Retrieval Augmented Generation of Large Language Models for scRNAseq Analysis

1st Cory Henn
University of New Mexico
Department of Biology
Albuquerque, NM, USA
coryhenn@unm.edu

2nd Alex Hartel
University of New Mexico
Department of Computer Science
Albuquerque, NM, USA
ahartel@unm.edu

3rd David Dominguez
University of New Mexico
Department of Computer Science
Albuquerque, NM, USA
dmdominguez17@unm.edu

***Abstract*—Single cell RNA sequencing generates high-dimensional data that require accurate and scalable gene-level interpretation of cell clusters to gain biological insight. Here, we propose a novel framework that utilizes a large language model augmented with unstructured retrieval capabilities to dynamically interact with heterogeneous datasets. Our approach integrates unsupervised machine learning of gene expression data with curated gene ontology and unstructured data, enabling the large language model to retrieve relevant biological context from cell clusters. Experiments on single cell RNA sequencing datasets demonstrate that our method achieves 81% accuracy in cell-type annotation, matching or surpassing traditional methods, while providing context and interpretable cluster-level summaries based on gene expression. This work bridges bioinformatics and natural language processing, offering a scalable and fluid solution for cell cluster characterization to accelerate discoveries in single-cell transcriptomics. Our framework represents a significant step toward the integration of language models in biology, opening new avenues for automated and informative analysis of complex biological datasets.**

***Index Terms*—Large Language Model (LLM), machine learning, bioinformatics, single-cell RNA sequencing**

I. Introduction

Single cell RNA sequencing (scRNA-seq) is widely used in biology to explore cellular complexity at high resolution. This technology enables the sequencing of individual cells to measure gene expression, generating data that is used to classify and understand cellular identities. Traditional machine learning approaches are applied to scRNA-seq data to group similar cells into clusters based on their gene expression profiles. These clusters are subsequently annotated with cell type labels using reference libraries or atlases.

However, annotation remains a challenging task. Manual annotation is labor-intensive and time-consuming, while automated methods often struggle to adapt to the diversity of datasets encountered in real-world studies. Beyond annotation, researchers frequently seek to answer complex biological questions about cell clusters, necessitating additional downstream analyses. The lack of adaptable, multimodal tools for comprehensive cell cluster analysis represents a significant bottleneck in advancing biological discovery.

Transformer models pose an interesting solution to the problems faced in traditional cluster analysis. Large transformers are pretrained on a large corpus of knowledge (some including biology). scRNA-seq generates gene expression information that can be provided to a transformer to perform cell type annotation and interpretation of cell clusters. However, vanilla large transformers are trained on a wide breadth of knowledge and lack specificity for scRNA-seq analysis. Some transformer models, such as BioBERT [5], address this by fine-tuning a vanilla large transformer model (BERT) [1] on domain-specific biology content. While BioBERT is well-suited for biomedical classification tasks, it would need to be further fine-tuned on large sets of scRNA-seq data to improve its ability to classify cell types in scRNA-seq datasets effectively. Additionally, BioBERT is not designed for generative question answering, which could be a valuable feature for researchers seeking to interact with data in a more dynamic and flexible manner.

Instead, we use OpenAI's [8] generative Large Language Model (LLM), GPT-4, to annotate cell types and characterize cell clusters from scRNA-seq data. GPT-4 enabled us to leverage natural language processing to dynamically interact with heterogeneous datasets, while also providing question-and-answer capabilities for cell type annotation and cluster characterization. Recognizing that GPT-4 also suffers from lack of biological specificity, we provided gene ontology and implemented Retrieval Augmented Generation (RAG) as a means to strengthen the model's ability to accurately annotate cell types and characterize cell clusters.

Our model's prompt provides GPT-4 with the top fifty differentially expressed genes from a cluster, along with gene ontology data and relevant NCBI [6] journal articles, thereby enriching GPT-4 with dataset-specific context.

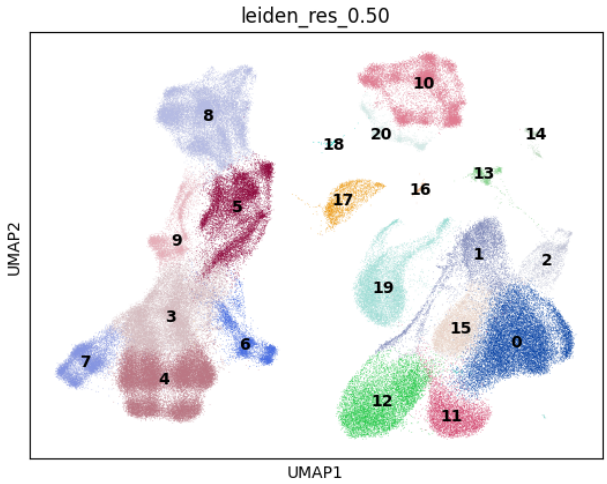
II. Results

To ensure consistency and reproducibility, we used a publicly available Peripheral Blood Monocytes (PBMC) dataset [11] for scRNA-seq analysis. After processing the data and clustering, we checked the cell types of each cell in the clusters for accuracy against the documented cell type of each cell in the dataset. We found that 45% of clusters exhibited accuracy $\geq 90\%$ and 60% of clusters had an accuracy $\geq 80\%$ I.

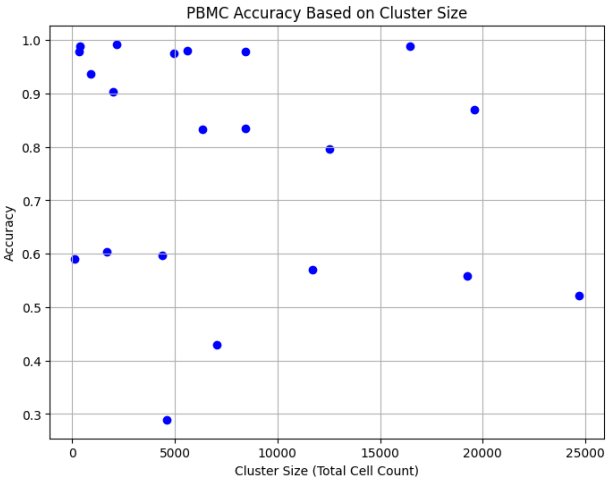
Since we observed both varying sizes of clusters and accuracy of cell types within the clusters, we also wondered if the two

Cluster	Cell Type	Cell Type Count	Total Cell Count	Accuracy
0	CD14 Mono	16293	16476	0.9889
1	CD16 Mono	5294	6351	0.8336
10	B naive	6673	11714	0.5697
11	CD14 Mono	4847	4966	0.9760
12	CD14 Mono	8264	8450	0.9780
13	pDC	857	914	0.9376
14	Plasmablast	363	367	0.9891
15	CD14 Mono	5515	5628	0.9799
16	HSPC	322	329	0.9787
17	Platelet	2153	2169	0.9926
18	Eryth	82	139	0.5899
19	CD14 Mono	7025	8420	0.8343
2	cDC2	1804	1996	0.9038
20	B naive	1029	1704	0.6039
3	CD4 TCM	10748	19253	0.5583
4	CD4 Naive	12899	24697	0.5223
5	CD8 TEM	9984	12527	0.7970
6	CD8 Naive	1333	4606	0.2894
7	CD4 Naive	3029	7050	0.4296
8	NK	17054	19613	0.8695
9	MAIT	2621	4395	0.5964

TABLE I: PBMC cluster accuracy based on majority cells per cluster/max cells per cluster



(a) Leiden UMAP visualization of clusters



(b) Accuracy compared to Cluster Size

Fig. 1: a) UMAP of PBMC cell clusters using the Leiden algorithm. b) Accuracy by cluster size based on Leiden algorithm using max cells per cluster.

conditions correlated. We plotted the cluster sizes against cell type accuracy and found no correlation between the two 1. This suggests that the representation of differentially expressed genes within the cluster is independent of cell number. Based on the necessity for accurate cell clustering to further validate the model, we chose to proceed with clusters that exhibited an accuracy $\geq 90\%$ compared to the ground truth. We selected nine (9) clusters from the PBMC dataset to utilize in testing our model.

We chose the bioinformatics suite CellTypist [13] to semi-automatically annotate the PBMC dataset and clusters for comparison to the GPT-based models. CellTypist performed well on the PBMC clusters, and was able to achieve an overall 78% accuracy 3 at broadly annotating our nine cluster test set. However, CellTypist did struggle to accurately predict cell types overall when compared to the PBMC cell annotations 2.

To further validate our GPT-4o-RAG model, we compared its performance against non-RAG variations, including GPT-4 and GPT-4o. To ensure consistency, we used the same prompt across all three models. GPT-4 achieved an accuracy of 30%, while GPT-4o improved the accuracy to 70% based on an accuracy scoring metric of exact match, partial match, slight mismatch, and no match 3.

We subsequently ran our GPT-4o-RAG model using the same gene sets and biological processes data. The RAG component included 5000 articles retrieved from NCBI, which were embedded, vectorized, and selectively incorporated into the input prompt for GPT-4o. Our GPT-4o-RAG model was able to achieve 81% accuracy 3.

III. Materials and Methods

GPT

A. Preprocessing

We used the Scanpy [12] library in Python [3] to preprocess the PBMC dataset [7]. The data was turned into an AnnData object for ease of processing. The $m \times n$ matrix consisted of genes as instances and indvidual cells as features. The dataset was first checked for cell quality, then reduced through Principal Component Analysis (PCA). The reduced data was processed through a nearest-neighbors graph algorithm, followed by Uniform Manifold Approximation and Projection (UMAP) for visualization. Finally, the Leiden clustering algorithm was applied. The Leiden resolution was set at 0.50 to achieve the desired clustering. Top differentially expressed genes were extracted from each cluster for downstream analysis.

B. Prompt

We constructed our prompt in accordance with the guidelines set forth by Hou and Zhicheng [4], ensuring the inclusion of terminology compatible with our RAG implementation of NCBI articles. The prompt used was as follows: *"Identify the cell type of human peripheral blood mononuclear cells using*

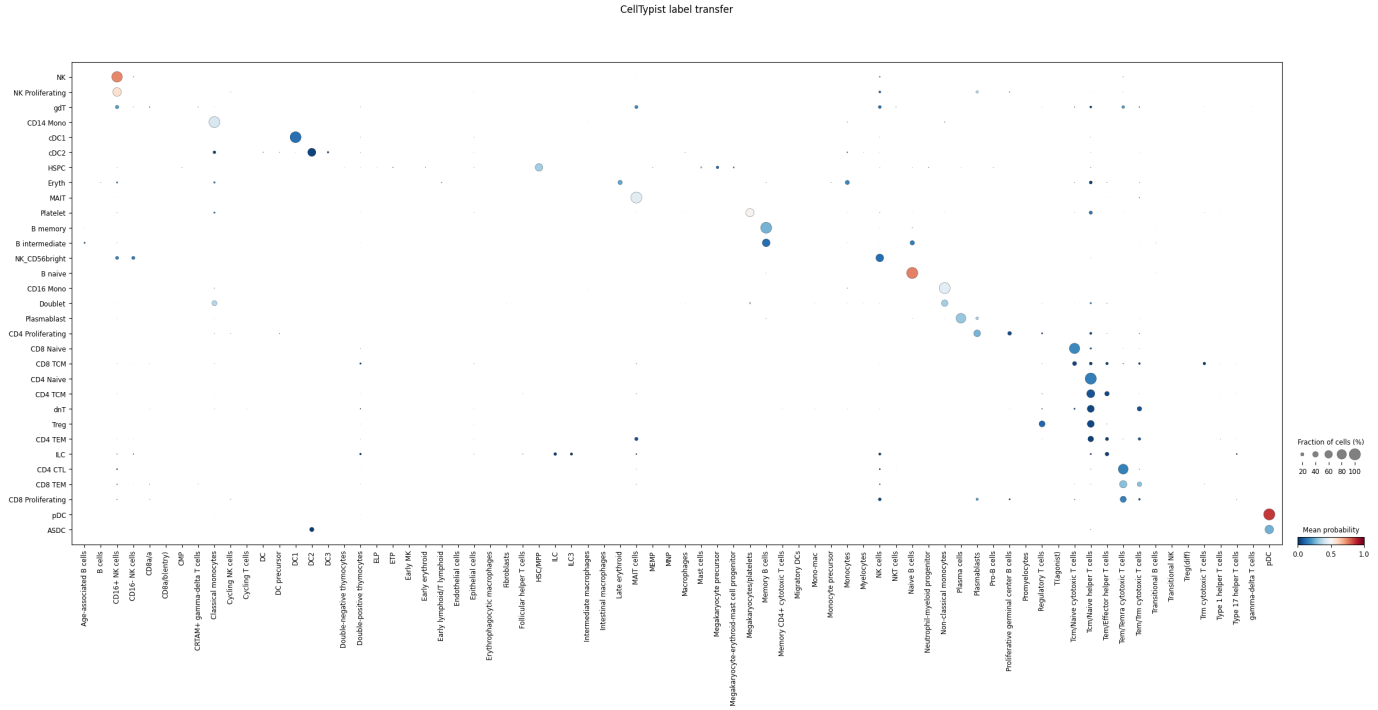


Fig. 2: CellTypist cell label predictions versus PBMC manual annotations

the following genes, biological processes, and PMC journal articles."

C. GPT non-RAG

For our GPT without the RAG, we had first extracted our top fifty differentially expressed genes from each cluster from the PBMC dataset and used the Database for Annotation Visualization and Integrated Discovery (DAVID) [2, 10] to get the gene ontology. Once we obtained the information from DAVID we extracted the gene ontologiy biological processes to a CSV file. Finally we have what we need to feed the GPT to start training it.

We created a script for processing the two CSV files containing information on gene markers and biological processes. The script first loads the data from these files. It then constructs a detailed prompt by iterating over the clusters of gene markers, gathering the top genes and their associated biological processes (if available). The genes are formatted into strings, and any missing biological process data is handled appropriately. For each cluster, the script assembles information on the top genes and biological processes into a structured text format, which is finally printed as a prompt for further bioinformatics analysis.

We then have a second script where we use OpenAI's [8] GPT model to analyze cluster information related to gene markers and biological processes. The script first sets up the OpenAI client with an API key and defines a prompt that asks GPT to analyze the data and infer a cell type. It iterates over the gene marker clusters, gathers the top genes, and associated biological processes, formatting them into a structured prompt. It then counts the number of tokens in the prompt to ensure it fits within the model's limit. Finally, the script sends the prompt to GPT-4 (or another specified model) using the OpenAI API, receives a response, and prints the result.

D. GPT + RAG

For our RAG portion we use the same methods above for integration of gene names and gene ontology biological processes. Then, it starts by searching for articles related to "Peripheral Blood Monocytes" within a specific date range. After gathering a list of relevant article IDs (PMIDs), it fetches the full-text XML of these articles in batches. The script validates and parses the XML content, extracting the article titles and text while filtering out unwanted phrases. The article content is then chunked into smaller text pieces for further analysis. Next, it generates embeddings for each article using a pre-trained model and normalizes these embeddings. Finally, the script creates a FAISS [9] index to facilitate efficient similarity searches and saves the index and article titles for later use.

Then finally in our last script we use OpenAI's GPT-4 to analyze gene clusters and retrieve relevant information from NCBI articles. First, it loads a pre-existing FAISS index and article titles, which were previously created using embeddings from a SentenceTransformer model. The script then defines a function to query the FAISS index by encoding the user's query and finding the most similar articles, filtered by a similarity threshold. The retrieved articles, along with their titles and content, are incorporated into a prompt along with gene cluster information and biological processes. The prompt is sent to GPT-4, which is expected to analyze the data and infer a cell

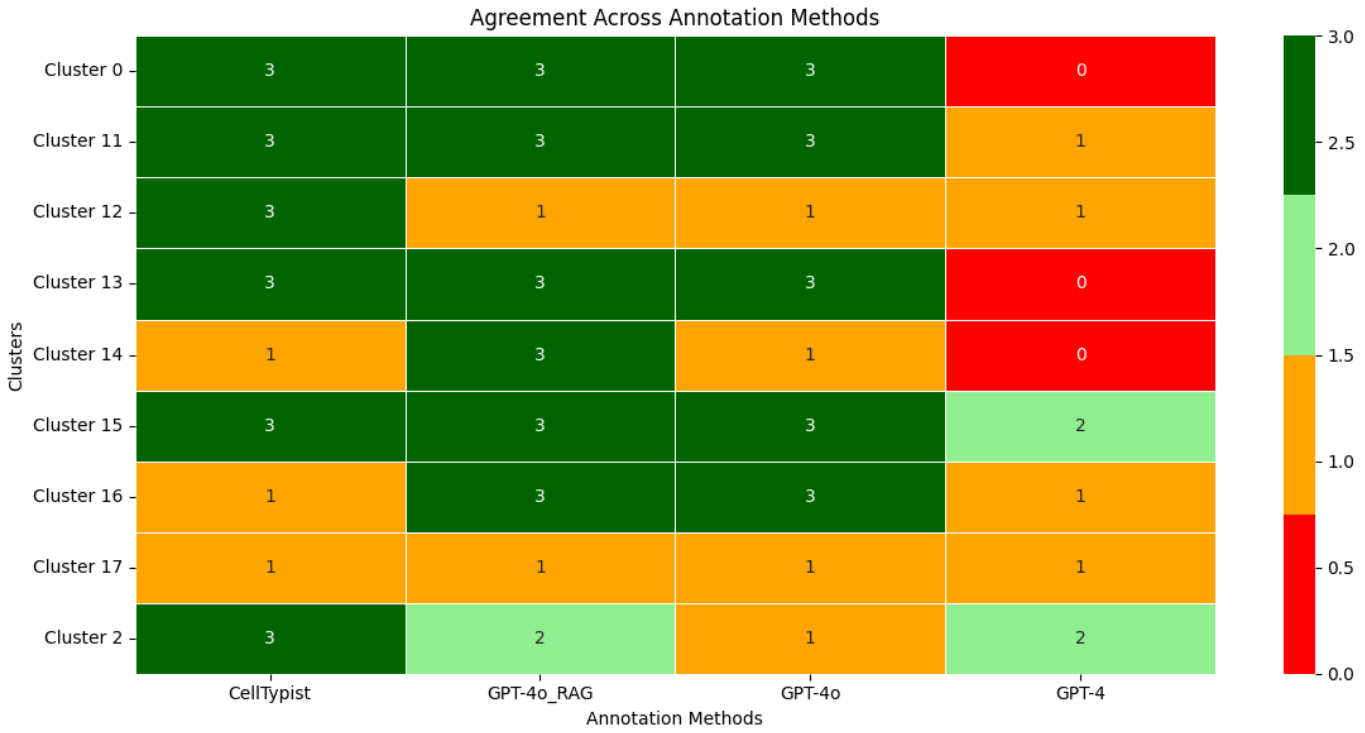


Fig. 3: Accuracy of annotation across methods using semi-automatic annotation, GPT models and GPT + RAG

type based on the provided context. Finally, the script prints the response generated by GPT-4 into a place where we save the response.

IV. Discussion

The result of our study has highlighted the use of single cell RNA-sequencing analysis using the retrieval-augmented generation (RAG) framework with LLMs. Our approach achieved high accuracy with clear gene expression signatures. By utilizing GPT-4o + RAG, we helped bridge the gap between natural language processing and bioinformatics to provide meaningful and interpretable cell-type annotation.

The high accuracy demonstrates our model can effectively handle biological data. Beyond just raw cell-type predictions, our model is framed to provide contextual summaries that are enriched with biological knowledge retrieved from NCBI articles. This provides deeper understanding of the biological processes underlying each cluster by pulling additional information from scientific articles.

Utilizing different retrieval mechanisms has allowed our model a more dynamic adaptation to diverse datasets without having to perform extensive fine-tuning. This makes the framework suitable for large-scale scRNA-seq studies spanning multiple conditions and tissue types with small adjustments. The model is flexible for various scRNA-seq studies.

Looking ahead, several key steps can enhance our framework:

1) **Benchmarking Protocols**: Developing standardized evaluation frameworks using publicly available, well-annotated scRNA-seq datasets to validate and compare performance.
2) **Integration of Data Sets**: Incorporating complementary data types, such as protein analysis or other specific biological analysis, to enrich cell-type characterization and improve cluster interpretations.
3) **Interactive Tool Development**: Design a user-friendly, interactive software tool that allows researchers work with the raw data, query the system, and present visualizations of the results.
4) **Community Engagement**: Collaborating with biologists, bioinformaticians, and computer scientists to gather feedback on usability and identify any areas that could be improved.

By focusing on these steps we aim to refine our method further, making it a practical tool for single cell research, accelerating discoveries and fostering interdisciplinary collaboration.

Acknowledgment

We gratefully acknowledge the support and guidance of Dr. Trilce Estrada, Dr. Eric Denkers, and Nidia Vaquera, whose expertise and mentorship were invaluable throughout this research. We extend our thanks to the Department of Biology and the Department of Computer Science at the University of New Mexico for providing access to computational resources and laboratory facilities.

Lastly, we are deeply grateful to our friends and families for their encouragement and support throughout this journey.

REFERENCES

[1] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *arXiv preprint arXiv:1810.04805* (2018). URL: https://arxiv.org/abs/1810.04805.

[2] Huang DW, Sherman BT, and Lempicki RA. "Systematic and integrative analysis of large gene lists using DAVID Bioinformatics Resources". In: *Nature Protoc.* 4.1 (2009). [PubMed], pp. 44–57. DOI: 10.1038/nprot.2008.211. URL: https://doi.org/10.1038/nprot.2008.211.

[3] Python Software Foundation. *Python Programming Language*. 2024. URL: https://www.python.org/.

[4] W. Hou and Z. Ji. "Assessing GPT-4 for cell type annotation in single-cell RNA-seq analysis". In: *Nature Methods* 21 (2024), pp. 1462–1465. DOI: 10.1038/s41592-024-02235-4. URL: https://doi.org/10.1038/s41592-024-02235-4.

[5] Jinhyuk Lee et al. "BioBERT: a pre-trained biomedical language representation model for biomedical text mining". In: *arXiv preprint arXiv:2001.09977* (2020). URL: https://arxiv.org/abs/2001.09977.

[6] National Center for Biotechnology Information. *NCBI Home Page*. 2024. URL: https://www.ncbi.nlm.nih.gov/.

[7] Gene Expression Omnibus. *GSE164378: Dataset Title (replace with actual title if available)*. Accessed: 2024-12-05. 2024. URL: https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE164378.

[8] OpenAI. *OpenAI: GPT-3 and other models*. 2023. URL: https://www.openai.com/.

[9] Facebook AI Research. *FAISS: A Library for Efficient Similarity Search and Clustering of Dense Vectors*. 2024. URL: https://github.com/facebookresearch/faiss.

[10] B.T. Sherman et al. "DAVID: a web server for functional enrichment analysis and functional annotation of gene lists (2021 update)". In: *Nucleic Acids Research* 50.W1 (2022). [PubMed], W216–W223. DOI: 10.1093/nar/gkac194. URL: https://doi.org/10.1093/nar/gkac194.

[11] Tim Stuart et al. "Comprehensive Integration of Single-Cell Data". In: *Resource* 177.7 (June 2019). Open Archive, 1888–1902.e21. DOI: 10.1016/j.cell.2019.04.003. URL: https://doi.org/10.1016/j.cell.2019.04.003.

[12] Alexander Wolf et al. *Scanpy: Large-scale single-cell gene expression data analysis*. https://scanpy.readthedocs.io/en/stable/. 2021.

[13] Dominik Paul Zych, Geoffrey Schiebinger, Martin Rosvall, et al. "CellTypist: reference single-cell type annotations leveraging numerous publicly available datasets". In: *Genome Biology* 23.1 (2022), p. 134.