Using k-mer Embeddings Learned from a Skip-Gram Based Neural Network as Effective Feature Representation for Building a Cross-Species Prediction Model to Identify DNA N6-Methyladenine Sites in Plant Genomes

Duong Nguyen Trinh Trung (khucnam@yahoo.com)
Yuan Ze University https://orcid.org/0000-0001-5793-649X
Van-Ngu Trinh
Soonchunhyang University
Nguyen Quoc Khanh Le
Taipei Medical University
Yu-Yen Ou
Yuan Ze University

Research Article

Keywords: DNA N6-methyladenine site prediction, k-mer embeddings,natural language processing, ensemble tree-based algorithms

Posted Date: June 30th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-553304/v1

Abstract

Identification of DNA N6-methyladenine sites has been a very active topic of computational biology due to the unavailability of suitable methods to identify them accurately, especially in plants. Substantial results were obtained with a great effort put in extracting, heuristic searching, or fusing a diverse types of features, not to mention a feature selection step. We considered DNA, the human life book, as a book corpus for training DNA language models. K-mer embeddings then were generated from Skipgram neural networks and input into several ensemble tree-based algorithms. We trained the prediction model on Rosaceae genome dataset and performed a comprehensive test on 3 plant genome datasets. Our proposed method shows promising performance with AUC performance approaching an ideal value on Rosaceae dataset (0.99), a high score on Rice dataset (0.95) and improved performance on Rice dataset while enjoying an elegant, yet efficient feature extraction process.

Introduction

DNA N6-methyladenine is a non-canonical DNA modification at the 6th nitrogen position of an adenine ring, catalyzed by DNA methyltransferases (DNMTs) [1]. Since it was not detectable in earlier studies, DNA 6mA modification was considered absent in eukaryotes. With the development of deep sequencing techniques, 6mA was found to be present in the genomes of diverse species [2–5]. Recently, it has been resurfaced as a possible reversible epigenetic mark that correlated with a series of biological processes, such as DNA replication, DNA repair, and transcription. [2, 6–10]. While assumed to be absent due to its undetectable degree, considerable 6mA distribution is identified across the plant genome suggesting its roles in plant development, tissue differentiation, and regulations in gene expression. Furthermore, a previous rice genome study indicates that 6mA is associated with the environmental alteration. The levels of 6mA are positively correlated with the expression of key genes related to the stress response [5].

Despite the potential roles of 6mA suggested from the abovementioned studies, the biological function of 6mA in plants remains elusive. The experimental methods for identifying DNA N6-Methyladenine sites such as next-generation sequencing with coupling immunoprecipitation [11], DNA immunoprecipitation with 6mA antibodies restriction enzyme-assisted sequencing with DpnI-assisted [12], mass spectrometry, single-molecule real-time (SMRT) sequencing and ultrahigh-performance liquid chromatograph [13] generally require substantial time and effort, yet they can only cover a partial 6mA epigenetics landscape. Thus, characterizing the distribution of 6mA sites across the genome using computational approach is essential to understand its functions.

As the amount of genomic data is soaring, building efficient computational approaches, especially those based on machine learning methods, for identify 6mA sites in different genomes becomes an attractive topic for a lot of research groups. A central question is how to define an effective and distinctive feature sets that can help enhance the prediction performance. For long time, many groups have relied on various encoding scheme for effective data representation and feature learning in DNA methylation pattern detection. In an extensive review published in 2018 [14], Yu et al. have collected, analyzed, and

summarized the existing encoding schemes of genome sequence. In this review, the authors grouped the used encoding schemes into 5 groups namely biochemical properties, primary-structure properties, cartesian-coordinate properties, binary and information encoding, graphical representation. We also found that in order to obtain high prediction performance for DNA methylation sites, a great effort must be spent on extracting, heuristic searching, or combining a diverse types of features [15–18]. In several works, a two-step feature selection were also required. The work of Hasan et al. [19] is a great example in which the authors explored 10 different feature encoding schemes before finding five best feature encoding schemes based on physicochemical and position-specific information. Although the authors can achieved promising results, we believe that effective prediction model can also be constructed using a delicate, yet efficient feature selection method.

DNA is the giant book of life in which the arrangement of 4 nucleotide matters. This human life book is, in several aspects, similar to other normal books containing characters appearing in a certain order which follows some grammar rules to convey intended meanings. In this view, we can employ natural language processing techniques to decipher hidden biological meanings in DNA sequences. Thus, to address the above challenges and utilize the interrelations between biological sequences and human language, we employed advanced method in natural language processing area to formulate the representation of DNA. Specifically, we constructed a bag of nucleotide n-gram with Skip-gram architecture neural network to learn from DNA segments. This neural network generated numerical vectors for DNA k-mers which were called k-mer embeddings. In the next step, we concatenated word embeddings of various k-mers to form a real-valued vectors to input into machine learning-based prediction models. We offered several fusions of features all of which are based on k-mer embeddings when searching for best k-mer embedding-based feature sets.

Apart from a discriminative feature sets, a second important part to build an efficient prediction model is to employ a good machine learning algorithms. In this study, we exploited and compared 3 efficient ensemble tree-based machine learning algorithms namely Deep Forest, XGBoost, and Random Forest in predicting DNA N^6-Methyladenine sites with the proposed feature representation method. Among 3 algorithms, Random Forest is used as a classifier in 6 out of 12 recent 6mA sites prediction [16, 20–24]. It uses an ensemble of trees to gather “the wisdom of crowds” for effective prediction. Regarding the two other ensemble tree-based machine learning algorithms, while Deep Forest is a special machine learning algorithm that get inspired from deep neural networks and ensemble learning, XGBoost a decision-tree-based ensemble machine learning algorithm that uses a gradient boosting framework and it has been widely used in machine learning community due its efficiency in terms of both performance and processing time.

We applied our research on DNA N6- Methyladenine data of plant genome. Specifically, we chose Rosaceae genome to train and evaluate the prediction performance. Moreover, we also use two recent plant genome datasets (Rice and Arabidopsis Thaliana) for the independent tests. Compared to the state-of-the-art predictor, on independent tests, our approach obtained promising results with AUC performance approaching the ideal value on Rosaceae genome dataset at 0.99 and 0.953 on Rice dataset.

Furthermore, the statistical rates were also improved on 4 surveyed metrics on Rice genome dataset. Without suffering from the laborious process for extracting, searching and combining traditional sequence-derived features and other encoding schemes, our experimental-confirmed results demonstrate that our approach is straightforward, yet efficient and our study could open a new method in bioinformatics modeling using natural language processing technique.

Materials & Methods

We present a cross-species prediction model to identify N^6-Methyladenine sites based on k-mer embedding-based features. Figure 1 displays the flowchart of our study which can be roughly divided into 4 main sub-processes: data collection, feature generation, classification and performance evaluation.

2.1 Data collection

It is always essential to construct a high-quality and unbiased benchmark dataset to establish a strong supervised learning classification. We therefore re-used the benchmark dataset from the latest study [19] to solve this problem. The data in this dataset were obtained from different sources for different species. First, the Rosaceae were obtained from MDR, MethSMRT, and the Gene Expression Omnibus (GEO) databases [25–27]. Specifically, the Fragaria vesca and Rosa chinensis DNA genomes under the Rosaceae family were retrieved with 40,574 samples containing 41 base pairs centering the adenine. Then, to avoid overestimate the prediction performance, the author have removed redundant sequences with the cutoff value of 90% resulting in 36,537 positive samples. For the negative sample, the authors followed the procedure described in [16, 17, 20, 28] with two other plant species datasets (rice and A. thaliana). Table 1 provides the statistics of the surveyed datasets.

Table 1
The statistics of the surveyed dataset.

Species	Dataset	6mA sites	Non 6mA sites
Rosaceae	Training	29,237	29,433
Rosaceae	Independent test	7,300	7,300
Rice		154,000	154,000
Arabidopsis thaliana		31,873	31,873

2.2 Training a bag of nucleotide n-gram with Skip-gram architecture neural network as a DNA language model

DNA sequences can be viewed as a language comprising of 4 letters (A, T, G, and C). Therefore, in this study, we regarded each sequence (DNA segment with the length of 41, to be more exact) as a ”DNA sentence” by splitting it into a series of k-mers (overlapping is allowed) with k in the range from 1 to 5. This means that after the splitting step, we obtained 5 text corpuses for building 5 different language models. In each case, the k-mer is corresponding to a word in human language which enable the use of natural language processing technique to extract the hidden information. The upper part of the feature extraction section of Fig. 1 illustrates how DNA segment CTTATGG is split into 5 “DNA sentences” comprising of 1-mers, 2-mers, 3-mers, 4-mers, and 5-mers, respectively.

Apart from the corpuses of text, to construct a language model for word embedding generation, we also need an embedding method. In this study, we used a Skip-gram neural network [29] which has one hidden layer to learn from the “DNA sentences” described above. The input to this Skip-gram neural network is the target k-mer while the output is context k-mers and the learning is in an unsupervised manner. Figure S1 in the Supplementary materials give an example of a Skip-gram neural network learning from a “DNA sentence”.

We further applied another technique which is call bag of nucleotide n-gram. In this technique, each k-mer was represented as a bag of nucleotide n-grams. For example, a 4-mer CTAT was a bag of C, T, A, CT, TA, AT, CTA, TAT, CTAT, and special 6-mer < CTAT>. The vector representation for CTAT then was the sum of the representations associated to each nucleotide n-gram.

We used fastText [30–32] to create the bag of nucleotide n-gram based with Skip gram (BoN-Skipgram) described above. fastText is an open source, free and lightweight library that enables fast and efficient text representation learning. After training, the embedding can be retrieved from BoN-Skipgram by looking up in the mapping between k-mers and the weight vectors. In Table 2, we provided a hyperparameters for training a BoN-Skipgram model.

Table 2
Training hyperparameters of BoN-Skipgram model

Hyperparameters	**fastText model**
Network architecture	Skipgram
Corpus	DNA sequences of Rosaceae
Context Words	1-mers, 2-mers, 3-mers, 4-mers, 5-mers
Minimal number of k-mer ocurences	1 (default)
Epoch	5 (default)
Learning rate	0.1 (default)

2.3 Hybrid features from k-mer embeddings

We obtained k-mer embeddings from the Skip-gram neural network trained on DNA sequences. Each k-mer was presented by a dense continuous numerical vector assumed to distill the most information about that k-mer in the relation of it and its neighboring k-mers. In natural language processing, this kind of word representation has been proved to be more efficient than that of bag-of-word or one-hot encoding. Unlike the traditional bag-of-words model where each word or nucleotide was represented by a high-dimensional sparse vector based on the occurrence counts of words within a document, word embedding holds the power of generalization, which can catch some similarities among contextual features [16].

In many NLP applications, the dimension of the k-mer embedding vectors is an important hyperparameters. Applications using bigger datasets tend to have the dimension of word or sentence embedding vectors of several hundred or even several thousand. However, the bigger the dimension, the larger the size of the feature vectors. As our samples contain only 41 nucleotides, we assumed that a scalar can present the condensed information of the k-mers in such very short DNA segments. Furthermore, using many real values to present each k-mers may lead to redundant information, thus causing overfitting problems. Therefore, all our k-mer embedding vectors have the length of 1.

We further combined different k-mer embedding-based features to create a diverse feature sets to find the optimal one. Table 3 presents our combination methods using concatenation methods and figure S2 in the Supplementary materials represents a visualization of these feature sets.

Table 3
Hybrid feature generation

Feature combination	Dimension of feature vectors
1-mer, 2-mer	41 + 40
1-mer, 2-mer, 3-mer	41 + 40 + 39
1-mer, 2-mer, 3-mer, 4-mer	41 + 40 + 39 + 38
1-mer, 2-mer, 3-mer, 4-mer, 5 mers	41 + 40 + 39 + 38 + 37

2.4 Classification

2.4.1 Deep Forest

Deep Forest is a special machine learning algorithm that get inspired from deep neural networks and ensemble learning [29]. Deep Forest integrates best ideas from these two approaches namely layer-by-layer processing and model diversity. In layer-by-layer processing, Deep Forest exploits a cascade

structure where each level of cascade receives feature information from its preceding layer and provides its processing outcome to the next layer. Furthermore, each layer is an ensemble of decision tree forest translating the whole model into an ensemble of ensembles. To encourage the diversity, different types of classifiers (LogisticRegression, RandomForest, and ExtraTreesClassifier) were included as diversity is crucial for ensemble construction.

Equation (1) below illustrates the error-ambiguity decomposition:

E = E - A (1)

Where E denotes the error of an ensemble, E and A denote the average error of individual classifiers in the ensemble and the average ambiguity (or diversity) among the individual classifiers, respectively.

2.4.2 XGBoost

XGBoost is a generalized boosting technique that enable the optimization of an arbitrarily specialized loss function. It is rooted from boosting technique, which is an iterative ensemble method that trained models sequentially. These models can be considered as "weak learners" (usually are decision trees) since they perform basic prediction rules that only execute slightly better than a random guess. The basic principle behind boosting is to concentrate on the "hard" examples, or the examples that the model fails to trustfully predict correctly. These examples are given more emphasis by skewing the distribution of observations to make such examples appear in a sample probable. As such, the next weak learner will be more focused on correcting these hard examples correct. Since we want our learner is always doing better than random, from sequential training round, we will always get some degree of information. Combining all the simple prediction rules into one overarching model, a powerful predictor is obtained. There are several algorithms based on boosting techniques. Uniquely, in order to avoid over-fitting, XGBoost uses a more regularized model formalization, which yields better performance.

2.4.3 Random Forest

Random Forest is an ensemble algorithm based on many decision trees. It inherits the benefits of a decision tree model such as scaling well to larger datasets and being robust against to irrelevant features. Furthermore, it also improves the performance by reducing the variance which is one of the downsides of decision trees. The “random” part in the Random Forest was implemented via the sampling process from the training data. This means that we trained a group of different decision trees on different randomly-picked samples from training data and we also sample different subsets of features among all available ones. This added randomness in the splitting process with an aim to reduce variance in the final model.

2.5 Model setting and evaluation metrics

We used sensitivity, specificity, accuracy, and Matthews Correlation Coefficient (MCC) as the evaluation metrics. Given the training data, for a certain model, we performed the 5-fold cross-validation technique to lower the risk of overestimating or underestimating the real performance of our prediction model.

Additionally, 3 independent tests on Rosaceae, Rice and Arabidopsis thaliana were performed with optimal model after the hyperparemeter tuning process employed with this cross-validation technique to evaluate the prediction performance on unseen data. The definition of evaluation measurements are defined as follows:

$$\text{Sensitivity} = TP/(TP + FN) \quad (1)$$

$$\text{Specificity} = TN/(TN + FP) \quad (2)$$

$$\text{Accuracy} = (TP + TN)/(TP + FP + TN + FN) \quad (3)$$

$$MCC = (TP*TN - FP*FN)/\sqrt{((TP + FP)(TP + FN)(TN + FP)(TN + FN))} \quad (4)$$

where TP, TN, FP, and FN respectively denote true positives, true negatives, false positives, and false negative. Moreover, we also plotted receiver operating characteristic (ROC) curve and calculated the area under the ROC curve (AUC) as a single metric to evaluate the overall performance of this binary classification. These metrics are based on dynamic positive-negative prediction thresholds instead of a static threshold as seen in accuracy. It is the area under ROC that plots true positive rates (sensitivity) against false positive rates using different prediction thresholds. AUC is a real value between zero and one. The higher AUC, the better the classifier and a perfect classifier, meanwhile, has AUC equal to one.

Results & Discussions

3.1 Hyperparameter optimization

Hyperparameter optimization plays a very important role to achieve high-quality models in most machine learning and deep learning tasks (e.g., in [20, 21]). We performed grid search on possible values of hyperparemeters for each classifiers. We tuned XGBoost model on 3 hyperparameters namely, max_depth (maximum depth of a tree), learning rate and n_estimators (number of trees). For Random Forest, values of max_features (max number of features considered for splitting a node), and n_estimators (number of trees in the forest) was searched for. In deep forest, we fixed the different types of classifiers (LogisticRegression, RandomForest, and ExtraTreesClassifier) to add the diversity to the model. Then, we tuned on 3 hyper-parameters: n_estimators max_feature, and max_depth (max number of levels in each decision tree). The ranges of these hyperparameters were given in Table 4.

Table 4
Hyperparameters search range for each learning algorithm

Learning algorithm	Hyperparameter range
XGBoost	learning rate = [0.1, 0.2]
	n_estimators = 50 □ 300
	max_depth = 25 □ 150
Random Forest	n_estimators = 50 □ 300
	max_feature = 25 □ 150
Deep Forest	n_estimators = 50 □ 150
	max_layers = 25 □ 100
	max_dept = 25 □ 100

3.2 Overall performance

As mentioned above, we concatenated k-mer embeddings with k from 1 to 5 to create several feature sets. We used these feature sets to input into Deep Forest, Random Forest, and XGBoost classifiers. The performance scores for for the different feature sets from 5-fold cross validation process are presented in Table 5. We show both mean and standard deviation of the scores for 6 random experiments. We observed that although the results are stable (low standard variations) across 6 runs, there were not much difference in terms of prediction performance among these feature sets. However, among 3 classification algorithms, Deep Forest yields the best performance with highest accuracy, sensitivity, specificity, and MCC on all feature types. In contrast, Random Forest classifier obtains the lowest performance. Furthermore, we consider the 1-2-3-4mers feature set comprising of 1-mer, 2-mer, 3-mer, and 4-mer embeddings the most effective one. Thus, we continued to use the 1-2-3-4mers feature set in the next experiments.

Table 5
Overall performance

	5-fold cross-validation				
1-2mers	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
Deep forest	**94.1 ± 0.134**	**94.7 ± 0.197**	**93.5 ± 0.153**	**0.882 ± 0.003**	0.980 ± 0.001
Random Forest	93.0 ± 0.095	93.1 ± 0.376	92.8 ± 0.319	0.859 ± 0.002	0.975 ± 0.001
XGBoost	94.1 ± 0.306	94.5 ± 0.284	93.7 ± 0.445	0.882 ± 0.006	**0.982 ± 0.002**
1-2-3mers	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
Deep forest	**94.6 ± 0.122**	**94.5 ± 0.711**	**94.7 ± 0.527**	**0.892 ± 0.002**	0.982 ± 0.000
Random Forest	93.1 ± 0.094	93.6 ± 0.136	92.5 ± 0.303	0.861 ± 0.002	0.976 ± 0.001
XGBoost	94.1 ± 0.332	94.2 ± 0.210	94.1 ± 0.587	0.883 ± 0.007	**0.983 ± 0.002**
1-2-3-4mers	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
Deep forest	**94.6 ± 0.043**	**94.6 ± 0.149**	**94.5 ± 0.142**	**0.892 ± 0.001**	0.982 ± 0.001
Random Forest	93.2 ± 0.042	93.4 ± 0.388	93.1 ± 0.398	0.864 ± 0.001	0.977 ± 0.001
XGBoost	94.2 ± 0.301	94.3 ± 0.285	94.2 ± 0.565	0.885 ± 0.006	**0.983 ± 0.002**
1-2-3-4-5mers	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
Deep forest	**94.5 ± 0.043**	**94.9 ± 0.281**	**94.1 ± 0.260**	**0.891 ± 0.001**	0.981 ± 0.000
Random Forest	93.2 ± 0.072	93.5 ± 0.251	92.9 ± 0.330	0.865 ± 0.001	0.977 ± 0.001
XGBoost	94.2 ± 0.286	94.4 ± 0.526	93.9 ± 0.423	0.884 ± 0.006	**0.983 ± 0.002**

**It should be noted that we report the average result in format: m ± d, where m is the average and d is the standard deviation across the many runs. Highest scores are in bold. We used this convention throughout this paper*,

3.3 Prediction performance of the surveyed ensemble tree-based algorithms on independent datasets

We reported the prediction performance of Deep Forest, XGBoost, and RandomForest classifiers on 3 independent datasets with 1-2-3-4mers feature set in Table 6a, 6b, 6c. These tables shows the prediction outcomes on Rosaceae, Rice and A. thaliana independent datasets, respectively. It is interesting to

observer (from Table 6a) that for Rosaceae data, the independent test performance scores are higher in almost all metrics for all 3 classifiers (marked by arrows) compared to the 5-fold cross validation ones presented in Table 5. This indicates that our 3 classifiers do not undergo overfitting problem. In addition, for Rice independent dataset, our 3 classifiers consistently obtains AUC scores of abound 0.95 which is promising as we train our model on Rosaceae genome data. However, such scores on corresponding metrics on A. thaliana independent dataset is not satisfied as our models can only achieve an accuracy of around 80%. From this comparison, we considered Deep Forest as the best classifier and used it for further experiments.

Table 6
a: Prediction performance on Rosaceae independent data

	Rosaceae independent dataset				
Classifiers	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
Deep forest	**95.9 ± 0.110↑**	**96.3 ± 0.110↑**	**95.5 ± 0.245↑**	**0.918 ± 0.219↑**	**0.990 ± 0.020↑**
Random Forest	94.7 ± 0.117↑	95.0 ± 0.299↑	94.4 ± 0.371↑	0.894 ± 0.223	0.985 ± 0.075↑
XGBoost	95.6 ± 0.308↑	95.8 ± 0.248↑	95.5 ± 0.513↑	0.913 ± 0.618↑	0.990 ± 0.118↑

Table 6
b: Prediction performance on Rice independent data

	Rice independent dataset				
Classifiers	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
Deep forest	**90.6 ± 0.041**	**89.7 ± 0.226**	**91.6 ± 0.293**	**0.813 ± 0.041**	0.947 ± 0.065
Random Forest	90.0 ± 0.089	89.0 ± 0.271	91.0 ± 0.175	0.801 ± 0.002	0.947 ± 0.001
XGBoost	90.1 ± 0.103	89.2 ± 0.335	90.9 ± 0.175	0.802 ± 0.002	**0.950 ± 0.001**

Table 6
c: Prediction performance on A. thaliana independent data

	A. thaliana independent dataset				
Classifiers	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
Deep forest	**79.1 ± 0.248**	88.2 ± 0.458	70.0 ± 0.366	**0.592 ± 0.005**	0.863 ± 0.001
Random Forest	78.8 ± 0.308	**89.1 ± 0.609**	68.6 ± 0.804	0.590 ± 0.006	0.861 ± 0.003
XGBoost	79.1 ± 0.264	87.5 ± 0.987	**70.7 ± 1.367**	0.591 ± 0.004	**0.872 ± 0.003**

3.4 Survey about the combination of k-mer embedding-based features and well-known encoding schemes

In [19], Hasan et al have presented their great effort in surveying and detecting the best feature sets for 6mA sites prediction problems. We have reused their datasets for this study. Therefore, it is interesting to combine the optimal features they have found with our proposed features. Thus, we created a new hybrid feature sets comprising of our best types (1-2-3-4mers) with their best 5 encoding schemes (MBE, DBE, EIIP, DPP, and NCP). Then, we utilized the same hyperparameter settings as we did in the experimental mentioned above. We reported the average results of 10 random runs with Deep Forest classifier in Table 7 and plotted the ROC curves in Fig. 2.

Table 7
Prediction performance on hybrid features comprising of k-mer embeddings and optimal encoding schemes.

	5-fold cross validation				
	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
Rosaceae	94.6 ± 0.0	94.9 ± 0.2↑	94.3 ± 0.2	0.893 ± 0↑	0.983 ± 0↑
	Independent dataset				
	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
Rosaceae	96.0 ± 0.1↑	96.4 ± 0.1↑	95.6 ± 0.2↑	0.921 ± 0.001↑	0.991 ± 0↑
Rice	90.9 ± 0.1	89.3 ± 0.3	92.3 ± 0.3↑	0.817 ± 0.001↑	0.953 ± 0↑
Arabi	78.8 ± 0.2	92.4 ± 0.7↑	65.2 ± 1.2	0.598 ± 0.003↑	0.875 ± 0↑

In Table 7, where applicable, we have put an arrow indicating an improvement compared to the k-mer embedding-based features presented in Table 6a on Deep Forest performance. It is easily to see that with this hybrid feature set, many performance scores are a little bit higher. For example, in the independent

test on Rosaceae data, the scores on all 5 metrics are improved. In addition, for Rice and A.thaliana, we observed the improvement for scores in 3 metrics each. Figure 2 shows the ROC curves of Deep Forest on this hybrid feature set. It is obvious that our model can approach an ideal AUC score of 0.99 on independent test on Rosaceae data and obtain a high AUC value of 0.953 on Rice data. Furthermore, the performance are stable across 10 random runs.

3.5 Comparison to previous works on DNA N^6-Methyladenine sites prediction

As we mentioned earlier, researchers working on DNA N^6-Methyladenine sites prediction has employed a lot of encoding schemes in a laborious manner to obtain promising results. Here, we would like to compare the effectiveness of our feature extraction method with Meta-i6mA, the latest predictor by Hasan et al. Thus, we used the prediction performances obtained with our best k-mer embedding based feature set (1-2-3-4mer) and Deep Forest for this comparison. (We also recalculated the prediction performance of Meta-i6mA where needed). Table 8a shows the prediction comparison on cross-validation data while Table 8b, 8c, 8d display the results on 3 surveyed independent test datasets.

As shown in Table 8a, our predictor yields a better performance in all metrics except for the specificity score. On Rosaceae independent test, we achieved better accuracy, sensitivity and MCC. Furthermore, on Rice independent test (Table 8b), our accuracy, sensitivity, and MCC are better. It is noteworthy that we improved the MCC from 0.797 to 0.813 which indicating our proposed features is efficient even when being using in a cross-species prediction manner.

Overall, from this comparison and based on the fact that we did not need to go through the laborious process for extracting and selecting features but still can obtain same level prediction outcome with state-of-the-art 6mA site predictor. We therefore confidently conclude that our extraction method, based on the application of NLP technique, particularly k-mer embeddings generating from DNA language models, on this N6-methyladenine site prediction problems is straightforward, yet efficient.

Table 8
a: Performance comparison between our proposed model and Meta-i6mA on cross-validation data

	5-fold cross-validation				
Classifiers	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
k-mer embedding	**94.6 ± 0.043**	94.6 ± 0.149	**94.5 ± 0.142**	**0.892 ± 0.001**	**0.982 ± 0.001**
Hasan et al.	93.6	**95**	92.1	0.891	0.958

Table 8
b: Performance comparison between our proposed model and Meta-i6mA on Rosaceae independent data

	Rosaceae independent dataset				
Classifiers	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
k-mer embedding	**95.9 ± 0.110**	**96.3 ± 0.110**	95.5 ± 0.245	**0.918 ± 0.219**	0.990 ± 0.020
Hasan et al.	95.8	95.7	**96**	0.917	-

Table 8
c: Performance comparison between our proposed model and Meta-i6mA on Rice independent data

	Rice independent dataset				
Classifiers	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
k-mer embedding	**90.6 ± 0.041**	89.7 ± 0.226	**91.6 ± 0.293**	**0.813 ± 0.041**	0.947 ± 0.065
Hasan et al.	89.9	**90.5**	89.2	0.797	-

Table 8
d: Performance comparison between our proposed model and Meta-i6mA on Arabidopsis thaliana independent data

	Arabidopsis thaliana independent dataset				
Classifiers	Acc(%)	Spec(%)	Sen(%)	MCC	AUC
k-mer embedding	79.1 ± 0.248	88.2 ± 0.458	70.0 ± 0.366	0.592 ± 0.005	0.863 ± 0.001
Hasan et al.	**80.9**	**91.0**	**70.8**	**0.631**	-

***Note: "-" sign indicates that the value is not shown*

3.6 Feature sets and source code availability

To ensure the reproducibility of the results, we provided the feature sets learned from fasText model (k-mer embedding) and the source code for constructing prediction model. The interested users can go to our repository at https://github.com/khucnam/Deep_Emb_6mA to examine our methods.

Conclusion

Predicting DNA methylation patterns has drawn a lot of attention, especially for DNA N^6-Methyladenine sites on plant genome, due to its biological functions. The computational methods based on machine learning for such problems need to search and combine for optimal feature sets from a wide range of

encoding schemes. Such approaches also neglect the analogies of DNA sequences and human language. Therefore, we eliminated all the old encoding scheme and adopted the new one based on advanced technique for decipher textual information in natural language processing. Furthermore, we employed 3 different ensemble tree-based classifiers, to extensively survey the efficiency of the proposed feature extraction method. As we have showed, our approach obtains promising results with AUC performance approaching the ideal value on Rosaceae genome dataset (0.99) and a high AUC score on Rice dataset (0.95). Furthermore, an improvement on the MCC score on Rice genome dataset was observed. This demonstrates that our proposed feature representation is elegant, yet efficient. We believe that our method could open a new way to extract useful features using advanced NLP technique.

Declarations

Funding.

This work was partially supported by the Ministry of Science and Technology, Taiwan, R.O.C. under Grant no. MOST 109-2811-E-155-505 and no. MOST 109-2221-E-155-045.

Competing interests. The authors declare no competing interests.

Availability of data and material The original data is published at http://kurata14.bio.kyutech.ac.jp/Meta-i6mA/help.php

Code availability Source code is provided at https://github.com/khucnam/Deep_Emb_6mA

Authors' contributions Conceptualization, N.T.T.D, T.V.N, N.Q.K.L. and Y.Y.O.; methodology, N.T.T.D and N.Q.K.L.; formal analysis, T.T.D.N.; writing—original draft preparation, T.T.D.N.; writing—review and editing, T.T.D.N., T.V.N, N.Q.K.L., and Y.Y.O.; supervision, Y.Y.O.; funding acquisition, Y.Y.O. All authors have read and agreed to the published version of the manuscript.

Ethics approval Not applicable

Consent to participate Not applicable

Consent for publication All authors have read and agreed to the published version of the manuscript.

References

[1] Z. K. O'Brown, and E. L. Greer, "N6-methyladenine: a conserved and dynamic DNA mark," DNA Methyltransferases-Role and Function, pp. 213-246: Springer, 2016.

[2] E. L. Greer, M. A. Blanco, L. Gu, E. Sendinc, J. Liu, D. Aristizábal-Corrales, C.-H. Hsu, L. Aravind, C. He, and Y. Shi, "DNA methylation on N6-adenine in C. elegans," Cell, vol. 161, no. 4, pp. 868-878, 2015.

[3] G. Zhang, H. Huang, D. Liu, Y. Cheng, X. Liu, W. Zhang, R. Yin, D. Zhang, P. Zhang, and J. Liu, “N6-methyladenine DNA modification in Drosophila,” Cell, vol. 161, no. 4, pp. 893-906, 2015.

[4] C. Zhou, C. Wang, H. Liu, Q. Zhou, Q. Liu, Y. Guo, T. Peng, J. Song, J. Zhang, and L. Chen, “Identification and analysis of adenine N 6-methylation sites in the rice genome,” Nature plants, vol. 4, no. 8, pp. 554-563, 2018.

[5] Q. Zhang, Z. Liang, X. Cui, C. Ji, Y. Li, P. Zhang, J. Liu, A. Riaz, P. Yao, and M. Liu, “N6-Methyladenine DNA methylation in Japonica and Indica rice genomes and its association with gene expression, plant development, and stress responses,” Molecular plant, vol. 11, no. 12, pp. 1492-1508, 2018.

[6] G.-Z. Luo, M. A. Blanco, E. L. Greer, C. He, and Y. Shi, “DNA N 6-methyladenine: a new epigenetic mark in eukaryotes?,” Nature reviews Molecular cell biology, vol. 16, no. 12, pp. 705-710, 2015.

[7] P. J. Pukkila, J. Peterson, G. Herman, P. Modrich, and M. Meselson, “Effects of high levels of DNA adenine methylation on methyl-directed mismatch repair in Escherichia coli,” Genetics, vol. 104, no. 4, pp. 571-582, 1983.

[8] D. Roberts, B. Hoopes, W. McClure, and N. Kleckner, “IS10 transposition is regulated by DNA adenine methylation,” Cell, vol. 43, no. 1, pp. 117-130, 1985.

[9] D. Ratel, J. L. Ravanat, F. Berger, and D. Wion, “N6-methyladenine: the other methylated base of DNA,” Bioessays, vol. 28, no. 3, pp. 309-315, 2006.

[10] J. Karanthamalai, A. Chodon, S. Chauhan, and G. Pandi, “DNA N6-Methyladenine Modification in Plant Genomes—A Glimpse into Emerging Epigenetic Code,” Plants, vol. 9, no. 2, pp. 247, 2020.

[11] Z. D. Smith, and A. Meissner, “DNA methylation: roles in mammalian development,” Nature Reviews Genetics, vol. 14, no. 3, pp. 204-220, 2013.

[12] G.-Z. Luo, F. Wang, X. Weng, K. Chen, Z. Hao, M. Yu, X. Deng, J. Liu, and C. He, “Characterization of eukaryotic DNA N 6-methyladenine by a highly sensitive restriction enzyme-assisted sequencing,” Nature communications, vol. 7, no. 1, pp. 1-6, 2016.

[13] G. Fang, D. Munera, D. I. Friedman, A. Mandlik, M. C. Chao, O. Banerjee, Z. Feng, B. Losic, M. C. Mahajan, and O. J. Jabado, “Genome-wide mapping of methylated adenine residues in pathogenic Escherichia coli using single-molecule real-time sequencing,” Nature biotechnology, vol. 30, no. 12, pp. 1232-1239, 2012.

[14] N. Yu, Z. Li, and Z. Yu, “Survey on encoding schemes for genomic data representation and feature learning—from signal processing to machine learning,” Big Data Mining and Analytics, vol. 1, no. 3, pp. 191-210, 2018.

[15] M. Zhang, J.-W. Sun, Z. Liu, M.-W. Ren, H.-B. Shen, and D.-J. Yu, “Improving N6-methyladenosine site prediction with heuristic selection of nucleotide physical–chemical properties,” Analytical biochemistry, vol. 508, pp. 104-113, 2016.

[16] H. Xu, R. Hu, P. Jia, and Z. Zhao, “6mA-Finder: a novel online tool for predicting DNA N6-methyladenine sites in genomes,” Bioinformatics, vol. 36, no. 10, pp. 3257-3259, 2020.

[17] W. Chen, H. Lv, F. Nie, and H. Lin, “i6mA-Pred: Identifying DNA N6-methyladenine sites in the rice genome,” Bioinformatics, vol. 35, no. 16, pp. 2796-2800, 2019.

[18] J. Khanal, D. Y. Lim, H. Tayara, and K. T. Chong, “i6mA-stack: A stacking ensemble-based computational prediction of DNA N6-methyladenine (6mA) sites in the Rosaceae genome,” Genomics, 2020.

[19] M. M. Hasan, S. Basith, M. S. Khatun, G. Lee, B. Manavalan, and H. Kurata, “Meta-i6mA: an interspecies predictor for identifying DNA N6-methyladenine sites of plant genomes by exploiting informative features in an integrative machine-learning framework,” Briefings in Bioinformatics, 2020.

[20] S. Basith, B. Manavalan, T. H. Shin, and G. Lee, “SDM6A: A web-based integrative machine-learning framework for predicting 6mA sites in the rice genome,” Molecular Therapy-Nucleic Acids, vol. 18, pp. 131-141, 2019.

[21] M. M. Hasan, B. Manavalan, W. Shoombuatong, M. S. Khatun, and H. Kurata, “i6mA-Fuse: improved and robust prediction of DNA 6 mA sites in the Rosaceae genome by fusing multiple feature representation,” Plant molecular biology, pp. 1-10, 2020.

[22] X. Wang, and R. Yan, “RFAthM6A: a new tool for predicting m 6 A sites in Arabidopsis thaliana,” Plant molecular biology, vol. 96, no. 3, pp. 327-337, 2018.

[23] P. Feng, H. Yang, H. Ding, H. Lin, W. Chen, and K.-C. Chou, “iDNA6mA-PseKNC: Identifying DNA N6-methyladenosine sites by incorporating nucleotide physicochemical properties into PseKNC,” Genomics, vol. 111, no. 1, pp. 96-102, 2019.

[24] M. Tahir, H. Tayara, and K. T. Chong, “iDNA6mA (5-step rule): Identification of DNA N6-methyladenine sites in the rice genome by intelligent computational model via Chou's 5-step rule,” Chemometrics and Intelligent Laboratory Systems, vol. 189, pp. 96-101, 2019.

[25] E. Clough, and T. Barrett, "The gene expression omnibus database," Statistical genomics, pp. 93-110: Springer, 2016.

[26] S. W. Taju, and Y.-Y. Ou, "Using deep learning with position specific scoring matrices to identify efflux proteins in membrane and transport proteins." pp. 101-108 %@ 1509038345.

[27] Z.-Y. Liu, J.-F. Xing, W. Chen, M.-W. Luan, R. Xie, J. Huang, S.-Q. Xie, and C.-L. Xiao, “MDR: an integrative DNA N6-methyladenine and N4-methylcytosine modification database for Rosaceae,” Horticulture research, vol. 6, no. 1, pp. 1-7, 2019.

[28] C. Pian, G. Zhang, F. Li, and X. Fan, “MM-6mAPred: identifying DNA N6-methyladenine sites based on Markov model,” Bioinformatics, vol. 36, no. 2, pp. 388-392, 2020.

[29] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality." pp. 3111-3119.

[30] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” Transactions of the Association for Computational Linguistics, vol. 5, pp. 135-146 %@ 2307-387X, 2017.

[31] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, “Bag of tricks for efficient text classification,” arXiv preprint arXiv:1607.01759, 2016.

[32] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov, “Fasttext. zip: Compressing text classification models,” arXiv preprint arXiv:1612.03651, 2016.

Figures

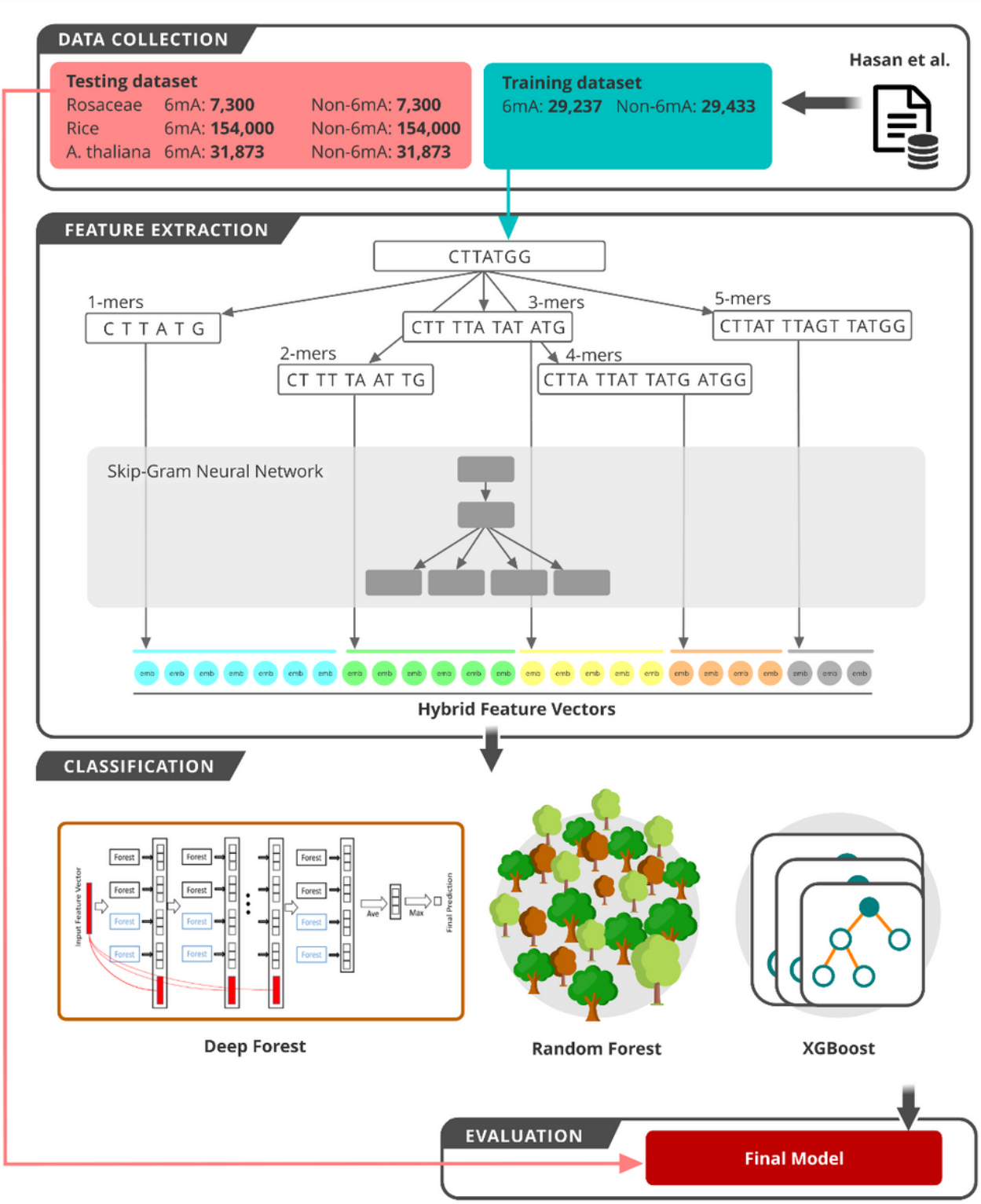

Figure 1

The flowchart of our study. For illustration purpose, the hybrid features combining 5 types of k-mer embeddings were used

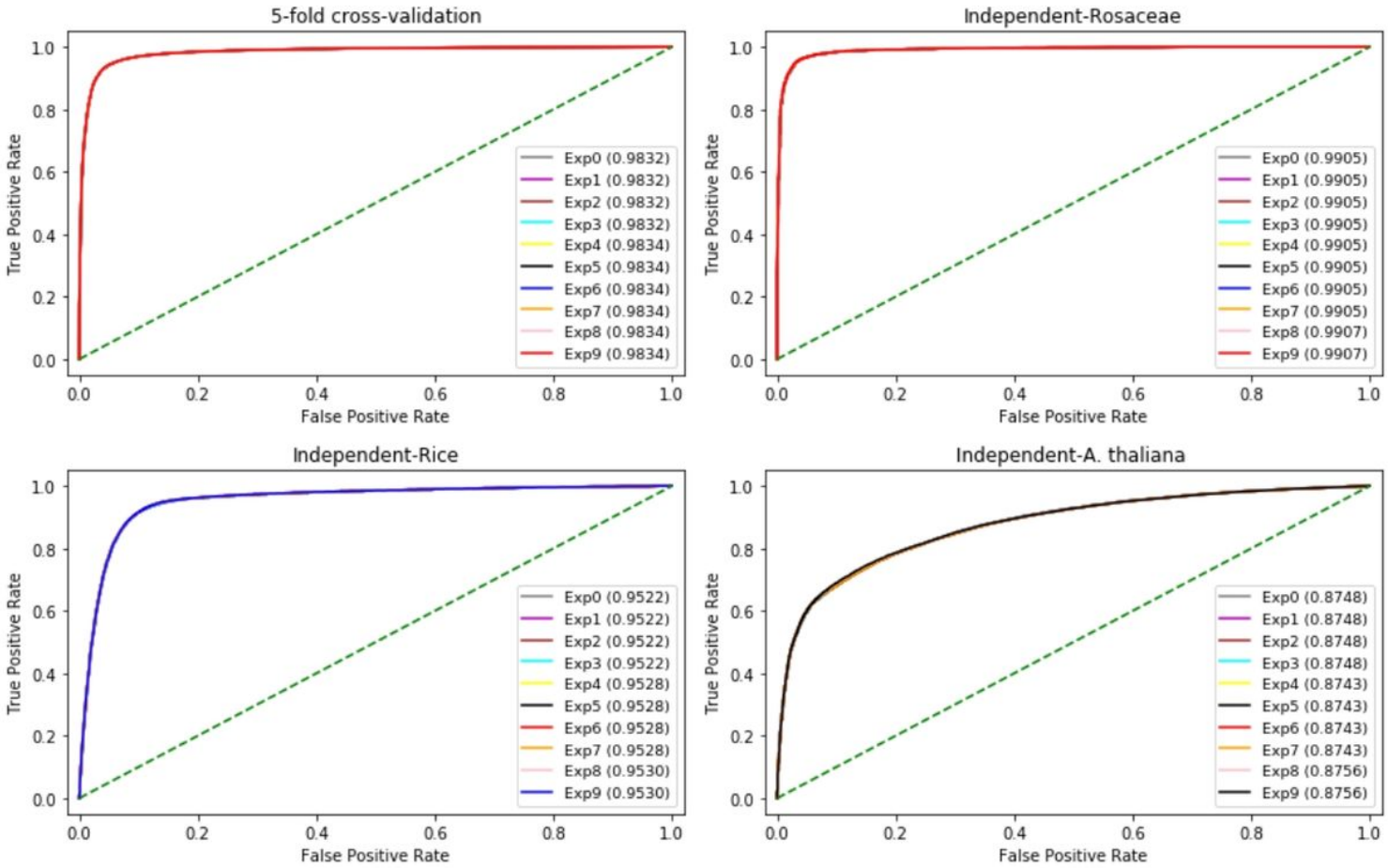

Figure 2

ROC curves of Deep Forest on hybrid features. From let to right: First row: ROC curves on cross-validation data and independent test data of Rosaceae; Second row: ROC curves on independent test data of Rice and A. tha-liana

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementaryInformation.docx