Supplementary information of RNAdegformer: Accurate Prediction of mRNA Degradation at Nucleotide Resolution with Deep Learning

Investigating structural preferences of mRNA degradation via model interpretation

To elucidate structural preferences of mRNA degradation, we analyze attention weights of our best single model ($k = 6$) and its attention weights right before outputting predictions. While it's not surprising to see that stem loops have low reactivity and our model predictions have low mean absolute error on stem loops, taking the column-wise sum of attention weights across different loop types, we see strikingly that a disproportionate amount of attention paid to dangling ends (Table S1). Inspecting some examples of attention weights of the last transformer layer, we see that the model pays lots of attention to the beginning of sequences, which are typically dangling ends (Figure S1). This can perhaps be explained by dangling ends in the beginning loosely intra-molecularly interacting with the nucleotides in the rest of the entire sequence, which is probably very difficult to capture; therefore, the model dedicates a disproportionate amount of attention to understanding these interactions. Comparing the Pearson R correlation coefficients of our model predictions of loop types, we see that our model shows the best correlation with dangling ends, while excluding the first 5 positions when doing the same analysis shows decreased correlation on dangling ends reactivity (Table S1). These results indicate that indeed the model is correctly learning the complex interactions of dangling ends in the beginning that contribute quite a bit to the degradation/stability of mRNA [1, 2].

Selection of hyperparameters and robustness against different hyperparameters

During the competition, we first selected an initial set of hyperparameters based on experience and then experimented with batch size, epochs trained, etc. We experimented with one hyperparameter at a time (e.g. for batch size we tried 8,16,32), and kept using the best one while testing the rest. Compared to grid search, which would involve many more sets of hyperparameters to test, our approach is much more efficient, especially during the competition, where time was very limited. For instance, if we want to test 3 different hyperparameters, each with 3 values, our approach would be to test one a time 3 times for a total of 9 sets of hyperparameters (3+3+3), whereas grid search would mean a total of 27 (3x3x3). Because the competition had an independent private test set that is bigger and more diverse, our relatively less aggressive approach also resulted in less overfitting to the validation and public test set, as evidenced by our improved placing in private test compared to public. Following the competition, we only made small adjustments to hyperparameters. By far, we have found k to have the biggest impact on performance, since larger k-mer convolutions can enable the model to learn more high level interactions but also run the risk of overfitting (Figure S3). In practice, we found ensembling different ks by simply averaging predictions from models with differernt ks to provide a good boost to performance. In addition, we tested the (unsupervised+supervised) RNAdegformer's robustness against different hyperparameters by rerunning training of RNADegformer while changing hyperparameters one at a time and see that RNAdegformer is relatively robust against different hyperparameters. Batch sizes of 8, 16, and 32 all provide similar performance (Figure S3). Further, we varied the number of epochs trained during supervised learning and found that RNAdegformer's performance remains good across different numbers of epochs trained (Figure S4). As for α and β, we found that RNAdegformer can perform well under different degree of error weighting but using $\alpha = 0.5$ and $\beta = 5$ gives slightly better results in both public and private test set (Figure S5). Similarly, RNAdegformer is robust against different weight decay values but 0.1 seems to be the optimal value by a small margin (Figure S6).

Bibliography

[1] Salvatore Bommarito, Nicolas Peyret, and John Santalucia Jr. Thermodynamic parameters for dna sequences with dangling ends. *Nucleic Acids Research*, 28(9):1929–1934, 2000. doi: 10.1093/nar/28.9.1929.

[2] Tatsuo Ohmichi, Shu-Ichi Nakano, Daisuke Miyoshi, and Naoki Sugimoto. Long rna dangling end has large energetic contribution to duplex stability. *Journal of the American Chemical Society*, 124(35):10367–10372, 2002. doi: 10.1021/ja0255406.

loop type	avg_reactivity	avg_attention_weight	loop_avg_ubpp	reactivity_mae	bpp_pearsonr	react_pearsonr
bulge	0.8979±0.5929	0.7873±0.1291	0.8791±0.1617	0.2988	0.2944	0.7676
dangling End	0.6133±0.4434	2.6260±5.2492	0.7303±0.2732	0.2099	0.5490	0.7945
Hairpin	0.7919±0.5936	0.7842±0.1283	0.8477±0.2043	0.2733	0.3092	0.7529
Internal	0.5288±0.4401	0.7859±0.1401	0.8090±0.2280	0.2217	0.2002	0.7016
Multi	0.6075±0.4144	0.7925±0.1325	0.7830±0.2296	0.2215	0.3775	0.7201
Stem	0.1330±0.2073	0.7945±0.3489	0.1073±0.1236	0.0782	0.5427	0.7012
eXternal	0.5264±0.3657	0.8057±0.1412	0.7204±0.2623	0.1830	0.4799	0.7790

Table S1: Analysis of attention weights against different loop types. The columns correspond to avg_reactivity: average SHAPE reactivity for the paricular loop type, avg_attention: average attention(column) sum for the paricular loop type, loop_avg_ubpp: average unpaired probability for the paricular loop type, NT_mae: mean absolute error of RNAdegformer predictions for the particular loop type, upp_pearsonr: Pearson R correlation of unpaired probability with SHAPE reactivity, NT_pearsonr: Pearson R correlation of RNAdegformer predictions with SHAPE reactivity

loop type	avg_reactivity	avg_attention_weight	loop_avg_ubpp	reactivity_mae	bpp_pearsonr	react_pearsonr
bulge	0.8975±0.5932	0.7867±0.1272	0.8792±0.1618	0.2987	0.2950	0.7677
dangling End	0.4734±0.3856	0.7791±0.1358	0.6326±0.2773	0.1718	0.4861	0.7749
Hairpin	0.7918±0.5937	0.7842±0.1283	0.8476±0.2044	0.2734	0.3092	0.7530
Internal	0.5286±0.4403	0.7847±0.1298	0.8087±0.2281	0.2218	0.1999	0.7016
Multi	0.6075±0.4144	0.7925±0.1325	0.7830±0.2296	0.2215	0.3775	0.7201
Stem	0.1327±0.2070	0.7883±0.1313	0.1069±0.1231	0.0779	0.5417	0.7019
eXternal	0.5264±0.3657	0.8057±0.1412	0.7204±0.2623	0.1830	0.4799	0.7790

Table S2: Analysis of attention weights against different loop types, while excluding the first 5 positions

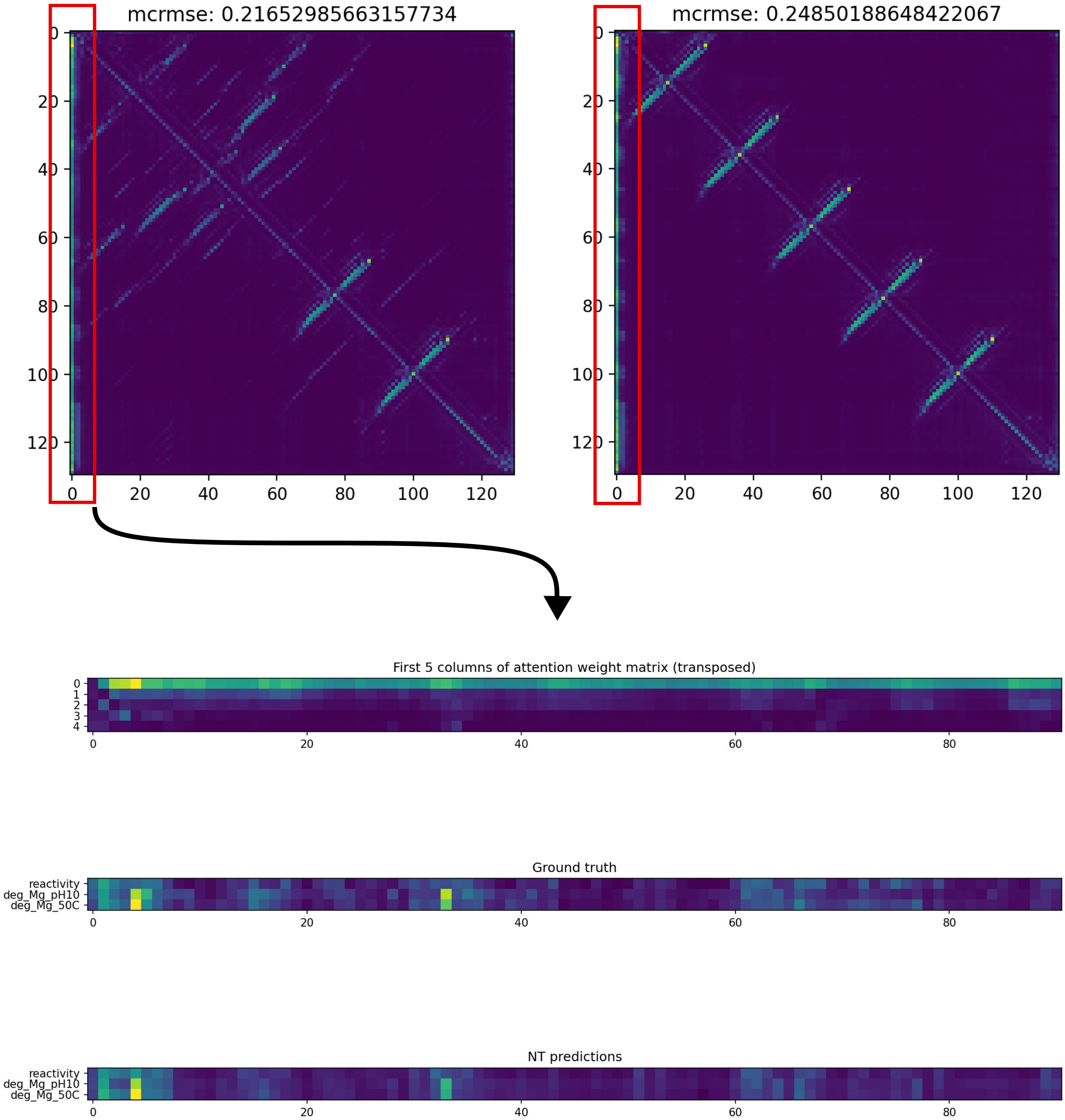

Figure S1: Examples of the transformer scanning the whole sequnece in the first few positions which are dangling ends. The transformer dedicates much of its attention to the beginning of the sequence which then interacts with the entire sequence. In the first example, transposing and visualizing the first 5 columns, we notice that the network paying attention to positions of high degradation in the beginning, while scanning the rest of the sequence. Note that only ground truth values for first 91 positions are available.

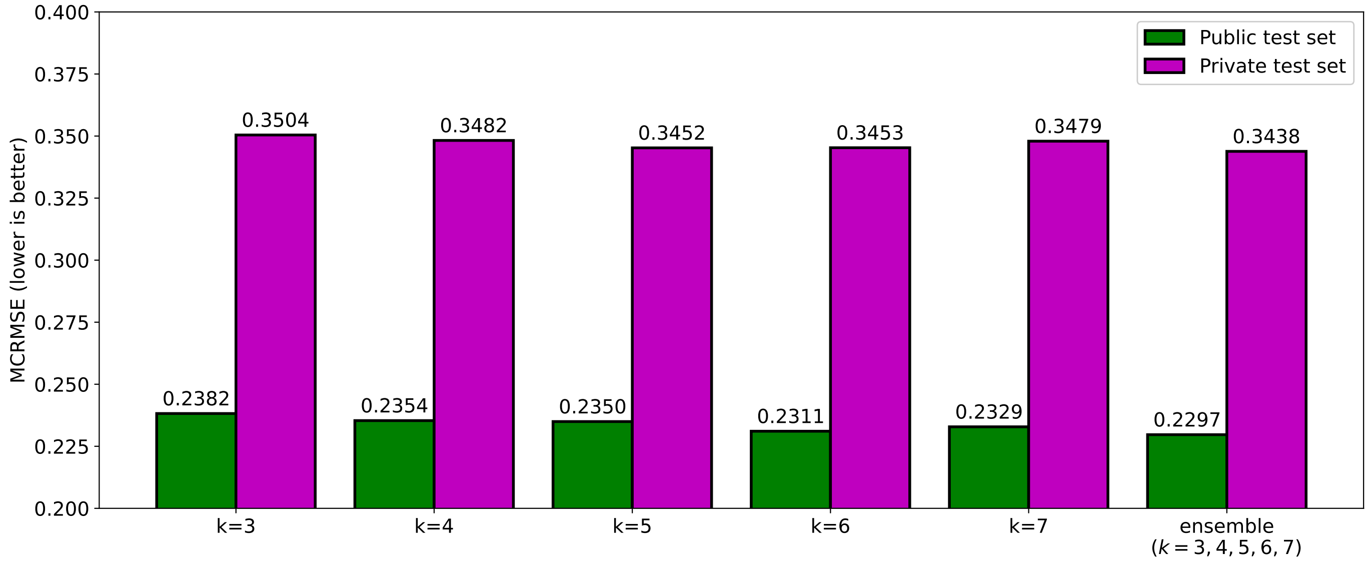

Figure S2: RNAdegformer's performance with different ks.

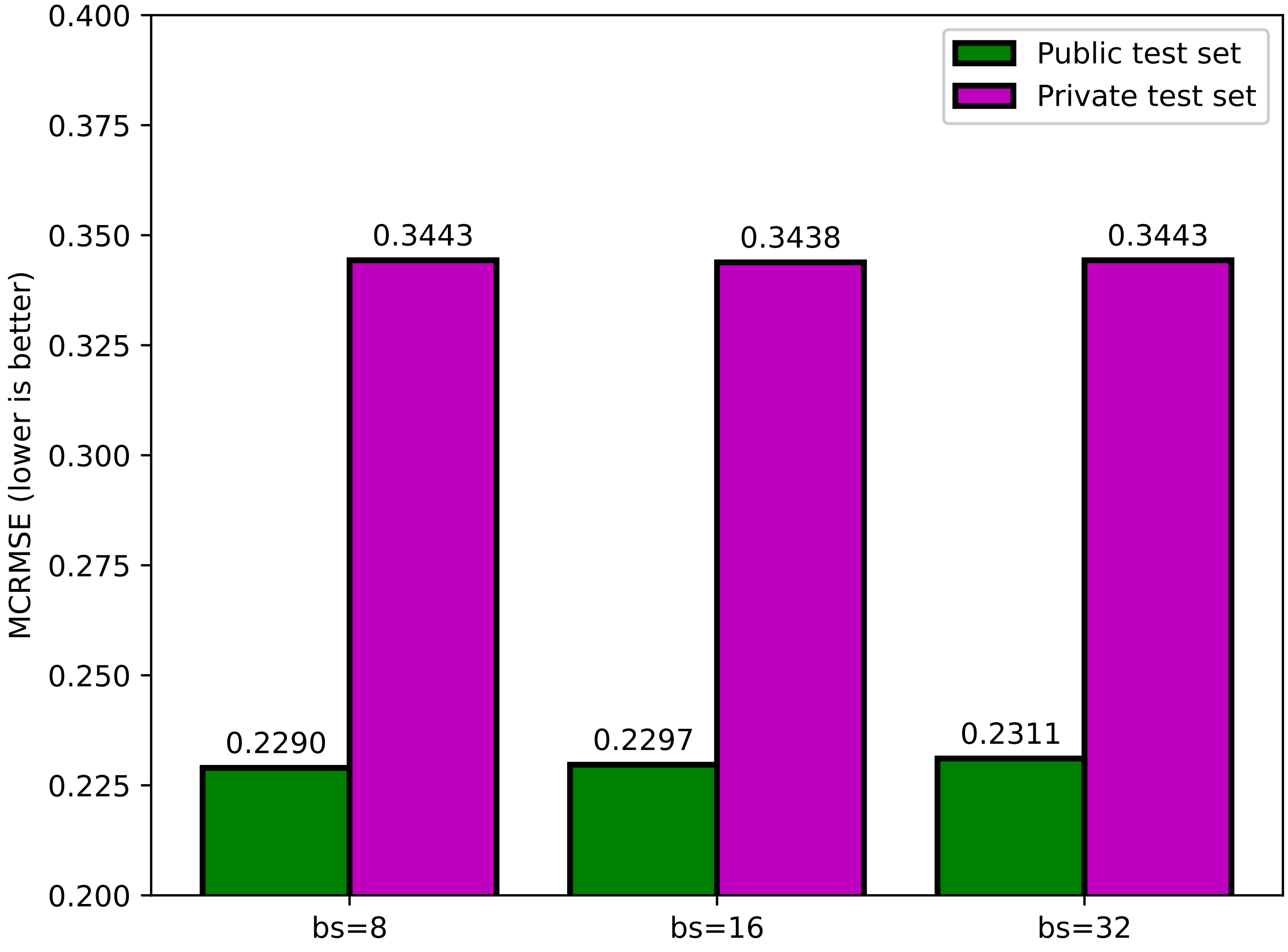

Figure S3: Robustness of RNAdegformer against batch size used.

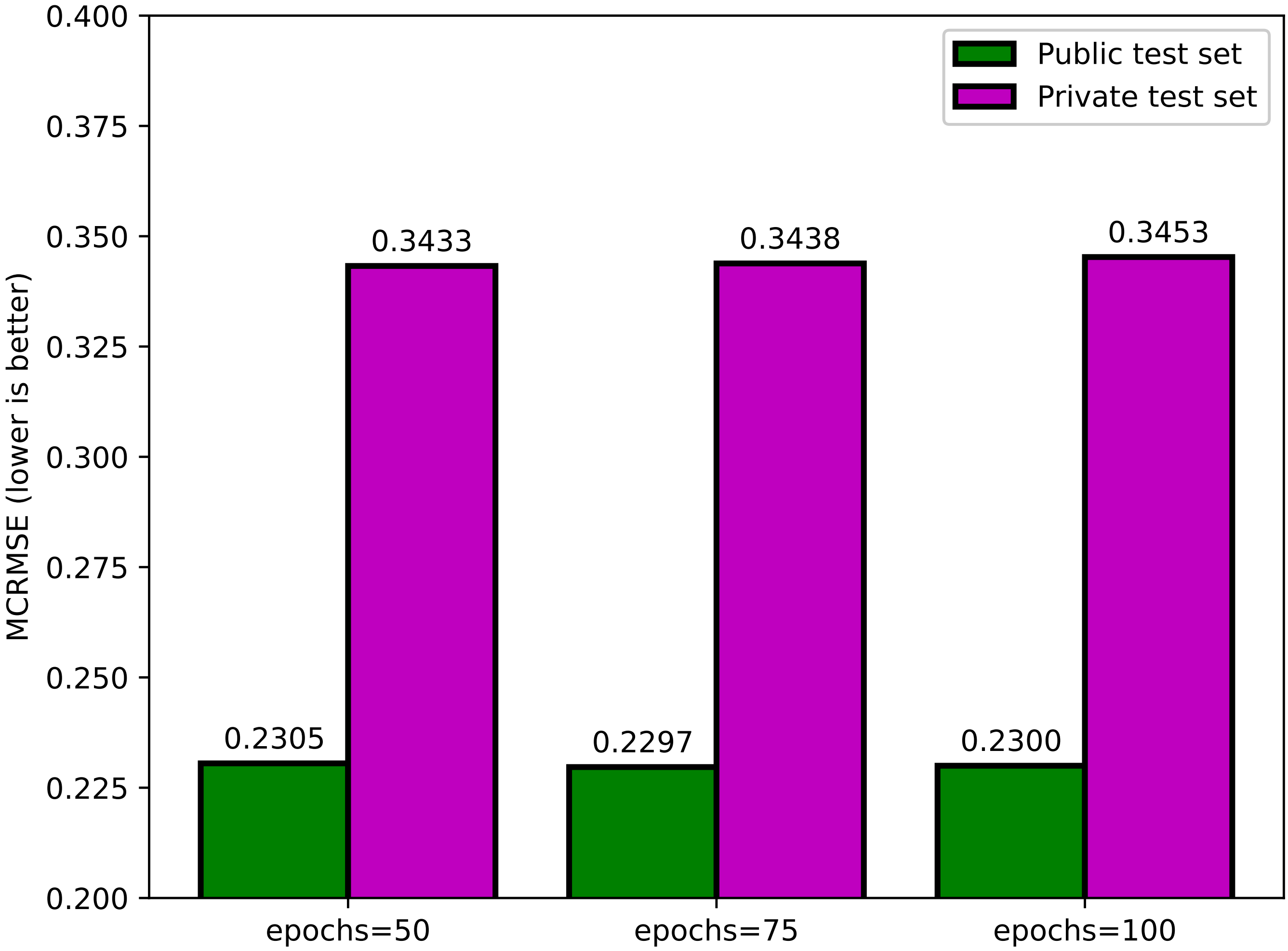

Figure S4: Robustness of RNAdegformer against number of epochs trained.

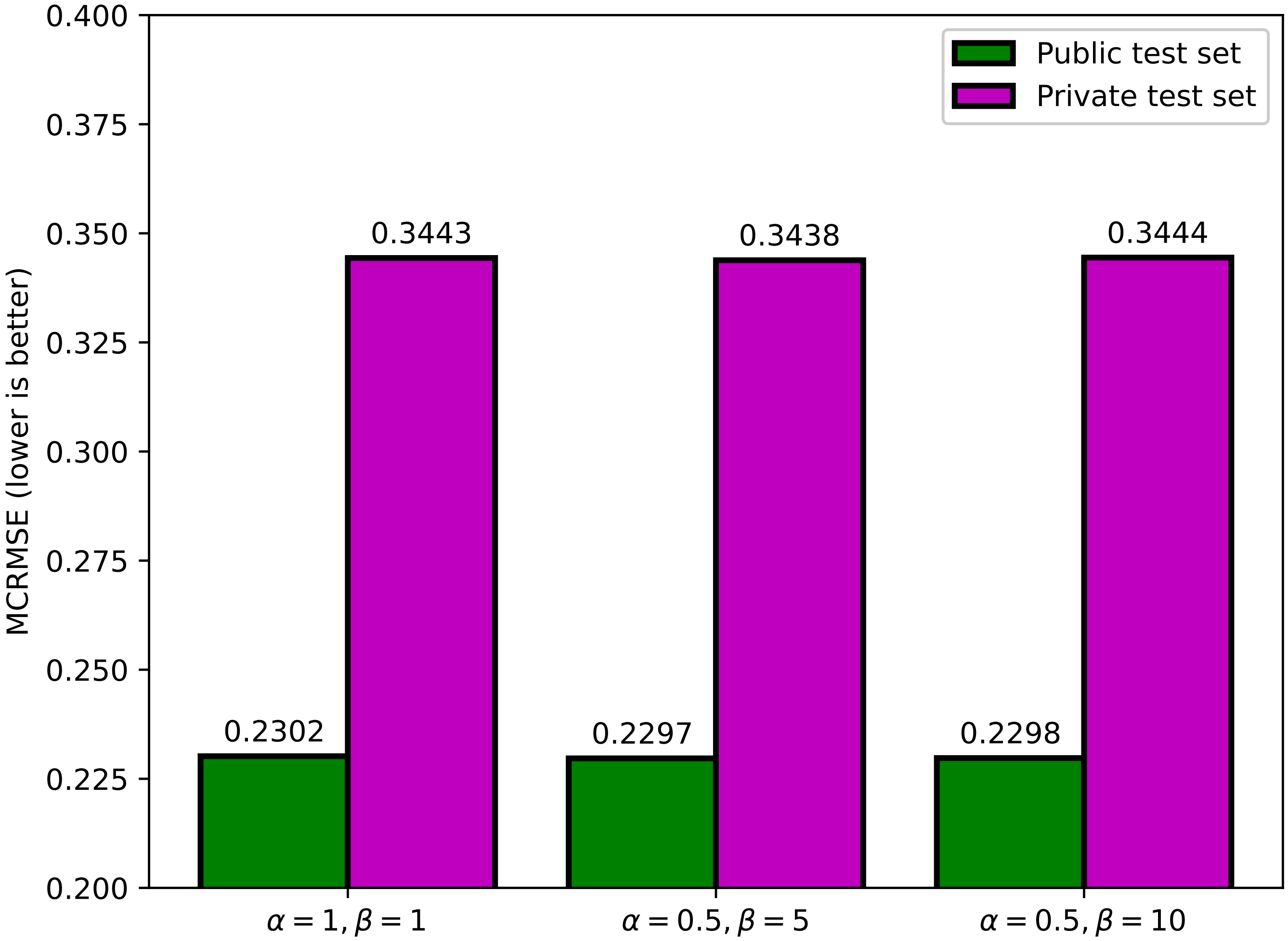

Figure S5: Robustness of RNAdegformer against different α and β values.

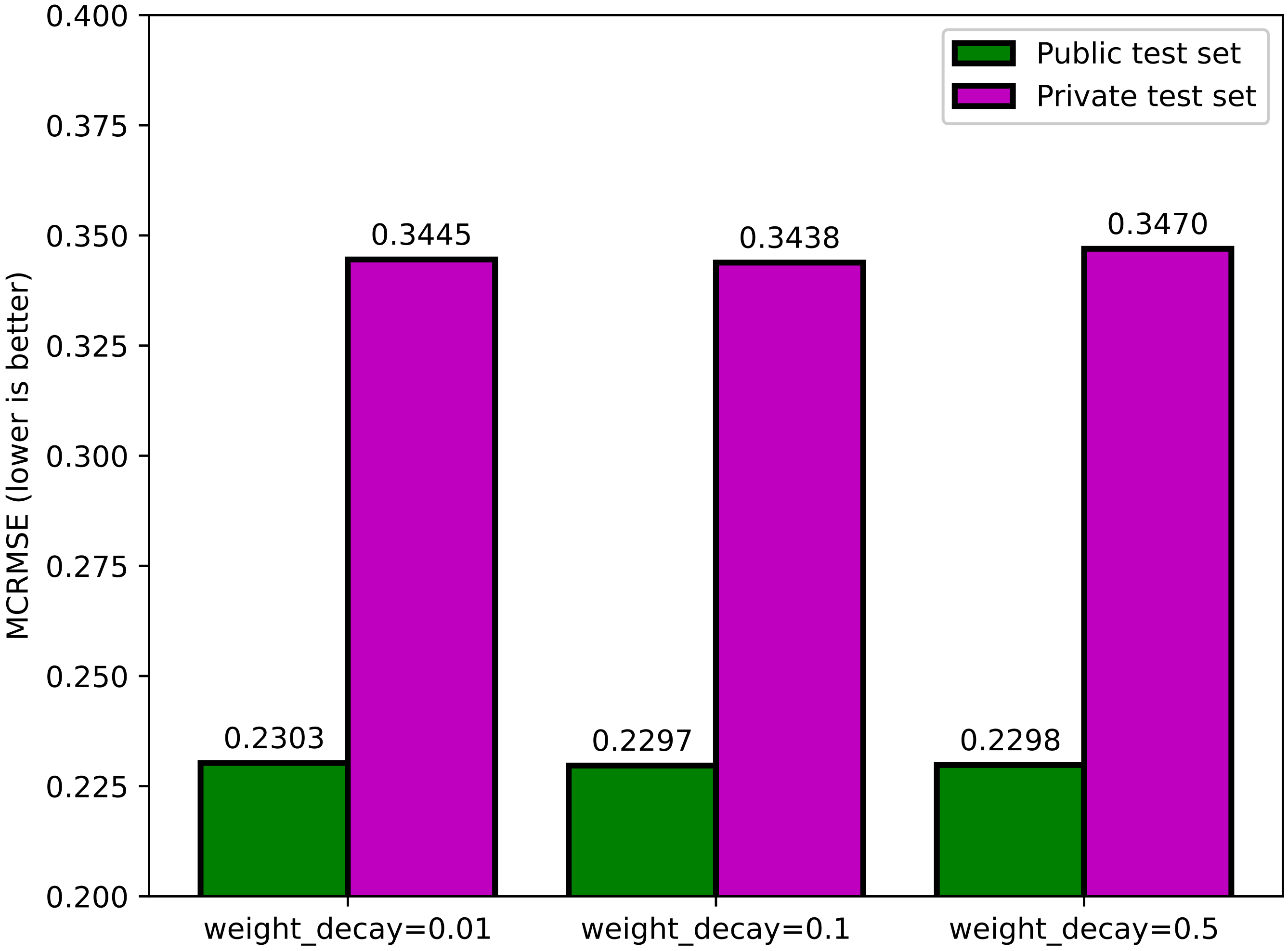

Figure S6: Robustness of RNAdegformer against different weight decay values.