R-STEINER: Generation Method of 5'UTR for Increasing the Amount of Translated Proteins

Hiroaki Tanaka[1,a) Yu Suzuki[1,b) Shotaro Yamasaki[1,c) Koichiro Yoshino[1,d) Ko Kato[1,e)
Satoshi Nakamura[1,f)

Abstract: Protein production in plants is a hot topic because there are many benefits relative to bacteria, yeasts, and animals, but the amount of protein expression in plants is less. It is argued that editing 5'UTRs increases the amount of translated proteins. However, obtaining such 5'UTRs is difficult due to the cost, time and effort required in experiments. To solve this, we predict the amount of translated proteins by machine learning. In this paper, we propose a method, named "R-STEINER", that generates 5'UTRs that increase the amount of proteins of a given gene. The proposed process involves building a model for predicting the amount of translated proteins, generating 5'UTRs, selecting them and increasing the proteins according to the model. This method enables us to obtain 5'UTRs that increase the amount of translated proteins without real synthesis experiments, resulting in reduced cost, time and effort. In our study, we built a prediction model for Oryza sativa and synthesized the 5'UTRs generated by R-STEINER. We confirmed that the model can predict the amount of translated proteins with a correlation coefficient of 0.89.

Keywords: machine learning, bioinformatics, protein production,ensemble learning, mRNA

1. Introduction

Protein production by plants is a hot topic [14]. There are options for platforms used to produce proteins, and each platform has advantages and disadvantages. We take the example of developing a vaccine. Obviously, safety is most important. If we develop a vaccine by using human cells, the vaccine has the possibility of containing matter that is harmful to human beings. However, if we use plants or plants' cells, this possibility becomes infinitesimally small [30].

The protein expression of a plant is less than that of other hosts, e.g., bacteria, yeasts and animals [10], [24], [30]. As a solution, we focus on the area 5'UTR (5'-untranslated region). It is known that gene expression is affected by 5'UTR sequence [29], and some researchers have tried to discover 5'UTR sequences that increase the amount of translated proteins of certain genes. However, real experiments done to discover the translation enhancer—referring to the 5'UTR sequences which increase the amount of translated proteins—involve significant costs and require time and effort.

As a solution, we propose a method, R-STEINER, that enable us to obtain the translation enhancers of a given gene without real experiments, resulting in reduced cost, time and effort. The proposed process is composed of the following parts: building a model for predicting the amount of translated proteins, generating 5'UTR sequences randomly and selecting those that increase the amount of proteins according to the model. The parts of sequence generation and selection involve performing the real experiment; conventionally, we could obtain the amount of translated proteins only in real experiments, but the model gives us the predicted amount of proteins.

With R-STEINER, we build a model for predicting the amount of translated proteins. We have to evaluate the model not only in the traditional way of evaluating a machine learning model but also through real synthesis experiments because the 5'UTR sequences generated in the generation part are completely artificial. As the model built in R-STEINER is learned by natural sequences, there is the possibility that it cannot predict the amount of translated proteins from artificial sequences. In addition to this concern, we are concerned that it might not be possible to synthesise some artificial sequences; presumably, 5'UTR sequences are made by some unknown rule, but sequences generated by R-STEINER are not. Given the above two concerns, we have to evaluate whether the model can predict the amount of translated proteins even for artificial 5'UTRs. For this evaluation, we performed real synthesis experiments and evaluated the model in Sect. 6.

2. Basic Knowledge

We will introduce basic knowledge to aid in understanding our study.

We show the process from a gene to proteins in Fig. 1. "Translation" indicates the process from mRNA to proteins. An mRNA consists of three areas: 5'UTR, CDS (coding DNA sequence) and

[1] Nara Institute of Science and Technology
a) tanaka.hiroaki.sy2@is.naist.jp
b) ysuzuki@is.naist.jp
c) s-yamasaki@bs.naist.jp
d) koichiro@is.naist.jp
e) kou@bs.naist.jp
f) s-nakamura@is.naist.jp

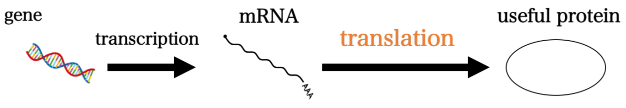

Fig. 1 Process from the gene to proteins

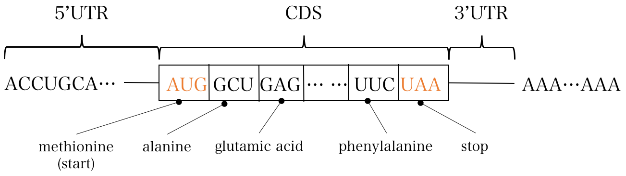

Fig. 2 Structure of mRNA

3'UTR (3'-untranslated region) (see Fig. 2). Then, the mRNA is decoded in a ribosome to produce a specific amino acid chain or protein. Strictly, only some combinations in CDS are decoded to amino acids; for example, the combination CAU is decoded to $C_6H_9N_3O_2$ (see Fig. 2). More details on translation are given in Alberts et al. [1].

In our study, we aim to make a large amount of proteins from one mRNA. If any mRNA is decoded to protein actively, many ribosomes are attached to the mRNA, and the ribosomes and mRNA form a complex called a "polysome". In other words, as the ratio of mRNAs that form polysomes increases, the amounts of proteins generated from the mRNAs increase. Therefore, we use the ratio of mRNA as the criterion of the amount of translated proteins, i.e., how many proteins are generated from an mRNA. We call the ratio the "PR-value" (polysome ratio value).

It is known that an area of 5'UTR affects the amount of translated proteins [25]. According to this report, we assume that it is possible to increase the amount of translated proteins by controlling the sequence of 5'UTR.

We have two reasons for controlling only the 5'UTR sequence. First, controlling CDS is not reasonable for applications. If we change the CDS sequence, the proteins produced by mRNA are changed. This is not desirable for applications. Second, controlling the 3'UTR sequence is almost impossible. 3'UTR contains information on where it will break. Therefore, if we try to control the 3'UTR sequence, the synthesised 3'UTR may be an unexpected sequence. For these two reasons, we do not change the CDS and 3'UTR sequences.

We introduce one more piece of basic knowledge on mRNA. mRNA usually makes a secondary structure, i.e., mRNA does not lie on a straight line but makes a complex structure (see Fig. 3). We can estimate the possible forms and free energy of them from the nucleotide sequence. Free energy indicates how strongly a nucleotide sequence makes a secondary structure—as the nucleotide sequence makes the secondary structure stronger, free energy increases. In this study, we calculated the free energy by using the ViennaRNA Package [11].

3. Related Work

In this section, we introduce related work on the relationships among features of the mRNA sequence and the amount of translated proteins.

Kawaguchi and Bailey-Serres [9] analysed the relationships between ribosome loading[*1], three features, the length of 5'UTR, CDS and 3'UTR, and the contents of A, U, G, C, AU, GC, CU, AG, GU and AC. Ribosome loading represents the amount of translated proteins. However, they analysed the relationships between only one feature and the amount of translated proteins. Therefore, they could not reveal the relationships among the amount of translated proteins and some features.

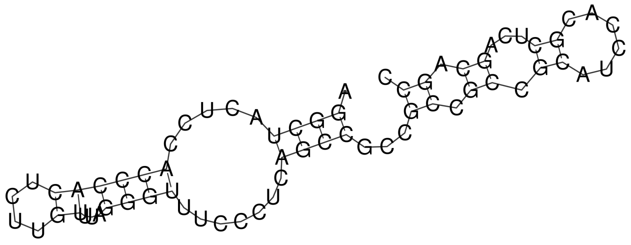

Fig. 3 Example of secondary structure

Matsuura et al. [13] built a model for predicting relative F-Luc activity that uses PLS regression. The relative F-Luc activity represents how strongly heat-stress conditions affect the amount of translated proteins. The PLS regression model can take into account the relationships between some features; therefore, the problem that remains by Kawaguchi et al. [9] is solved. However, the prediction precision is not sufficient in the case of predicting the PR-value.

4. R-STEINER: Proposed Method

In this section, we propose our method, R-STEINER (generate nucleotide sequences Randomly and Select a TrEmendous 5'-untranslated region that Increases the amount of traNslated protEins of a ceRtain gene) to discover the translation enhancers. R-STEINER is split into two steps: the B-step where we build a model for predicting PR-value and the G-step where we yield the translation enhancers followed by selecting top-*k* sequences that increase the amount of translated proteins of a given gene. Details on B-step and G-step are given in Sect. 4.1 and Sect. 4.2, respectively.

4.1 B-step

The B-step consists of two steps:

(B1) feature engineering,

(B2) building the prediction model.

In step (B1), we transform an mRNA sequence to a feature vector that we can throw into machine learning models (Sect. 4.1.1). In step (B2), we build a model for predicting the PR-value by using an ensemble of random forest [3], gradient boosting [8], and XGBoost [4] (Sect. 4.1.2).

We have two datasets that are in two conditions. The first dataset is for a normal condition. In this condition, the cells proliferate activity, and their matter production is active. We represent this dataset as *Con*. The second dataset is for a heat-stress

[*1] Ribosome loading is one criteria of the amount of translated proteins.

Table 1 Summary of *Con* dataset ($N_{Con} = 24915$)

Gene ID	5'UTR	CDS	3'UTR	PR-value
1	GUU...GAG	AUGU...AUGA	UGA...UGC	0.9229
2	GAA...UAU	AUGA...GUAA	GAG...GUC	1.0054
⋮	⋮	⋮	⋮	⋮
N_{Con}	$s^{5'\mathrm{UTR}}_{N_{Con}}$	$s^{\mathrm{CDS}}_{N_{Con}}$	$s^{3'\mathrm{UTR}}_{N_{Con}}$	$y_{N_{Con}}$

Table 2 Summary of *HS* dataset ($N_{HS} = 21786$)

Gene ID	5'UTR	CDS	3'UTR	PR-value
1	GAA...UAU	AUGA...GUAA	GAG...GUC	1.1293
2	AGG...GCC	AUGG...UUGA	GUG...UUC	0.7600
⋮	⋮	⋮	⋮	⋮
N_{HS}	$s^{5'\mathrm{UTR}}_{N_{HS}}$	$s^{\mathrm{CDS}}_{N_{HS}}$	$s^{3'\mathrm{UTR}}_{N_{HS}}$	$y_{N_{HS}}$

condition. In this condition, the activities of cells are restrained, and, in general, their matter production is also reduced. We represent this dataset as *HS*. Summaries of the datasets for our analysis are shown in **Table 1** and **Table 2**. Generally, the lengths of each area are different, i.e., the length of 5'UTR of gene n is different from that of gene $n + 1$. We split the datasets into a training set (50%), validation set (25%) and test set (25%). We trained predictors with the training set and tuned hyperparameters with the validation set. Then, in the aggregation step, we aggregate the three models, random forest, gradient boosting and XGBoost, because there is no difference among the prediction precision of these models (Sect. 5.2).

4.1.1 (B1) Feature Engineering

In (B1), we engineer features for regression models. We use three types of features:

(F1) lengths of 5'UTR, CDS and 3'UTR,

(F2) secondary energy of 5'UTR, CDS and 3'UTR,

(F3) counts of 3-gram acids of 3'UTR, CDS and 3'UTR.

It is known that (F1) and (F2) affect the amount of translated proteins [9] under experimental settings. Using these features, we construct the following feature vector:

$$\boldsymbol{x} = \mathrm{concat}\left[\boldsymbol{x}_{F_1} \quad \boldsymbol{x}_{F_2} \quad \boldsymbol{x}_{F_3}\right] \in \mathbb{R}^{238}, \tag{1}$$

where

$$\boldsymbol{x}_{F_1} = \left[\mathrm{len}(5'\mathrm{UTR}) \quad \mathrm{len}(\mathrm{CDS}) \quad \mathrm{len}(3'\mathrm{UTR})\right] \in \mathbb{R}^3, \tag{2}$$

$$\boldsymbol{x}_{F_2} = \left[G(5'\mathrm{UTR}) \quad G(\mathrm{CDS}) \quad G(3'\mathrm{UTR})\right] \in \mathbb{R}^3, \tag{3}$$

$$\boldsymbol{x}_{F_3} = \mathrm{concat}\left[\boldsymbol{c}^{5'\mathrm{UTR}} \quad \boldsymbol{c}^{\mathrm{CDS}} \quad \boldsymbol{c}^{3'\mathrm{UTR}}\right] \in \mathbb{R}^{232}. \tag{4}$$

Here, we use following notations. len(R) and $G(R)$ represent the length of any area R and the free energy of R, respectively. $\boldsymbol{c}^{5'\mathrm{UTR}}$ and $\boldsymbol{c}^{3'\mathrm{UTR}}$ contain counters of A, U, G, C, AA, AU, ..., UU, AAA, AAU, AAU, ..., UUU on 5'UTR and 3'UTR, respectively. $\boldsymbol{c}^{\mathrm{CDS}}$ represents AAA, AAU, ..., UUU on CDS.

In CDS, three continuous codons correspond to an amino acid; therefore, we assumed that the counters of two continuous codons and one codon do not need to be counted. Kawaguchi et al. [9] also analysed the relationships between A, U, G and C contents and the amount of translated proteins. In our study, we use more various count features than the features used by Kawaguchi et al. [9].

4.1.2 (B2) Prediction Models

We build a model predicting the PR-value by using an ensemble model of six models comprising random forest models, gradient boosting models and XGBoost models, i.e., we estimate the PR-value of a given mRNA by using Eq. (7).

$$\hat{h}^{(\mathrm{HS})}(\boldsymbol{x}^*) = \frac{1}{3}\left(h^{(\mathrm{HS})}_{\mathrm{rf}}(\boldsymbol{x}^*) + h^{(\mathrm{HS})}_{\mathrm{gb}}(\boldsymbol{x}^*) + h^{(\mathrm{HS})}_{\mathrm{xgb}}(\boldsymbol{x}^*)\right), \tag{5}$$

$$\hat{h}^{(\mathrm{Con})}(\boldsymbol{x}^*) = \frac{1}{3}\left(h^{(\mathrm{Con})}_{\mathrm{rf}}(\boldsymbol{x}^*) + h^{(\mathrm{Con})}_{\mathrm{gb}}(\boldsymbol{x}^*) + h^{(\mathrm{Con})}_{\mathrm{xgb}}(\boldsymbol{x}^*)\right), \tag{6}$$

$$\hat{h}(\boldsymbol{x}^*) = \frac{1}{2}\left(\hat{h}^{(\mathrm{HS})}(\boldsymbol{x}^*) + \hat{h}^{(\mathrm{Con})}(\boldsymbol{x}^*)\right), \tag{7}$$

where $\boldsymbol{x}^* \in \mathbb{R}^{238}$ is a feature vector of the given mRNA, $h^{(\mathrm{HS})}_{\mathrm{rf}}(\cdot), h^{(\mathrm{HS})}_{\mathrm{gb}}(\cdot), h^{(\mathrm{HS})}_{\mathrm{xgb}}(\cdot)$ mean the prediction model built by random forest, gradient boosting and XGBoost in *HS*, respectively and $h^{(\mathrm{Con})}_{\mathrm{rf}}(\cdot), h^{(\mathrm{Con})}_{\mathrm{gb}}(\cdot), h^{(\mathrm{Con})}_{\mathrm{xgb}}(\cdot)$ also mean the prediction models in *Con* vice versa.

4.2 G-step

The G-step is also split into three steps: random generation, selecting good feature vectors, and selecting sequences. In the random-generation step, we generate ℓ nucleotide acids (A, U, G or C) randomly and concatenate them, resulting in obtaining 5'UTR sequences on a computer. In the step for selecting good feature vectors, we transform the sequences obtained in the previous step into the feature vectors and predict the PR-value by using the ensemble prediction model Eq. (7). Then, we select the top k feature vectors whose PR-values are largest. In the step of selecting a sequence, we select sequences corresponding to the selected feature vectors.

4.2.1 Algorithm of Sequence Generation

In the G-step, we generate B 5'UTR sequences and combine them with certain CDS and 3'UTR. Then, for the combined mRNA, we predict the PR-value by using the prediction model and select the top k mRNA sequences. We cannot use subsequences AUG and AAUAAU. Thus, we generate 5'UTR by using Algorithm 1. In this paper, we use $B = 2 \times 10^6$.

Algorithm 1 Algorithm of mRNA Generation

```
Require: $\hat{h}(\cdot)$: prediction model defined by Eq. (7)
Ensure: $k$ sequences of 5'UTR $s^{5'\mathrm{UTR}}$
make one part of feature vectors $\boldsymbol{x}_{F_2}$ and $\boldsymbol{x}_{F3}$
for $t = 1, 2, \cdots, B$ do
    fix $L \in \mathbb{N}$ randomly in interval $(22, 49)$
    generate acids $\{s_\ell\}_{\ell=1}^{L}$, where $s_\ell$ is selected from $\{\mathrm{A, U, G, C}\}$ randomly
    $S_t^{5'\mathrm{UTR}} \leftarrow \mathrm{concat}\{s_\ell\}_{\ell=1}^{L}$
    make one part of feature vector $\boldsymbol{x}_{F_3}$ of $S_t^{5'\mathrm{UTR}}$
    make the feature vector $\boldsymbol{x}_t^*$ by concatination of $\boldsymbol{x}_{F_1}, \boldsymbol{x}_{F_2}$ and $\boldsymbol{x}_{F_3}$
    estimate PR-value of the sequence $S_t^{5'\mathrm{UTR}}$ at current step by
        $\hat{y}_t = \hat{h}(\boldsymbol{x}_t^*)$
end for
sort $\{S_t^{5'\mathrm{UTR}}\}_{t=1}^{T}$ in descending order of $\{\hat{y}_t\}_{t=1}^{T}$
return top $k$ sequences of $\{S_t^{5'\mathrm{UTR}}\}_{t=1}^{T}$
```

5. Preliminary Evaluation of Prediction Models

In this section, we compare models developed with random forest, gradient boosting and XGBoost with the models developed by linear regression, PLS regression [27], linear lasso [26], [5], [6] and neural network [15]. Then, we declare that the tree-based prediction models are better than the others. As the feature vector $\boldsymbol{x}$ contains both—discrete and continuous—variables, tree-based models will show better performance than the other models [7].

5.1 Hyperparameter Tuning

Here, we explain how to tune the hyperparameters of the models. Each prediction model has hyperparameters. The PLS regression model has the hyperparameter η, which represents the number of principal components. The feed-forward neural network has many hyperparameters for the architecture of the model such as the number of layers, the number of units in each layer and the activation functions in each layer. In addition to these hyperparameters, we should decide the learning rate, the way of optimization, whether should we drop out some units and so on. Generally, such an architecture, hyperparameters and various network tunings are determined on the basis of previous research of the same domain. However, we could not find such research; therefore, we used a simple feed-forward neural network as our prediction model. The hyperparameters of random forest, gradient boosting and XGBoost that we tuned were the number of regression trees M and the maximum depth of each regression tree $d_{\max}$.

These hyperparameters were determined by using the validation set. There are various methods for optimizing hyperparameters, such as grid search, random search [19], [21], [22] and Bayesian optimization [16], [17], [18]. Generally, for some loss function that is not represented clearly, we cannot obtain a strictly optimal solution for minimizing the function. Therefore, we seek the hyperparameter in a given search area, i.e., even if we adopt any of the above methods, we have to determine the area where hyperparameters are searched. If the search area is out of focus, the selected hyperparameters are far from the true optimal hyperparameters. Specifically, at the worst, we were concerned that the selected hyperparameters will not even be a local minimum point along one axial direction. To avoid such an unfortunate situation, we sought hyperparameters with the following steps.

step 1. Set initial value of hyperparameters as $\left(\theta_1, \theta_2, \cdots, \theta_p\right) = \left(\hat{\theta}_1, \hat{\theta}_2, \cdots, \hat{\theta}_p\right)$.

step 2. Update $\theta_1 = \hat{\theta}_1$ such that it satisfies

$$\hat{\theta}_1 = \underset{\theta_1 \in I_1}{\arg\min} \frac{1}{|\mathcal{D}_{\mathrm{vali}}|} \sum_{\boldsymbol{x}_n \in \mathcal{D}_{\mathrm{vali}}} l\left(h(\boldsymbol{x}_n), y_n\right), \qquad (8)$$

where $\mathcal{D}_{\mathrm{vali}}$ represents the validation set, l is the squared loss, the other hyperparameters contained in the loss function l are fixed and interval I_1 contains the local minimum point along the θ_1 axial direction.

step 3. Update $\theta_2 = \hat{\theta}_2, \cdots, \theta_p = \hat{\theta}_p$ in a similar manner to previous step.

step 4. Finally, determine $\theta_1, \theta_2, \cdots, \theta_p$ by grid search in area $\prod_{i=1}^{p} [\theta_i - \varepsilon_i, \theta_i + \varepsilon_i]$.

By following these steps, we can focus narrowly on the area that contains at least one (local) minimum point, i.e., we can avoid the situation where the searched area does not contain any (local) minimum points. Certainly, if the prediction model has only one hyperparameter, all you need to do is search for the optimal value along the θ_1 axial direction.

If the prediction models have hyperparameters, they are tuned in the previous steps, except for the neural network. The areas for grid search and determined values in each prediction model are shown in **Table 4**. We adopted the Bayesian optimization to determine each hyperparameter, the number of units and dropout rates in each layer, because the combination pattern is too large to adopt the grid search in the neural network. The architecture of the neural network is described in **Table 3**.

5.2 Evaluation of Prediction Models

Here, we evaluate the prediction models by training the models with the training set, tuning the hyperparameters in the manner described in Sect. 5.1 with the validation set and comparing the models with the test set. The results are shown in Fig. 4. As is shown, the tree-based prediction models were better than the other prediction models. In addition, we apply a statistical test:

Null Hypothesis $H_0 \quad \rho_{\mathrm{rf}} = \rho_{\mathrm{gb}} = \rho_{\mathrm{xgb}}$,

Alternative Hypothesis $H_1 \quad \neg H_0$,

where ρ_{rf}, ρ_{gb}, and ρ_{xgb} mean the correlation coefficients obtained by using random forest, gradient boosting and XGBoost, respectively. We cannot reject the null hypothesis, i.e., there is the potential for no differences in the correlation coefficients. Therefore, we aggregated all three models in Sect. 4.1.2.

6. Synthesis Experiment

In the G-step, we selected 5'UTR sequences by using the prediction model as the utility function, i.e., we selected the top-k 5'UTRs that maximized the predicted PR-values in the generated sequences. However, the prediction model is learned to fit the natural 5'UTRs, so the model does not always predict the PR-value of the artificial 5'UTRs accurately. Therefore, we had to conduct synthesis experiments in order to make sure that the predicted PR-values of the artificial 5'UTRs were close to the true amounts of translated proteins. If the predicted PR-values are close to the true amounts of the proteins, we can obtain high-performance 5'UTRs by increasing the iterations of sequence generation in Algorithm 1.

In the experiments, we adopted the criterion F/R-luc activity, which represents the translated proteins of the 5'UTR. It is known that $\log_{10}$(F/R-luc activity) bears a linear relationship with PR-value [23]. Hence, we evaluated whether the predicted PR-values were close to the observed $\log_{10}$(F/R-luc activity) with the correlation coefficient. We chose 5'UTRs that were made in the experiments as follows.

Table 3 Hyperparameters of neural network

layer	hyperparameter	candidate area	selected (*HS*)	selected (*Con*)
input	activation	None	tanh	tanh
hidden	number of units	$\{256, 512, 1024, 2148\}$	2048	512
	drop out rate	$[0, 0.5]$	0	0.27
	activation function	None	relu	relu
hidden	number of units	$\{256, 512, 1024, 2148\}$	1024	512
	drop out rate	$[0, 0.5]$	0	0
	activation function	None	linear	linear
output	number of units	None	1	1

Table 4 Hyperparameters

Model	hyperparameter	searched area (*HS*)	determined value (*HS*)	searched area (*Con*)	determined value (*Con*)
PLS regression	η	$\{2, 3, \cdots, 221\}$	123	$\{2, 3, \cdots, 221\}$	90
linear lasso	α	$\{2^i \mid i = 1, 0, \cdots, -4\}$	2^{-4}	$\{2^i \mid i = 1, 0, \cdots, -4\}$	2^{-4}
random forest	M	$\{10, 20, \cdots, 100\}$	94	$\{270, 275, \cdots, 285\}$	285
	$d_{\max}$	$\{1, 6, \cdots, 31\}$	20	$\{11, 12, \cdots, 20\}$	16
gradient boosting	M	$\{35, 40, \cdots, 50\}$	50	$\{180, 185, \cdots, 195\}$	195
	$d_{\max}$	$\{9, 10, \cdots, 14\}$	9	$\{1, 2, \cdots, 10\}$	5
XGBoost	M	$\{2^i \mid i = 11, 12, \cdots, 15\}$	2^{11}	$\{974, 984, \cdots, 1074\}$	1054
	$d_{\max}$	$\{1, 6, \cdots, 46\}$	11	$\{1, 2, \cdots, 10\}$	3

(S1) Generate 5'UTR sequences as use R-STEINER, i.e., execute the algorithm until the 9-th line.

(S2) Select three 5'UTRs whose PR-values are highest, two 5'UTRs whose PR-values are lowest and four 5'UTRs randomly. Note that these four 5'UTRs are not same as previous three and two 5'UTRs.

We show the 5'UTR sequences that were in the above way in **Table 5**. We synthesised the selected 5'UTRs and calculated the correlation coefficient between the predicted PR-value and observed $\log_{10}$(F/R-luc activity).

We clarify how the synthesis experiments were performed in 6.1 - 6.7. Then, we show the evaluation in 6.8.

6.1 Plant Materials, Culture Conditions, and Growth Conditions

Oryza sativa L. cv. Nipponbare suspension cells [20] were cultured in R2S medium with rotary shaking at 90 rpm at 30°C in a dark condition. For genome-wide analysis of the polysome association, cells cultured for three days and cells cultured for three days and incubated at 41°C for 15 min were collected as the control (*Con*) sample and heat-stress (*HS*) sample, respectively. In addition, Oc suspension cells from roots of *Oryza sativa* L. accession C5924 [2], that is, the suspension cells of the easy-to-isolate protoplast, were cultured under the same condition, and Oc cells cultured for three days were used for transient expression assay. Both suspension cell cultures were maintained with sub-culturing every week.

6.2 Polysome Fractionation Assays and RNA Isolation from Sucrose Gradients

Polysome fractionation analysis was performed according to the previously described method in Yamasaki et al. [28]. Cell extracts were layered on a 26.25–71.25% sucrose density gradient and centrifuged. After centrifugation, the gradients were separated into two fractions by using a piston gradient fractionator (BioComp Instruments, Fredericton, NB, Canada). The second fraction of the bottom half (polysome fraction) and both fractions (total fraction) were individually collected and pooled into tubes containing guanidine hydrochloride (final concentration, 5.5 M). RNA was precipitated by the addition of an equal volume of ethanol, overnight incubation at −20°C and centrifugation at 10,000 rpm for 45 min in a JA-20 rotor (Beckman Coulter, Fullerton, CA, USA). The resulting precipitate was washed with 85% ethanol. RNA was purified by using an RNeasy kit (Qiagen, Hilden, Germany) with on-column DNase I treatment according to the manufacturer's instructions. RNA was eluted with 100 µl of RNase-free water, and the RNA integrity was examined with an Agilent Bioanalyzer 2100 (Agilent Technologies). We prepared RNA in two independent biological replicates.

6.3 Cap Analysis of Gene Expression (CAGE) and Data Analysis

nAnT-iCAGE libraries preparation, sequencing, filtering, mapping and gene annotation were performed on the basis of the previously described methods in Yamasaki et al. [29]. For this analysis, to more accurately identify the transcription start site (TSS), additional quality control was performed to remove tags with mismatches within three bases from the 5'end. In addition, tag counts were converted to tag per million (TPM) values at each TSS level as TPM_TSS and averaged between two replicates. Finally, we calculated the polysome ratio at each TSS level (PR_TSS) as an indicator of polysome association by using the following formula Eq. (9). To obtain more reliable data, we used a limited TSS that mapped more than 50 tags in the total fraction data in both replicates for calculating PR_TSS. Genome information from IRGSP-1.0 in The Rice Annotation Project Database[*2] was used as a reference for rRNA tag removal, mapping and annotation.

$$\mathrm{PR\_TSS} = \frac{\text{TPM\_TSS in polysome fraction data}}{\text{TPM\_TSS in total fraction data}} \tag{9}$$

6.4 Plasmid Construction

A reporter plasmid for transient expression assay with PEG-mediated protoplast transformation using in vitro synthesised RNA was constructed by modifying plasmid pFL-

[*2] http://rapdb.dna.affrc.go.jp

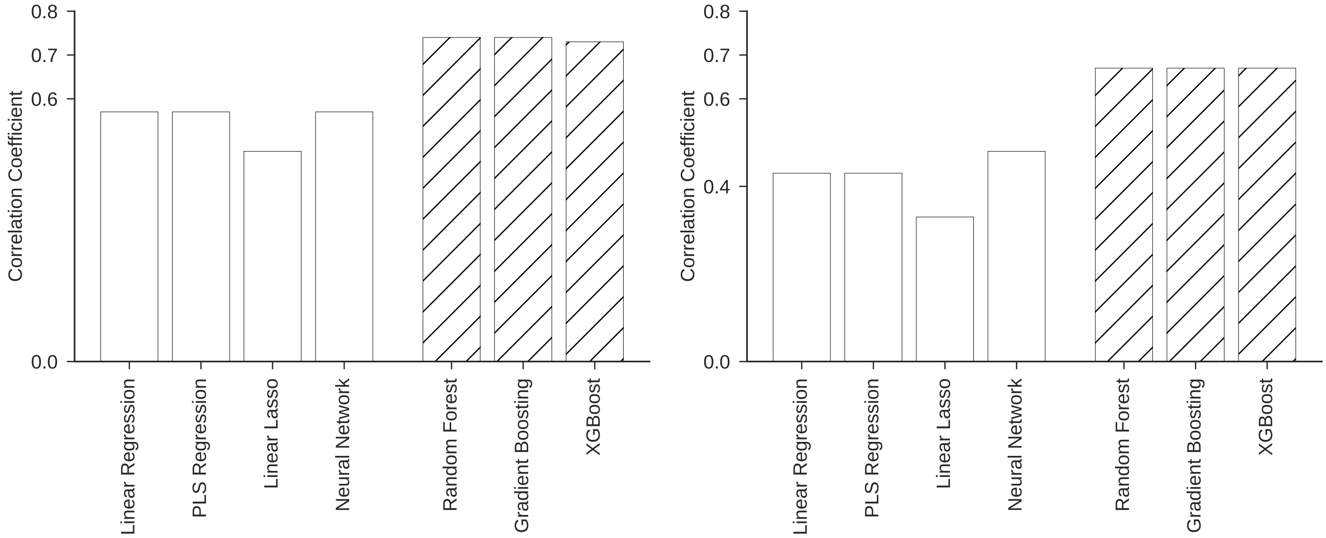

Fig. 4 Correlation coefficients between observed and predicted value. Left is in *HS*; right is in *Con*.

Table 5 Gene-specific primers for *in vitro* synthesised RNA

5’UTR name	Primer sequence (5’ to 3’)
GS1	ATTAATTGTACAGCACAGCAAATTGTCAACTATTTTTTCAGACAGATGGAA
GS2	AAGTATCATTAAGTTTGATTTATTTGTCAAGGAACAAGTATCTTAGATGGAA
GS3	ACAAACGAACTCTCATTCACATCATTAGATAAATTAATTTTAATATGGAA
GS3	TTCCGGATCGTACCGGTCAGGATGGAA
GS4	TACCAGCCCTAGGTGGAAACATTCTAGCCGCCATAATGGAA
GS5	ATTTCCCCTTCTATCACGCTCGCGACCTAGGCGTTGGGGCTCGTATGGAA
GS6	TAACGCCCGGGGTCTGTGTGTCGCTCCCTAAATGGAA
GS7	CGGCGCTCCGCGGCCGTTGGAGGGGCCGCCGATGGAA
GS8	CCCGGACGAGCCGGGCCGGCCGGATGGAA

pA [13]. The test 5’UTRs were synthesised by oligonucleotide annealing that included a part of the T3 promoter—AATTAACCCTCACTAAAGG—with NcoI site: CATGG, and a part of the F-luc coding region with the AatII site: GACGT. In other words, 5’UTRs actually synthesised are sequences concatenated as follows:

concatenate [NcoI, T3 promoter, GS*i*, AatII],

where GS*i* represent the generated sequence in Table 5. Each annealed oligonucleotide was introduced into pFL-pA at the NcoI/AatII sites to generate the plasmids pT3-5’-UTR-FL-pA. Insert DNA fragments were verified by sequencing.

6.5 Synthesis of Reporter mRNAs In Vitro

RNA synthesis was performed in vitro from plasmids pT3-5’UTR-FL-pA containing the test 5’UTR and pT3-RL-pA [12] as described previously [13].

6.6 Protoplast Isolation

Three-day-old Oc suspension cells were collected and gently shaken in protoplastization enzyme solution (4% Cellulase RS, 1% macerozyme R10, 0.1% $CaCl_2 \cdot 6\,H_2O$, 0.1% MES, 0.4 M mannitol, pH 5.6) at 30°C for 3 h. The isolation solutions containing crude protoplasts were filtered through a 40-μm nylon sieve, and the same volume of W5 solution (154 mM of NaCl, 125 mM of $CaCl_2$, 5 mM of KCl, 2 mM of Mes-KOH, pH 5.6) was added to the solutions. After centrifugation for 4 min at 800 rpm, pelleted protoplasts were collected and washed once more in the W5 solution by centrifugation. Pelleted protoplasts were added into the W5 solution and incubated on ice for 30 min. The final protoplast density was adjusted to 1×10^6 protoplasts ml^{-1}.

6.7 Protoplast Transient Expression Assay

Two μg of capped F-Luc mRNAs harboring a 5’UTR that contained 0.4 μg of capped R-Luc mRNAs (internal control) were mixed with 1.9×10^6 protoplasts, and an equal volume of polyethylene glycol-CMS (PEG-CMS) solution [200 mM mannitol, 0.1 M $Ca(NO_3)_2$, 40% PEG 4000] was then added to each sample. The protoplast mixture was incubated at room temperature for 20 min, and 1 ml of protoplast medium (400 mM of mannitol supplemented with R2S) was added. The transiently transfected protoplasts were then incubated at 30°C for 20 min, lysed in Passive Lysis Buffer (Promega) and assayed for R-Luc and F-Luc activities by using the Dual-Luciferase Reporter Assay System (Promega, Madison, WI, USA) and a plate reader (TriStar LB 941: Berthold Technologies, Bad Wildbad, Germany).

6.8 Evaluation

The results are shown in Fig. 5. The right figure is a result of a reproductive experiment, i.e., we did the same synthesis experiment twice in order to certain that we did not make mistakes in the first experiment. Note that the ranges of the vertical axis of the two figure are not same, because the activities of the cells used to the real experiments are different—it is impossible to make the activities even in the experiments—. As shown in Fig. 5, the correlation coefficients were very high (0.89 and 0.91). Therefore, as predicted, the PR-value became larger, and the true amount of translated proteins became larger, i.e., the prediction model worked well even for the artificial mRNA. In addition, the two scatter plots were similar to each other; hence, the result of this synthesis experiment is reproducible.

Considering the synthesis experiments, the prediction model built in Sect. 4 can predict the amounts of translated proteins of artificial mRNAs accurately. Therefore, in Algorithm 1, increasing the number B allows us to obtain translation enhancers.

On the other hand, in the each cluster of Fig. 5, there are variations in each point. Therefore, we cannot obtain the 5'UTR which *maximizes* the amount of translated proteins of a certain gene, but we can obtain the 5'UTRs which *increase* the amounts of the proteins.

7. Conclusion

We proposed R-STEINER. With R-STEINER, we can discover the translation enhancers of a certain gene. In the B-step, we built a model for predicting the PR-value. The best models were three tree-based ensemble models: the random forest, gradient boosting and XGBoost. This is because tree-based methods are robust for count features, and all of the features, except for secondary energy, engineered in Sect. 4.1.1 are discrete-type features. This result was common between *HS* and *Con*; therefore, the fact that the tree-based ensemble models are the best prediction models does not depend on the condition. Then, using rice, we clarified that the prediction model used in R-STEINER can predict the amount of translated proteins even for artificial mRNA. From this result, it is clear that the prediction model can predict the amount of translated proteins of 5'UTRs that are generated in G-step. Therefore, R-STEINER generates translation enhancers by increasing the iteration B in Algorithm 1. Hence, we can perform real synthesis experiments for the generated 5'UTRs by R-STEINER, resulting in reduction of the cost, time and effort.

The point that should be improved for R-STEINER is sequence generation. In the G-step, we generate nucleotides randomly and yield mRNA sequences by combining the generated nucleotides. We should solve the optimization problem

$$\boldsymbol{x}' = \arg\max_{\boldsymbol{x}} \hat{h}(\boldsymbol{x}) \tag{10}$$

and generate a 5'UTR sequence corresponding to $\boldsymbol{x}'$. However, for this approach, we have the following two difficulties. The first is that solving Eq. (10) is difficult. The second is that one feature vector does not correspond to one 5'UTR sequence. As can be seen in Sect. 4.1.1, one feature vector corresponds to some 5'UTR sequences.

To solve the first problem, we should develop a method for solving the optimization problem of the function whose variable is a 238-dimensional vector and that we cannot write in an explicit form. After the first is solved, we need a method for generating one sequence with one feature vector. Specifically, we need to develop a method for selecting one sequence from candidates of sequences by using some kind of criterion. If these two problems are solved, we could generate the translation enhancers of a certain gene in shorter time.

In addition to the above suggestions, we considered the importance of features that were calculated by using the ensemble predicted model. We calculated the importance by calculating the sample mean among the three tree-based models: random forest, gradient boosting and XGBoost (see Fig. 6). As can be seen in Fig. 6, the features of 5'UTR affected the prediction more strongly than the features of CDS and 3'UTR. This result agrees with the previous research [29], but the second important feature—the counts of GAC—is not mentioned. Some features of CDS may affect the amount of translated proteins.

References

[1] Alberts, B., Bray, D., Hopkin, K., Johnson, A., Lewis, J., Raff, M., Roberts, K. and Walter, P.: *Essential cell biology*, Garland Science (2013).

[2] Baba, A., Hasezawa, S. and Syōno, K.: Cultivation of rice protoplasts and their transformation mediated by Agrobacterium spheroplasts, *Plant and cell physiology*, Vol. 27, No. 3, pp. 463–471 (1986).

[3] Breiman, L.: Random forests, *Machine learning*, Vol. 45, No. 1, pp. 5–32 (2001).

[4] Chen, T. and Guestrin, C.: Xgboost: A scalable tree boosting system, *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, ACM, pp. 785–794 (2016).

[5] Friedman, J., Hastie, T., Höfling, H., Tibshirani, R. et al.: Pathwise coordinate optimization, *The Annals of Applied Statistics*, Vol. 1, No. 2, pp. 302–332 (2007).

[6] Friedman, J., Hastie, T. and Tibshirani, R.: Regularization paths for generalized linear models via coordinate descent, *Journal of statistical software*, Vol. 33, No. 1, p. 1 (2010).

[7] Friedman, J., Hastie, T. and Tibshirani, R.: The elements of statistical learning (2001).

[8] Friedman, J. H.: Stochastic gradient boosting, *Computational Statistics & Data Analysis*, Vol. 38, No. 4, pp. 367–378 (2002).

[9] Kawaguchi, R. and Bailey-Serres, J.: mRNA sequence features that contribute to translational regulation in Arabidopsis, *Nucleic Acids Research*, Vol. 33, No. 3, pp. 955–965 (2005).

[10] Kazuhito, F.: Challenge for Medical Protein Production by Using Plant.

[11] Lorenz, R., Bernhart, S. H., Zu Siederdissen, C. H., Tafer, H., Flamm, C., Stadler, P. F. and Hofacker, I. L.: ViennaRNA Package 2.0, *Algorithms for Molecular Biology*, Vol. 6, No. 1, p. 26 (2011).

[12] Matsuura, H., Shinmyo, A. and Kato, K.: Preferential translation mediated by Hsp81-3 5'-UTR during heat shock involves ribosome entry at the 5 ′ -end rather than an internal site in Arabidopsis suspension cells, *Journal of bioscience and bioengineering*, Vol. 105, No. 1, pp. 39–47 (2008).

[13] Matsuura, H., Takenami, S., Kubo, Y., Ueda, K., Ueda, A., Yamaguchi, M., Hirata, K., Demura, T., Kanaya, S. and Kato, K.: A computational and experimental approach reveals that the 5 ′ -proximal region of the 5 ′ -UTR has a Cis-regulatory signature responsible for heat stress-regulated mRNA translation in Arabidopsis, *Plant and cell physiology*, Vol. 54, No. 4, pp. 474–483 (2013).

[14] Maxmen, A. et al.: Drug-making plant blooms, *Nature*, Vol. 485, No. 7397, p. 160 (2012).

[15] McCulloch, W. S. and Pitts, W.: A logical calculus of the ideas immanent in nervous activity, *The bulletin of mathematical biophysics*, Vol. 5, No. 4, pp. 115–133 (1943).

[16] Močkus, J.: On Bayesian methods for seeking the extremum, *Optimization Techniques IFIP Technical Conference*, Springer, pp. 400–404 (1975).

[17] Mockus, J.: On Bayesian Methods for Seeking the Extremum and their Application., *IFIP Congress*, pp. 195–200 (1977).

[18] Mockus, J.: *Bayesian approach to global optimization: theory and applications*, Vol. 37, Springer Science & Business Media (2012).

[19] Rastrigin, L.: The convergence of the random search method in the extremal control of a many parameter system, *Automaton & Remote Control*, Vol. 24, pp. 1337–1342 (1963).

[20] Saotome, A., Kimura, S., Mori, Y., Uchiyama, Y., Morohashi, K. and Sakaguchi, K.: Characterization of four RecQ homologues from rice (Oryza sativa L. cv. Nipponbare), *Biochemical and biophysical research communications*, Vol. 345, No. 4, pp. 1283–1291 (2006).

[21] Schrack, G. and Choit, M.: Optimized relative step size random searches, *Mathematical Programming*, Vol. 10, No. 1, pp. 230–244 (1976).

[22] Schumer, M. and Steiglitz, K.: Adaptive step size random search, *IEEE Transactions on Automatic Control*, Vol. 13, No. 3, pp. 270–276 (1968).

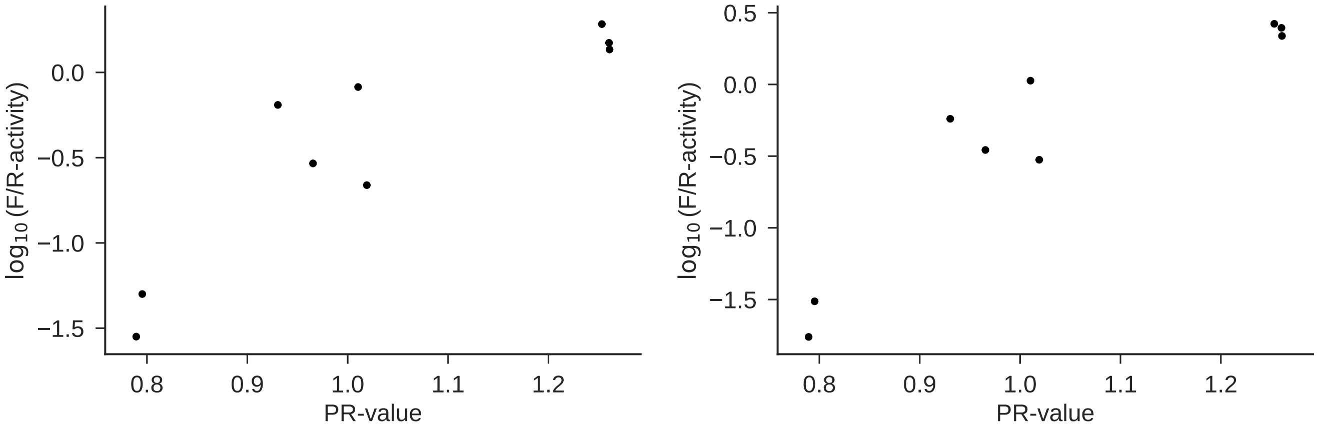

Fig. 5 Correlation coefficients: x-axis is predicted PR-values, and y-axis is observed $\log_{10}$(F/R-luc activity). Correlation coefficients on left are 0.89, and those on right are 0.91. Right is result of reproductive experiment.

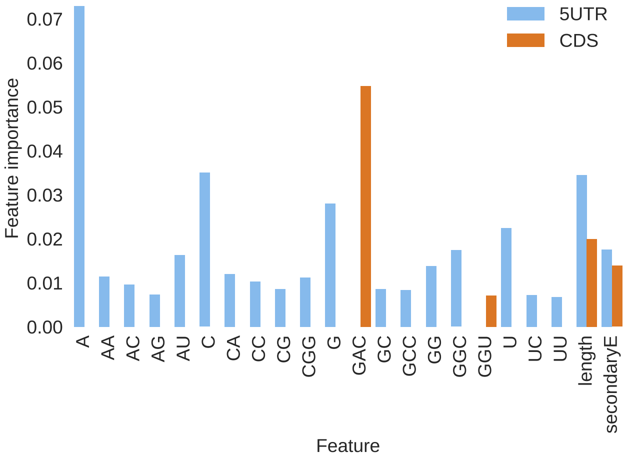

Fig. 6 Top–90 percentile important features

[23] Shotaro, Y.: Research on mechanism of mRNA translation in Arabidopsis thaliana, PhD Thesis, Nara Institute of Science and Technology (2016).

[24] Shotaro, Y., Kiyotaka, U. and Ko, K.: Development of Plant Expression Vector on Considering of Environmental Stress.

[25] Sugio, T., Satoh, J., Matsuura, H., Shinmyo, A. and Kato, K.: The 5'-untranslated region of the Oryza sativa alcohol dehydrogenase gene functions as a translational enhancer in monocotyledonous plant cells, *Journal of bioscience and bioengineering*, Vol. 105, No. 3, pp. 300–302 (2008).

[26] Tibshirani, R.: Regression shrinkage and selection via the lasso, *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288 (1996).

[27] Wold, H.: Estimation of principal components and related models by iterative least squares. Multivariate Analysis. Edited by: Krishnaiah PR. 1966.

[28] Yamasaki, S., Matsuura, H., Demura, T. and Kato, K.: Changes in polysome association of mRNA throughout growth and development in Arabidopsis thaliana, *Plant and Cell Physiology*, Vol. 56, No. 11, pp. 2169–2180 (2015).

[29] Yamasaki, S., Sanada, Y., Imase, R., Matsuura, H., Ueno, D., Demura, T. and Kato, K.: Arabidopsis thaliana cold-regulated 47 gene 5'-untranslated region enables stable high-level expression of transgenes, *Journal of Bioscience and Bioengineering* (2017).

[30] Yao, J., Weng, Y., Dickey, A. and Wang, K. Y.: Plants as factories for human pharmaceuticals: applications and challenges, *International journal of molecular sciences*, Vol. 16, No. 12, pp. 28549–28565 (2015).