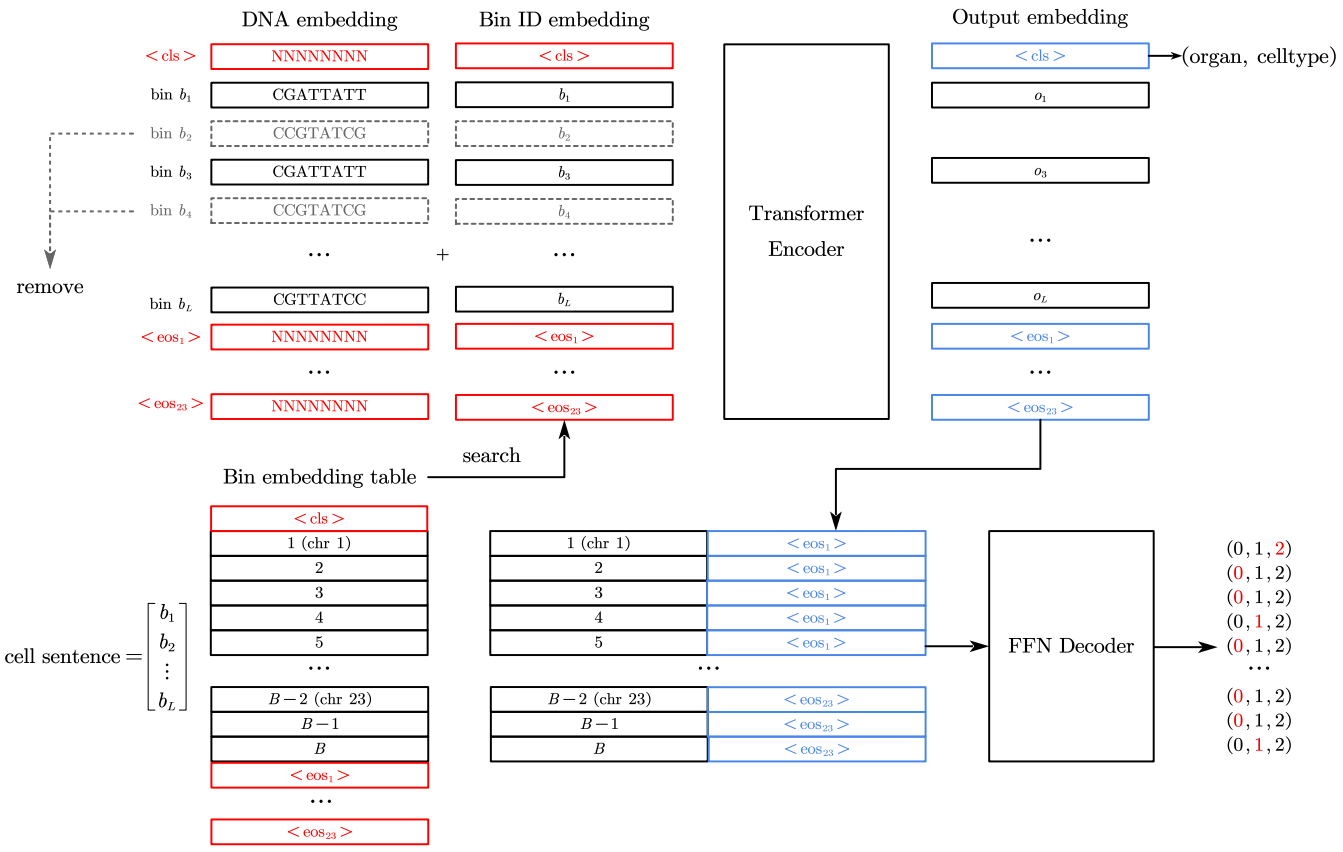
atacFormer Summary

GE Muyang

Feb 04, 2025

1 Current Framework



Suppose that we have B different ATAC bins in total. Each bin corresponds to a DNA sequence with length 5000, as in SnapATAC. In each cell, suppose that ATAC bins $b_1, b_2, \cdots, b_L$ are open. The cell sentence is formulated as $(b_1, b_2, \cdots, b_L)$. As the input to the model, some open bins will be removed from the sentence (as masked tokens in BERT).

Bin ID embedding

Since the vocabulary size is very large, to reduce the number of parameters, we use **embedding factorization**: The embedding matrix E (of size $V \times d$, where V is the vocabulary size and d is the embedding dimension) is factorized into:

- E_1: A smaller embedding matrix of size $V \times k$ (where $k \ll d$);
- E_2: A projection matrix of size $k \times d$, which maps the reduced k-dimensional embeddings to the final d-dimensional space.

In our case, we take $V = B + 25$, where 25 is for the embeddings of $< cls >$, $< pad >$ and $< eos_1 >, \cdots, < eos_{23} >$ for 23 chromosomes. We take $k = 64$ and $d = 512$.

DNA embedding

We plan to use **Nucleotide Transformer** to build DNA embedding table for all 5000-bp DNA sequence of each bin. For special tokens, the DNA embedding can be fixed to zero vector.

Transformer encoder

12 transformer encoder layers built by Flash Attention 1. The maximum sequence length (of cell sentence) is fixed to 5000.

Deocder and MLM loss

The output embeddings of $< eos_1 >, \cdots, < eos_{23} >$ (512 dimensional) are first compressed to 64 dimensional, and then concatenated with bin ID embeddings on the corresponding chromosome (e.g. $< eos_1 >$ is concatenated with all bins on chromosome 1) into 128-dimensional feature vectors. These feature vectors are feed into a feed-forward network (FFN) decoder to output the probability of 3 states: 0 (not in the cell sentence), 1 (in the cell sentence) and 2 (in the cell sentence but be removed).

Supervised loss

The output embedding corresponding to $< cls >$ is used to predict the organ and the celltype of the cell using FFN.

2 Future Perspective

Decoder by genomic regions

The $< eos >$ can be designed for different genomic regions instead of chromosomes. The following are possible annotated regions for ATAC bins.

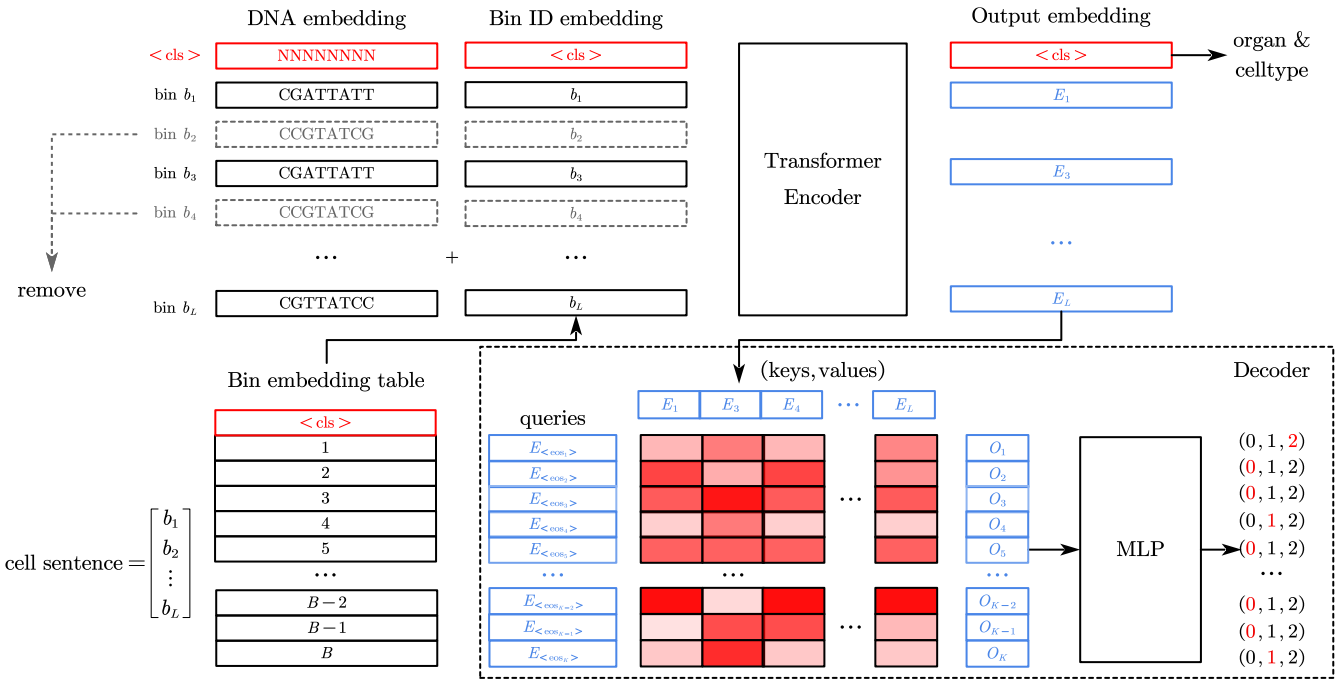
1. **Gene-related Regions**
   - **Promoter**: Located in the gene promoter region (typically 2 kb upstream to 0.5 kb downstream of the transcription start site).
   - **Exon**: Located in the exon region of a gene.
   - **Intron**: Located in the intron region of a gene.
   - **5' UTR**: Located in the 5' untranslated region of a gene.
   - **3' UTR**: Located in the 3' untranslated region of a gene.
   - **TTS (Transcription Termination Site)**: Located near the transcription termination site of a gene.
2. **Non-gene-related Regions**
   - **Intergenic**: Located in intergenic regions, far from any known genes.
   - **Non-coding RNA (ncRNA)**: Located in non-coding RNA gene regions.
3. **Regulatory Elements**
   - **Enhancer**: Located in known or predicted enhancer regions.
   - **Insulator**: Located in insulator regions.
   - **CTCF Binding Site**: Located in CTCF binding site regions.
4. **Other Features**
   - **Repeat**: Located in repeat regions (e.g., LINE, SINE, LTR).
   - **Conserved**: Located in evolutionarily conserved regions.

We can use **HOMER** to annotate ATAC bins. For each class c of bins, we use one $< eos_c >$ to predict the open/closed state of ATAC bins in this class.

Deocder by cross attention

Use independent $< eos >$ embedding vectors (not in the cell sentence).

- Use a Transformer encoder to encode the cell sentence, obtaining embeddings $E_{\text{encoder}} \in \mathbb{R}^{L \times d}$.
- Define $< eos_k >$ tokens corresponding to different regions. Each $< eos_k >$ token has an embedding $E_{<eos_k>} \in \mathbb{R}^d$.
- For each $< eos_k >$ token, compute cross-attention: $\text{Output}^k = \text{Attention}(E_{<eos_k>}, E_{\text{encoder}}, E_{\text{encoder}})$. Each output Output^k represents the interaction between the $< eos_k >$ token and the cell sentence.
- For each Output^i, predict the added/dropped bins in the corresponding functional region: $\text{Prediction}^i = \text{MLP}(\text{Output}^i)$



The advantages of this strategy are:

- Compared with self-attention, each $< eos >$ token can focus on a specific functional area and dynamically select bins related to it through cross-attention.
- Each $< eos >$ token can interact with the encoder output independently, which can avoid $< eos >$ tokens interacting with each other in the self-attention.

Contrastive learning

Randomly drop some tokens in the scATAC-seq data x_n of each cell n. Then find one cell m in the nearest neighborhood of cell n. Tokens in x_m but not in x_n are added to x'_n.

This data x'_n will serve as a positive sample of the original cell. The other cells will serve as negative samples. The data were then modeled by contrastive learning.

Another decoder will be responsible for predicting dropped/added tokens based on $< eos_k >$ tokens (in/not in the cell sentence).