miRe2e: a full end-to-end deep model based on Transformers for prediction of pre-miRNAs

J. Raad, L. Bugnon, D.H. Milone and G. Stegmayer

Research Institute for Signals, Systems and Computational Intelligence sinc(i)
(FICH-UNL/CONICET), Ciudad Universitaria, Santa Fe, Argentina.

Supplementary Material

PyTorch[1] was used to build and train the deep learning models. Our models were trained on a Nvidia Titan V GPU with 12 Gb of RAM. The architecture of the neural models are detailed in the following tables. We evaluated several loss functions, optimizers and learning rates on training data.

The selected loss functions were: Mean Squared Error (MSE) for the Structure prediction model and the MFE estimation model; and Focal Loss (FL) (Lin *et al.*, 2017) for the pre-miRNA classifier. The FL adds an extra factor to the standard cross entropy criterion, which allows reducing the relative loss for well-classified examples and puts more focus on hard, misclassified examples. The FL used here was

$$FL(p_\tau) = -\alpha(1 - p_\tau)^\gamma \log(p_\tau), \tag{1}$$

where p_τ is the predicted probability (output score) for the sequence under analysis, the parameter γ can be used to increase or reduce the weight given to those samples that are correctly classified, and α is a weighting factor to address class imbalance. We have used $\alpha = 1.0$ and $\gamma = 4.0$.

The optimizer selected was Stochastic Gradient Descent (SGD) with Nesterov momentum (Sutskever *et al.*, 2013), and a learning rate of 10^{-3}. Regarding the training process, each module was pre-trained separately and in cascade, using the outputs of the previous pre-trained model as inputs to the next model. No fine-tuning of the complete model was required. More implementation details are provided in the following tables and the source code[2].

[1] https://pytorch.org/
[2] https://github.com/sinc-lab/miRe2e

Table 1: Structure predictor.

Layer (type)	Output shape	Param #
ReLU-1	[4, 100]	0
BatchNorm1d-2	[4, 100]	8
Conv1d-3	[111, 100]	1,443
ReLU-4	[111, 100]	0
BatchNorm1d-5	[111, 100]	222
Conv1d-6	[111, 100]	37,074
ReLU-7	[111, 100]	0
BatchNorm1d-8	[111, 100]	222
Conv1d-9	[111, 100]	37,074
ResNet-10	[111, 100]	0
ReLU-11	[111, 100]	0
BatchNorm1d-12	[111, 100]	222
Conv1d-13	[111, 100]	37,074
ReLU-14	[111, 100]	0
BatchNorm1d-15	[111, 100]	222
Conv1d-16	[111, 100]	37,074
ResNet-17	[111, 100]	0
ReLU-18	[111, 100]	0
BatchNorm1d-19	[111, 100]	222
Conv1d-20	[111, 100]	37,074
ReLU-21	[111, 100]	0
BatchNorm1d-22	[-1, 111, 100]	222
Conv1d-23	[111, 100]	37,074
ResNet-24	[111, 100]	0
EncoderStr-25	[111, 100]	0
MultiheadAttention-26	[[2, 222], [100, 100]]	0
Dropout-27	[2, 222]	0
LayerNorm-28	[2, 222]	444
Linear-29	[2, 888]	198,024
Dropout-30	[2, 888]	0
Linear-31	[2, 222]	197,358
Dropout-32	[2, 222]	0
LayerNorm-33	[2, 222]	444
TransformerEncoderLayer-34	[2, 222]	0
MultiheadAttention-35	[[2, 222], [100, 100]]	0
Dropout-36	[2, 222]	0
LayerNorm-37	[2, 222]	444
Linear-38	[2, 888]	198,024
Dropout-39	[2, 888]	0
Linear-40	[2, 222]	197,358
Dropout-41	[2, 222]	0
LayerNorm-42	[2, 222]	444
TransformerEncoderLayer-43	[2, 222]	0
MultiheadAttention-44	[[2, 222], [100, 100]]	0
Dropout-45	[2, 222]	0
LayerNorm-46	[2, 222]	444
Linear-47	[2, 888]	198,024
Dropout-48	[2, 888]	0
Linear-49	[2, 222]	197,358
Dropout-50	[2, 222]	0
LayerNorm-51	[2, 222]	444
TransformerEncoderLayer-52	[2, 222]	0
MultiheadAttention-53	[[2, 222], [100, 100]]	0
Dropout-54	[2, 222]	0
LayerNorm-55	[2, 222]	444
Linear-56	[2, 888]	198,024
Dropout-57	[2, 888]	0
Linear-58	[2, 222]	197,358
Dropout-59	[2, 222]	0
LayerNorm-60	[2, 222]	444
TransformerEncoderLayer-61	[2, 222]	0
MultiheadAttention-62	[[2, 222], [100, 100]]	0
Dropout-63	[2, 222]	0
LayerNorm-64	[2, 222]	444
Linear-65	[2, 888]	198,024
Dropout-66	[2, 888]	0

Linear-67	[2, 222]	197,358
Dropout-68	[2, 222]	0
LayerNorm-69	[2, 222]	444
TransformerEncoderLayer-70	[2, 222]	0
MultiheadAttention-71	[[2, 222], [100, 100]]	0
Dropout-72	[2, 222]	0
LayerNorm-73	[2, 222]	444
Linear-74	[2, 888]	198,024
Dropout-75	[2, 888]	0
Linear-76	[2, 222]	197,358
Dropout-77	[2, 222]	0
LayerNorm-78	[2, 222]	444
TransformerEncoderLayer-79	[2, 222]	0
TransformerEncoder-80	[2, 222]	0
Dropout-81	[100, 222]	0
Linear-82	[100, 100]	22,300
ELU-83	[100, 100]	0
Dropout-84	[100, 100]	0
Linear-85	[100, 10]	1,010
ELU-86	[100, 10]	0
Linear-87	[100, 1]	11
Tanh-88	[100, 1]	0

Table 2: MFE estimation model.

Layer (type)	Output shape	Param #
ReLU-1	[5, 100]	0
BatchNorm1d-2	[5, 100]	10
Conv1d-3	[64, 100]	1,024
ReLU-4	[64, 100]	0
BatchNorm1d-5	[64, 100]	128
Conv1d-6	[64, 100]	12,352
ReLU-7	[64, 100]	0
BatchNorm1d-8	[64, 100]	128
Conv1d-9	[64, 100]	12,352
ResNet-10	[64, 100]	0
AvgPool1d-11	[64, 50]	0
ReLU-12	[64, 50]	0
BatchNorm1d-13	[64, 50]	128
Conv1d-14	[64, 50]	12,352
ReLU-15	[64, 50]	0
BatchNorm1d-16	[64, 50]	128
Conv1d-17	[64, 50]	12,352
ResNet-18	[64, 50]	0
AvgPool1d-19	[64, 25]	0
ReLU-20	[64, 25]	0
BatchNorm1d-21	[64, 25]	128
Conv1d-22	[64, 25]	12,352
ReLU-23	[64, 25]	0
BatchNorm1d-24	[64, 25]	128
Conv1d-25	[64, 25]	12,352
ResNet-26	[64, 25]	0
AvgPool1d-27	[64, 12]	0
Encoder-28	[64, 12]	0
Linear-29	[100]	76,900
ELU-30	[100]	0
BatchNorm1d-31	[100]	200
Linear-32	[30]	3,030
ELU-33	[30]	0
BatchNorm1d-34	[30]	60
Linear-35	[1]	31
ELU-36	[1]	0

Table 3: Pre-miRNA classifier.

Layer (type)	Output shape	Param #
ReLU-1	[5, 100]	0
BatchNorm1d-2	[5, 100]	10
Conv1d-3	[64, 100]	1,024
ReLU-4	[64, 100]	0
BatchNorm1d-5	[64, 100]	128
Conv1d-6	[64, 100]	12,352
ReLU-7	[64, 100]	0
BatchNorm1d-8	[64, 100]	128
Conv1d-9	[64, 100]	12,352
ResNet-10	[64, 100]	0
ReLU-11	[64, 100]	0
BatchNorm1d-12	[64, 100]	128
Conv1d-13	[64, 100]	12,352
ReLU-14	[64, 100]	0
BatchNorm1d-15	[64, 100]	128
Conv1d-16	[64, 100]	12,352
ResNet-17	[64, 100]	0
ReLU-18	[64, 100]	0
BatchNorm1d-19	[64, 100]	128
Conv1d-20	[64, 100]	12,352
ReLU-21	[64, 100]	0
BatchNorm1d-22	[64, 100]	128
Conv1d-23	[64, 100]	12,352
ResNet-24	[64, 100]	0
AvgPool1d-25	[64, 50]	0
ReLU-26	[64, 50]	0
BatchNorm1d-27	[64, 50]	128
Conv1d-28	[64, 50]	12,352
ReLU-29	[64, 50]	0
BatchNorm1d-30	[64, 50]	128
Conv1d-31	[64, 50]	12,352
ResNet-32	[64, 50]	0
ReLU-33	[64, 50]	0
BatchNorm1d-34	[64, 50]	128
Conv1d-35	[64, 50]	12,352
ReLU-36	[64, 50]	0
BatchNorm1d-37	[64, 50]	128
Conv1d-38	[64, 50]	12,352
ResNet-39	[64, 50]	0
ReLU-40	[64, 50]	0
BatchNorm1d-41	[64, 50]	128
Conv1d-42	[64, 50]	12,352
ReLU-43	[64, 50]	0
BatchNorm1d-44	[64, 50]	128
Conv1d-45	[64, 50]	12,352
ResNet-46	[64, 50]	0
AvgPool1d-47	[64, 25]	0
ReLU-48	[64, 25]	0
BatchNorm1d-49	[64, 25]	128
Conv1d-50	[64, 25]	12,352
ReLU-51	[64, 25]	0
BatchNorm1d-52	[64, 25]	128
Conv1d-53	[64, 25]	12,352
ResNet-54	[64, 25]	0
ReLU-55	[64, 25]	0
BatchNorm1d-56	[64, 25]	128
Conv1d-57	[64, 25]	12,352
ReLU-58	[64, 25]	0
BatchNorm1d-59	[64, 25]	128
Conv1d-60	[64, 25]	12,352
ResNet-61	[64, 25]	0
ReLU-62	[64, 25]	0
BatchNorm1d-63	[64, 25]	128
Conv1d-64	[64, 25]	12,352
ReLU-65	[64, 25]	0
BatchNorm1d-66	[64, 25]	128

Conv1d-67	[64, 25]	12,352
ResNet-68	[64, 25]	0
AvgPool1d-69	[64, 12]	0
ReLU-70	[64, 12]	0
BatchNorm1d-71	[64, 12]	128
Conv1d-72	[64, 12]	12,352
ReLU-73	[64, 12]	0
BatchNorm1d-74	[64, 12]	128
Conv1d-75	[64, 12]	12,352
ResNet-76	[64, 12]	0
ReLU-77	[64, 12]	0
BatchNorm1d-78	[64, 12]	128
Conv1d-79	[64, 12]	12,352
ReLU-80	[64, 12]	0
BatchNorm1d-81	[64, 12]	128
Conv1d-82	[64, 12]	12,352
ResNet-83	[64, 12]	0
ReLU-84	[64, 12]	0
BatchNorm1d-85	[64, 12]	128
Conv1d-86	[64, 12]	12,352
ReLU-87	[64, 12]	0
BatchNorm1d-88	[64, 12]	128
Conv1d-89	[64, 12]	12,352
ResNet-90	[64, 12]	0
AvgPool1d-91	[64, 6]	0
Encoder-92	[64, 6]	0
PositionalEncoder-93	[6, 64]	0
MultiheadAttention-94	[[2, 64], [6, 6]]	0
Dropout-95	[2, 64]	0
LayerNorm-96	[2, 64]	128
Linear-97	[2, 256]	16,64
Dropout-98	[2, 256]	0
Linear-99	[2, 64]	16,448
Dropout-100	[2, 64]	0
LayerNorm-101	[2, 64]	128
TransformerEncoderLayer-102	[2, 64]	0
MultiheadAttention-103	[[2, 64], [6, 6]]	0
Dropout-104	[2, 64]	0
LayerNorm-105	[2, 64]	128
Linear-106	[2, 256]	16,64
Dropout-107	[2, 256]	0
Linear-108	[2, 64]	16,448
Dropout-109	[2, 64]	0
LayerNorm-110	[2, 64]	128
TransformerEncoderLayer-111	[2, 64]	0
MultiheadAttention-112	[[2, 64], [6, 6]]	0
Dropout-113	[2, 64]	0
LayerNorm-114	[2, 64]	128
Linear-115	[2, 256]	16,64
Dropout-116	[2, 256]	0
Linear-117	[2, 64]	16,448
Dropout-118	[2, 64]	0
LayerNorm-119	[2, 64]	128
TransformerEncoderLayer-120	[2, 64]	0
TransformerEncoder-121	[2, 64]	0
BatchNorm1d-122	[385]	770
Linear-123	[1000]	386
ELU-124	[1000]	0
BatchNorm1d-125	[1000]	2
Dropout-126	[1000]	0
Linear-127	[1000]	1,001,000
Linear-128	[1000]	1,001,000
Linear-129	[2]	2,002
Softmax-130	[2]	0

References

Lin, T.-Y. *et al.* (2017). Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2980–2988.

Sutskever, I. *et al.* (2013). On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28, pages 1139–1147.