ACS OMEGA

Open Access

http://pubs.acs.org/journal/acsodf

Article

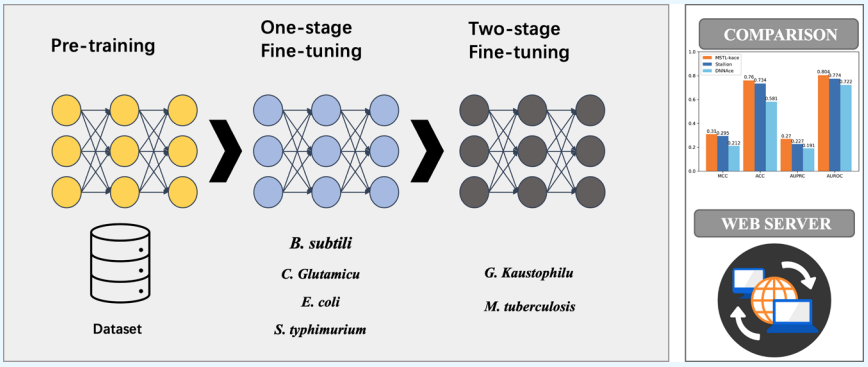
MSTL-Kace: Prediction of Prokaryotic Lysine Acetylation Sites Based on Multistage Transfer Learning Strategy

Gang-Ao Wang, Xiaodi Yan, Xiang Li, Yinbo Liu, Junfeng Xia, and Xiaolei Zhu*

Cite This: *ACS Omega* 2023, 8, 41930−41942

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information



ABSTRACT: As one of the most important post-translational modifications (PTM), lysine acetylation (Kace) plays an important role in various biological activities. Traditional experimental methods for identifying Kace sites are inefficient and expensive. Instead, several machine learning methods have been developed for Kace site prediction, and hand-crafted features have been used to encode the protein sequences. However, there are still two challenges: the complex biological information may be under-represented by these manmade features and the small sample issue of some species needs to be addressed. We propose a novel model, MSTL-Kace, which was developed based on transfer learning strategy with pretrained bidirectional encoder representations from transformers (BERT) model. In this model, the high-level embeddings were extracted from species-specific BERT models, and a two-stage fine-tuning strategy was used to deal with small sample issue. Specifically, a domain-specific BERT model was pretrained using all of the sequences in our data sets, which was then fine-tuned, or two-stage fine-tuned based on the training data set of each species to obtain the species-specific BERT models. Afterward, the embeddings of residues were extracted from the fine-tuned model and fed to the different downstream learning algorithms. After comparison, the best model for the six prokaryotic species was built by using a random forest. The results for the independent test sets show that our model outperforms the state-of-the-art methods on all six species. The source codes and data for MSTL-Kace are available at https://github.com/leo97king/MSTL-Kace.

1. INTRODUCTION

Protein post-translational modifications (PTM) play pivotal roles in bioregulatory processes, including cellular metabolism, DNA repair, gene activation, gene regulation, and signaling processes.[1−3] Common types of PTM include crotonylation, methylation, phosphorylation, ubiquitination, and acetylation. As one of the most important PTMs, lysine acetylation (Kace) exists in both prokaryotes and eukaryotes and is universally present in the nucleus and cytoplasm. According to previous studies, Kace is involved in several important physiological functions including transcriptional regulation, regulation of signaling pathways, metabolic regulation, and regulation of protein stability.[4]

Identification of lysine acetylation sites is essential in the study of acetylation, and it is necessary to develop efficient methods to detect Kace sites. Although traditional experimental methods can identify Kace sites, they are inefficient and expensive. Therefore, researchers have developed computational methods to predict lysine acetylation sites.[5−14] In earlier years, most predictors were developed for identifying Kace in eukaryotes, and studies on prokaryotes are lacking.

In recent years, with the development of artificial intelligence technology and its application to bioinformatics, several methods have been developed for predicting Kace sites of prokaryotes based on machine learning (ML) and deep learning (DL). Among them, the representative ones are STALLION[15] and DNNAce.[16] In 2018, Chen et al. developed a model, ProAcePred, to predict Kace sites of 9 prokaryotic

Received: September 16, 2023
Revised: October 11, 2023
Accepted: October 13, 2023
Published: October 25, 2023

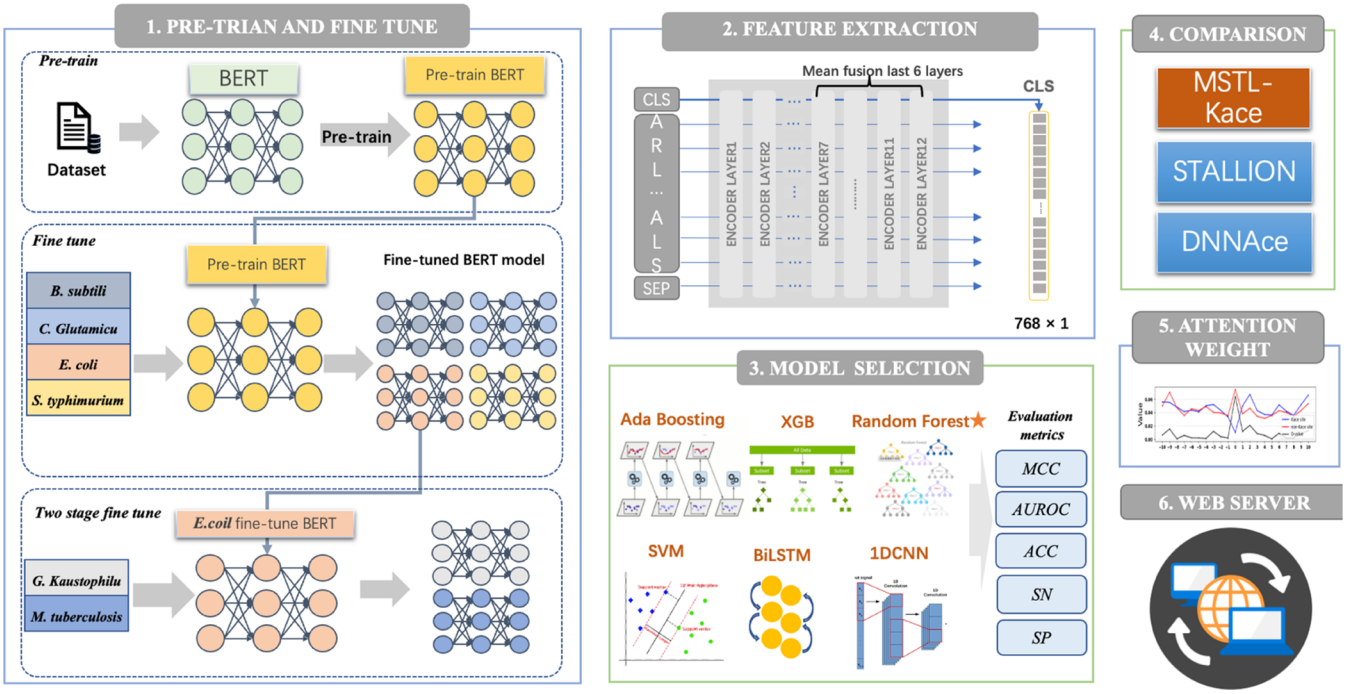


Figure 1. Overall flow of analysis in the present study. Our workflow is composed of pretraining and fine-tuning, feature extraction, machine learning-based, and deep learning-based prediction, evaluation, analysis of attention weights, comparison with existing methods, and web server implementation.

species.[12] In their work, the authors extracted 7 kinds of features which were then optimized by elastic net. Based on the optimized feature subsets, SVM was used to build the models. Yu et al. constructed a deep learning model, DNNAce, for acetylation sites prediction. They used the same data sets as in Chen et al. work.[12] For building DNNAce, six types of features were extracted from protein sequences, and Group Lasso was used to remove the irrelevant features. Based on the selected features, a feedforward neural network (FNN) was used to build the models. STALLION was developed based on the data set from ProAcePred 2.0,[13] an updated version of ProAcePred, which includes six prokaryotic species. STALLION is a stacking ensemble-based[17,18] predictor. For building the model, 11 types of encoding schemes were used to extract the features from the protein sequences. Three feature selection approaches were employed to carefully select the optimal feature set for each of the five different tree-based ensemble algorithms and to construct their respective base learners for each species. Then, the model was trained and evaluated with an appropriate classifier using the predicted information from the base learners and fivefold cross-validation.

Although the above methods have made considerable progress in identifying Kace sites, there are still some limitations. The features used for building these methods are basically hand-crafted features that were extracted from protein sequences. The hand-crafted features are mainly based on experience, which may not be able to represent all of the biological information contained in the protein sequences. Therefore, to eliminate the reliance on hand-crafted features for prediction performance and to mine the biological information in sequences more deeply, bidirectional encoder representations from transformers[19] (BERT) have been used to represent the high-level embeddings of protein sequences.

The BERT[20] model was proposed by Google in 2019 which has shown outstanding performance in the field of natural language processing (NLP), and upon its introduction, it achieved state-of-the-art new results on 11 NLP tasks. After the emergence of the BERT model, many scholars have applied it to the analysis of biological sequences.[21,22] By considering the protein sequences as sentences, the BERT model has been successfully applied to the prediction of protein PTM sites. Qiao et al. proposed a model named BERT-Kcr,[23] for predicting protein lysine crotonylation (Kcr) sites. The model was developed by using a transfer learning method with pretrained BERT models. The features encoded by BERT were extracted and then fed into a BiLSTM network to build their final model. Similarly, Liu et al. proposed a model named BERT-Kgly[24] to predict protein lysine glycation sites by extracting features from BERT. This suggests that using sequence information and NLP pretrained models directly can be an effective method for identifying protein PTM sites. However, the two methods mentioned above utilize only the embedding feature of the last encoder layer without considering the information complementarity among multiple embedding layers. In addition, the small sample problem for predicting PTMs was also not addressed in the two previous works.

In this study, we proposed a new approach called MSTL-Kace to predict the Kace sites of proteins. By considering the protein sequences as natural language sentences, a domain-specific BERT model was pretrained using all of the sequences in our data sets. Then, we fine-tuned the pretrained model in one or two stages based on the training data sets of different species. Then, the embedding of token "CLS" for each sentence (sequence) was extracted from the fine-tuned model, which was fed to the different downstream learning algorithms. After comparison, a random forest was chosen as the optimal classifier. The results for the independent test set show that our

https://doi.org/10.1021/acsomega.3c07086
ACS Omega 2023, 8, 41930−41942

model outperforms the state-of-the-art methods on all six prokaryotic species. The flowchart of the experiment is shown in Figure 1.

2. MATERIALS AND METHODS

2.1. Data Sets. To fairly compare the performance with other methods, we used the same data sets as those used in STALLION. The data sets include peptides from six species, *Bacillus subtilis*, *Corynebacterium glutamicum*, *Escherichia coli*, *Geobacillus kaustophilus*, *Mycobacterium tuberculosis*, and *Salmonella typhimurium*. The data sets were constructed by Chen et al.,[13] based on the PLMD database.[25] The homologous sequences were eliminated by using CD-HIT[26] based on a threshold of sequence identity as 30%. Each sequence is a peptide of 21 residues in length centered with a residue K. If a residue is missing in the sequence, the missing position is replaced with a pseudoresidue "O". Ultimately, this data set consists of a total of 11 138 nonredundant positive (Kace) samples and 14 843 nonredundant negative (non-Kace) samples. Then, 10 484 nonredundant positive (Kace) and the same number of negative samples were selected as the training set, and the remaining samples were used as the independent test set. The specific number of samples for each species is shown in Table 1.

Table 1. Training and Independent Test Sets for Six Species

	train set		independent test set	
species	positive	negative	positive	negative
B. subtilis	1571	1571	125	1165
C. glutamicum	1052	1052	83	830
E. coli	6592	6592	361	1384
G. kaustophilus	206	206	17	192
M. tuberculosis	865	865	68	575
S. typhimurium	198	198	10	217

2.2. Methods. *2.2.1. BERT Model.* Bidirectional encoder representations from transformers (BERT), which was proposed by Devlin et al.,[20] is a formidable language representation model. The model is based on the original transformer model that was proposed by Vaswani et al.[19] BERT model has achieved excellent results on a wide range of NLP tasks including answering questions and linguistic inference. The architecture of BERT is a multilayer bidirectional Transformer encoder that learns the information contained within the text using both the left and the right side of the text. The network structure is the same for all encoder layers, and each encoder layer consists of two main sublayers: a multihead self-attention layer and a feedforward neural network layer. In addition, a residual connection is added to each sublayer. Then, the output is normalized by using LayerNorm.[27] Thus, the output of each sublayer is LayerNorm $(x + \text{Sublayer}(x))$. The schematic diagram of the BERT model structure is shown in Figure S1. When a sentence is input to the BERT model, each word is encoded by three embeddings: token embedding, segment embedding, and positional embedding. Generally, when training a BERT model, special tokens such as "CLS" token and "SEP" token are added as the beginning and the end of a sentence. Each word of an input text is fed into the token embedding layer, which is thus converted to a vector. The segment embedding is used to distinguish whether the word belongs to the first half or the second half of the sentence. The positional embedding contains information about the position of the input token, the formula for the position vector[19] is listed as follows

$$\mathrm{PE}_{(\mathrm{pos},2i)} = \sin\left(\frac{\mathrm{pos}}{10000^{2i/d_{\mathrm{model}}}}\right)$$

$$\mathrm{PE}_{(\mathrm{pos},2i+1)} = \cos\left(\frac{\mathrm{pos}}{10000^{2i/d_{\mathrm{model}}}}\right)$$

These embeddings are then processed by the Transformer encoder for which the core is multihead self-attention layers. Multihead attention involves the integration of multiple attentional mechanisms, resulting in the generation of queries, keys, and values by employing attention functions and linear transformations. Each individual attention function operates on a specific subset of the output sequence, ensuring their independence from one another. There are several studies on PTM site prediction that have achieved good results by incorporating attention mechanisms.[22,23,28,29] By generating different attention weights, the multihead attention mechanism can enhance model accuracy and stability. This is calculated using the following formulas

$$\mathrm{softmax}(x_i) = \frac{\exp(x_i)}{\sum_{j=1}^{n}\exp(x_j)}$$

$$Q_i = XW_i^{\mathrm{q}},\ K_i = XW_i^{\mathrm{k}},\ V_i = XW_i^{\mathrm{v}}$$

$$\mathrm{head}_i = \mathrm{attention}(Q_i, K_i, V_i) = \mathrm{softmax}\left(\frac{Q_i K_i^{\mathrm{T}}}{\sqrt{d_{\mathrm{k}}}}\right)V_i$$

$$\mathrm{multihead}(Q, K, V) = \mathrm{concat}(\mathrm{head}_1, \ldots, \mathrm{head}_n)W^{\mathrm{c}}$$

where Q, K, and V represent query, key, and value matrices, respectively; X represents input matrices; and W^{q}, W^{k}, W^{v}, and W^{c} represent the trained weight matrices.

In this study, we pretrained the BERT model with the protein sequences in the training data sets of all species to obtain a domain-specific BERT model. Then, the domain-specific BERT was fine-tuned based on the training data set for each species to obtain a species-specific BERT model. Afterward, the embeddings were extracted from species-specific BERT model, which were fed to the downstream classifiers to build our final models.

2.2.2. Two-Stage Fine-Tuning. In the field of natural language processing, generally, the larger the data set used to train the model, the better the model's performance and generalization ability. However, in our case, the sample sizes of some species are small. To address the issue of poor performance on the species with small sample sizes, Tsukiyama et al.[22] used a two-stage fine-tuning strategy. Specifically, the fine-tuned model on the species with large sample size was fine-tuned again on the species with small sample size. In this study, we also adopt a two-stage transfer learning approach to deal with the small sample size issue. Specifically, we used the fine-tuned model on the species with large data set as an intermediate model and then fine-tuned the intermediate model on the small data set to improve performance. This method is used to mitigate the impact of limited data availability.

2.3. Features and Classifiers. Because the embedding features of different encoder layers can represent different

https://doi.org/10.1021/acsomega.3c07086
ACS Omega 2023, 8, 41930−41942

levels of information, we extracted the embedding features of the "CLS" tokens from the different encoder layers of the fine-tuned BERT models, and a mean fused strategy was used to fuse these features. Based on mean fused features, four machine learning classifiers random forest (RF),[30] AdBoost (ADB),[31] XGBoost (XGB),[32] support vector machines (SVM)[33] and two deep learning classifiers 1D-CNN[34] (Figure S2) and BiLSTM[35] (Figure S3) were used to build our models. Details of the six classifiers can be found in the Supporting Information.

2.4. Model Evaluation Parameters. In order to validate the performance of the model and compare it more intuitively with other methods, we chose the area under the receiver operating characteristic (AUROC) curve and the area under the precision-recall curve (AUPRC) as the main evaluation metrics. The receiver operating characteristic (ROC) curve is an important index to measure the robustness of the model. Considering that the independent tests used in this study were unbalanced data sets, the area under the precision-recall curve (AUPRC) was also used to evaluate the models. In addition, we have also chosen specificity (SP), sensitivity (SN), accuracy (ACC), and Matthew's correlation coefficient (MCC) as evaluation parameters, which are defined as follows

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

$$SP = \frac{TN}{TN + FP}$$

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

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

where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively.

3. RESULTS AND DISCUSSION

3.1. Performance of the Fine-Tuned BERT Models. After we pretrained the BERT model with the samples in the training data sets of all species, we fine-tuned the model to predict the Kace sites for each species. During fine-tuning, the performance of the model can be improved by adjusting hyperparameters such as epoch, learning rate, optimizer, etc. We selected hyperparameters suitable for different species through fivefold cross-validation. The fine-tuned model was then used to make predictions for the test set. The results on the fivefold cross-validation and the test set are shown in Table 2, respectively. The cross-validation AUROC of *B. subtilis* is 0.71. For *C. glutamicum*, it is 0.707. And the AUROC values of *E. coli*, *G. kaustophilus*, *M. tuberculosis*, and *S. typhimurium* are 0.702, 0.604, 0.712, and 0.528, respectively. In addition, the values of AUROC for six species on the independent test set are 0.736, 0.746, 0.721, 0.709, 0.734, and 0.580.

Table 2. Results Predicted by the Fine-Tuned BERT Model on the Validation and Test Sets

	validation set			independent test set		
species	MCC	ACC	AUROC	MCC	ACC	AUROC
B. subtilis	0.306	0.657	0.710	0.290	0.691	0.736
C. glutamicum	0.342	0.631	0.707	0.305	0.735	0.746
E. coli	0.325	0.709	0.702	0.359	0.669	0.721
G. kaustophilus	0.130	0.555	0.604	0.229	0.565	0.709
M. tuberculosis	0.371	0.663	0.712	0.323	0.734	0.734
S. typhimurium	0.072	0.531	0.528	0.069	0.652	0.580

Table 3. Fivefold Cross-Validation AUROC Values of the Models by Using the Embedding Feature from Pretrained BERT Model

	ADB	RF	XGB	SVM	BiLSTM	1D-CNN
B. subtilis	0.632	0.675	0.638	0.643	0.645	0.638
C. glutamicum	0.644	0.657	0.687	0.68	0.677	0.681
E. coli	0.624	0.630	0.673	0.665	0.67	0.684
G. kaustophilus	0.585	0.563	0.564	0.561	0.572	0.568
M. tuberculosis	0.646	0.658	0.67	0.677	0.669	0.679
S. typhimurium	0.563	0.587	0.582	0.541	0.569	0.583

3.2. Models Built Based on the Embedding Features Extracted from Pretrained Domain-Specific BERT Model. Besides fine-tuning pretrained BERT model for downstream tasks, the embedding features can be extracted from BERT model to build models based on other learning algorithms.[21,36] We extracted the embedding features of token "CLS" from the pretrained model and fed them into several downstream classifiers such as AdaBoost (ADB), random forest (RF), XGBoost (XGB), support vector machine (SVM), BiLSTM, and 1D-CNN. Table 3 shows the fivefold cross-validation AUROC values of different models on the training data sets. The results indicate that there is no significant improvement compared to the cross-validation AUROC values in Table 2. Therefore, we proceeded with extracting features from the fine-tuned model for subsequent experiments.

3.3. Models Built Based on the Embedding Features Extracted from Fine-Tuned BERT Models. Furthermore, we extracted the embedding features of the "CLS" tokens from the different encoder layers of the fine-tuned BERT models. Note that for each round of cross-validation, the fourfold of data was used to fine-tune the model and the remaining fold of data as validation set was not used to fine-tune the model to avoid overfitting. Then, the embeddings were extracted based on the fine-tuned model. Considering the complementarity between the embeddings extracted from different encoder layers, the embedding features were combined by the mean fusion method, by which the embeddings extracted from different encoder layers were averaged as the final feature vector. Based on the mean fused features by combining the last encoder layer, the last 3 encoder layers, the last 6 encoder layers, the last 9 encoder layers, and all 12 encoder layers, we built models for different species by using three machine learning classifiers (ADB, XGB, SVM) and two deep learning classifiers (1D-CNN and BiLSTM). We used grid search to optimize the hyperparameters of each classifier, the ranges of the parameters and the parameters used for the final models are shown in Tables S1 and S2. The cross-validation AUROCs of the models based on fusion features from different encoding layers by using different learning algorithms are shown in Tables S3−S8. Figure 2 shows the cross-validation AUROCs of the models based on the random forest. The trends of all species are basically consistent and the highest AUROCs were obtained when using the fusion features from the last 6 layers.

https://doi.org/10.1021/acsomega.3c07086
ACS Omega 2023, 8, 41930−41942

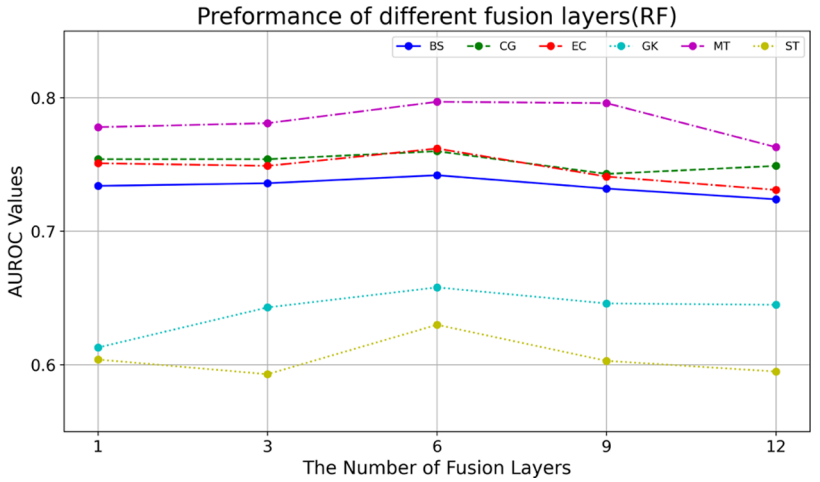


Figure 2. Cross-validation AUROCs of the models based on the fusion features from different encoding layers by using random forest.

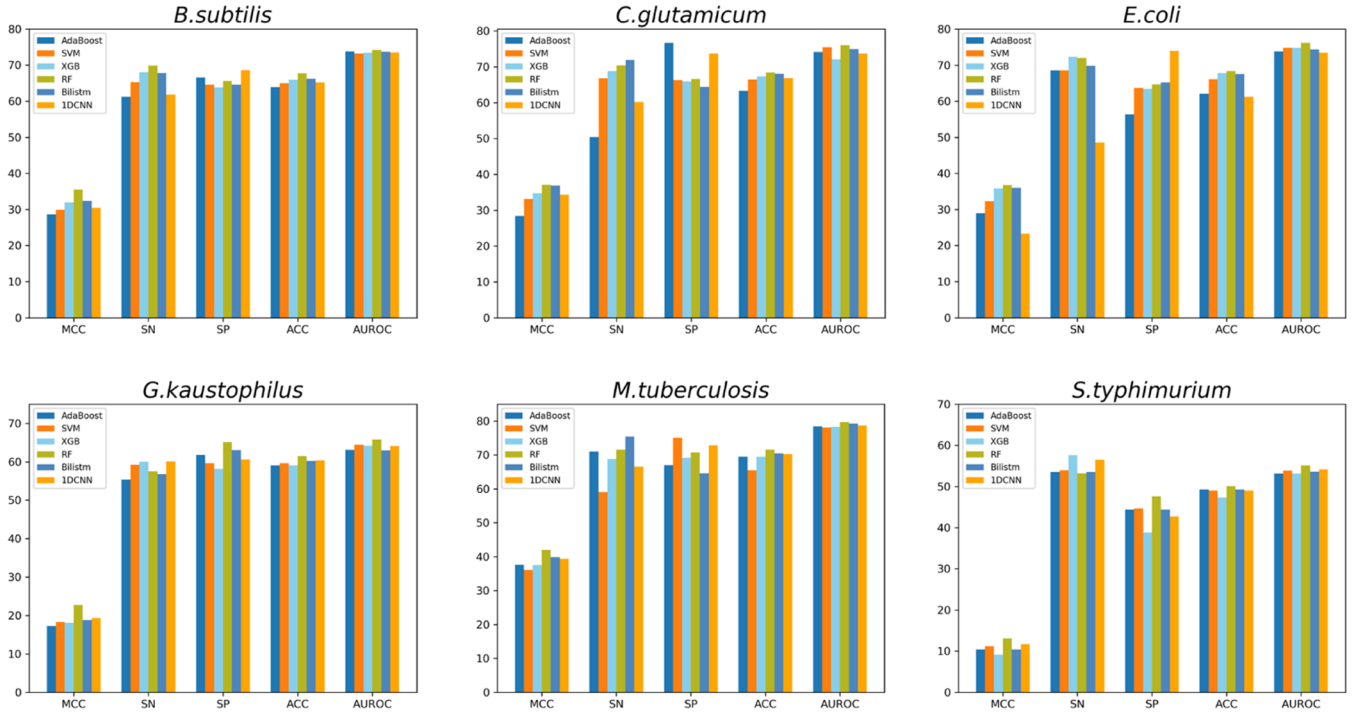


Figure 3. Fivefold cross-validation results of different models on the training sets.

Table 4. Performance of MSTL-Kace on Independent Test Sets of Different Species

species	MCC	SN	SP	ACC	AUROC	AUPRC
B. subtilis	0.31	0.700	0.765	0.76	0.804	0.270
C. glutamicum	0.347	0.759	0.777	0.775	0.829	0.338
E. coli	0.416	0.712	0.771	0.759	0.811	0.526
G. kaustophilus	0.379	0.765	0.818	0.813	0.796	0.290
M. tuberculosis	0.339	0.750	0.74	0.742	0.798	0.385
S. typhimurium	0.407	0.300	0.991	0.960	0.705	0.207

Table 5. Comparison of Fivefold Cross-Validation Results between One-Stage Fine-Tuned (OSF) and Two-Stage Fine-Tuned (TSF) Models on *G. kaustophilus*, *M. tuberculosis*, and *S. typhimurium*

species[a]	MCC	SN	SP	ACC	AUROC
G. kaustophilus (OSF)	0.165	**0.605**	0.555	0.568	0.605
G. kaustophilus (TSF)	**0.173**	0.603	**0.570**	**0.584**	**0.639**
M. tuberculosis (OSF)	0.415	0.711	0.702	0.706	0.776
M. tuberculosis (TSF)	**0.495**	**0.751**	**0.744**	**0.747**	**0.821**
S. typhimurium (OSF)	**0.105**	**0.577**	**0.527**	**0.545**	**0.601**
S. typhimurium (TSF)	0.098	0.526	0.572	0.544	0.555

[a]OSF: one-stage fine-tuning; TSF: two-stage fine-tuning.

The corresponding figures obtained by using different classifiers are presented in Figure S4, and they show the same trend as Figure 2.

Then, we utilized six classifiers mentioned above to build models based on the mean fused embedding features of the last 6 encoder layers. Totally, we built 36 models for predicting the Kace sites. The cross-validation results (Tables S9−S14) of these models on the six species are shown in Figure 3. The results indicate that the models built by using RF show the best

https://doi.org/10.1021/acsomega.3c07086
ACS Omega 2023, 8, 41930−41942

Table 6. Comparison of the Results on the Test Set between One-Stage and Two-Stage Fine-Tuned Model on *G. kaustophilus*, *M. tuberculosis*, and *S. typhimurium*

species[a]	MCC	SN	SP	ACC	AUROC	AUPRC
G. kaustophilus (OSF)	0.379	**0.765**	0.818	0.813	0.796	0.290
G. kaustophilus (TSF)	**0.404**	0.706	**0.865**	**0.852**	**0.8405**	**0.373**
M. tuberculosis (OSF)	0.339	**0.75**	0.74	0.742	0.798	0.385
M. tuberculosis (TSF)	**0.442**	0.735	**0.845**	**0.828**	**0.834**	**0.474**
S. typhimurium (OSF)	**0.407**	0.300	**0.991**	**0.960**	0.705	**0.207**
S. typhimurium (TSF)	0.186	**0.400**	0.894	0.872	**0.720**	0.121

[a]OSF: one-stage fine-tuning; TSF: two-stage fine-tuning.

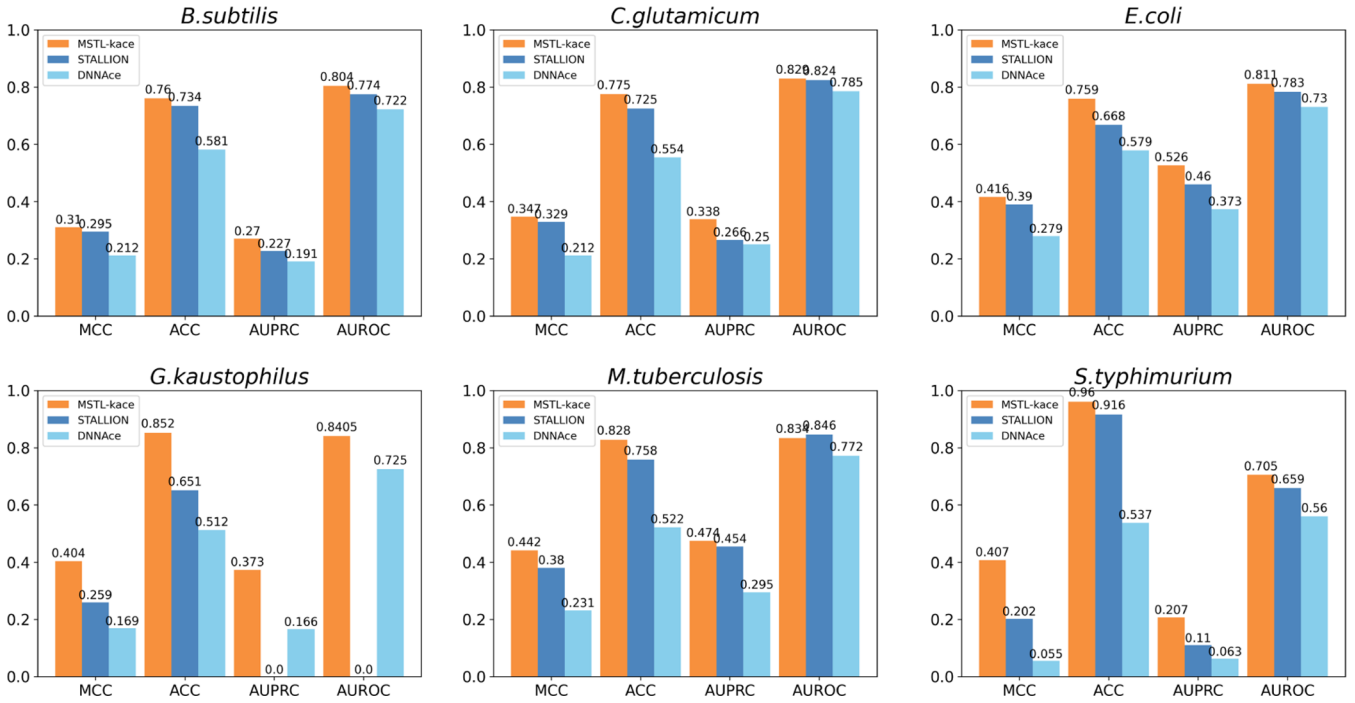


Figure 4. Performance comparison between MSTL-Kace and other methods for predicting Kace sites on independent test sets.

performance for all six species. The values of AUROC are 0.742, 0.760, 0.762, 0.658, 0.797, and 0.629 for *B. subtilis*, *C. glutamicum*, *E. coli*, *G. kaustophilus*, *M. tuberculosis*, and *S. typhimurium*, respectively. In addition, the values of MCC and ACC of the models built by using random forest are also higher than those of other classification methods. Based on the results of the fivefold cross-validation, we choose random forest as the classifier to build our models. Moreover, the performance of the models built based on the mean fused features by using RF is also significantly better than the performance of the fine-tuned models according to Figure 3 and Table 2.

These models were further evaluated on independent test sets of the six species. Table 4 shows that the AUROCs on the independent test sets are 0.804, 0.829, 0.811, 0.796, 0.798, and 0.705 for *B. subtilis*, *C. glutamicum*, *E. coli*, *G. kaustophilus*, *M. tuberculosis*, and *S. typhimurium*, respectively. And the AUPRCs are 0.270, 0.338, 0.526, 0.290, 0.385, and 0.207 for the six species, respectively.

3.4. Improve the Model Performance via Two-Stage Fine-Tuning. In this work, there are three species, *G. kaustophilus*, *M. tuberculosis*, and *S. typhimurium*, whose positive sample sizes of training data sets are less than 1000. For this small sample size issue, a previous study[22] shows that the performance of the model can be improved by introducing an intermediate model. In the case of small sample data, single-stage transfer learning may be limited by insufficient data. However, employing a multistage transfer learning strategy allows for gradual knowledge and feature transfer, thereby progressively improving model performance and generalization ability. In our work, we chose to use the fine-tuned model of *E. coli* as the intermediate model because the data set for *E. coli* in this experiment is the largest. Specifically, the two-stage fine-tuning strategy involves first fine-tuning the pretrained BERT model on the data set of *E. coli* to obtain an intermediate model and then fine-tuning the intermediate model on the data sets of the other three species. Then, the embedding features were extracted from the fine-tuned models of the second stage.

Table 5 shows the cross-validation results of the three species on the training data sets. Compared with the results in Figure 3, the table demonstrates a significant improvement in performance for *G. kaustophilus* and *M. tuberculosis*. Table 6 shows the predictive results for the independent test sets of the three species. Compared to the results in Table 4, the results demonstrate a better performance, particularly for *G. kaustophilus* and *M. tuberculosis*. We analyzed the possible reasons for our results. Related studies[37−41] have revealed partial similarities between *G. kaustophilus*, *M. tuberculosis*, and *E. coli* in terms of their growth conditions, metabolism, and molecular mechanisms. For example, both bacteria can grow

https://doi.org/10.1021/acsomega.3c07086
ACS Omega 2023, 8, 41930−41942

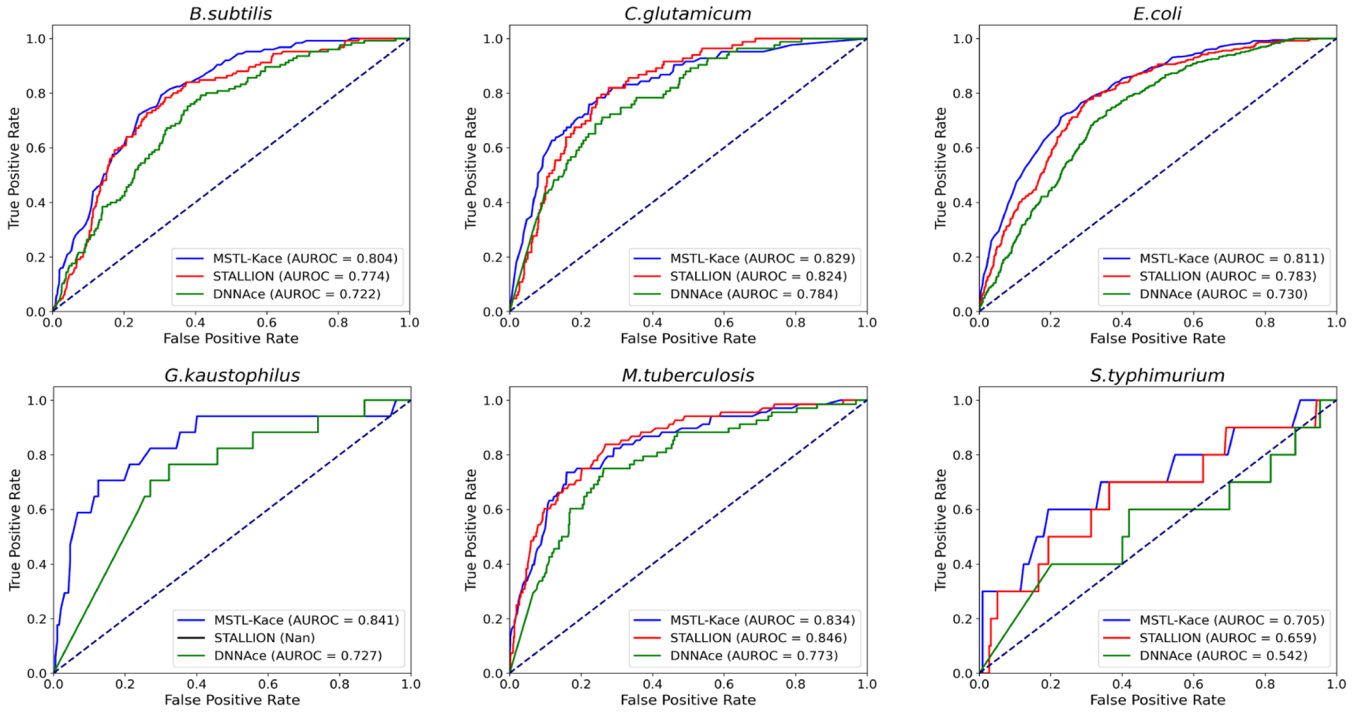


Figure 5. ROC curves generated by MSTL-Kace and other methods for the independent test sets.

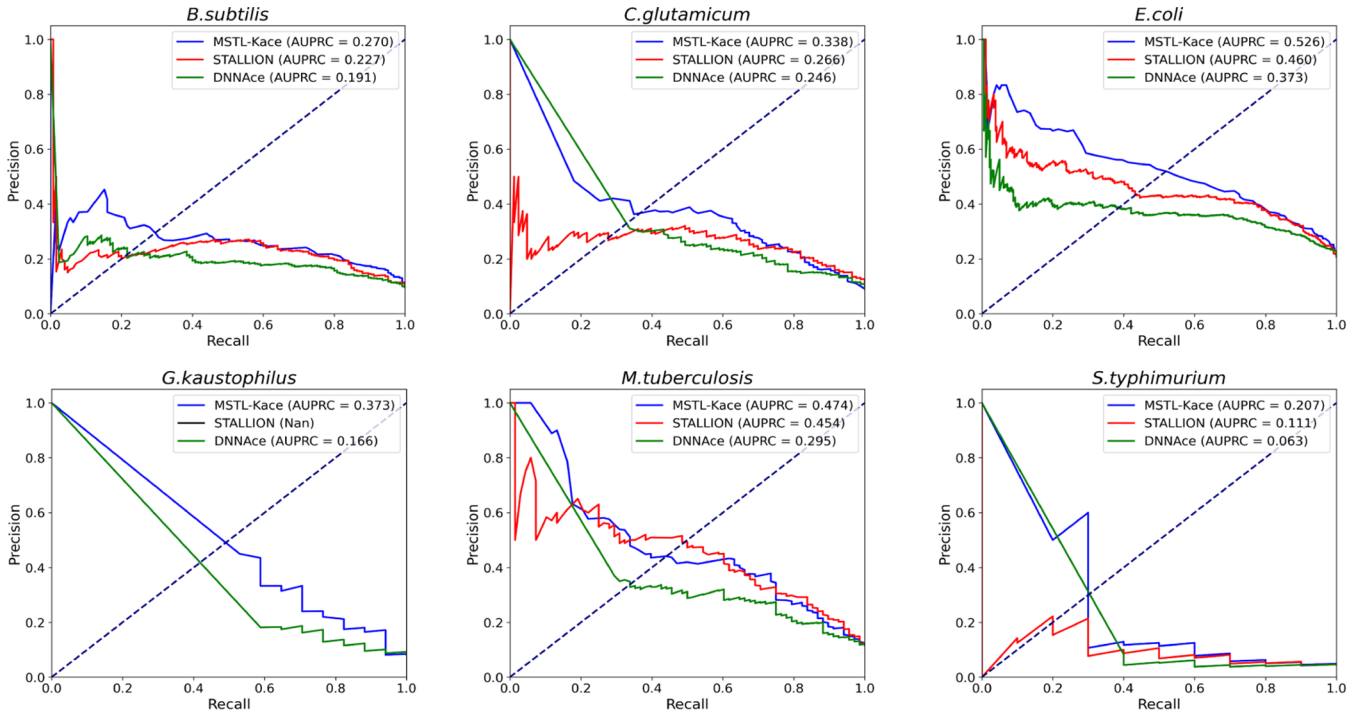


Figure 6. Precision-recall curves generated by MSTL-Kace and other methods for the independent test sets.

under aerobic conditions and have similar metabolic pathways for energy production. There are also some similarities in their molecular mechanisms such as the presence of mechanosensitive channels. In addition to the similarities in growth conditions, metabolism, and molecular mechanisms mentioned earlier, there are also some homologies and similarities between the two species, in terms of enzymes and proteins. For example, the architecture of the *E. coli bd* oxidases is highly similar to that of *G. kaustophilus*[42] and both *M. tuberculosis* and *E. coli* possess homologues of the enzyme isocitrate lyase, which is involved in the glyoxylate cycle.[43] Moreover, the two species have similar ribosomal proteins and DNA replication proteins.[44,45] These similarities may contribute to the transferability of the MSTL-Kace model between the two species.

3.5. Comparison with Other Prediction Tools on the Independent Test Sets. To show the superiority of our model, we compared our model with the other two state-of-the-art models, STALLION and DNNAce, on the same

https://doi.org/10.1021/acsomega.3c07086
ACS Omega 2023, 8, 41930−41942

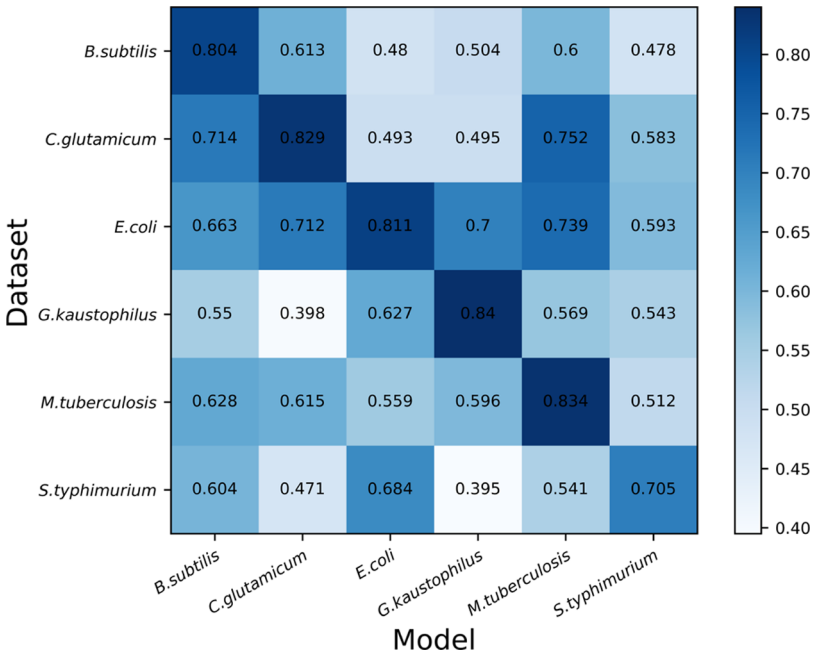


Figure 7. Heatmap showing the cross-species predictive AUROC.

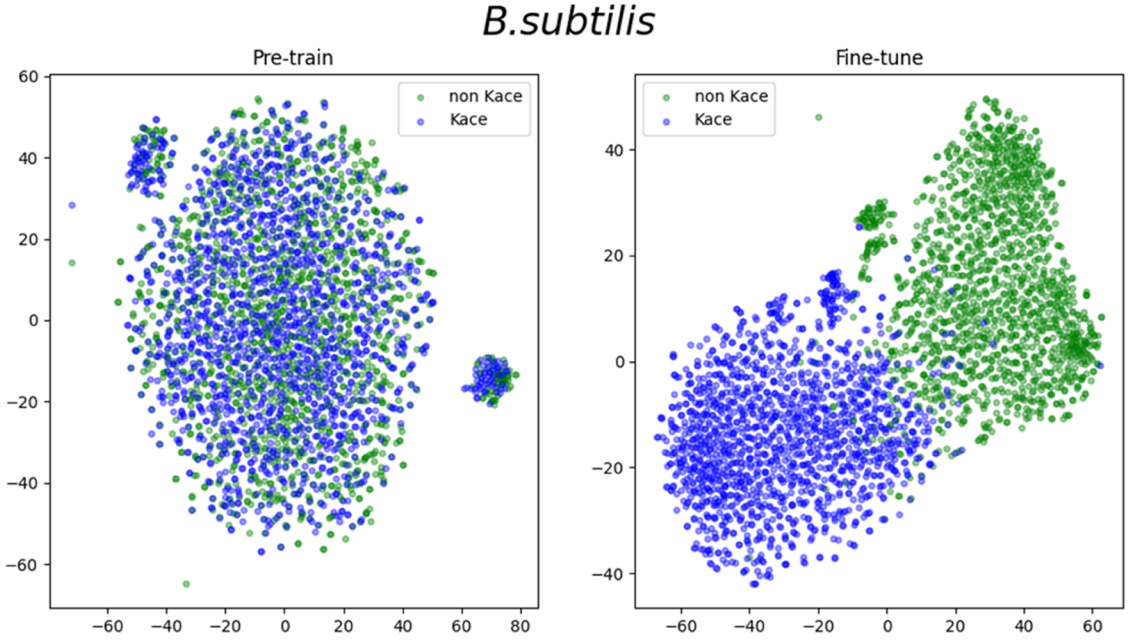


Figure 8. t-SNE illustration of the embedding features extracted from pretrain BERT and fine-tuned BERT.

independent test sets. STALLION is an advanced machine learning method based on the stacking integration strategy, which is superior to ProAcePred2.0 in the comparison of its experimental results. DNNAce is a predictor developed using deep learning and is based on the data used in ProAcePred. To compare fairly, we used the same training set as that in their paper to rebuild the DNNAce. The predictive results of STALLION, DNNAce, and our model are shown in Figure 4. By using MCC, ACC, AUPRC, and AUROC as the main evaluation metrics, for *B. subtilis*, MSTL-Kace obtained the highest MCC, ACC, AUPRC, and AUROC of 0.31, 0.76, 0.27, and 0.804, respectively. More specifically, AUROC of MSTL-Kace is 3% higher than STALLION and 8.2% higher than DNNAce. AUPRC of MSTL-Kace is 4.3% higher than STALLION and 7.9% higher than DNNAce. For *C. glutamicum*, the values of MCC, ACC, AUPRC, and AUROC were 0.347, 0.775, 0.338, and 0.829, respectively. For *E. coli*, the ACC of MSTL-Kace was 0.759 which is 9.1% higher than that of STALLION. It is also slightly higher than other methods in AUPRC and AUROC. Because of the disfunction of STALLION web server, we cannot get the predictive results and calculate AUPRC and AUROC for *G. kaustophilus*. So, the evaluation metrics reported in their paper were used here. According to other metrics, our model achieved an MCC of 0.379, which is 12% higher than that of STALLION, and an ACC of 0.813, which is 16.2% higher than that of STALLION. For *M. tuberculosis*, MSTL-Kace performed lower than STALLION by 1.2% in terms of

https://doi.org/10.1021/acsomega.3c07086
ACS Omega 2023, 8, 41930−41942

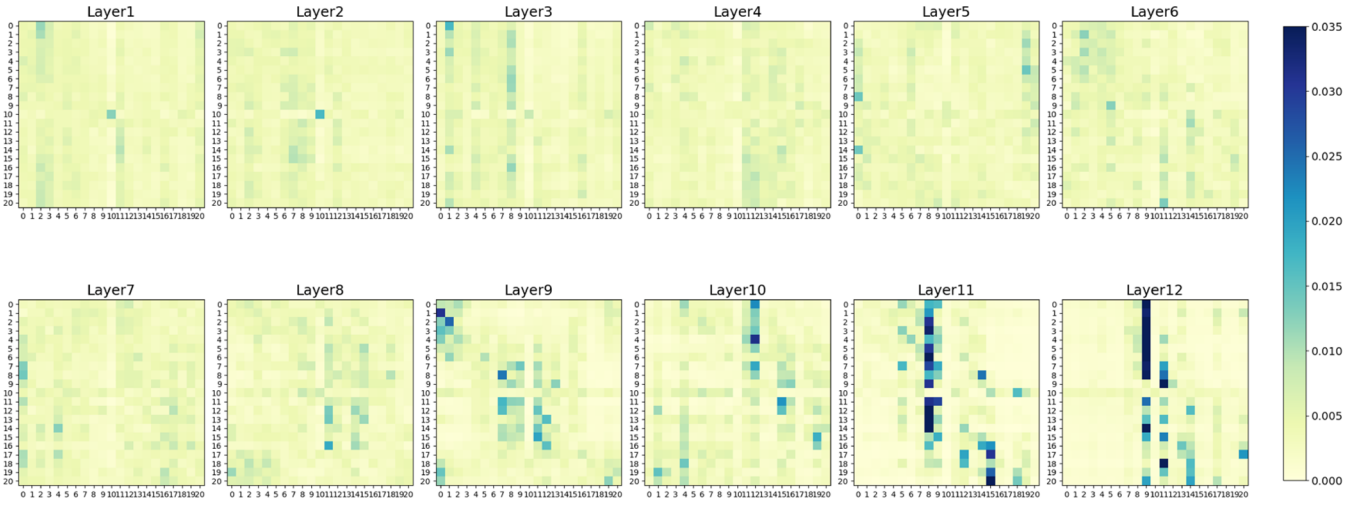


Figure 9. Attention weights of different positions in 12 encoder layers.

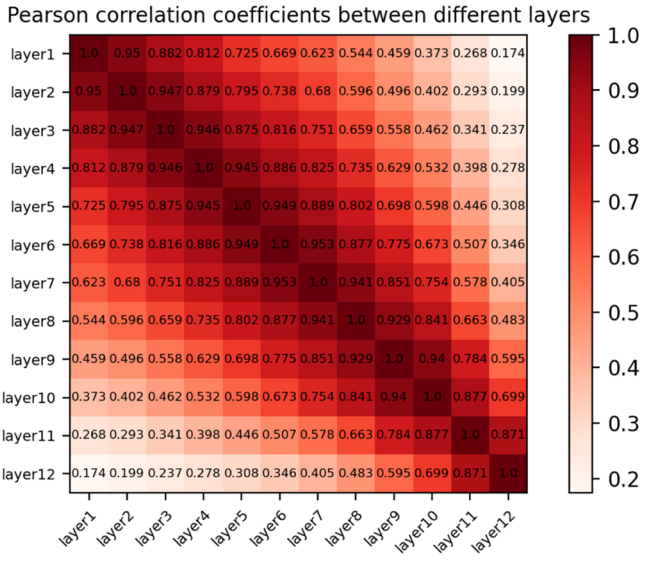


Figure 10. Pearson correlation coefficient between embedding feature of 12 encoder layers.

AUROC, while outperforming STALLION in all other metrics. This is the only metric in which MSTL-Kace falls behind STALLION. For *S. typhimurium*, as the result shows in the figure, MSTL-Kace also outperforms the other two methods in various metrics, especially in MCC and AUPRC, with improvements of 20.5 and 9.6% compared to STALLION, and 35.5 and 14.4% compared to DNNAce, respectively. Summing up the above results, MSTL-Kace can achieve an excellent performance more than STALLION and DNNAce. The receiver operating characteristic curve and the precision-recall curve are shown in Figures 5 and 6.

3.6. Cross-Species Evaluation. In this study, we ultimately established six species-specific models. It is interesting to test the specificities of these models to evaluate if it is possible to build a generic model for predicting Kace sites for prokaryotes. Thus, we conducted cross-species prediction on the six independent test sets by using six species-specific models. A heatmap (Figure 7) was plotted to show the AUROC values of these models on the six test sets. It is clear that the highest AUROC values are obtained at the diagonal positions, which indicates that the species-specific predictive results are significantly better than the cross-species predictive results. According to our results, it is better to build species-specific models to predict Kace sites for different species.

3.7. Visualization of Embeddings of Pretrained and Fine-Tuned BERT Models. Fine-tuning is a common technique in transfer learning.[46] Transfer learning is a method of identifying and applying knowledge and skills learned in a previous domain or task to a novel domain or task. Researchers usually pretrain a model on a data-rich task first and then fine-tune it for downstream tasks. In this experiment, we have pretrained a protein language model based on 10 484 positive samples and 10 484 negative samples. The pretrained model was fine-tuned to build models for predicting Kace of different species based on the data for different species. To observe the difference of the embeddings obtained between the pretrained and the fine-tuned models, we use t-SNE[47] to visualize the features learned from the data. Figure 8 shows the results for *B. subtilis*, and it is obvious that there is a clear distinction after fine-tuning. The t-SNE figures for other species are shown in Figure S5.

3.8. Visualization of Attention Weights between Different Layers. The present study is based on the BERT-Base model, which consists of 12 encoder layers. The embedding features of different encoder layers represent information at different levels of natural language. The combination of the embedding features from different encoder layers might be beneficial for the performance of the models. Our results indicate the model based on the mean fusion features of the last six encoder layers achieves the best performance.

To explain these results, we analyzed the attention weights of different encoder layers by using the data of *B. subtilis* as example, and other species in Figure S6. The BERT model uses a multihead self-attention mechanism to calculate the attention weights for each position,[48] representing the importance of that position to the current query. We calculated the attention weights for each position of the positive samples in the training set and plotted the average attention heatmap.

As shown in Figure 9, the attention in the shallow layers of the BERT model is relatively scattered and the attention weights is low. With the increase of the model layers, the attention weights increase and become concentrated near

https://doi.org/10.1021/acsomega.3c07086
ACS Omega 2023, 8, 41930−41942

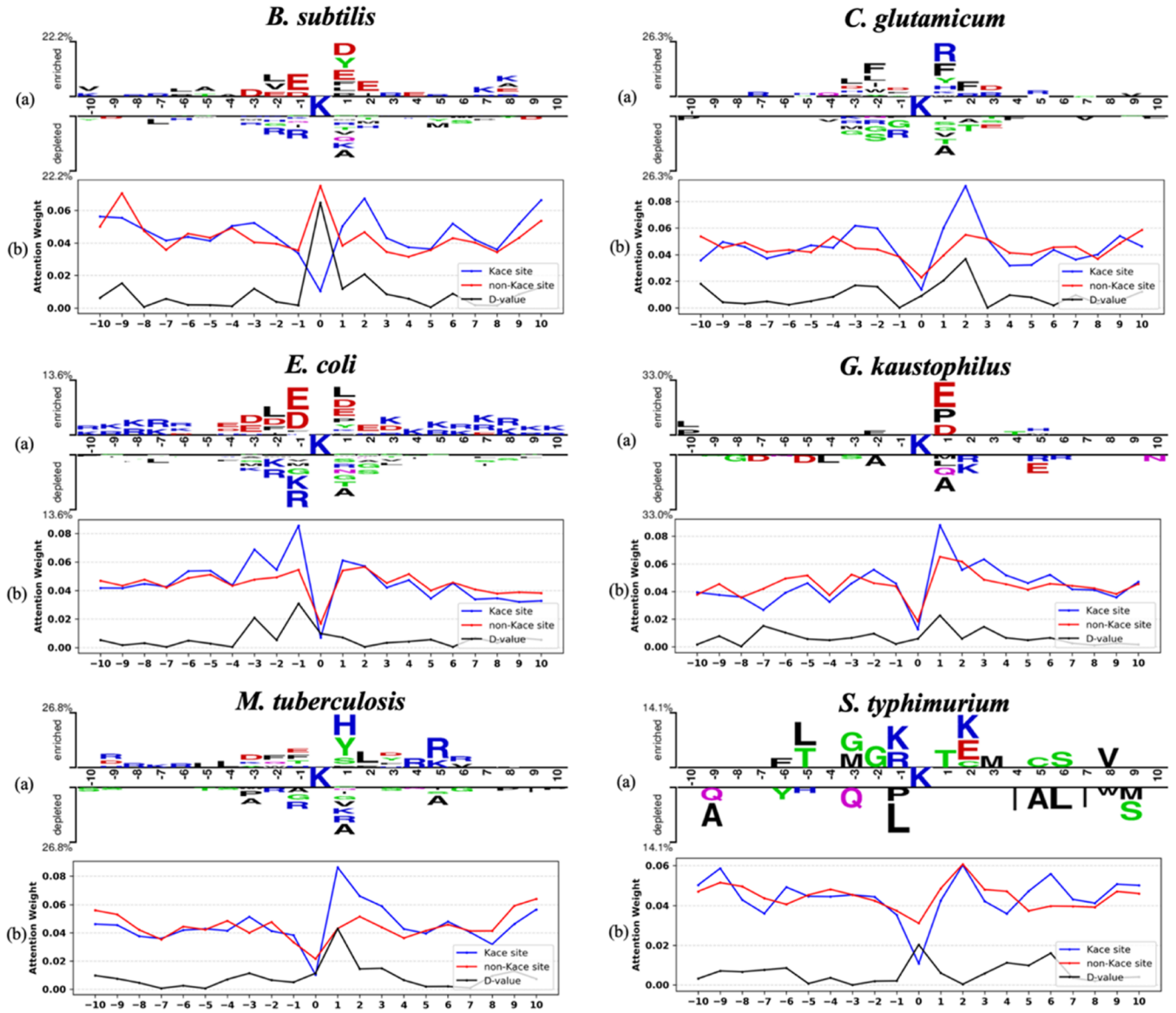


Figure 11. (a) Comparison of the amino acid composition between positive and native samples. (b) Differences in the averaged position weights between the positive and negative samples.

position 10, where the Kace site is located. In layer 12, positions 9, 10, and 11 receive the highest attention weights, indicating that the attention mechanism of the BERT model can identify the interaction between the Kace site and other amino acid residues in the protein sequence.

When the embedding features of the last layer alone are used, the overfitting caused by the highest attention weights may lead to poor results on the validation set. Furthermore, when features from all layers are fused, the redundant information contained in the shallow features can also affect the prediction results. From the attention heatmap, it can be seen that the last six layers contain higher attention weights. Therefore, selecting the embedding features from the last six layers for fusion is a better choice.

Furthermore, we performed Pearson correlation analysis on the embedding features of all 12 encoder layers. As shown in Figure 10, it can be seen that the correlation coefficients are high between adjacent layers, but the correlation coefficient is low between distant layers, which indicates the complementarity between different embedding features obtained from different layers.

3.9. Sequence and Attention Mechanism Analysis.

The results in Figure 4 show that MSTL-Kace outperforms STALLION and DNNAce on all species. MSTL-Kace used fewer types of features and provided better performance than STALLION and DNNAce, which implies that embeddings extracted from BERT are relevant to the sequence information. This demonstrates that the attention mechanism of the BERT model is capable of focusing the partial information on the input. The attention weights of the token "CLS" have been used to demonstrate the importance of different positions.[23] So, we calculated the average "CLS" attention weights of different protein sequence positions and analyzed the relationship between sequence motifs[49,50] and attention weights obtained by BERT.[48]

As shown in Figure 11, we take the intermediate residue K as the origin, and Figure 11(a) shows the proportion of amino acid types in the protein sequence. We visualized the attention maps in Figure 11(b), which shows the attention weights of

https://doi.org/10.1021/acsomega.3c07086
ACS Omega 2023, 8, 41930−41942

token "CLS" in sequence positions. The blue polyline represents the fraction of the Kace sequence, the red polyline represents the fraction of the non-Kace sequence, and the black represents the absolute value of the difference between these two fractions. What is evident is the significant difference in attention weights between positive and negative samples near the center position. For example, there is a significant difference in attention weights between *G. kaustophilus* and *M. tuberculosis* at position 1. In *G. kaustophilus*, the positive samples show a higher probability of residue E, P, and D, while the negative samples have a higher probability of A, L, and Q. Similarly, in *M. tuberculosis*, the positive samples exhibit a higher probability of residue H, Y, S, while the negative samples have a higher probability of K, R, and A. Similar observations were made in other species, where positions with larger attention weight differences also exhibited substantial differences in residue distribution.

These results indicate that MSTL-Kace can effectively identify the interactions between Kace sites and other amino acids in the sequence and assign higher attention to them. This could assist biologists in gaining further insights into the mechanisms and physicochemical information about lysine acetylation.

3.10. Web Server. As demonstrated above, we proposed a model, MSTL-Kace, which has outstanding performance in predicting lysine acetylation sites. For the convenience of researchers, we offered the web service for the model, which allows site prediction by simply entering the protein sequence. The address of this web server is: http://MSTL-Kace.zhulab.org.cn/. Hopefully, these things may provide thoughts and help relevant researchers.

4. CONCLUSIONS

In this article, we proposed a novel method to predict the Kace sites of prokaryotes. By using the protein sequences from different species, we pretrained a domain-specific BERT model. After fine-tuning the model with species-specific sequences, we extracted the sequential embeddings from the model, which were used to build the models for different species based on different downstream learning algorithms. The results of the independent test sets show that our models outperform the other state-of-the-art methods for all six species. Through multistage transfer learning, the model can progressively transfer knowledge and features, thereby enhancing predictive performance on small sample data. This strategy helps alleviate the limitations caused by data scarcity and provides an improved generalization ability and robustness. By analyzing the attention weights of the fine-tuned model, we demonstrated that the sequence patterns can partially be obtained by the BERT model. For the convenience of related researchers, we provided a stable prediction web site service to speed up the identification of Kace sites for other researchers.

In the process of this study, we found that MSTL-Kace still needs improvement. We will introduce contrastive learning and multitask learning to improve our model.

■ ASSOCIATED CONTENT

Data Availability Statement

The source codes and data for MSTL-Kace are available at https://github.com/leo97king/MSTL-Kace.

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acsomega.3c07086.

Details of classifier; hyperparameters used (Tables S1 and S2); cross-validation results based on fused embedding features of different layers (Tables S3−S8); cross-validation results of models built by using different learning algorithms (Table S9−S14); model structure of BERT (Figure S1); structure diagram of BiLSTM (Figure S2); structure diagram of 1D-CNN (Figure S3); performance of fusion features from different encoding layers by using different learning algorithms (Figure S4); t-SNE illustrations of the embedding features extracted from pretrain BERT and fine-tuned BERT (Figure S5); and attention weights heatmaps in 12 encoder layers for different species (Figure S6) (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Xiaolei Zhu − *School of Sciences, Anhui Agricultural University, Hefei 230036 Anhui, China;* orcid.org/0000-0002-1967-2806; Email: xlzhu_mdl@hotmail.com

Authors

Gang-Ao Wang − *School of Sciences, Anhui Agricultural University, Hefei 230036 Anhui, China*

Xiaodi Yan − *School of Sciences, Anhui Agricultural University, Hefei 230036 Anhui, China*

Xiang Li − *School of Sciences, Anhui Agricultural University, Hefei 230036 Anhui, China*

Yinbo Liu − *School of Sciences, Anhui Agricultural University, Hefei 230036 Anhui, China*

Junfeng Xia − *Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Institutes of Physical Science and Information Technology, Anhui University, Hefei 230601 Anhui, China;* orcid.org/0000-0003-3024-1705

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.3c07086

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This work was supported by a grant from National Natural Science Foundation of China (21403002, U22A2038), the National Key Research and Development Program of China (2020YFA0908700), and the Young Wanjiang Scholar Program of Anhui Province.

■ REFERENCES

(1) Soffer, R. L. Post-translational modification of proteins catalyzed by aminoacyl-tRNA-protein transferases. *Mol. Cell. Biochem.* **1973**, *2* (1), 3−14.

(2) Wold, F. In vivo chemical modification of proteins (post-translational modification). *Annu. Rev. Biochem.* **1981**, *50*, 783−814.

(3) Grotenbreg, G.; Ploegh, H. Chemical biology: dressed-up proteins. *Nature* **2007**, *446* (7139), 993−995.

(4) Bradner, J. E.; West, N.; Grachan, M. L.; Greenberg, E. F.; Haggarty, S. J.; Warnow, T.; Mazitschek, R. Chemical phylogenetics of histone deacetylases. *Nat. Chem. Biol.* **2010**, *6* (3), 238−243.

(5) Basith, S.; Chang, H. J.; Nithiyanandam, S.; Shin, T. H.; Manavalan, B.; Lee, G. Recent Trends on the Development of Machine Learning Approaches for the Prediction of Lysine Acetylation Sites. *Curr. Med. Chem.* **2022**, *29*, 235−250, DOI: 10.2174/0929867328999210902125308.
(6) Suo, S.-B.; Qiu, J.-D.; Shi, S.-P.; Sun, X.-Y.; Huang, S.-Y.; Chen, X.; Liang, R.-P. Position-Specific Analysis and Prediction for Protein Lysine Acetylation Based on Multiple Features. *PLoS One* **2012**, *7* (11), No. e49108.
(7) Shao, J.; Xu, D.; Hu, L.; Kwan, Y. W.; Wang, Y.; Kong, X.; Ngai, S. M. Systematic analysis of human lysine acetylation proteins and accurate prediction of human lysine acetylation through bi-relative adapted binomial score Bayes feature representation. *Mol. Biosyst.* **2012**, *8* (11), 2964−2973.
(8) Shi, S.-P.; Qiu, J.-D.; Sun, X.-Y.; Suo, S.-B.; Huang, S.-Y.; Liang, R.-P. PLMLA: prediction of lysine methylation and lysine acetylation by combining multiple features. *Mol. BioSyst.* **2012**, *8* (5), 1520−1527.
(9) Suo, S.-B.; Qiu, J.-D.; Shi, S.-P.; Chen, X.; Huang, S.-Y.; Liang, R.-P. Proteome-wide analysis of amino acid variations that influence protein lysine acetylation. *J. Proteome Res.* **2013**, *12* (2), 949−958.
(10) Li, Y.; Wang, M.; Wang, H.; Tan, H.; Zhang, Z.; Webb, G. I.; Song, J. Accurate in silico identification of species-specific acetylation sites by integrating protein sequence-derived and functional features. *Sci. Rep.* **2014**, *4*, No. 5765, DOI: 10.1038/srep05765.
(11) Lu, C.-T.; Lee, T.-Y.; Chen, Y.-J.; Chen, Y.-J. An intelligent system for identifying acetylated lysine on histones and nonhistone proteins. *BioMed Res. Int.* **2014**, *2014*, No. 528650.
(12) Chen, G.; Cao, M.; Luo, K.; Wang, L.; Wen, P.; Shi, S. ProAcePred: prokaryote lysine acetylation sites prediction based on elastic net feature optimization. *Bioinformatics* **2018**, *34* (23), 3999−4006.
(13) Chen, G.; Cao, M.; Yu, J.; Guo, X.; Shi, S. Prediction and functional analysis of prokaryote lysine acetylation site by incorporating six types of features into Chou's general PseAAC. *J. Theor. Biol.* **2019**, *461*, 92−101.
(14) Wang, H. Q.; Yan, Z. L.; Liu, D.; Zhao, H.; Zhao, J. MDC-Kace: A Model for Predicting Lysine Acetylation Sites Based on Modular Densely Connected Convolutional Networks. *IEEE Access* **2020**, *8*, 214469−214480.
(15) Basith, S.; Lee, G.; Manavalan, B. STALLION: a stacking-based ensemble learning framework for prokaryotic lysine acetylation site prediction. *Briefings Bioinf.* **2022**, No. bbab376, DOI: 10.1093/bib/bbab376.
(16) Yu, B.; Yu, Z.; Chen, C.; Ma, A.; Liu, B.; Tian, B.; Ma, Q. DNNAce: Prediction of prokaryote lysine acetylation sites through deep neural networks with multi-information fusion. *Chemom. Intell. Lab. Syst.* **2020**, *200*, No. 103999, DOI: 10.1016/j.chemolab.2020.103999.
(17) Zhou, X.; Zhao, H.; Jiang, T. Adaptive analysis of optical fringe patterns using ensemble empirical mode decomposition algorithm. *Opt. Lett.* **2009**, *34* (13), 2033−2035.
(18) Wolpert, D. H. Stacked Generalization. *Neural Networks* **1992**, *5* (2), 241−259.
(19) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; Polosukhin, I.Attention Is All You Need *Adv. Neural inf. Pocessing Syst.* 2017; Vol. *30*.
(20) Devlin, J.; Chang, M. W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, Minnesota*; Association for Computational Linguistics: Minneapolis, MN, 2019; pp 4171−4186.
(21) Zhang, Y.; Zhu, G.; Li, K.; Li, F.; Huang, L.; Duan, M.; Zhou, F. HLAB: learning the BiLSTM features from the ProtBert-encoded proteins for the class I HLA-peptide binding prediction. *Briefings Bioinf.* **2022**, *23* (5), No. bbac173.
(22) Tsukiyama, S.; Hasan, M. M.; Deng, H.-W.; Kurata, H. BERT6mA: prediction of DNA N6-methyladenine site using deep learning-based approaches. *Briefings Bioinf.* **2022**, *23* (2), No. bbac053.
(23) Qiao, Y.; Zhu, X.; Gong, H. BERT-Kcr: prediction of lysine crotonylation sites by a transfer learning method with pre-trained BERT models. *Bioinformatics* **2022**, *38* (3), 648−654.
(24) Liu, Y.; Liu, Y.; Wang, G. A.; Cheng, Y.; Bi, S.; Zhu, X. BERT-Kgly: A Bidirectional Encoder Representations From Transformers (BERT)-Based Model for Predicting Lysine Glycation Site for Homo sapiens. *Front. Bioinf.* **2022**, *2*, No. 834153.
(25) Xu, H.; Zhou, J.; Lin, S.; Deng, W.; Zhang, Y.; Xue, Y. PLMD: An updated data resource of protein lysine modifications. *J. Genet. Genomics* **2017**, *44* (5), 243−250.
(26) Huang, Y.; Niu, B. F.; Gao, Y.; Fu, L. M.; Li, W. Z. CD-HIT Suite: a web server for clustering and comparing biological sequences. *Bioinformatics* **2010**, *26* (5), 680−682.
(27) Ba, J. L.; Kiros, J. R.; Hinton, G. E.Layer Normalization *arXiv preprint arXiv:1607.06450* 2016.
(28) Li, Z.; Fang, J.; Wang, S.; Zhang, L.; Chen, Y.; Pian, C. Adapt-Kcr: a novel deep learning framework for accurate prediction of lysine crotonylation sites based on learning embedding features and attention architecture. *Brief Bioinf.* **2022**, *23* (2), No. bbac037.
(29) Wang, X.; Ding, Z.; Wang, R.; Lin, X. Deepro-Glu: combination of convolutional neural network and Bi-LSTM models using ProtBert and handcrafted features to identify lysine glutarylation sites. *Briefings Bioinf.* **2023**, *24* (2), No. bbac631.
(30) Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45* (1), 5−32.
(31) Freund, Y.; Schapire, R. E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* **1997**, *55* (1), 119−139.
(32) Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System, KDD '16. In *The 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016−08−13*; ACM, 2016; pp 785−794.
(33) Sain, S. R. The Nature of Statistical Learning Theory. *Technometrics* **1996**, *38* (4), 409.
(34) Krizhevsky, A.; Sutskever, I.; Hinton, G. E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60* (6), 84−90.
(35) Zhang, S.; Zheng, D.; Hu, X.; Yang, M. Bidirectional Long Short-Term Memory Networks for Relation Classification. In *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation* 2015; pp 73−78.
(36) Yeung, W.; Zhou, Z.; Li, S.; Kannan, N. Alignment-free estimation of sequence conservation for identifying functional sites using protein sequence embeddings. *Briefings Bioinf.* **2023**, *24* (1), No. bbac599, DOI: 10.1093/bib/bbac599.
(37) Takami, H.; Takaki, Y.; Chee, G.-J.; Nishi, S.; Shimamura, S.; Suzuki, H.; Matsui, S.; Uchiyama, I. Thermoadaptation trait revealed by the genome sequence of thermophilic Geobacillus kaustophilus. *Nucleic Acids Res.* **2004**, *32* (21), 6292−6303.
(38) Fu, L. M.; Fu-Liu, C. S. Is Mycobacterium tuberculosis a closer relative to Gram-positive or Gram−negative bacterial pathogens? *Tuberculosis* **2002**, *82* (2), 85−90.
(39) Gordon, S. V.; Parish, T. Microbe Profile: Mycobacterium tuberculosis: Humanity's deadly microbial foe: This article is part of the Microbe Profiles collection. *Microbiology* **2018**, *164* (4), 437−439.
(40) Tenaillon, O.; Skurnik, D.; Picard, B.; Denamur, E. The population genetics of commensal *Escherichia coli*. *Nat. Rev. Microbiol.* **2010**, *8* (3), 207−217.
(41) Singleton, P. *Bacteria in Biology, Biotechnology, and Medicine*; John Wiley & Sons, 2004.
(42) Theßeling, A.; Rasmussen, T.; Burschel, S.; Wohlwend, D.; Kagi, J.; Muller, R.; Bottcher, B.; Friedrich, T. Homologous bd oxidases share the same architecture but differ in mechanism. *Nat. Commun.* **2019**, *10* (1), No. 5138.
(43) Maurer, J. A.; Elmore, D. E.; Lester, H. A.; Dougherty, D. A. Comparing and Contrasting *Escherichia coli*and Mycobacterium

tuberculosis Mechanosensitive Channels (MscL). *J. Biol. Chem.* **2000**, *275* (29), 22238−22244.
(44) Huang, H.-L.; Su, H.-T.; Wu, C.-H. H.; Tsai-Wu, J.-J. A Molecular Biological and Biochemical Investigation on Mycobacterium tuberculosis MutT Protein. *Jundishapur J. Microbiol.* **2014**, *7* (3), No. e9367.
(45) Kumar, P.; Krishna, K.; Srinivasan, R.; Ajitkumar, P.; Varshney, U. Mycobacterium tuberculosis and *Escherichia coli* nucleoside diphosphate kinases lack multifunctional activities to process uracil containing DNA. *DNA Repair* **2004**, *3* (11), 1483−1492.
(46) Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? *Adv. Neural Inf. Process. Syst.* 2014; Vol. *27*.
(47) van der Maaten, L.; Hinton, G. Visualizing Data using t-SNE *J. Mach. Learn. Res.* 2008; Vol. *2008*.
(48) Clark, K.; Khandelwal, U.; Levy, O.; Manning, C. D. What does bert look at? an analysis of bert's attention *arXiv preprint arXiv:1906.04341* 2019.
(49) O'Shea, J. P.; Chou, M. F.; Quader, S. A.; Ryan, J. K.; Church, G. M.; Schwartz, D. pLogo: a probabilistic approach to visualizing sequence motifs. *Nat. Methods* **2013**, *10* (12), 1211−1212.
(50) Wu, X.; Bartel, D. P. kpLogo: positional k-mer analysis reveals hidden specificity in biological sequences. *Nucleic Acids Res.* **2017**, *45* (W1), W534−w538.

https://doi.org/10.1021/acsomega.3c07086
ACS Omega 2023, 8, 41930−41942