Omics-based LLMs for Drug Discovery

CHRIS SOBCZAK

SFU

Based on Sheng et al. 2026 received in October 2025 and published in January 2026.

Outline

Drug discovery bottleneck

- Traditional drug discovery: target → screen → optimize
- Problems:
  - low success rate
  - high cost and time
  - struggles with complex diseases
- Shift from single targets → systems-level modeling

Omics

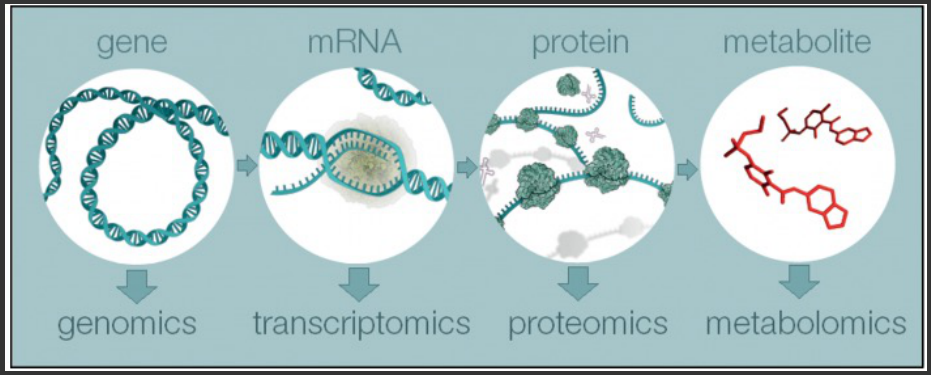

Figure 1: An overview of the research areas within *omics* (Tong and Bangzhuo 2023)

- Omics is a system-level measurement
- It provides high-dimensional, multi-layer biological state

Some challenges of omics

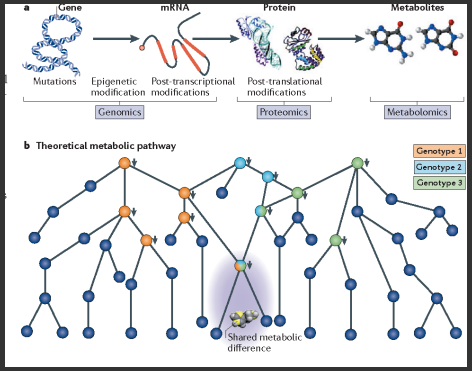

Figure 2: Graphical representation of omics (Yanes and Siuzdak 2012)

- Heterogeneity across data sources (multi-omics)
- Cross-scale integration (gene $\rightarrow$ cell $\rightarrow$ tissue)
- Noise and redundancy (these are highly correlated networks with a lot of noise)

New framing

- Can we treat omics data as a language?
- Can we apply LLMs to learn biological structure?

LLM architecture

- Input text → tokenization → embeddings → transformer → output
- Discrete → continuous → contextual → predictions

Tokens

Consider a vocabulary:

$$\mathcal{V} = \{\text{cat}, \text{dog}, \text{the}, \text{ferret}, \dots\} \tag{1}$$
$$= \{1, 2, 3, 4, \dots\} \tag{2}$$

- Each element in the vocabulary has an index
- The index of a token in vocabulary is converted into a vector $x_i \in \mathbb{R}^d$

$$t_i \in \{1, \dots, V\} \tag{3}$$

Tokens

- The x_is are arranged into the embedding matrix:

$$E \in \mathbb{R}^{V \times d}, \tag{4}$$

where V is the vocabulary size (ex: 50,000 tokens) and d is the embedding dimension (ex: 512, 1024)

- The embedding matrix is the collection of learned parameters
- It is initiallized randomly and iterated upon

Tokens

- A token is a discrete index into a finite vocabulary:
- In natural language processing (NLP), *cat* is associated with an index of a dictionary
- In genomics, the k-mer "ATCGA" is a token associated with another arbitrary index in a dictionary
- In scRNA the gene location and binned expression level is associated with a specific token
- Tokenization defines what measurable events exist
- What features are observable
- In NLP, these are words and phrases that carry context specific sentiments and meanings
- Tokenization identifies the signal

Tokens

- The *vector representation* of natural language is a discrete encoding of *tokens*
- Tokens in the context of natural language are words, phrases or parts of words that carry a contextual meaning
- To adapt LLMs for use in omics, a new tokenization scheme is required

Table 1 Representative omics data for language models.

Modality	Data form	Data source	Tokenization strategies
Genomics (DNA)	DNA sequences (A/T/C/G)	Human genome (∼2.75 B nucleotide bases)	Nucleotide as token: k-mer
		Human genome (∼2.75 B nucleotide bases), multi-species genome references (∼32.5 B nucleotide bases)	Nucleotide as token: byte-pair encoding (BPE)
Transcriptomics (mRNA)	Transcript sequences (A/U/C/G)	ncRNA sequences (∼23 M)	Nucleotide as token: one-hot encoding
Protein sequences/ structures	Amino acid sequences, MSAs, 3D structures	UniRef (∼65 M protein sequences)	Residue as token
Single-cell transcriptomics	Cell-by-gene expression matrix	PanglaoDB (∼1.1 M human cells, 74 tissues, 209 datasets)	Gene as token: binning for value
		CELLxGENE (∼33 M normal human cells)	
		Genecorpus-30 M (∼30 M human scRNA-seq cells from 561 datasets, multiple public sources)	Gene as token: Ranking for value
Spatial transcriptomics	Cell-by-gene expression matrix + 2D coordinates	∼11.4 M cells (∼8.7 M scRNA-seq, ∼2.7 M SRT) from public datasets (HCA, HTCA, GEO, CosMx SMI)	Cell as tokens

Figure 3: Omics tokenization strategies (Sheng et al. 2026)

Embedding

$$E = \begin{bmatrix} \text{embedding of token 1} \\ \text{embedding of token 2} \\ \vdots \\ \text{embedding of token } V \end{bmatrix} \tag{5}$$

If $\text{token}_i = 4$:

$$x_i = E[4],\ x_i \in \mathbb{R}^d \tag{6}$$

d is the number of features used to describe a token; latent variables or coordinates

Embedding

The model learns / estimates E such that tokens used in similar contexts are similar vectors:

$$||x_{\text{cat}} - x_{\text{dog}}|| \text{ is small} \tag{7}$$

$$||x_{\text{cat}} - x_{\text{hard drive}}|| \text{ is large} \tag{8}$$

The embedding encodes:

- co-occurrence structure
- functional similarity
- context

In omics, this could be:

- genes that co-express $\rightarrow$ mathematically similar embeddings
- k-mers in similar regulatory regions $\rightarrow$ similar embeddings

Embedding space

- $\mathbb{R}^{V \times d}$ is the space of all possible embedding matrices
- Each model comes up with an estimate $\hat{E}$ in this space
- Training the LLM is estimating $E \in \mathbb{R}^{V \times d}$
- Training the LLM constructs an embedding to create distance between unrelated objects (tokens), and position related ones close to each other
- Defines a *metric space* on tokens

In genomics:

- Before embedding, genes are unordered labels
- After embedding, genes are mapped into a continuous space
- Distances should represent biological similarity

Embedding

Each dimension of the embedding describes some kind of concept like:

- Animalness
- Evilness
- Action related
- Grammatical role

The vector encodes relative similarity, not absolute meaning

- Closeness of tokens could mean co-expression
- Shared pathways
- Similar regulatory roles
- Embedding = learned representation of gene function and context

The design of tokenization should be related to the selection of d

Transformer

We need to add contextual information to the tokens embedding:

$$x_i = \text{cat} \tag{9}$$

$$z_i = \text{cat} \mid \text{context} \tag{10}$$

$$z_i = f_\theta(x_1, \ldots, x_n) \tag{11}$$

$$z_i = \sum_j \alpha_{ij} \cdot v_j \tag{12}$$

$$f_\theta : (\mathbb{R}^d)^n \to (\mathbb{R}^d)^n \tag{13}$$

$$x_i \to z_i \tag{14}$$

The transformer:

1. computes relationships between tokens (attention)
2. mixes information across tokens
3. updates representations (x_i)

Self-attention

For each position i we want an updated representation that incorporates information from all other tokens:
Instead of:

$$x_i = \text{cat} \tag{15}$$

we want:

$$z_i = \text{cat in the context of the sentance} \tag{16}$$

For each pair (i, j):

$$\text{score}(i, j) = q_i^\top k_j \tag{17}$$

How related are tokens i and j?

Information parameters

- q_i: what token i is looking for
- k_j: what token j offers to token i
- v_j: information content of j
- α_{ij} how much j influences i

Self-attention

$$\alpha_{ij} = \frac{\exp(\text{score}(i,j))}{\sum_k \exp(\text{score}(i,k))} \tag{18}$$

For a fixed i, $\sum_j \alpha_{ij} = 1$
For each token i:

$$\alpha_{ij} = \text{weight assigned to token } j \tag{19}$$

$$z_i = \sum_j \alpha_{ij} v_j \tag{20}$$

Look at all tokens j, and take weighted averages with α_{ij} (high α_{ij} is high similarity, interaction)

Self-attention

- Learn which objects influence each other
- Instead of fixed graph or neighborhood
- In biology, gene networks are unknown
- Interactions are context dependent

Multi-head attention:

$$\text{Head}_h : (W_Q^h, W_K^h, W_V^h) \tag{21}$$

Different heads learn different interaction modes

- pathway-level
- regulatory motifs
- cell-state structure

Large language model architecture

- LLMs are designed to understand contextual information, predict subsequent words, and generate coherent text
- An encoder converts input text / tokens into a *vector representation*
- The LLM operates on this high dimensional *vector representation* object
- The decoder then converts the output back into the original data type

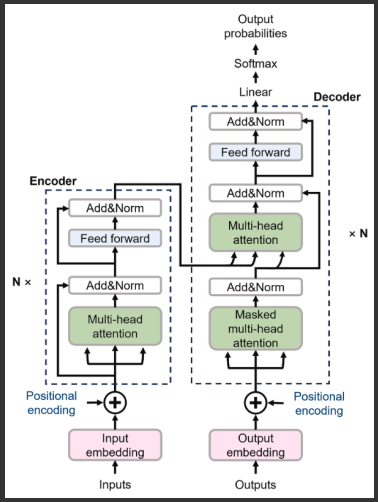

Figure 4: Architecture of transformer (Sheng et al. 2026)

Summary

LLM	Conventional Statistics
embedding	latent factor
attention	adaptive covariance
pretraining	unsupervised density estimation
fine-tuning	supervised regression

LLMs for omics

- LLMs in omics are self-supervised deep representation learning

What is new about this?

1. scaling between omics domains
2. self-supervision
3. flexible (but difficult) tokenization

Tokenization

The tokenization strategy defines the invariances, what correlations can be learned and what relationships or patterns get lost

1. k-mers (DNA)
   - tokens = local sequence patterns
   - model $\approx$ learns motif co-occurrence structure
2. genes as tokens (scRNA)
   - tokens = categorical variables (genes)
   - model $\approx$ learns co-expression geometry
3. ranked genes (Geneformer)
   - tokens = ordered features
   - model $\approx$ learns relative importance structure

Tokenization

- DNA and RNA sequences naturally have text-like properties, therefore one can use tokenization approaches like one-hot encoding of individual nucleotides, k-mer segmentation, overalapping strings (or some variant applied to BLAST type algorithm?)
- Protein models like ESM-2 treat amino acids as tokens
- In single-cell data, can treat a cell as a sentance and a gene as a word
- More examples:
  - scBERT: binning to discretize continuous expression values
  - scGPT: 3 types of tokens
    1. gene names representing itendity
    2. adaptively discretized expression
    3. condition tokens
  - Geneformer: ranking genes within each individual cell by expression level, therefore a sentance of genes
  - CellPLM: cells are tokens and tissues or spatial regions are sentances
- The idea is to preserve biologically meaningful structure and represent it in a way for the LLM architecture

Tokens

Table 1 Representative omics data for language models.

Modality	Data form	Data source	Tokenization strategies
Genomics (DNA)	DNA sequences (A/T/C/G)	Human genome (∼2.75 B nucleotide bases)	Nucleotide as token: k-mer
		Human genome (∼2.75 B nucleotide bases), multi-species genome references (∼32.5 B nucleotide bases)	Nucleotide as token: byte-pair encoding (BPE)
Transcriptomics (mRNA)	Transcript sequences (A/U/C/G)	ncRNA sequences (∼23 M)	Nucleotide as token: one-hot encoding
Protein sequences/ structures	Amino acid sequences, MSAs, 3D structures	UniRef (∼65 M protein sequences)	Residue as token
Single-cell transcriptomics	Cell-by-gene expression matrix	PanglaoDB (∼1.1 M human cells, 74 tissues, 209 datasets) CELLxGENE (∼33 M normal human cells)	Gene as token: binning for value
		Genecorpus-30 M (∼30 M human scRNA-seq cells from 561 datasets, multiple public sources)	Gene as token: Ranking for value
Spatial transcriptomics	Cell-by-gene expression matrix + 2D coordinates	∼11.4 M cells (∼8.7 M scRNA-seq, ∼2.7 M SRT) from public datasets (HCA, HTCA, GEO, CosMx SMI)	Cell as tokens

Figure 5: Omics tokenization strategies (Sheng et al. 2026)

LLMs in bioinformatics

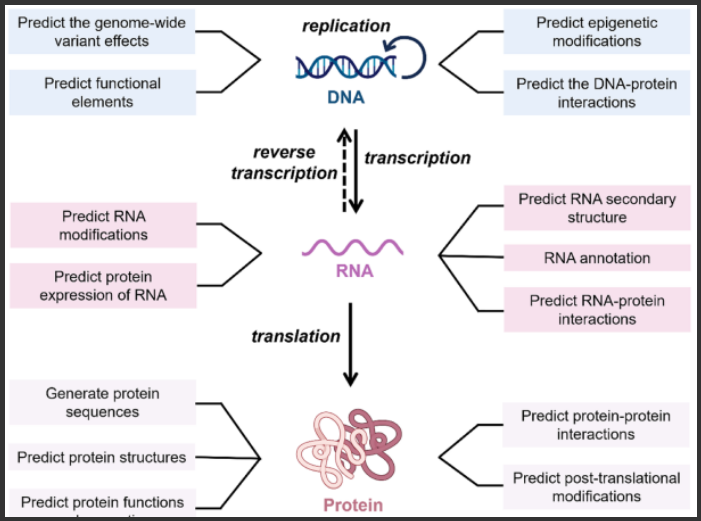

Figure 6: LLMs in omics (Sheng et al. 2026)

Why use LLMs in Omics

- Learn the joint representation of all variables
- The learned embedding may be an alternative to the estimate networks
- Help capture long-range dependencies and capture nonlinear relationships
- Alternative to graphical models and covariance estimation
- Most useful in integrating multi-omics

Limitations

- Noise is still a problem
- Tokenization is difficult
- Interpretability is weak
- Embeddings are not identifiable
- No quantification of uncertainty

Discussion

- Is the learned embedding geometry biologically meaningful?
- Can the transformer parigdim inform causality?
- How do these ideas complement *traditional* methods (graphical models, correlation and partial correlation estimation ...)?

Bibliography I

- Yanes, Óscar and Gary Siuzdak (2012). "Metabolomics: the apogee of the omics trilogy". In: URL: https://api.semanticscholar.org/CorpusID:17912313.
- Tong, Hua Zou and Bangzhuo (Dec. 2023). Data Analysis in Metabolomics. URL: https://xbiomeanalysis.github.io/Metabolomics_Aanlysis/.
- Sheng, Xia et al. (Jan. 2026). "Omics-based large language models: A new engine for drug discovery innovation". In: Acta Pharmaceutica Sinica B 16.1, pp. 122–136. ISSN: 2211-3835. DOI: 10.1016/j.apsb.2025.10.034. URL: http://dx.doi.org/10.1016/j.apsb.2025.10.034.