Weighted Diversified Sampling for Efficient Data-Driven Single-Cell Gene-Gene Interaction Discovery

Yifan Wu, Zirui Liu, Khushbu Pahwa, Xia Hu*
Department of Computer Science
Rice University
Houston, TX, USA
{yifan.wu, zirui.liu, khushbu.pahwa, xia.hu}@rice.edu

Yuntao Yang, Zhao Li, Rongbin Li, Wenjin Zheng*
UTHealth Houston
Houston, TX, USA
{yuntao.yang, zhao.li, rongbin.li, wenjin.zheng}@uth.tmc.edu

Zhaozhuo Xu*
Stevens Institute of Technology
Hoboken, NJ, USA
zhaozhuo.xu@stevens.edu

Abstract

Gene-gene interactions play a crucial role in the manifestation of complex human diseases. Uncovering significant gene-gene interactions is a challenging task. Here, we present an innovative approach utilizing data-driven computational tools, leveraging an advanced Transformer model, to unearth noteworthy gene-gene interactions. Despite the efficacy of Transformer models, their parameter intensity presents a bottleneck in data ingestion, hindering data efficiency. To mitigate this, we introduce a novel weighted diversified sampling algorithm. This algorithm computes the diversity score of each data sample in just two passes of the dataset, facilitating efficient subset generation for interaction discovery. Our extensive experimentation demonstrates that by sampling a mere 1% of the single-cell dataset, we achieve performance comparable to that of utilizing the entire dataset.

1 Introduction

Gene-gene interactions play a crucial role in the manifestation of complex human diseases, including multiple sclerosis [Brassat et al., 2006, Motsinger et al., 2007, Slim et al., 2022], pre-eclampsia [Li et al., 2022, Diab et al., 2021, Williams and Pipkin, 2011, Oudejans and Van Dijk, 2008], and Alzheimer's disease [Ghebranious et al., 2011, Hohman et al., 2016]. Computational tools equipped with machine learning (ML) prove effective in uncovering these significant gene interactions [McKinney et al., 2006, Cui et al., 2022, Yuan and Bar-Joseph, 2021b, Wei et al., 2024, Upstill-Goddard et al., 2013]. By learning an ML model on massive single-cell transcriptomic data, we can identify

*Corresponding authors: Xia Hu and Zhaozhuo Xu. Use footnote for providing further information about authors (webpage, alternative address)—*not* for acknowledging funding agencies. Funding acknowledgements go at the end of the paper.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

gene-gene interactions associated with complex but common human diseases. Existing models rely on prior knowledge such as transcription factors (TF) [Wang et al., 2019, Yuan and Bar-Joseph, 2021a, Chen et al., 2021a, Shu et al., 2021] or existing gene-gene interaction (GGI) networks [Ata et al., 2020, Yuan and Bar-Joseph, 2019a], to infer new relationships. Although GGI networks and TFs are crucial for mapping biological processes, they frequently suffer from high false-positive rates and biases, particularly in large-scale in vitro experiments [Mahdavi and Lin, 2007, Rasmussen and et al., 2021]. In response to these challenges, we propose that gene-gene interactions can be uncovered using purely data-driven methods.

The Rise of Transformers on Single-Cell Transcriptomic Data. Recent advances in natural language processing, particularly the development of Transformer models [Vaswani et al., 2017], have demonstrated significant potential in biological data analysis [Hao et al., 2023, Theodoris et al., 2023, Bian et al., 2024, Cui et al., 2024]. Transformer models are known for their ability to capture the dependencies between gene expressions. The information fused through the self-attention mechanism [Vaswani et al., 2017] is particularly suited for analyzing the intricate relationships in single-cell transcriptomic data. On the other hand, Transformer models also demonstrated superior performance when we scaled up their parameter size [Hao et al., 2023]. This scaling capacity raises the researcher's interest in training and deploying parameter-intensive Transformer models, denoted as single-cell foundation models [Cui et al., 2024]. We would like to take this advantage for better gene-gene interaction discovery by identifying feature interactions within Transformer models.

Data-Driven Gene-Gene Interaction via Attention. In this work, we would like to advance the gene-gene interaction discovery with the Transformer models that have demonstrated superior performance on single-cell transcriptomic data. We see the self-attention mechanism [Vaswani et al., 2017] as a pathway to facilitate the modeling of gene-gene interactions. In single-cell foundation models, the input to the model is a bag of m gene expressions for a single cell. Next, in each layer and each head of the Transformer, there will be an attention map with shape $m \times m$ generated for this cell. Each entry of this attention map represents the interaction between two genes in this layer and this head. Assuming that we have a perfect Transformer that takes a cell gene expressions and correctly predicts if it is infected by a disease, we view the attention map of this cell as a strong indicator of disease-oriented gene-gene interactions.

Efficiency Challenge in Data Ingestion. Despite the transformative capabilities of Transformer models, one significant challenge remains: the efficient ingestion and processing of massive volumes of single-cell transcriptomic data. We are utilizing Transformer models with parameter sizes that exceed the hardware capacity, particularly that of the graphics processing unit (GPU). As a result, given a pre-trained Transformer, we have to perform batch-size computation on a massive single-cell transcriptomic dataset for computing gene-gene interactions through attention maps. This batch-size computation significantly enlarges the total execution time for scientific discovery. Moreover, the hardware in the real-world deployment environment for gene-gene interaction detection may have even more limited resources. Therefore, the current computational framework cannot support gene-gene interaction discovery on real-world single-cell transcriptomic datasets.

Our Proposal: Two-Pass Weighted Diversified Sampling. In this paper, we introduce a novel weighted diversified sampling algorithm. This randomized algorithm computes the diversity score of each data sample in just two passes of the dataset. The proposed algorithm is highly memory-efficient and requires constant memory that is independent of the cell dataset size. Our theoretical analysis suggests that this diversity score estimates the density of the Min-Max kernel defined on the cell-level gene expressions, which provides the foundation and justification of the proposed strategy. Through extensive experiments, we demonstrate how the proposed sampling algorithm facilitates efficient subset generation for interaction discovery. The results show that by sampling a mere 1% of the single-cell dataset, we can achieve performance comparable to that of utilizing the entire dataset.

Our Contributions. We summarize our contributions as fellows.

- We introduce a computational framework that advances the discovery of significant gene-gene interactions with CelluFormer, our proposed Transformer model that is trained on single-cell transcriptomic data.
- We pinpoint the challenge in data ingestion for the data-driven gene-gene interaction. Moreover, we argue that we should perform diversified sampling that selects a representative subset of single-cell transcriptomics data to fulfill the objective.

- We develop a diversity score for every cell in the dataset based on the Min-Max kernel density. Moreover, we perform a randomized algorithm that efficiently estimates the Min-Max kernel density for each cell. Furthermore, we use the estimated density to generate an effective subset for gene-gene interaction.

2 Data-Driven Single-Cell Gene-Gene Interaction Discovery

In this section, we propose a computing framework to perform gene-gene interaction discovery on single-cell transcriptomic data. We start by introducing the format of single-cell transcriptomic data. Next, we propose the formulation of our CelluFormer model tailored to single-cell data. Next, we present our multi-cell-type training to build an effective transformer model on single-cell data. Finally, given a pre-trained transformer, we showcase how to perform gene-gene interaction discovery by analyzing the attention maps.

2.1 Single-Cell Transcriptomic Data

Single-cell transcriptomic is a technology that profiles gene expression at the individual cell level. The profiled results, namely single-cell transcriptomic data, provide a unique landscape of gene expressions. In contrast to traditional bulk RNA-seq analysis, single-cell transcriptomic data allows for cell-level sequencing, which captures the variability between individual cells [Ata et al., 2020]. Leveraging this high-resolution data allows scientists to gain insights into developmental processes, disease mechanisms, and cellular responses to environmental changes.

The single-cell transcriptomic data can be formulated as a set of high-dimensional and sparse feature vectors. We denote a single-cell transcriptomic dataset at X, where each cell $x \in X$ is a sparse vector with dimensionality $V \in \mathbb{N}_+$. Here V represents the total number of genes we can observe in X. Since cell $x \in \mathbb{R}^V$ is a sparse vector, we can represent x as a set $\{(i_1, v_1), (i_2, v_2), \cdots, (i_k, v_k)\}$. In this set, every tuple (i, v) represents the expression of gene $i \in [V]$ with expression level $v \in \mathbb{R}$. Besides we can also denote cell x as $[x_1, x_2, \cdots, x_V]$, where most of the x_is are zeros.

In this data formulation, single-cell transcriptomic data for each cell is represented as a set of gene expressions, with different cells expressing varying genes. Additionally, even when two cells express the same gene, their expression levels may differ. Our research objective is to identify gene-gene interactions within the vocabulary V that drive complex biological processes and disease mechanisms.

2.2 CelluFormer: A Single-Cell Transformer

Here, we propose our Transformer architecture, CelluFormer, to learn gene-gene interactions within single-cell transcriptomic data. Based on the set formulation of single-cell transcriptomic data, we believe that the order of genes is arbitrary and biologically meaningless. Similar to scGPT [Cui et al., 2024], and scFoundation [Hao et al., 2024], our method adopts a permutation-invariant design. We define our permutation-invariant condition as follows.

Condition 2.1. *Let X denote a single-cell transcriptomic dataset. Given a single-cell data of cell $x \in X$, denoted as a set $\{(i_1, v_1), (i_2, v_2), \cdots, (i_k, v_k)\}$, a function $f : X \to \mathbb{R}$ should satisfy that, for any permutation π, $f(x) = f(\pi(x))$.*

We see Condition 2.1 as a fundamental difference between the proposed Transformer and the sequence Transformers [Vaswani et al., 2017] widely used in natural language processing. For sequence Transformers, we have to ingest sequential masks during the training to ensure that the current token does not interact with the future token. Additionally, during the inference, the sequence Transformer should perform a step-by-step generation for each token. As a result, the sequence Transformer does not satisfy Condition 2.1. Moreover, the difference between CelluFormer and a vision Transformer [Dosovitskiy et al., 2020] is that the vision Transformer has a fixed sequence length for every input data sample. However, the number of genes expressed in each cell can vary

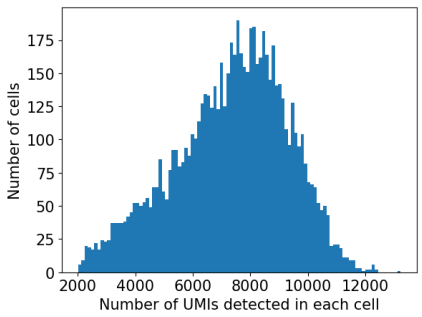

Figure 1: Distribution of Sequence Lengths in L6_CT Cell Type Data.

a lot. For example, according to Figure 1, the number of genes expressed in a single cell can be up to 12,000 or more. Thus, we utilize a padding mask for the classification downstream task. Additional details regarding the implementation of CelluFormer are provided in Appendix C.1.

2.3 Multi-Cell-Type Training of CelluFormer

We observe that there is a significant performance difference between Transformer models if we feed them with different styles of single-cell transcriptomic data. It is known that cells can be categorized into different types based on their functionality. For instance, neuronal cells represent the cell types that fire electric signals called action potentials across a neural network [Levitan and Kaczmarek, 2015]. Our study suggests that Transformers should be trained on single-cell transcriptomic data from various cell types to achieve better performance. We showcase an example in Table 1. We train a Transformer model to classify whether a cell is an Alzheimer's disease-infected cell or not. According to our study, CelluFormer proposed in Section 2.2 trained on neuronal cells outperforms traditional multilayer perceptron (MLP) with downstream training on a single cell type. However, we do not see this gap when we perform training of CelluFormer on a single cell type. As a result, we see that the Transformers generally prefer massive exposure to the single-cell transcriptomic data.

Table 1: Performance comparison of models on neuronal cell dataset.

Model	Training Dataset	F1 Score	Accuracy
MLP	Pax6	78.91	82.71
	L5_ET	62.02	73.31
	L6_CT	91.14	92.01
	L6_IT_Car3	95.34	95.51
	L6b	86.01	88.76
	Chandelier	81.66	84.56
	L5_6_NP	89.33	90.42
	All Neuronal Cell Types	97.23	97.25
CelluFormer	All Neuronal Cell Types	**98.12**	**98.12**

2.4 Gene-Gene Interaction Discovery via Attention Maps

In this paper, we would like to accomplish the following objective.

Objective 2.2 (Gene-gene interaction discovery)**.** *Let X denote a single-cell transcriptomic dataset. Let $\mathcal{V}$ denote the genes expressed in at least one $x \in X$. Let $f : X \to \mathbb{R}$ denote a permutation invariant (see Condition 2.1) CelluFormer. f can successfully predict whether any $x \in X$ is infected by disease D. We would like to find a gene-gene pair (v_1, v_2) that contributes the most to f's performance in X. Here $v_1, v_2 \in \mathcal{V}$.*

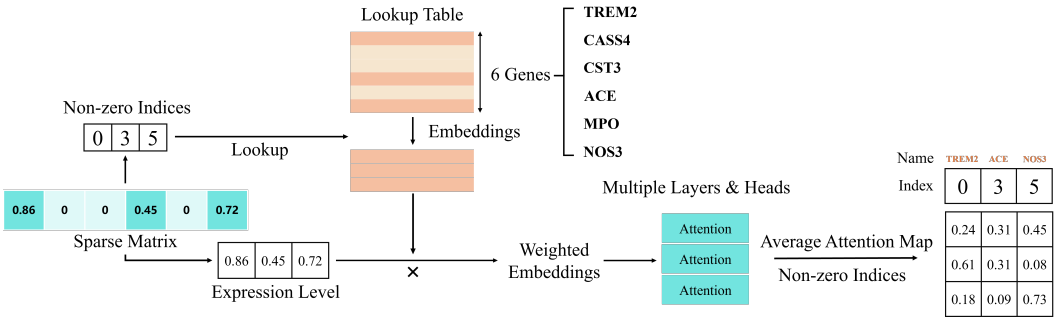

Figure 2: Gene-gene interaction modeling with attention maps.

We see the self-attention mechanism of Transformers on a cell's set style gene expressions as a pathway to model gene-gene interactions. CelluFormer takes a cell x's gene expressions and produces an attention map $A_{i,j} \in \mathbb{R}^{m \times m}$ at encoder block i and attention head j. Here m represents the number of genes expressed in cell x. Since Transformer architecture uses the Softmax function to produce $A_{i,j}$, we can view the pth row of $A_{i,j}$ as the interaction between gene p and all other genes in x. As a result, an attention map is a natural indicator of gene-gene interactions. Moreover, if we have a perfect Transformer that takes a cell x gene expressions and correctly predicts if it is infected by a disease, we view the attention map of this cell as an indicator of disease-oriented gene-gene interactions. Following this path, we propose a gene-gene interaction modeling approach

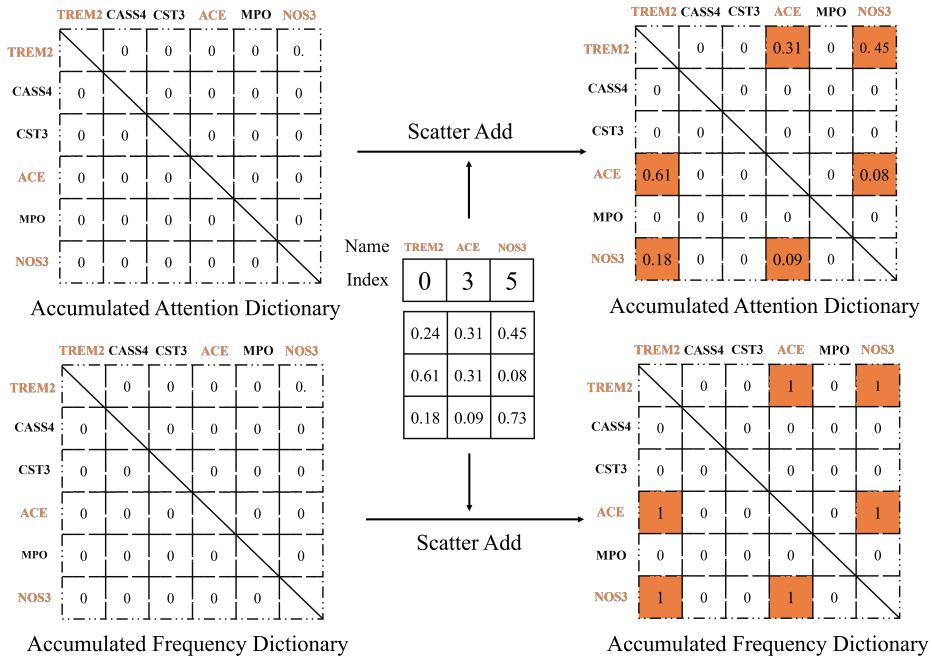

Figure 3: Accumulating multiple cells' average attention maps.

as illustrated in Figure 2. For each cell x, we represent it as a set and generate a bag of embeddings from the gene embedding table. Next, we use the expression levels of each gene as a scaling factor for each gene's embedding. Next, we take the average attention maps of all layers and all heads to obtain a gene-gene interaction map in this cell.

In Objective 2.2, we would like to see not only the gene-gene interactions just for cell x but also the statistical evidence of how two genes interact in the dataset X. As a result, we propose to accumulate multiple cells' averaged attention maps as illustrated in Figure 3. For X, we initialize $Z_0 \in 0^{V \times V}$ matrix as the overall attention map before aggregation and $M_0 \in 0^{V \times V}$ as the overall frequency dictionary before aggregation. Next, for each cell x in the dataset, we remove its diagonal value in its averaged attention map as it represents self-interaction. Next, we perform scatter addition operations that merge x's averaged attention map back to Z_0. We let Z_{ij} add the interaction value of gene v_i and v_j in the average attention map of cell x obtained in the Transformer model. Simultaneously, to eliminate the dataset bias of expressed genes, we count the number of appearances for each gene pair in the dataset. Once again, we perform scatter addition to record the counts back to M_0. This is done by updating M_0 through scatter addition, where $M_{ij} = M_{ij} + 1$ for every occurrence of the gene pair (v_i, v_j) in the dataset. Finally, we rank the off-diagonal values in Z where $Z_{ij} \leftarrow \frac{Z_{ij}}{M_{ij}}$ to retrieve the top gene-gene interaction.

3 Weighted Diversified Sampling

In this section, we start by showcasing the data-efficiency problem when we use the trained CelluFormer for gene-gene interaction discovery. Following this, we define a diversity score for each cell in the dataset and propose a two-pass randomized algorithm to efficiently compute it. Lastly, we propose a weighted diversified sampling strategy on massive single-cell data.

3.1 Data-Intensive Computation for Gene-Gene Interaction Discovery

As illustrated in Section 2.4, once we have a pre-trained CelluFormer that can successfully predict whether a cell is infected by a disease or not with its gene expressions, we can perform gene-gene interaction discovery by passing massive cells into this model and get the accumulated attention map as Figure 3. However, this process requires data-intensive computation. For every cell in the dataset,

we first need to compute the average attention map as illustrated in Figure 2. Next, we perform aggregations as shown in Figure 3. It is known that CelluFormer uses plenty of trainable parameters to achieve good disease infection classification performance. As a result, the computation complexity for generating a cell's averaged attention map is expensive. Moreover, since the attention map for cell x is $m \times m$, where m is the number of genes expressed in x. According to Figure 1, we see that m can be 12,000 or more. These giant attention maps consume the limited high bandwidth memory (HBM) in the graphics processing unit. Therefore, we have to perform batch-wise computation on a massive cell dataset for computing gene-gene interactions. Moreover, given the scale of the dataset, *any sampling algorithm with a runtime that grows exponentially with the dataset size is impractical.*

Algorithm 1 Two-Pass Algorithm for Estimating Min-Max Density

Input: Cell dataset X, 0-bit CWS function family $\mathcal{H}$ (see Definition 3.3), Hash range B, Rows R
Output: Min-Max density set w for every $x \in X$.
Initialize: $A \leftarrow 0^{R \times B}$
Generated R independent 0-bit CWS functions $h_1, \ldots, h_R$ from $\mathcal{H}$ with range B at Random.
{We set $R = O(\log |X|)$ following the theoretical analysis of Definition 3.3}
$W \leftarrow \emptyset$
for $x \in X$ **do**
  for $r = 1 \rightarrow R$ **do**
    $A_{r,h_r(x)} += 1$
  end for
end for
for $x \in X$ **do**
  for $r = 1 \rightarrow R$ **do**
    $w_x \leftarrow w_x + A_{r,h_r(x)}$
  end for
  $w_x \leftarrow w_x / R$ {w_x is the estimated Min-Max density for x.}
  $W \leftarrow \{w_x\}$
end for
return W

3.2 Two-Pass Randomized Algorithm for Computing Min-Max Density

In this work, we would like to address this data-efficiency challenge by raising and asking the following research question: *Can we find a representative and small subset from the large cell dataset and still perform successful gene-gene interaction discovery?* Moreover, we would like the procedure for finding this small subset as efficient as possible.

We would like to answer this question by proposing a diversity score of a cell in the dataset. To begin with, we introduce the Min-Max similarity between two cell's gene expressions.

Definition 3.1 (Min-Max Similarity)**.** Given two cell's gene expressions, denoted as $x, y \in \mathbb{R}^V$ (see Section 2.1), we define their Min-Max similarity as: $\text{Min-Max}(x, y) = \frac{\sum_i^V \min(x_i, y_i)}{\sum_i^V \max(x_i, y_i)}$.

According to the definition, $\text{Min-Max}(x, y) \in [0, 1]$. Higher Min-Max means that two cell's gene expressions are closer to each other. Min-Max is widely viewed as a kernel [Li, 2015b, Li et al., 2021, Li and Li, 2021] in statistical machine learning. In this paper, we would like to define a kernel density on top of the Min-Max similarity.

Definition 3.2 (Min-Max Density)**.** Given a cell dataset X, for every $q \in X$, we define its Min-Max density as: $\mathcal{K}(q) = \sum_{x \in X} \varphi(q, x)$, where $\varphi(q, x) : \mathbb{R} \rightarrow \mathbb{R}$ is a monotonic increasing function along with $\text{Min-Max}(q, x)$ similarity defined in Definition 3.1.

We view $\text{Min-Max}(q)$ density as an indicator of how diverse q is in X. Smaller $\text{Min-Max}(q)$ means that all other $x \in X$ may be less similar to q, making q a unique cell. On the other hand, higher $\text{Min-Max}(q)$ means that X has some cells that have similar gene expressions with q, making q less unique. However, to compute $\text{Min-Max}(q)$ for every $q \in X$ following Definition 3.2, we have to compute all pairwise $\text{Min-Max}(x, y)$ for any $x, y \in X$, which results in an unaffordable $O(n^2 \mathbb{NNZ}(X))$ time complexity, where n is the size of X and $\mathbb{NNZ}(X)$ is the maximum possible

number of genes expressed in a cell $x \in X$. To reduce this n^2 computation, we propose a randomized algorithm that takes advantage of 0-bit consistent weighted sampling (CWS) [Li, 2015a] hash functions.

Definition 3.3 (0-bit Consistent Weighted Sampling Hash Functions [Li, 2015a, Li et al., 2021])**.** Let $\mathcal{H}$ denote a randomized hash function family. If we pick a $h \in \mathcal{H}$ at random, for any two cell expressions $x, y \in \mathbb{R}^V$, we have $\Pr[h(x) = h(y)] = \mathsf{Min\text{-}Max}(x, y) + o(1)$. Here every $h \in \mathcal{H}$ is a hash function that maps any $x \in X$ to an integer in $[0, B)$. We denote B as the hash range.

Here the $o(1)$ is a minor additive term with complex form. For simplicity, we refer the readers to [Li et al., 2021], Theorem 4.4 for more details.

This work presents an efficient randomized algorithm that estimates Min-Max density $\mathcal{K}(q)$ (see Definition 3.2) for every $q \in X$. As showcased in Algorithm 1, we initialize an array A with all values as zeros. Next, we conduct a pass over X. In this pass, for every $x \in X$, we compute its hash values after R independent hash functions. Next, we increment $A_{r,h_r(x)}$ with 1. After this pass, we take another pass at the dataset, for every $x \in X$, we take an average over the $A_{r,h_r(x)}$ and build a density score w_x. We would like to highlight that Algorithm 1 requires only two linear scans of the dataset. The time complexity for this algorithm is $O(n\mathbb{NNZ}(X))$, which is linear to the dataset. Moreover, we show that Algorithm 1 produces an estimator to Min-Max density.

Theorem 3.4 (Min-Max Density Estimator, informal version of Theorem B.1)**.** *Given a cell dataset X, for every $q \in X$, we compute w_q following Algorithm 1. Next, we have $\mathbb{E}[w_q] = \sum_{x \in X}(\mathsf{Min\text{-}Max}(x, q) + o(1))$, where* Min-Max *is the* Min-Max *similarity defined in Definition 3.1. As a result, w_q is an estimator for* Min-Max *density $\mathcal{K}(q)$ defined in Definition 3.2 with $\varphi(q, x) = \mathsf{Min\text{-}Max}(x, q) + o(1)$.*

We provide the proof of Theorem 3.4 in the supplementary materials.

3.3 Weighted Diversified Sampling with Inverse Min-Max Density

We propose to use the inverse form of Min-Max density in Definition 3.2 as a score for diversity. We define it as normalized inverse Min-Max density as below.

Definition 3.5 (Inverse Min-Max Density (IMD))**.** Given a cell dataset X, for every $q \in X$, we define its normalized inverse Min-Max density as $\mathcal{I}(q) = \mathsf{Softmax}(1/\mathcal{K}(q))$, where $\mathcal{K}(q)$ is the Min-Max diversity for q in Definition 3.2, Softmax is the softmax function that takes over all cells in X.

We view the IMD $\mathcal{I}(q) \in [0, 1]$ as a monotonic increasing function for the diversity of q. Higher $\mathcal{I}(q)$ means that all other $x \in X$ may be less similar to q, making q a unique cell. Moreover, IMD can be directly used as a sample probability to generate a representative subset of X for Objective 2.2. Given X, we perform sampling without replacement to generate a subset $X_{\mathsf{sub}} \subset X$, where $x \in X$ has the sampling probability $\mathcal{I}(x)$. The advantages of sampling with IMD (see Definition 3.5) can be summarized as follows.

- The IMD $\mathcal{I}(q)$ can be an effective indicator for how diverse q is in dataset X.
- Computing IMD is an efficient one-shot preprocessing process with just two linear scans of X with time complexity $O(n\mathbb{NNZ}(X))$, where n and $\mathbb{NNZ}(X)$ is defined in Section 3.2.
- The memory complexity of computing IMD is $O(RB)$, which can be viewed as constant since it is independent of n and $\mathbb{NNZ}(X)$.

In the following section, we would like to evaluate the empirical performance of IMD in selecting a representative subset out of a massive single-cell transcriptomic dataset while still maintaining effective performance in data-driven gene-gene interaction discovery powered by CelluFormer.

Definition 3.6 (Estimated Interaction Score with WDS)**.** Let $Z_x(v_i, v_j)$ denote the interaction value of gene v_i and v_j in the average attention map of cell x obtained in the CelluFormer. For dataset X, we perform a sampling where each cell $x \in X$ is sampled with probability $\mathcal{I}(x)$ (see Definition 3.5) and get a subset X_s. Next, we define the estimated interaction score between gene v_i and v_j learned from X as:

$$\widetilde{Z}(v_i, v_j) = \frac{\sum_{x \in X_s} Z_x(v_i, v_j) \cdot \mathcal{I}(x)}{\sum_{x \in X_s} \mathcal{I}(x)},$$

where $\widetilde{Z}(v_i, v_j)$ is an unbiased estimator for the expectation of $Z(v_i, v_j)$ in distribution with density $\mathcal{I}(x)$. Formally,

$$\mathbb{E}[\widetilde{Z}(v_i, v_j)] = \mathbb{E}_{x\sim\mathcal{I}(x)}[Z_x(v_i, v_j)],$$

$$\mathbf{Var}[\widetilde{Z}(v_i, v_j)] = \frac{\sum_{x\in X_s}\mathcal{I}(x)^2}{(\sum_{x\in X_s}\mathcal{I}(x))^2}\mathbf{Var}_{x\sim\mathcal{I}(x)}[Z_x(v_i, v_j)].$$

4 Experiment

In this section, we want to validate the effectiveness of our gene-gene interaction pipeline as well as the two-pass diversified sampling algorithm 1. There are a few research questions we want to answer:

- **RQ1:** How does the proposed Transformer-based computing framework introduced in Section 2 perform in gene-gene interaction discovery?
- **RQ2:** How does the Min-Max density estimated by two-pass diversified sampling Algorithm 1 characterize the diversity of a cell in the whole dataset? Is this estimated Min-Max density useful?
- **RQ3:** How does the estimated Min-Max density perform in improving data-efficiency of gene-gene interaction discovery? How is the quality of the subset sampled according to the estimated Min-Max density?

4.1 Settings

Dataset: For the training dataset, we employ the Seattle Alzheimer's Disease Brain Cell Atlas (SEA-AD) [Gabitto et al., 2023], which includes single nucleus RNA sequencing data of 36,601 genes (as 36,601 features) from 84 senior brain donors exhibiting varying degrees of Alzheimer's Disease (AD) neuropathological changes. By providing extensive cellular and genetic data, SEA-AD enables in-depth exploration of the cellular heterogeneity and gene expression profiles associated with AD. To facilitate a comparative analysis between AD-affected and non-AD brains, we select cells from 42 donors classified within the high-AD category and 9 donors from the non-AD category, based on their neuropathological profiles. This selection criterion ensures a robust comparison, aiding in the identification of gene-gene interactions linked to AD progression [Gabitto et al., 2023]. The dataset is comprehensively annotated, covering 1,240,908 cells across 24 distinct cell types. We selected 18 neuronal cell types as our final training dataset since we believe neuronal cells are more likely to reveal explainable gene-gene interactions that are related to Alzheimer's Disease compared to non-neuronal cells. To better detect expression relationships among genes, we apply the Seurat Transformation Function [Stuart et al., 2019] to eliminate the problem of sequence depth difference.

Model: For the SEA-AD dataset, we designed a CelluFormer model as explained in 2.2 to predict labels indicative of Alzheimer's disease conditions. Further details can be found in the Appendix C.1.

Baselines: Our proposed algorithm leverages the attention maps of the Transformer models. Accordingly, we compare our method with three statistical methods, Pearson Correlation, CS-CORE, and Spearman's Correlation [Freedman et al., 2007, Su et al., 2023, De Smet and Marchal, 2010]. While these methods are widely adopted by biologists for gene co-expression analysis, gene co-expression values alone do not provide information about the relationship between gene pairs and Alzheimer's Disease. To identify gene-gene interactions relevant to Alzheimer's Disease, we apply these methods to subsets containing disease and non-disease cells respectively, and calculate their gene co-expression values. The difference in co-expression values between disease and non-disease cells is then used as a final score to rank the gene pairs. We also present more experiments in Appendix D.1 that demonstrate how Transformers aggregate data with varying labels.

Our baseline includes NID [Tsang et al., 2017], a traditional feature interpretation technique that extracts learned interactions from trained MLPs. NID identifies interacting features by detecting strongly weighted connections to a standard hidden unit in MLPs after training. We evaluated our CelluFormer model against the MLP model, with performance results presented in Table 1.

Additionally, to comprehensively evaluate RQ1, we utilized two existing single-cell large foundation models to assess our algorithm. Specifically, we fine-tuned two foundation models, scFoundation [Hao et al., 2024] and scGPT [Cui et al., 2023], to classify whether a cell is AD or non-AD (performance

results are provided in Table 4). We then applied our gene-gene interaction discovery pipeline using the attention maps of these foundation models.

In the sampling experiments, we compare WDS with uniform sampling since none of them requires preprocessing time exponential to the dataset size.

Evaluation Metric: For a comprehensive evaluation encompassing the entire ranked list of gene-gene interactions, we utilized the Kolmogorov-Smirnov test, which was facilitated by the `GSEApy` package [Fang et al., 2023] in Python. We select normalized enrichment score (NES) [Subramanian et al., 2005] as our evaluation metric. The ground truth dataset is sourced from *BioGRID* and *DisGenet* [Oughtred et al., 2019, Piñero et al., 2016]. For our experiments, we extract a subset of DisGenet that includes genes associated with Alzheimer's Disease. We then filter out genes in BioGRID that are not present in this DisGenet subset. Finally, we obtain a filtered BioGRID dataset containing only genes relevant to Alzheimer's Disease. We provide more explanations about our evaluation metrics in Appendix C.2.

4.2 The Effectiveness of Transformers in Gene-Gene Interaction Discovery (RQ1)

To evaluate the effectiveness of our Transformer-based framework for gene-gene interaction discovery, we performed feature selection across seven different cell types used as inference datasets. Additionally, we used a dataset encompassing all neuronal cell types to assess the overall performance of various models. As shown in Table 2, Transformer-based methods, including CelluFormer, scGPT and scFoundation, significantly outperformed other baselines.

This result indicates that our proposed Transformer-based framework is more effective and stable at extracting general and global gene-gene interaction information. In addition, the foundation models, scGPT and scFoundation, achieved comparable performances with other baselines across some of the datasets. We attribute this outcome to two main factors. **Overfitting to Pretrained Knowledge**: A foundation model, particularly a large one, might have learned very general or specific knowledge during its pretraining phase. When fine-tuning for a specific task, the model might overfit the preexisting knowledge, leading to poor generalization of the new task data. **Mismatch Between Pretraining and Fine-Tuning Data**: If the data distribution for fine-tuning is significantly different from the data on which the foundation model was trained, the model might struggle to adapt, resulting in worse performance. A model trained from scratch on the specific task data may perform better as it directly optimizes for that data distribution.

Table 2: Performance comparison of models on neuronal cell data. To evaluate different models on datasets with varying sizes, we further select 7 neuronal cell types from all neuronal cell types. CelluFormer, scGPT, scFoundation, MLP, Pearson Correlation, Spearman's Correlation, and CS-CORE were tested on 8 different datasets to obtain their gene pair rankings.

Dataset	CelluFormer	scFoundation	scGPT	NID	Pearson	CS-CORE	Spearman
L5_ET	1.15	1.04	**1.23**	0.90	0.50	1.11	0.91
L6_CT	1.18	1.03	1.17	**1.54**	-0.21	1.06	0.72
Pax6	**1.25**	0.82	1.01	1.04	0.93	0.95	1.15
L5_6_NP	1.21	1.06	**1.50**	1.49	0.87	0.92	0.95
L6b	1.13	0.99	**1.23**	0.62	0.75	0.62	1.08
Chandelier	**1.17**	1.16	1.09	1.07	0.94	1.06	0.96
L6_IT_Car3	**1.22**	0.90	0.66	1.19	0.59	1.08	0.86
All neuron data	**1.17**	1.02	0.99	0.86	1.01	1.06	1.04

4.3 Ablation Studies (RQ2 & RQ3)

We addressed these questions by comparing our weighted diversified sampling (WDS) method with uniform sampling across various sample sizes, ranging from 1% to 10% of the original dataset. We generated data subsets for each cell type using WDS and uniform sampling. We then applied our Transformer-based framework for feature selection at each sample size. Since CelluFormer consistently outperformed other baselines, we selected it as our base model. We repeated Each experiment five times and recorded the NES scores as the results. To evaluate the sampling methods, we calculated the average NES score across the five experiments. We also computed the Mean Square

Error (MSE) between the NES scores from the sampling experiments and the ground truth derived from the entire dataset, as shown in Table 2. The evaluation results are presented in Table 3. We note that WDS consistently produced higher NES scores compared to uniform sampling. As the sample size increased, the NES scores from uniform sampling began to converge with the ground truth. In contrast, the NES scores from WDS consistently remained close to the ground truth, even at smaller sample sizes. The result indicates that while WDS offers a significant advantage in small samples by enabling the Transformer to capture a broader range of genetic interactions, its benefits diminish as more data becomes available. Moreover, we find that for some cell types, smaller samples of data outperformed larger samples of data on NES. This suggests that: (1) single-cell transcriptomic data may contain noises that affect gene-gene interaction discovery, and, (2) some complex gene-gene interaction patterns in single-cell transcriptomic data cannot be interpreted directly through attention maps. We also provide a detailed study on the choice of parameter R in Algorithm 1 in Appendix D.2.

Table 3: Evaluation Results for the transformer over sample data. For each cell type, we performed 8 groups of down-sampling regarding 4 different sample sizes and 2 sampling methods. We let the transformer conduct inferences over the sample data and generate results.

Dataset	Sample Size	Mean of NES		MSE of NES	
		Uniform	WDS	Uniform	WDS
L5_ET	1%	0.90	**0.95**	0.0127	**0.0082**
	2%	0.89	**1.17**	0.0131	**0.0001**
	5%	1.02	**1.19**	0.0036	**0.0003**
	10%	0.87	**1.07**	0.0158	**0.0012**
L6_CT	1%	0.85	**1.19**	0.0207	**0.0000**
	2%	1.05	**1.18**	0.0030	**4.30e-05**
	5%	0.93	**1.23**	0.0122	**0.0006**
	10%	0.91	**1.21**	0.0136	**0.0002**
Pax6	1%	0.94	**1.08**	0.0184	**0.0053**
	2%	1.03	**1.18**	0.0098	**0.0009**
	5%	0.98	**1.20**	0.0139	**0.0004**
	10%	1.06	**1.17**	0.0072	**0.0012**
L5_6_NP	1%	0.90	**1.13**	0.0192	**0.0016**
	2%	**1.15**	1.11	**0.0009**	0.0021
	5%	1.02	**1.20**	0.0076	**4.54e-06**
	10%	1.01	**1.17**	0.0080	**0.0004**
L6b	1%	0.79	**1.17**	0.0226	**0.0004**
	2%	0.76	**1.14**	0.0266	**0.0000**
	5%	0.88	**1.20**	0.0121	**0.0009**
	10%	1.20	**1.21**	**0.0010**	0.0014
L6_IT_Car3	1%	0.78	**1.20**	0.0384	**0.0001**
	2%	0.87	**1.15**	0.0242	**0.0011**
	5%	0.97	**1.17**	0.0123	**0.0006**
	10%	0.97	**1.18**	0.0123	**0.0003**

5 Related Work

Single-Cell Transformer Models. Single-cell RNA sequencing (scRNA-seq), or single-cell transcriptomics, enables high-throughput insights into cellular systems, amassing extensive databases of transcriptional profiles across various cell types for the construction of foundational cellular models [Hao et al., 2023]. Recently, there has emerged a large number of transformer models pre-trained for single-cell RNA sequencing tasks, including scFoundation [Hao et al., 2023], Geneformer [Theodoris et al., 2023], scMulan [Bian et al., 2024], scGPT [Cui et al., 2024]. These foundation models have gained a progressive understanding of gene expressions and can build meaningful gene encodings over limited transcriptomic data. Yet, the previous work did not pay attention to pairwise gene-gene interactions. In our work, we would like to highlight a fundamental functionality of single-cell foundation models: we must use these models to perform data-driven scientific discovery.

Randomized Algorithms for Efficient Kernel Density Estimation. Kernel density estimation (KDE) is a fundamental task in both machine learning and statistics. It finds extensive use in real-

world applications such as outlier detection [Luo and Shrivastava, 2018, Coleman et al., 2020] and genetic abundance analysis [Coleman et al., 2022]. Recently, there has been a growing interest in applying hash-based estimators (HBE)[Charikar and Siminelakis, 2017, Backurs et al., 2019, Siminelakis et al., 2019, Coleman et al., 2020, Spring and Shrivastava, 2021] for KDE. HBEs leverage Locality Sensitive Hashing (LSH)[Indyk and Motwani, 1998, Datar et al., 2004, Li et al., 2019] functions, where the collision probability of two vectors under an LSH function is monotonic relative to their distance measure. This property allows HBE to perform efficient importance sampling using LSH functions and hash table-type data structures. Furthermore, [Liu et al., 2024] extend KDE algorithms as a sketch for the distribution. However, previous works have not considered LSH for weighted similarity. In this work, we focus on designing a new HBE that incorporates the Min-Max similarity [Li, 2015b], a weighted similarity measure.

6 Conclusion

Gene-gene interactions are pivotal in the development of complex human diseases, yet identifying these interactions remains a formidable challenge. In response, we have developed a pioneering approach that utilizes an advanced Transformer model to effectively reveal significant gene-gene interactions. Although Transformer models are highly effective, their extensive parameter requirements often impede efficient data processing. To overcome this limitation, we have introduced a weighted diversified sampling algorithm. This innovative algorithm efficiently calculates the diversity score of each data sample across just two passes of the dataset. With this method, we enable the rapid generation of optimized data subsets for interaction analysis. Our comprehensive experiments illustrate that by leveraging this method to sample a mere 1% of the single-cell dataset, we can achieve results that rival those obtained using the full dataset, significantly enhancing both efficiency and scalability.

References

Sezin Kircali Ata, Min Wu, Yuan Fang, Le Ou-Yang, Chee Keong Kwoh, and Xiao-Li Li. Recent advances in network-based methods for disease gene prediction. *Briefings in Bioinformatics*, 22 (4):bbaa303, 12 2020. ISSN 1477-4054. doi: 10.1093/bib/bbaa303. URL https://doi.org/10.1093/bib/bbaa303.

Arturs Backurs, Piotr Indyk, and Tal Wagner. Space and time efficient kernel density estimation in high dimensions. *Advances in Neural Information Processing Systems*, 32, 2019.

Haiyang Bian, Yixin Chen, Xiaomin Dong, Chen Li, Minsheng Hao, Sijie Chen, Jinyi Hu, Maosong Sun, Lei Wei, and Xuegong Zhang. scmulan: a multitask generative pre-trained language model for single-cell analysis. *bioRxiv*, 2024. doi: 10.1101/2024.01.25.577152. URL https://www.biorxiv.org/content/early/2024/01/29/2024.01.25.577152.

D Brassat, Alison A Motsinger, SJ Caillier, HA Erlich, K Walker, LL Steiner, BAC Cree, LF Barcellos, MA Pericak-Vance, S Schmidt, et al. Multifactor dimensionality reduction reveals gene–gene interactions associated with multiple sclerosis susceptibility in african americans. *Genes & Immunity*, 7(4):310–315, 2006.

Moses Charikar and Paris Siminelakis. Hashing-based-estimators for kernel density in high dimensions. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1032–1043. IEEE, 2017.

Jiaxing Chen, ChinWang Cheong, Liang Lan, Xin Zhou, Jiming Liu, Aiping Lyu, William K Cheung, and Lu Zhang. DeepDRIM: a deep neural network to reconstruct cell-type-specific gene regulatory network using single-cell RNA-seq data. *Briefings in Bioinformatics*, 22(6):bbab325, 08 2021a. ISSN 1477-4054. doi: 10.1093/bib/bbab325. URL https://doi.org/10.1093/bib/bbab325.

Jiaxing Chen, ChinWang Cheong, Liang Lan, Xin Zhou, Jiming Liu, Aiping Lyu, William K Cheung, and Lu Zhang. Deepdrim: a deep neural network to reconstruct cell-type-specific gene regulatory network using single-cell rna-seq data. *Briefings in Bioinformatics*, 22(6), 2021b. doi: 10.1093/bib/bbab325. URL https://doi.org/10.1093/bib/bbab325.

Benjamin Coleman, Anshumali Shrivastava, and Richard G Baraniuk. Race: Sub-linear memory sketches for approximate near-neighbor search on streaming data. *arXiv preprint arXiv:1902.06687*, 2019.

Benjamin Coleman, Richard Baraniuk, and Anshumali Shrivastava. Sub-linear memory sketches for near neighbor search on streaming data. In *International Conference on Machine Learning*, pages 2089–2099. PMLR, 2020.

Benjamin Coleman, Benito Geordie, Li Chou, RA Leo Elworth, Todd Treangen, and Anshumali Shrivastava. One-pass diversified sampling with application to terabyte-scale genomic sequence streams. In *International Conference on Machine Learning*, pages 4202–4218. PMLR, 2022.

Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, and Bo Wang. scgpt: Towards building a foundation model for single-cell multi-omics using generative ai. *bioRxiv*, 2023. doi: 10.1101/2023.04.30.538439. URL https://www.biorxiv.org/content/early/2023/07/02/2023.04.30.538439.

Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, and Bo Wang. scgpt: toward building a foundation model for single-cell multi-omics using generative ai. *Nature Methods*, pages 1–11, 2024.

Tianyu Cui, Khaoula El Mekkaoui, Jaakko Reinvall, Aki S Havulinna, Pekka Marttinen, and Samuel Kaski. Gene–gene interaction detection with deep learning. *Communications Biology*, 5(1):1238, 2022.

Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the 20th ACM Symposium on Computational Geometry (SoCG)*, pages 253–262, Brooklyn, NY, 2004.

Riet De Smet and Kathleen Marchal. Advantages and limitations of current network inference methods. *Nature Reviews Microbiology*, 8(10):717–729, October 2010. doi: 10.1038/nrmicro2419. URL https://www.nature.com/articles/nrmicro2419.

Nicholas S Diab, Syndi Barish, Weilai Dong, Shujuan Zhao, Garrett Allington, Xiaobing Yu, Kristopher T Kahle, Martina Brueckner, and Sheng Chih Jin. Molecular genetics and complex inheritance of congenital heart disease. *Genes*, 12(7):1020, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

Sinan Erten, Gurkan Bebek, and Mehmet Koyutürk. Vavien: an algorithm for prioritizing candidate disease genes based on topological similarity of proteins in interaction networks. *Journal of Computational Biology*, 18:1561–1574, 2011. doi: 10.1089/cmb.2011.0178. URL https://doi.org/10.1089/cmb.2011.0178.

Zhuoqing Fang, Xinyuan Liu, and Gary Peltz. Gseapy: a comprehensive package for performing gene set enrichment analysis in python. *Bioinformatics*, 39, 2023.

David Freedman, Robert Pisani, and Roger Purves. Statistics (international student edition). *Pisani, R. Purves, 4th edn. WW Norton & Company, New York*, 2007.

Mariano I. Gabitto, Kyle J. Travaglini, Victoria M. Rachleff, Eitan S. Kaplan, Brian Long, Jeanelle Ariza, Yi Ding, et al. Integrated multimodal cell atlas of alzheimer's disease. *Research Square*, 2023.

Nader Ghebranious, Bickol Mukesh, Philip F Giampietro, Ingrid Glurich, Susan F Mickel, Stephen C Waring, and Catherine A McCarty. A pilot study of gene/gene and gene/environment interactions in alzheimer disease. *Clinical Medicine & Research*, 9(1):17–25, 2011.

Minsheng Hao, Jing Gong, Xin Zeng, Chiming Liu, Yucheng Guo, Xingyi Cheng, Taifeng Wang, Jianzhu Ma, Le Song, and Xuegong Zhang. Large scale foundation model on single-cell transcriptomics. *bioRxiv*, 2023. doi: 10.1101/2023.05.29.542705. URL https://www.biorxiv.org/content/early/2023/06/21/2023.05.29.542705.

Minsheng Hao, Jing Gong, Xin Zeng, Chiming Liu, Yucheng Guo, Xingyi Cheng, Taifeng Wang, Jianzhu Ma, Xuegong Zhang, and Le Song. Large-scale foundation model on single-cell transcriptomics. *Nature Methods*, 21(8):1481–1491, 2024. ISSN 1548-7105. doi: 10.1038/s41592-024-02305-7. URL https://doi.org/10.1038/s41592-024-02305-7.

Timothy J Hohman, William S Bush, Lan Jiang, Kristin D Brown-Gentry, Eric S Torstenson, Scott M Dudek, Shubhabrata Mukherjee, Adam Naj, Brian W Kunkle, Marylyn D Ritchie, et al. Discovery of gene-gene interactions across multiple independent data sets of late onset alzheimer disease from the alzheimer disease genetics consortium. *Neurobiology of aging*, 38:141–150, 2016.

Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing (STOC)*, pages 604–613, Dallas, TX, 1998.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning*, pages 3744–3753. PMLR, 2019.

Irwin B Levitan and Leonard K Kaczmarek. *The neuron: cell and molecular biology*. Oxford University Press, USA, 2015.

Ping Li. 0-bit consistent weighted sampling. In *Proceedings of the 21th ACM SIGKDD International conference on knowledge discovery and data mining*, pages 665–674, 2015a.

Ping Li. Min-max kernels. *arXiv preprint arXiv:1503.01737*, 2015b.

Ping Li, Xiaoyun Li, and Cun-Hui Zhang. Re-randomized densification for one permutation hashing and bin-wise consistent weighted sampling. *Advances in Neural Information Processing Systems*, 32, 2019.

Ping Li, Xiaoyun Li, Gennady Samorodnitsky, and Weijie Zhao. Consistent sampling through extremal process. In *Proceedings of the Web Conference 2021*, pages 1317–1327, 2021.

Xiaomei Li, Lin Liu, Clare Whitehead, Jiuyong Li, Benjamin Thierry, Thuc D Le, and Marnie Winter. Identifying preeclampsia-associated genes using a control theory method. *Briefings in Functional Genomics*, 21(4):296–309, 2022.

Xiaoyun Li and Ping Li. Rejection sampling for weighted jaccard similarity revisited. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 4197–4205, 2021.

Zichang Liu, Zhaozhuo Xu, Benjamin Coleman, and Anshumali Shrivastava. One-pass distribution sketch for measuring data heterogeneity in federated learning. *Advances in Neural Information Processing Systems*, 36, 2024.

Chen Luo and Anshumali Shrivastava. Arrays of (locality-sensitive) count estimators (ace) anomaly detection on the edge. In *Proceedings of the 2018 World Wide Web Conference*, pages 1439–1448, 2018.

Mahmoud A. Mahdavi and Yen-Han Lin. False positive reduction in protein-protein interaction predictions using gene ontology annotations. *BMC Bioinformatics*, 8(1):262, 2007. ISSN 1471-2105. doi: 10.1186/1471-2105-8-262. URL `https://doi.org/10.1186/1471-2105-8-262`.

Brett A McKinney, David M Reif, Marylyn D Ritchie, and Jason H Moore. Machine learning for detecting gene-gene interactions: a review. *Applied bioinformatics*, 5:77–88, 2006.

Alison A Motsinger, David Brassat, Stacy J Caillier, Henry A Erlich, Karen Walker, Lori L Steiner, Lisa F Barcellos, Margaret A Pericak-Vance, Silke Schmidt, Simon Gregory, et al. Complex gene–gene interactions in multiple sclerosis: a multifactorial approach reveals associations with inflammatory genes. *Neurogenetics*, 8:11–20, 2007.

CBM Oudejans and M Van Dijk. Placental gene expression and pre-eclampsia. *Placenta*, 29:78–82, 2008.

Rose Oughtred, Chris Stark, Bobby-Joe Breitkreutz, Jennifer Rust, Lorrie Boucher, Christie Chang, Nadine Kolas, et al. The BioGRID interaction database: 2019 update. *Nucleic Acids Research*, 47, 2019.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Janet Piñero, Àlex Bravo, Núria Queralt-Rosinach, Alba Gutiérrez-Sacristán, Jordi Deu-Pons, Emilio Centeno, Javier García-García, Ferran Sanz, and Laura I. Furlong. DisGeNET: a comprehensive platform integrating information on human disease-associated genes and variants. *Nucleic Acids Research*, 45(D1):D833–D839, 10 2016. ISSN 0305-1048. doi: 10.1093/nar/gkw943. URL `https://doi.org/10.1093/nar/gkw943`.

Ananthakrishnan Rao, Sagar VG, Tony Joseph, Anand Bhattacharya, and R. Srinivasan. Phenotype-driven gene prioritization for rare diseases using graph convolution on heterogeneous networks. *BMC Medical Genomics*, 11:57, 2018. doi: 10.1186/s12920-018-0372-8. URL `https://doi.org/10.1186/s12920-018-0372-8`.

Sebastian Rasmussen and et al. Predicting protein interactions in plants: High confidence comes at a cost. *Journal of Experimental Botany*, 73(12):3866–3876, 2021. doi: 10.1093/jxb/erab332. URL `https://academic.oup.com/jxb/article/73/12/3866/6565416`.

Hantao Shu, Jingtian Zhou, Qiuyu Lian, Han Li, Dan Zhao, Jianyang Zeng, and Jianzhu Ma. Modeling gene regulatory networks using neural network architectures. *Nature Computational Science*, 1(7):491–501, 2021. ISSN 2662-8457. doi: 10.1038/s43588-021-00099-8. URL `https://doi.org/10.1038/s43588-021-00099-8`.

Paris Siminelakis, Kexin Rong, Peter Bailis, Moses Charikar, and Philip Levis. Rehashing kernel evaluation in high dimensions. In *International Conference on Machine Learning*, pages 5789–5798. PMLR, 2019.

Vikash Singh and Pietro Lio'. Towards probabilistic generative models harnessing graph neural networks for disease-gene prediction, 2019. URL https://arxiv.org/abs/1907.05628.

Lotfi Slim, Clément Chatelain, Hélène de Foucauld, and Chloé-Agathe Azencott. A systematic analysis of gene–gene interaction in multiple sclerosis. *BMC Medical Genomics*, 15(1):100, 2022.

Ryan Spring and Anshumali Shrivastava. Mutual information estimation using lsh sampling. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 2807–2815, 2021.

Tim Stuart, Andrew Butler, Paul Hoffman, Christoph Hafemeister, Efthymia Papalexi, William M Mauck III, Yuhan Hao, Marlon Stoeckius, Peter Smibert, and Rahul Satija. Comprehensive integration of single-cell data. *Cell*, 177:1888–1902, 2019. doi: 10.1016/j.cell.2019.05.031. URL https://doi.org/10.1016/j.cell.2019.05.031.

Chang Su, Zichun Xu, Xinning Shan, Biao Cai, Hongyu Zhao, and Jingfei Zhang. Cell-type-specific co-expression inference from single cell rna-sequencing data. *Nature Communications*, 14(1):4846, Aug 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-40503-7. URL https://doi.org/10.1038/s41467-023-40503-7.

Aravind Subramanian, Pablo Tamayo, Vamsi K. Mootha, Sayan Mukherjee, Benjamin L. Ebert, Michael A. Gillette, Amanda Paulovich, Scott L. Pomeroy, Todd R. Golub, Eric S. Lander, and Jill P. Mesirov. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–15550, Oct 2005. ISSN 0027-8424. doi: 10.1073/pnas.0506580102. URL https://www.pnas.org/doi/10.1073/pnas.0506580102. Epub 2005 Sep 30.

Christina V. Theodoris, Ling Xiao, Anant Chopra, Mark D. Chaffin, Zeina R. Al Sayed, Matthew C. Hill, Helene Mantineo, Elizabeth M. Brydon, Zexian Zeng, X. Shirley Liu, and Patrick T. Ellinor. Transfer learning enables predictions in network biology. *Nature*, 618(7965):616–624, 06 2023. doi: 10.1038/s41586-023-06139-9. URL https://doi.org/10.1038/s41586-023-06139-9.

Michael Tsang, Dehua Cheng, and Yan Liu. Detecting statistical interactions from neural network weights. *arXiv preprint arXiv:1705.04977*, 2017.

Emil Uffelmann, Qin Qin Huang, Nchangwi Syntia Munung, Jantina de Vries, Yukinori Okada, Alicia R. Martin, Hilary C. Martin, Tuuli Lappalainen, and Danielle Posthuma. Genome-wide association studies. *Nature Reviews Methods Primers*, 1(1):59, 2021. doi: 10.1038/s43586-021-00056-9. URL https://doi.org/10.1038/s43586-021-00056-9.

Rosanna Upstill-Goddard, Diana Eccles, Joerg Fliege, and Andrew Collins. Machine learning approaches for the discovery of gene–gene interactions in disease data. *Briefings in bioinformatics*, 14(2):251–260, 2013.

Orit Vanunu, Oranit Magger, Eytan Ruppin, Tomer Shlomi, and Roded Sharan. Associating genes and protein complexes with disease via network propagation. *PLoS Computational Biology*, 6:1–9, 2010. doi: 10.1371/journal.pcbi.1000641. URL https://doi.org/10.1371/journal.pcbi.1000641.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Zuo-Teng Wang, Chen-Chen Tan, Lan Tan, and Jin-Tai Yu. Systems biology and gene networks in alzheimer's disease. *Neuroscience & Biobehavioral Reviews*, 96:31–44, 2019.

Qingyue Wei, Md Tauhidul Islam, Yuyin Zhou, and Lei Xing. Self-supervised deep learning of gene–gene interactions for improved gene expression recovery. *Briefings in Bioinformatics*, 25(2): bbae031, 2024.

Paula J Williams and Fiona Broughton Pipkin. The genetics of pre-eclampsia and other hypertensive disorders of pregnancy. *Best practice & research Clinical obstetrics & gynaecology*, 25(4):405–417, 2011.

Ye Yuan and Ziv Bar-Joseph. Deep learning for inferring gene relationships from single-cell expression data. *Proceedings of the National Academy of Sciences*, 116(52):27151–27158, 2019a. doi: 10.1073/pnas.1911536116. URL https://www.pnas.org/doi/abs/10.1073/pnas.1911536116.

Ye Yuan and Ziv Bar-Joseph. Deep learning for inferring gene relationships from single-cell expression data. *Proceedings of the National Academy of Sciences*, 116(52):27151–27158, 2019b. doi: 10.1073/pnas.1911536116. URL https://www.pnas.org/doi/abs/10.1073/pnas.1911536116.

Ye Yuan and Ziv Bar-Joseph. Deep learning of gene relationships from single cell time-course expression data. *Briefings in Bioinformatics*, 22(5):bbab142, 04 2021a. ISSN 1477-4054. doi: 10.1093/bib/bbab142. URL https://doi.org/10.1093/bib/bbab142.

Ye Yuan and Ziv Bar-Joseph. Deep learning of gene relationships from single cell time-course expression data. *Briefings in bioinformatics*, 22(5):bbab142, 2021b.

Appendix

A More Related Work on Gene-Gene Interaction Discovery

In this section, we provide a more detailed review of the existing work on gene-gene interaction discovery. For gene-gene interaction network construction, genome-wide association studies (GWAS) are widely adopted by biologists to study gene associations using single-nucleotide polymorphisms (SNPs) [Uffelmann et al., 2021]. However, GWAS have high computational costs and are simply based on direct genotype-phenotype associations instead of wired graph structure. To address this problem, many graphic models have emerged in recent years [Ata et al., 2020]. Network-based methods regard gene-gene interaction discovery as a task to construct a homogeneous graph among genes. For example, PRINCE [Vanunu et al., 2010] and VAVIEN [Erten et al., 2011] apply random work to predict new edges on existing protein-protein interaction (PPI) or gene-gene interaction (GGI) knowledge graphs. VGAE [Singh and Lio', 2019] and GCAS [Rao et al., 2018] explore the potential to incorporate GNN and auto-encoder structure in the GGI network. In addition, existing work like DeepDRIM [Chen et al., 2021b] and CNNC [Yuan and Bar-Joseph, 2019b] successfully improve the construction of GGI through inferencing gene associations on scRNA sequencing data and known transcription factors (TF). While GGI networks and TFs are instrumental for mapping biological processes, they are often plagued by high false-positive rates and context-dependent inaccuracies, especially when derived from large-scale in vitro experiments [Mahdavi and Lin, 2007, Rasmussen and et al., 2021]. Existing methods that rely on pre-established protein-protein interaction (PPI) or transcription factor (TF) networks are prone to bias because they tend to reinforce known interactions, making it difficult to objectively uncover novel gene-gene interactions. In contrast, our method circumvents this issue by directly discovering GGIs from scRNA-seq data without dependence on prior network knowledge.

B Proofs of Theorem 3.4

Theorem B.1 (Min-Max Density Estimator, formal version of Theorem 3.4)**.** *Given a cell dataset X, for every $q \in X$, we compute w_q following Algorithm 1. Next, we have*

$$\mathbb{E}[w_q] = \sum_{x \in X} (\textsf{Min-Max}(x, q) + o(1)),$$

where Min-Max *is the* Min-Max *similarity defined in Definition 3.1. As a result, w_q is an estimator for* Min-Max *density $\mathcal{K}(q)$ defined in Definition 3.2 with $\varphi(q, x) = \textsf{Min-Max}(x, q) + o(1)$.*

Proof. According to Theorem 2 in [Coleman et al., 2019], the expectation of w_q should be:

$$\mathbb{E}[w_q] = \sum_{x \in X} \Pr_{h \sim \mathcal{H}}[h(q) = h(x)]$$

According to Definition 3.3, we have

$$\Pr_{h \sim \mathcal{H}}[h(q) = h(x)] = \textsf{Min-Max}(x, q) + o(1)$$

.

As a result,

$$\mathbb{E}[w_q] = \sum_{x \in X} (\textsf{Min-Max}(x, q) + o(1))$$

Moreover, since $\textsf{Min-Max}(x, q) + o(1)$ is a monotonic increasing function of $\textsf{Min-Max}(x, q)$. We say that w_q is an estimator for Min-Max density $\mathcal{K}(q)$ defined in Definition 3.2 with $\varphi(q, x) = \textsf{Min-Max}(x, q) + o(1)$. □

C Experiment Details

C.1 Model Implementations

Transformer Configurations: In this work, we used the standard multi-head self-attention introduced in [Vaswani et al., 2017]. We do not see the potential of the proposed blocks in [Lee et al., 2019] in our setting. Moreover, we perform padding on each batch of training and inference of single-cell data. Accordingly, we introduce a padding mask in the attention mechanism to avoid computation on the padded position. For each input sequence, we represent them as embedding by a lookup table that maps a vocabulary of 36,601 genes to 128-dimensional vectors. Subsequently, the embedded data passes through 4 transformer encoder blocks. Each encoder block features 8 attention heads, to capture complex, non-linear relationships within the data. Finally, the output is fed into a linear layer that classifies the data labels. Here the label for the cell can be disease-oriented, such as whether this cell is from an Alzheimer's disease patient. We represent each input sequence by employing a lookup table that transforms a comprehensive vocabulary of 36,601 genes into 128-dimensional embedding vectors. These vectors are subsequently processed through a series of 4 Transformer encoder blocks. Each encoder block is equipped with 8 attention heads, a 512-dimensional feedforward layer, and a dropout layer in a ratio of 0.1. The processed outputs are then directed to a linear classification layer, which is tasked with predicting labels indicative of Alzheimer's disease conditions. We adopted the Adam Optimization Algorithm to minimize the loss function Kingma and Ba [2017]. The model is trained under a learning rate of 1e-5 and the batch size of our data-loader is set as 128. The testing results for the transformer after 3 epochs of training are given in Table 1.

MLP Configurations: The MLP consists of 2 hidden layers, with 128 and 256 hidden units respectively. Each hidden layer is followed by a dropout and a Softplus module. The MLP is trained under a learning rate of 1e-4 and the batch size of our data-loader is set as 128. We adopted the Adam Optimization Algorithm to minimize the loss function Kingma and Ba [2017]. The testing results for the MLP after 80 epochs of training are given in Table 1.

Table 4: Complete Performance comparison of models on neuronal cell data.

Model	Training Dataset	F1 Score	Accuracy
MLP	Pax6	78.91	82.71
	L5_ET	62.02	73.31
	L6_CT	91.14	92.01
	L6_IT_Car3	95.34	95.51
	L6b	86.01	88.76
	Chandelier	81.66	84.56
	L5_6_NP	89.33	90.42
	All Neuronal Cell Types	97.23	97.25
CelluFormer	All Neuronal Cell Types	**98.12**	**98.12**
scGPT	All Neuronal Cell Types	93.85	94.32
scFoundation	All Neuronal Cell Types	97.38	97.39

Fine-tuning configurations for scFoundatoin and scGPT: For fine-tuning scGPT, we use an LR of 1e-4 and a batch size of 64. We utilize a step scheduler down to 90% of the original learning rate every 10 steps. The training process converges after 6 epochs. For scFoundation, we use an LR of 1e-4 and a batch size of 32. We fine-tune scFoundation for 10 epochs. The performances of scFoundation and scGPT on classifying disease cells are shown in Table 4.

Implementation and Computation Resources: Our codebase and workflow are implemented in PyTorch Paszke et al. [2017]. We trained and tested our workflow on a server with 8 Nvidia Tesla V100 GPU and a 44-core/88-thread processor (Intel(R) Xeon(R) CPU E5-2699A v4 @ 2.40GHz).

C.2 Evaluation Metrics

The normalized enrichment score (NES) is the main metric used to analyze gene set enrichment outcomes Subramanian et al. [2005]. This score quantifies the extent of over-representation of a ground truth dataset at the top of the ranked list of gene-gene interactions. That is, the higher the

better. We can calculate NES by starting at the top of the ranked list and moving through it, adjusting a running tally by increasing the score for each gene-gene interaction in the ground truth dataset and decreasing it for others based on each gene-gene interaction's rank. This process continues until we evaluate the entire ranked list to identify the peak score, which is the enrichment score. The BioGRID Dataset provides human protein/genetic interactions. Specifically, *BioGRID* contributes $204,831$ protein/genetic interactions that help verify the enrichment of genuine biological interactions in a ranked list of gene-gene interactions. DisGenet contains 429,036 gene-disease associations (GDAs), connecting 17,381 genes to 15,093 diseases, disorders, and abnormal human phenotypes Oughtred et al. [2019], Piñero et al. [2016].

D More Experments

D.1 Contrastive Ranking

Here, we also explore alternative strategies for aggregating attention maps. While Pearson Correlation, Spearman's Correlation, and CS-CORE themselves cannot capture the information between gene pairs the the target disease, we believe Transformers learn the difference among data with varying labels. Hence, we do not need to calculate the difference between attention maps aggregated on data with varying labels. However, given that the Transformer is trained to classify disease cells, we hypothesize that it likely assigns significant attention to specific gene pairs within disease cells. To evaluate this, we applied our pipeline to three distinct datasets. The experimental results summarized in Table 5 show that our pipeline achieves improved NES when both disease and non-disease cells are used as inputs. These findings suggest that the Transformer benefits from data both positive and negative labels to provide a more comprehensive understanding of features.

Table 5: This experiment involves three groups. In the first group, the Transformer only takes the disease cells for inference. We directly evaluate the ranked list given by aggregated attention map across disease cells. In the second group, we calculate the aggregated attention maps on the disease cells and the non-disease cells respectively. The final attention map is obtained by subtracting these two attention maps. The third group is to aggregate attention maps across the whole dataset.

Strategy	L5_ET	L6_CT	Pax6	L5_6_NP	L6b	Chandelier	L6_IT_Car3
AD cells	1.09	1.09	0.98	0.78	1.13	0.90	0.89
AD cells - Non-AD cells	1.08	0.89	1.05	0.76	0.82	0.65	**1.39**
All cells	**1.15**	**1.18**	**1.25**	**1.21**	**1.13**	**1.17**	1.22

D.2 Empirical Study on Parameter R in Algorithm 1

Table 6: The Mean value of NES results across 5 experiments on L5_ET, L6_CT, and Pax6 cell type datasets.

Dataset	Sample Size	Mean of NES			
		Uniform	WDS with R=100	WDS with R=200	WDS with R=500
L5_ET	1%	0.90	**1.02**	0.95	0.93
	2%	0.89	**1.17**	**1.17**	0.97
	5%	1.02	0.97	**1.19**	1.11
	10%	0.87	1.01	**1.07**	**1.07**
L6_CT	1%	0.85	**1.19**	**1.19**	1.11
	2%	1.05	**1.21**	1.18	1.09
	5%	0.93	1.13	**1.23**	1.21
	10%	0.91	**1.23**	1.21	1.20
Pax6	1%	0.94	**1.13**	1.08	1.17
	2%	1.03	**1.22**	1.18	1.19
	5%	0.98	**1.21**	1.20	1.19
	10%	1.06	1.19	1.17	**1.22**

During our experiments on WDS, we observed that the value of R (see Algorithm 1) has a noticeable impact on NES performance. In Table 6 and Table 7, we evaluate three different R values ranging

from 100 to 500. The results demonstrate that increasing R leads to a significant decline in NES. Although WDS with smaller R values yields relatively higher NES, it tends to diverge from the NES calculated on the entire dataset.

Table 7: The MSE of NES results across 5 experiments on L5_ET, L6_CT, and Pax6 cell type datasets. The MSE values are calculated according to the results in Table 2.

Dataset	Sample Size	MSE of NES			
		Uniform	WDS with R=100	WDS with R=200	WDS with R=500
L5_ET	1%	0.0636	0.0408	**0.0178**	0.0477
	2%	0.0653	0.0005	**0.0004**	0.0339
	5%	0.0181	**0.0014**	0.0310	0.0018
	10%	0.0790	**0.0062**	0.0192	0.0064
L6_CT	1%	0.1033	0.0002	0.0002	0.0046
	2%	0.0151	**0.0001**	0.0014	0.0070
	5%	0.0610	0.0028	**0.0025**	0.0013
	10%	0.0681	0.0011	0.0031	**0.0007**
Pax6	1%	0.0920	0.0264	0.0135	**0.0057**
	2%	0.0488	0.0047	**0.0006**	0.0027
	5%	0.0695	0.0022	**0.0015**	0.0027
	10%	0.0362	0.0058	0.0028	**0.0008**