1. Encoder Stacks (Siamese structure):
   * RNA Encoder Stack: • Multiple layers of self-attention and feed-forward networks • Layer normalization and residual connections
   * RBP Encoder Stack: • Identical structure to RNA Encoder Stack • Shared weights with RNA Encoder Stack
2. Latent Space:
   * Projection of encoder outputs into a shared latent space
3. Cross-Attention Layer:
   * Bidirectional attention between RNA and RBP representations in latent space
4. Fusion Layer:
   * Combine latent representations from both encoder stacks
5. Decoder Stack (T5-like):
   * Multiple layers of self-attention, cross-attention, and feed-forward networks
   * Layer normalization and residual connections
   * Takes fused embedding as input for conditional generation
6. Generator (LM Head):
   * Linear layer projecting decoder output to vocabulary size
   * Softmax activation for probability distribution over amino acids
7. Training Objectives:
   * Triplet loss for Siamese network training
   * Reconstruction loss for autoencoder-like behavior
   * Generation loss for producing novel RBP sequences

Generative Aspect: The network becomes generative in the following ways:

1. Reconstruction: During training, the decoder stack learns to reconstruct the original RBP sequences, acting as an autoencoder.
2. Conditional Generation: After training, given an RNA motif embedding:
   * The RNA encoder processes the motif
   * The fused latent representation (combining RNA info and a learnable "template" RBP representation) is fed to the decoder
   * The decoder generates a novel RBP sequence tailored to the input RNA motif
3. Novelty in Generation: The shared latent space and the fusion of RNA and RBP information allow the model to generate RBP sequences that are novel yet relevant to the given RNA motif.
4. ESM Decoding: The generated sequences can be passed through the ESM model's decoder to obtain full protein representations.

Training Process:

1. Encode RNA and RBP sequences through their respective encoder stacks
2. Apply triplet loss in the latent space
3. Fuse latent representations
4. Decode and reconstruct original RBP sequences
5. Generate novel RBP sequences conditioned on RNA motifs
6. Apply reconstruction and generation losses

Training Process:

1. Prepare data:
   * RNA sequences and their corresponding RBP (RNA-binding protein) sequences
   * Positive pairs (known interactions) and negative pairs (non-interactions)
2. Define loss functions:
   * Contrastive loss (for Siamese network)
   * Cross-entropy loss (for sequence generation)
   * Optionally, add reconstruction loss
3. Training loop:
   * Forward pass through the model
   * Calculate losses
   * Backpropagate and update weights
4. Evaluation:
   * Binding prediction accuracy
   * Generated sequence quality (e.g., using perplexity or biological metrics)

Flow of data (a->b->c):

a) Input:

* RNA sequence (e.g., "AUGCUAGCUAGC")
* RBP sequence (e.g., "MKVLWAALLVTFLAGCQAKVEQAVETEPEPELRQQTEWQSGQRWELALGRFWDYLRWVQTLSEQVQEELLSSQVTQELRALMDETMKELKAYKSELEEQLTPVAEETRARLSKELQAAQARLGADVLASHGRLVQYRGEVQAMLGQSTEELRVRLASHLRKLRKRLLRDADDLQKRLAVYQAGAREGAERGLSAIRERLGPLVEQGRVRAATVGSLAGQPLQERAQAWGERLRARMEEMGSRTRDRLDEVKEQVAEVRAKLEEQAQQRLPAAAAPGAAQPRLPWELRLEARRRLAAPRGAHAQKRPLMTSAVSQALRQAAQQQAWPAQQQMQAATAAWAQAWQAQWQVWQAAQAAWLQAQQ")

b) Encoding:

1. Embed RNA and RBP sequences using shared embedding layer
2. Pass through T5 encoder stacks (self-attention + feed-forward)
3. Project encoded representations to latent space

c) Interaction and Fusion:

1. Apply cross-attention between RNA and RBP latent representations
2. Fuse the cross-attended representations

d) Decoding/Generation:

1. Use fused representation as input to the T5 decoder stack
2. Generate RBP sequence through iterative decoding

e) Output:

* Latent representations of RNA and RBP (for binding prediction)
* Generated RBP sequence (for protein design)

Losses:

1. Contrastive Loss (Siamese aspect): L\_contrastive = y \* D² + (1 - y) \* max(margin - D, 0)² Where:
   * D is the Euclidean distance between RNA and RBP latent representations
   * y is 1 for positive pairs, 0 for negative pairs
   * margin is a hyperparameter (e.g., 1.0)
2. Cross-Entropy Loss (Generation aspect): L\_ce = -Σ(t\_i \* log(p\_i)) Where:
   * t\_i is the true amino acid at position i
   * p\_i is the predicted probability for the correct amino acid at position i
3. Optional Reconstruction Loss: L\_rec = MSE(original\_sequence, reconstructed\_sequence)

Total Loss: L\_total = α \* L\_contrastive + β \* L\_ce + γ \* L\_rec

(α, β, γ are weighting factors)

This architecture allows for both binding prediction (using the latent representations and contrastive loss) and protein sequence generation (using the decoder and cross-entropy loss). The model learns to encode RNA and protein sequences into a meaningful latent space while also learning to generate protein sequences conditioned on RNA input.

Copy