2020 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE) | 978-1-6654-1917-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/WIECON-ECE52138.2020.9398044

COVID-19 mRNA Vaccine Degradation Prediction using Regularized LSTM Model

Sheikh Asif Imran ⋆, Md. Tariqul Islam †, Celia Shahnaz, Md. Tafhimul Islam, Omar Tawhid Imam, Moinul Haque
Department of Electrical and Electronic Engineering
Bangladesh University of Engineering and Technology
Dhaka, Bangladesh
email: ⋆shouborno@ieee.org, †tisbuet@gmail.com

***Abstract*—Due to the advantages of mRNA vaccines such as potency, safety, and production feasibility, recent researches in vaccinology has seen strong focus in mRNA vaccines. As leading researches involving COVID-19 mRNA vaccine candidates are being carried out, the challenge of overcoming the stability tradeoff of mRNA vaccines stand between the production and effective mass distribution stages. With the help of the OpenVaccine RNA database with degradation rate measurements provided by Stanford researchers, we developed an artificial recurrent neural network model to help bioinformatics researchers identify whether and where mRNAs might be unstable and prone to degrade under certain incubation measures. For this purpose, we've prepared a regularized LSTM model which minimizes mean columnwise root mean squared error for several degradation rates. We've found that recurrent algorithms perform better than tree-based algorithms.**

***Index Terms*—Bioinformatics, Recurrent neural networks, RNA, Vaccines, Root mean square.**

I. Introduction

Messenger RNA or mRNA vaccinces are novel technologies of vaccinology. They use RNA sequences that are translated in human or host cells, producing antigens that trigger the body to produce antibodies to fight the disease. Other newer variation of mRNA vaccine involves encoding complete monoclonal antibodies, which also has recently started phase 1 clinical trial with [1]. mRNA vaccines are more potent and easier to mass produce quickly compared to conventional vaccines, and are safe to use, as explained in [2] summarising works so far in the field of mRNA vaccines. Researches for COVID-19 vaccines have also so far been leaded by mRNA vaccine researches, starting early as shown in [3]. However, as [2] notes, one of the challenges of preparing mRNA vaccines is the stability of the vaccines. mRNA vaccines can spontaneously degrade under various conditions including temperature or environment. Damage anywhere along the RNA sequence can render its purpose useless, failing to be translated properly. Hence, to be able to effectively distribute a COVID-19 vaccine worldwide to vaccinate large populations, it's important to find stable RNA molecules. For this purpose, DAS Laboratory of Stanford and Eterna development team launched OpenVaccine [4] .

A. Related works

COVID-19 mRNA vaccine research started early, with Moderna starting the clinical trial of mRNA-1273 two months from the discovery of the sequence, reporting the preliminary findings in [5]. They also initiated phase 3 clinical trials recently, as identified at ClinicalTrials.gov in [6].

On the other hand, keeping the challenges of distributing an unstable mRNA vaccine in mind, stable vaccine candidates are also being researched, such as thermostable phase 1 candidate ARCoV as presented in [7]. ARCoV is claimed to be stable at room temperature for 1 week.

In general, to discover mRNA vaccines with notable stability and in-vivo efficiency, models predicting degradation rates along various positions of RNA sequence can help computational biochemists make significant progress, as explained in [4].

II. Problem formulation

The Stanford University scientists collected the data of 6034 RNA sequences and provided them in [4]. They designated 2400 sequences for training, 629 sequences for public testing and the rest for private scoring during the competition period, all of which are available now with labels post competition. The labels contain the degradation rates measured at different locations of the RNA sequence, namely reactivity values (reactivity) and degradation rates at base or linkage after incubating at high pH (deg_pH10), at high temperature (deg_50C), at high pH with Magnesium (deg_Mg_pH10) and at high temperature with Magnesium (deg_Mg_50C).

A. Dataset overview

The lengths of the RNA sequences were 107 bases for train and public test sets and 130 bases for private test set. For each RNA sequence, the following data were provided.

1) Sequence: This parameter contains the sequence of the nitrogenous bases of guanine (G), uracil (U), adenine (A), and cytosine (C) that composed the RNA. These bases convey the genetic information of the mRNA vaccine that would be decoded within human body.

2) Structure: This parameter contains a sequence of '.', '(', ')', indicating whether bases are paired or unpaired. For example, '(..)..' means that the first base is paired with the fourth base but not with the second, third, fifth or sixth bases.

3) Predicted loop type: This data contains the probable loop type each base belongs to in the structural context. They used bpRNA tool documented in [8], which predicts the loop types as the following labels: paired stem (S), hairpin loop (H), multiloop (M), internal loop (I), bulge (B), external loop (X), and dangling end (E).

4) Base pairing probability matrices (BPPs): Stanford University scientists used their recently developed algorithm EternaFold as presented in [9] to calculate the BPPs for the RNAs. These are symmetric square matrices with the same lengths as the sequences. This matrix gives the probability that each pair of nucleotides in the RNA forms a base pair given a particular model of RNA folding. This basically suggests the distributions of probabilities indicating the possibilities of structures alternative to the one provided in the structure parameter described earlier.

They also provided the labels for each sequence as we had mentioned earlier. However, the Stanford scientist could carry out the degradation rate measurements for the first 68 bases of the train and the public test sequences and 91 bases of the private test sequences. They also provided the result of a filter which indicates the quality of each data. The filter uses the criteria that the minimum value of the five labels should be above -0.5 and the signal to noise ratio (SNR) should all be above 1. They also clustered similar sequences together and used only the clusters with low number of members in the test set.

B. Evaluation metrics

For each sequence, the performance of the models are scored for 68 bases of the public test set and 91 bases of the private test set. The evaluation metric is mean columnwise root mean squared error (MCRMSE), as represented by the following equation.

$$\text{MCRMSE} = \frac{1}{N_t}\sum_{j=1}^{N_t}\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_{ij}-\hat{y}_{ij})^2} \tag{1}$$

Here, N_t is the number of scored ground truth target columns, y is the ground truth value, and $\hat{y}$ is the predicted value.

Among the five degradation rate parameters to be predicted, only three are scored, namely reactivity, deg_Mg_50C, and deg_Mg_pH10, leaving out the parameters measured for degradation rate when incubated without Magnesium.

For evaluation of the performance of our model, lower MCRMSE indicates better prediction of the degradation rates.

III. Proposed method

A. Data preparation

Alongside base pair sequences and provided pair-unpair structure array, we utilized the biophysically inspired features provided by Stanford, comprised of BPPs generated via EternaFold and predicted loop types generated via bpRNA. Firstly, we converted the A-U-G-C base pair sequences to integer class vectors, and then we converted them to categorical binary class matrix, resulting in 4 features. We also treated the structure array similarly, resulting in 3 features for each position of the RNA, corresponding to structural context. Predicted loop typed similarly resulted in 7 features. We also used the BPPs as square matrices, indicating an additional number of features equal to the length of the RNA sequence. Since the highest number of known ground truth degradation values of an RNA sequence is 91 as in the private test set, we truncated the sequences to the length of 91, corresponding to 91 base pairs, each of which now contained the concatenated features of 4 one hot encoded features corresponding to nitrogenous base pairs, 3 one hot encoded features corresponding to structural pair-unpair sequence, 7 one hot encoded features corresponding to predicted loop type according to bpRNA, and 91 features corresponding to base pairing probabilities with the 91 bases of the truncated RNA sequence. Hence, for each RNA sequence, we now had 105 features for the truncated 91 positions.

B. Target preparation and explanation

We experimented with both filtering out training data that did not pass the SNR filter and using all training data regardless of the SNR. In both cases, we opted to train our model for all 5 target values - reactivity, deg_50C, deg_Mg_50C, deg_pH10, deg_Mg_pH10, although the final evaluation would not be carried out for the two degradation rates corresponding to incubating the RNA molecules without Mg. Without filtering the training data, the minimum and maximum value across all five target values throughout the training set are -44.5153 and 44.5212 respectively. For filtered data, this range becomes -0.49 to 10.487 . It is notable that these values are unitless. The methods Stanford used to measure these values and the nature of these values are explained in [10]. The negative values arise from the attenuation correction and background subtraction step explained there. A value of 0 anywhere on the RNA sequence indicates that the position is inert or non-reactive. Lower values indicate more resistance to degradation, with negative values indicating that the signal at that position is lower than that of the background.

C. Loss function

When preparing the custom loss function, we kept in mind the standard evaluation metric of choice, MCRMSE, as explained in equation (1) . It's essentially the average of the RMSEs of the five degradation rate measurements. We also kept in mind that the degradation rates could not be measured for the last 39 sites of the 107 bases long 2400 RNAs designated in the training set. Hence, for each of the five targets, we tuned our loss function for the first 68 sites of the RNAs, resulting in the following custom MCRMSE loss function equation.

$$\text{Loss} = \frac{1}{5}\sum_{j=1}^{5}\sqrt{\frac{1}{68}\sum_{i=1}^{68}(y_{ij}-\hat{y}_{ij})^2} \tag{2}$$

The goal of the model would be to minimize the above loss function for shuffled validation set over a considerable number of epochs.

D. Model formulation

Due to the sequential context involved in the data as we have discussed so far, we primarily chose LSTM as the hypothesis space for starting to build the model. When we inspect how LSTM cells presented in [11] process sequences under the hood, we see that it utilizes memory cell to simulate the idea of forgetting some information while adding new information. The first major challenge we noticed was that due to the nature of the data, the loss failed to converge unless the target values were min-max scaled to a range of -1 to 1. Due to the nature of the original target and the relevance of the loss function for the original scale of the degradation measurements, that solution wasn't feasible. Hence, we opted to deal with the problem by choosing suitable activation functions for the layers of the model. For input layer, we preserved the output of the nodes by using linear activation function, essentially $A(x) = x$ or identity function. For the rest of the layers, we used $tanh$ activation function, essentially $A(x) = \tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}}$, which has the range $(-1, 1)$. For output layer, we chose Leaky Rectified Linear Unit (Leaky RELU) activation function, which corresponds to the following function.

$$f(x) = \begin{cases} 0.01x & \text{for} \quad x < 0 \\ x & \text{for} \quad x \geq 0 \end{cases} \tag{3}$$

With these activation functions, the loss and validation loss of the model started to converge.

Also, to chose the validation set over which the model won't be trained over the epochs but rather evaluate the loss function to minimize it, we firstly shuffled the total training set and split 33% of the set for validation data. We also enabled shuffling the training set after the validation split while fitting the model to minimize the custom MCRMSE loss function.

For optimizer of the model, we chose RMSprop for faster convergence, which was first proposed in [12]. [13] discusses the convergence condition of RMSprop from both theoretical and experimental context while also comparing it with Adam optimizer. We used 0.001 as the learning rate for the optimizer. We trained the models over 300 epochs with batch size 128.

For regularization purpose, we used recurrent dropout of 50% in our LSTM layers. This dropout is applied to LSTM cell update gates. This method of regularization was introduced in [14] . It averts the loss of long-term memory of the previously used method of feed-forward dropout. We also opted to return the full sequence through the LSTM layers instead of returning the latest output only.

We also used an one-dimensional (1D) spatial dropout layer followed by a normalization layer after the first three LSTM layers. The utility of spatial dropout for model regularization is discussed in [15]. It applies dropout to entire 1D feature maps instead of individual elements, promoting independence among feature maps while helping avoid overfitting.

For the output layer, we used a recurrent time distributed dense layer. It treats the input in the same manner as of a time series signal, applying the dense layer to each temporal slices. This way we arrive at a predicted output of the same length as of the input RNA sample containing the five predicted degradation values.

We present the schematic of our best model in Fig. 1. After tuning the parameters of the model for the best performance, the dimensionality of the output space was 5 for *lstm* and *lstm_4* layers, 50 for *lstm_1* and *lstm_3* layers, and 250 for *lstm_2* layer.

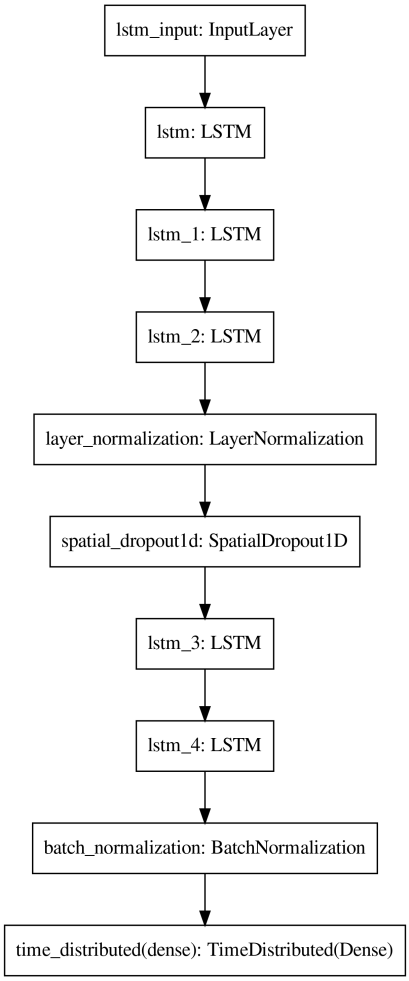

Fig. 1. Schematic of our regularized LSTM model

IV. Performance evaluation

The training and validation MCRMSE losses converged steadily, as presented in Fig. 2. The training loss converged to 0.4904 and validation loss converged to 0.5165 . Here, we used 1608 RNAs for training and 792 RNAs for validation, as explained earlier. The training was carried out for 300 epochs, which required 84 minutes on NVIDIA TESLA P100 GPU. As we have mentioned earlier, these 2400 RNAs were 107 bases long, among which the degradation rates of the first 68

bases were known. Hence, this loss corresponds to the loss function mentioned in equation (2).

Afterwards, for performance evaluation and scoring of the model, the public test set consisting of 629 RNAs of 107 base length, and private test set consisting of 3005 RNAs of 130 base lengths were used. The evaluation metrics, as explained previously, was used to calculate MCRMSE for reactivity, deg_Mg_50C, and deg_Mg_pH10, against the known ground truths, i.e. the first 68 bases of the public set RNAs and the first 91 bases of the private set RNAs. The score calculation only involved the RNAs that passed the SNR filter, i.e. RNAs for which the signal to noise ratio of the degradation rates were consistently above 1, to avoid using noisy data for scoring so that a reliable score can be measured.

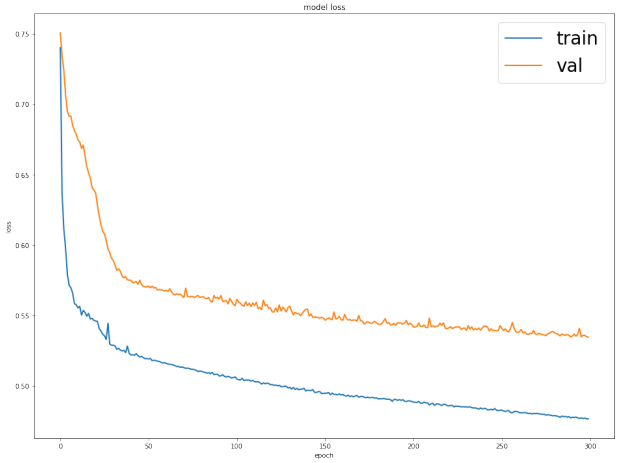

Fig. 2. Training and validation loss curve for our final model

Our MCRMSE score was 0.38796 on the public set and 0.51044 on the private set. We can compare the result with other methods as presented in the following table to confirm that our regularized LSTM model performs better than simple tree-based methods.

TABLE I
PERFORMANCE COMPARISON WITH OTHER METHODS

Method	Public MCRMSE	Private MCRMSE
Proposed LSTM model	0.38796	0.51044
XGBoost [16]	0.51191	0.76412
HistGradientBoosting [17]	0.47839	0.56141

We can confirm that, due to the sequential nature of the RNA data where sequential memory and context is relevant, our proposed recurrent neural network model performs better than simple tree-based methods.

V. CONCLUSION

For the search of stable mRNA vaccine candidates, computational biochemists and researchers of bioinformatics can find degradation prediction models helpful. We have demonstrated and explained the utility of recurrent neural network method over tree-based methods through our regularized LSTM model for this problem. Furthermore, some steps such as data augmentation and cross-validation can help improve the performance of the model even further. For instance, other methods such as ContraFold or Vienna can be used to generate additional BPPs for the RNAs, besides using the more recent EternaFold BPPs which we have explained previously. Also, other loop type prediction methods can be used to further augment the data. Such efficient degradation rate prediction methods can help simulate the stability of possible mRNA vaccines, eventually enabling vaccinology researchers to find mRNA vaccines that can be stored, distributd, and applied in-vivo efficiently.

REFERENCES

[1] A. Patel, M. A. Bah, and D. B. Weiner, "In vivo delivery of nucleic acid-encoded monoclonal antibodies," *BioDrugs*, pp. 1–21, 2020.

[2] N. Pardi, M. J. Hogan, F. W. Porter, and D. Weissman, "mrna vaccines—a new era in vaccinology," *Nature reviews Drug discovery*, vol. 17, no. 4, p. 261, 2018.

[3] T. T. Le, Z. Andreadakis, A. Kumar, R. G. Roman, S. Tollefsen, M. Saville, and S. Mayhew, "The covid-19 vaccine development landscape," *Nat Rev Drug Discov*, vol. 19, no. 5, pp. 305–306, 2020.

[4] Stanford University, "Openvaccine: Covid-19 mrna vaccine degradation prediction," Sep. 2020. [Online]. Available: https://www.kaggle.com/c/stanford-covid-vaccine

[5] L. A. Jackson, E. J. Anderson, N. G. Rouphael, P. C. Roberts, M. Makhene, R. N. Coler, M. P. McCullough, J. D. Chappell, M. R. Denison, L. J. Stevens *et al.*, "An mrna vaccine against sars-cov-2—preliminary report," *New England Journal of Medicine*, 2020.

[6] ModernaTX Inc., "A study to evaluate efficacy, safety, and immunogenicity of mrna-1273 vaccine in adults aged 18 years and older to prevent covid-19." [Online]. Available: https://ClinicalTrials.gov/show/NCT04470427

[7] N.-N. Zhang, X.-F. Li, Y.-Q. Deng, H. Zhao, Y.-J. Huang, G. Yang, W.-J. Huang, P. Gao, C. Zhou, R.-R. Zhang *et al.*, "A thermostable mrna vaccine against covid-19," *Cell*, vol. 182, no. 5, pp. 1271–1283, 2020.

[8] P. Danaee, M. Rouches, M. Wiley, D. Deng, L. Huang, and D. Hendrix, "bprna: large-scale automated annotation and analysis of rna secondary structure," *Nucleic acids research*, vol. 46, no. 11, pp. 5381–5394, 2018.

[9] H. K. Wayment-Steele, W. Kladwang, E. Participants, and R. Das, "Rna secondary structure packages ranked and improved by high-throughput experiments," *BioRxiv*, 2020.

[10] M. G. Seetin, W. Kladwang, J. P. Bida, and R. Das, "Massively parallel rna chemical mapping with a reduced bias map-seq protocol," in *RNA Folding*. Springer, 2014, pp. 95–117.

[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.

[12] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.

[13] F. Zou, L. Shen, Z. Jie, W. Zhang, and W. Liu, "A sufficient condition for convergences of adam and rmsprop," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019, pp. 11 127–11 135.

[14] S. Semeniuta, A. Severyn, and E. Barth, "Recurrent dropout without memory loss," *arXiv preprint arXiv:1603.05118*, 2016.

[15] S. Lee and C. Lee, "Revisiting spatial dropout for regularizing convolutional neural networks," *Multimedia Tools and Applications*, pp. 1–13, 2020.

[16] Arnab Khare, "Covid19 feature engineering xgboost," Sep. 2020. [Online]. Available: https://www.kaggle.com/arnabark/covid19-feature-engineering-xgboost/notebook

[17] Túlio de Freitas Castro, "Histgradientboosting baseline," Sep. 2020. [Online]. Available: https://www.kaggle.com/tuliofc/histgradientboosting-baseline