, Lawrence JB. XIST RNA: a window into the broader role of RNA in nuclear chromosome architecture. *Philos Trans R Soc B: Biol Sci* 2017;**372**(1733):20160360.
4. Almeida M, Pintacuda G, Masui O, *et al*. PCGF3/5–PRC1 initiates Polycomb recruitment in X chromosome inactivation. *Science* 2017;**356**(6342):1081–4.
5. Xing YH, Yao RW, Zhang Y, *et al*. SLERT regulates DDX21 rings associated with pol I transcription. *Cell* 2017;**169**(4):664–78.
6. Postepska-Igielska A, Giwojna A, Gasri-Plotnitsky L, *et al*. LncRNA Khps1 regulates expression of the proto-oncogene SPHK1 via triplex-mediated changes in chromatin structure. *Mol Cell* 2015;**60**(4):626–36.
7. Poliseno L, Salmena L, Zhang J, *et al*. A coding-independent function of gene and pseudogene mRNAs regulates tumour biology. *Nature* 2010;**465**(7301):1033.
8. Morán I, Akerman İ, Van De Bunt M, *et al*. Human β cell transcriptome analysis uncovers lncRNAs that are tissue-specific, dynamically regulated, and abnormally expressed in type 2 diabetes. *Cell Metab* 2012;**16**(4):435–48.
9. Kondo T, Hashimoto Y, Kato K, *et al*. Small peptide regulators of actin-based cell morphogenesis encoded by a polycistronic mRNA. *Nat Cell Biol* 2007;**9**(6):660.
10. Matsumoto A, Pasut A, Matsumoto M, *et al*. mTORC1 and muscle regeneration are regulated by the LINC00961-encoded SPAR polypeptide. *Nature* 2017;**541**(7636):228.
11. Couso JP, Patraquim P. Classification and function of small open reading frames. *Nat Rev Mol Cell Biol* 2017;**18**(9):575.
12. Kong L, Zhang Y, Ye ZQ, *et al*. CPC: assess the protein-coding potential of transcripts using sequence features and support vector machine. *Nucleic Acids Res* 2007;**35**(suppl_2):W345–9.
13. Lin MF, Jungreis I, Kellis M. PhyloCSF: a comparative genomics method to distinguish protein coding and non-coding regions. *Bioinformatics* 2011;**27**(13):i275–82.
14. Wang L, Park HJ, Dasari S, *et al*. CPAT: coding-potential assessment tool using an alignment-free logistic regression model. *Nucleic Acids Res* 2013;**41**(6):e74–4.
15. Li A, Zhang J, Zhou Z. PLEK: a tool for predicting long non-coding RNAs and messenger RNAs based on an improved k-mer scheme. *BMC Bioinf* 2014;**15**(1):311.
16. Sun L, Luo H, Bu D, *et al*. Utilizing sequence intrinsic composition to classify protein-coding and long non-coding transcripts. *Nucleic Acids Res* 2013;**41**(17):e166–6.
17. Schneider HW, Raiol T, Brigido MM, *et al*. A support vector machine based method to distinguish long non-coding RNAs from protein coding transcripts. *BMC Genomics* 2017;**18**(1):804.
18. Tripathi R, Patel S, Kumari V, *et al*. DeepLNC, a long non-coding RNA prediction tool using deep neural network. *Network Model Anal Health Inf Bioinf* 2016;**5**(1):21.
19. Wucher V, Legeai F, Hedan B, *et al*. FEELnc: a tool for long non-coding RNA annotation and its application to the dog transcriptome. *Nucleic Acids Res* 2017;**45**(8):e57–7.
20. Hill ST, Kuintzle R, Teegarden A, *et al*. A deep recurrent neural network discovers complex biological rules to decipher RNA protein-coding potential. *Nucleic Acids Res* 2018;**46**(16):8105–13.
21. Tong X, Liu S. CPPred: coding potential prediction based on the global description of RNA sequence. *Nucleic Acids Res* 2019;**47**(8):e43–3.
22. Baek J, Lee B, Kwon S, *et al*. Lncrnanet: long non-coding RNA identification using deep learning. *Bioinformatics* 2018;**34**(22):3889–97.
23. Yang C, Yang L, Zhou M, *et al*. LncADeep: an ab initio lncRNA identification and functional annotation tool based on deep learning. *Bioinformatics* 2018;**34**(22):3825–34.
24. Han S, Liang Y, Ma Q, *et al*. LncFinder: an integrated platform for long non-coding RNA identification utilizing sequence intrinsic composition, structural information and physicochemical property. *Brief Bioinform* 2019;**20**(6):2009–27.
25. Camargo AP, Sourkov V, Carazzolle MF. RNAsamba: coding potential assessment using ORF and whole transcript sequence information. *BioRxiv* 2019;620880.
26. O'Leary NA, Wright MW, Brister JR, *et al*. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Res* 2015;**44**(D1):D733–45.
27. Hunt SE, McLaren W, Gil L, *et al*. Ensembl variation resources. *Database* 2018;**2018**:bay119.
28. Li W, Godzik A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics* 2006;**22**(13):1658–9.
29. Claverie JM, Poirot O, Lopez F. The difficulty of identifying genes in anonymous vertebrate sequences. *Comput Chem* 1997;**21**(4):203–14.

30. Deonier RC, Tavaré S, Waterman MS. *Computational Genome Analysis: An Introduction*. Berlin: Springer Science+Business Media, 2005.
31. Cao F, Fullwood MJ. Inflated performance measures in enhancer–promoter interaction-prediction methods. *Nat Genet* 2019;**51**(8):1196–8.
32. Fickett JW. Recognition of protein coding regions in DNA sequences. *Nucleic Acids Res* 1982;**10**(17):5303–18.
33. Nakagawa S, Niimura Y, Gojobori T, *et al*. Diversity of preferred nucleotide sequences around the translation initiation codon in eukaryote genomes. *Nucleic Acids Res* 2007;**36**(3):861–71.
34. Kochetov AV. AUG codons at the beginning of protein coding sequences are frequent in eukaryotic mRNAs with a suboptimal start codon context. *Bioinformatics* 2004;**21**(7): 837–40.
35. Pisarev AV, Kolupaeva VG, Pisareva VP, *et al*. Specific functional interactions of nucleotides at key-3 and+ 4 positions flanking the initiation codon with components of the mammalian 48S translation initiation complex. *Genes Dev* 2006;**20**(5):624–36.
36. Volkova OA, Kochetov AV. Interrelations between the nucleotide context of human start AUG codon, N-end amino acids of the encoded protein and initiation of translation. *J Biomol Struct Dyn* 2010;**27**(5):611–8.
37. Zou Q, Zeng J, Cao L, *et al*. A novel features ranking metric with application to scalable visual and bioinformatics data classification. *Neurocomputing* 2016;**173**:346–54.
38. Kullback S, Leibler RA. On information and sufficiency. *Ann Math Stat* 1951;**22**(1):79–86.
39. Zou Q, Xing P, Wei L, *et al*. Gene2vec: gene subsequence embedding for prediction of mammalian N6-methyladenosine sites from mRNA. *RNA* 2019;**25**(2): 205–18.
40. Wei L, Ding Y, Su R, *et al*. Prediction of human protein subcellular localization using deep learning. *J Parallel Distributed Comput* 2018;**117**:212–7.
41. Peng H, Long F, Ding C. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans Pattern Anal Mach Intell* 2005;**8**:1226–38.
42. Kang YJ, Yang DC, Kong L, *et al*. CPC2: a fast and accurate coding potential calculator based on sequence intrinsic features. *Nucleic Acids Res* 2017;**45**(W1):W12–6.
43. Hanada K, Akiyama K, Sakurai T, *et al*. sORF finder: a program package to identify small open reading frames with high coding potential. *Bioinformatics* 2009;**26**(3):399–400.
44. Zhu S, Wang J, He Y, *et al*. Peptides/proteins encoded by non-coding RNA: a novel resource bank for drug targets and biomarkers. *Front Pharmacol* 2018;**9**:1295.
45. Huang JZ, Chen M, Chen D, *et al*. A peptide encoded by a putative lncRNA HOXB-AS3 suppresses colon cancer growth. *Mol Cell* 2017;**68**(1):171–84.
46. Zhang M, Huang N, Yang X, *et al*. A novel protein encoded by the circular form of the SHPRH gene suppresses glioma tumorigenesis. *Oncogene* 2018;**37**(13):1805–14.
47. Yang Y, Gao X, Zhang M, *et al*. Novel role of FBXW7 circular RNA in repressing glioma tumorigenesis. *J Nat Cancer Inst* 2018;**110**(3):304–15.
48. Lu XW, Xu N, Zheng YG, *et al*. Increased expression of long noncoding RNA LINC00961 suppresses glioma metastasis and correlates with favorable prognosis. *Eur Rev Med Pharmacol Sci* 2018;**22**(15):4917–24.
49. Nelson BR, Makarewich CA, Anderson DM, *et al*. A peptide encoded by a transcript annotated as long noncoding RNA enhances SERCA activity in muscle. *Science* 2016;**351**(6270):271–5.
50. Anderson DM, Anderson KM, Chang CL, *et al*. A micropeptide encoded by a putative long noncoding RNA regulates muscle performance. *Cell* 2015;**160**(4):595–606.
51. Legnini I, Di Timoteo G, Rossi F, *et al*. Circ-ZNF609 is a circular RNA that can be translated and functions in myogenesis. *Mol Cell* 2017;**66**(1):22–37.
52. D'Lima NG, Ma J, Winkler L, *et al*. A human microprotein that interacts with the mRNA decapping complex. *Nat Chem Biol* 2017;**13**(2):174–80.
53. Zdobnov EM, Bork P. Quantification of insect genome divergence. *Trends Genet* 2007;**23**(1):16–20.