Molecule Retrieval with Natural Language Queries

Kyllian Asselin de Beauville
Institut Polytechnique de Paris
kyllian.asselin-de-beauville@polytechnique.edu

Aymane Rahmoune
Institut Polytechnique de Paris
aymane.rahmoune@polytechnique.edu

Abstract

This report describes the approach and the methods we used for the challenge of retrieving molecules using natural language queries, as presented in the ALTEGRAD 2023 Data Challenge. We investigate the integration of two very different data modalities—natural language and molecular structures (graphs)—to address the complex task of identifying relevant molecules based on textual descriptions. Our approach leverages contrastive self-supervised learning to co-train a text encoder and a molecule encoder, aiming to bridge the semantic gap between text descriptions and molecular graph representations. By mapping text-molecule pairs into a shared representation space, we facilitate the retrieval of molecules corresponding to natural language queries without direct reference information.

1 Introduction

Integrating natural language processing with molecular graph analysis to retrieve molecules based on textual descriptions presents a significant challenge in chemistry that could improve drug discovery and design. This report addresses the **ALTEGRAD 2023 Data Challenge**, focusing on leveraging contrastive self-supervised learning to bridge the semantic gap between textual queries and molecular graphs. Inspired by the paper that introduced the **Text2Mol** [6] task, we develop a model that co-trains a text encoder and a molecule encoder, thereby facilitating accurate molecule retrieval by learning a shared representation space that enhances the semantic alignment between textual descriptions and molecular structures. Our approach, methodology, and results aim to advance this novel task of cross-modal molecule retrieval, contributing to the evolving intersection of chemistry and artificial intelligence.

2 Dataset

The dataset, **ChEBI-20** [6], contains 33,010 compound-description pairs derived from PubChem [10] and Chemical Entities of Biological Interest (ChEBI) [5, 7], keeping molecules with descriptions of more than 20 words. It is split into training (26,408 samples), validation (3,301 samples), and test sets (3,301 samples), ensuring comprehensive coverage for model training and evaluation. The training and validation data are provided in tab-separated values files, alongside a token embedding dictionary that maps substructure tokens, facilitating the intricate task of learning representations. Test files, `testtext.txt` and `testcids.txt`, contain 3,301 descriptions and corresponding graph IDs, respectively, allowing for precise matching of descriptions to molecules. This dataset is invaluable for training and evaluating models designed to retrieve molecules based on natural language queries.

3 Methodology

Our approach combines advanced pre-trained text encoders with diverse graph neural network (GNN) architectures and carefully chosen loss functions to co-train our models. This strategy is designed to effectively bridge the semantic gap between textual descriptions and molecular structures. By leveraging the strengths of each component, we aim to create a robust system capable of accurately matching natural language queries to their corresponding molecular graphs, enhancing the precision of molecule retrieval.

3.1 Text Encoders

Our exploration of text encoders was guided by the need to effectively process and understand natural language descriptions of molecular structures.

DistilBERT We began with DistilBERT [13], a lightweight variant of BERT, for its efficiency in training and deployment. This choice allowed us to capture complex relationships within textual descriptions, setting a baseline for understanding molecular information in natural language.

BioBERT Moving towards domain-specific encoders, we evaluated BioBERT [12], optimized for biomedical texts. Its pre-training on biomedical literature made it a strong candidate for enhancing our model's ability to interpret descriptions of molecular structures and functions, crucial for our retrieval task.

SciBERT Lastly, SciBERT [1] was selected for its focus on scientific texts, enabling our model to adeptly navigate and extract meaning from specialized literature. This progression towards increasingly specialized encoders underpinned our strategy to refine the semantic alignment between text descriptions and molecular graphs.

3.2 Graph Encoders

Our investigation into graph neural networks (GNNs) [14, 21] began with foundational architectures, progressing to more advanced variants to optimize molecular graph representation.

GCN We started with the Graph Convolutional Network (GCN) [11, 3], chosen for its fundamental approach to capturing graph-based structures through neighbor node information aggregation. This baseline allowed us to understand local and global dependencies within molecular graphs effectively.

GIN We then explored the Graph Isomorphism Network (GIN) [19], attracted by its theoretical capability to learn graph-level representations crucial for identifying molecular similarities. Despite its potential, GIN did not deliver the performance improvements we anticipated, leading us to pivot our focus.

GAT and GATv2 Our attention turned to the Graph Attention Network (GAT) [18], and its variant GATv2 [2], for their use of attention mechanisms to prioritize information from significant nodes. This adaptability showed promise in discriminating molecular features more effectively.

GAT with Residuals Inspired by the success of ResNet [8] in other domains, we introduced residuals to GAT to enhance information flow and mitigate vanishing gradients, thus supporting deeper model architectures.

GAT with Gated Residuals Progressing further, we incorporated gated residuals into GAT, drawing inspiration from UniMP [15] and AMAN [20]. This allowed for selective information propagation, highlighting the relevance of specific nodes and pathways in the graph.

RGAT-LSTM Lastly, inspired by the innovative combinations of attention mechanisms and dynamic evaluation in graph neural networks, we implemented RGAT-LSTM [9]. This architecture blends GAT's attention mechanism with the dynamic layer-wise information processing capability, offering a refined approach to graph representation learning.

3.3 Loss Functions

In our co-training process, we experimented with various loss functions to optimize the alignment between text and graph embeddings.

Contrastive Loss Initially, we used the contrastive loss function to minimize the distance between similar text-molecule pairs and maximize it for dissimilar pairs in the shared representation space. This foundational approach aimed to directly align with our objective of improving molecule retrieval accuracy.

Negative Sampling Contrastive Loss To enhance model robustness, we extended the contrastive loss with negative sampling, introducing additional non-matching pairs. This modification was intended to make our model more discerning by effectively handling a greater variety of non-corresponding pairs.

InfoNCE Loss Experiencing limited success with the previous methods, we adopted the InfoNCE [17] loss, which provided a significant performance boost. The inclusion of noise contrastive estimation allowed for a more effective differentiation between matching and non-matching pairs by leveraging a broader context within the embedding space.

$$\mathcal{L}_{\text{InfoNCE}}(\mathbf{v}_1, \mathbf{v}_2) = -\log \frac{\exp(\mathbf{v}_1 \cdot \mathbf{v}_2/\tau)}{\sum_{k=1}^{N} \mathbb{1}_{[k\neq 1]} \exp(\mathbf{v}_1 \cdot \mathbf{v}_k/\tau)} \tag{1}$$

NT-Xent Loss The NT-Xent (Normalized Temperature-Scaled Cross Entropy) [16, 4] loss, incorporating the cosine similarity ($\text{sim}(\mathbf{v}_1, \mathbf{v}_2) = \frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{\|\mathbf{v}_1\| \|\mathbf{v}_2\|}$) between embeddings, became our choice for further refinement. This loss function, with its temperature scaling, refined our model's sensitivity to the nuances of alignment between text and graph embeddings, leveraging the same metric we used to assess retrieval performance.

$$\mathcal{L}_{\text{NT-Xent}}(\mathbf{v}_1, \mathbf{v}_2) = -\log \frac{\exp(\text{sim}(\mathbf{v}_1, \mathbf{v}_2)/\tau)}{\sum_{k=1}^{N} \mathbb{1}_{[k\neq 1]} \exp(\text{sim}(\mathbf{v}_1, \mathbf{v}_k)/\tau)} \tag{2}$$

3.4 Similarity Metrics

In our quest to accurately measure the alignment between textual description embeddings and molecular graph embeddings, we investigated various similarity metrics, each offering unique perspectives on similarity:

Cosine Similarity Initially, we used cosine similarity for its ability to measure the cosine of the angle used two vectors. This metric is advantageous because it evaluates similarity based on vector orientation, disregarding magnitude, which is ideal for high-dimensional data like embeddings. Its normalization aspect makes it particularly suitable for our application, ensuring that the similarity measure remains within a bounded range.

$$\text{Cosine Similarity}(\mathbf{v}_1, \mathbf{v}_2) = \frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{\|\mathbf{v}_1\| \cdot \|\mathbf{v}_2\|} \tag{3}$$

Euclidean Similarity We explored Euclidean similarity, which assesses the direct distance between two points in vector space. While intuitive for spatial distance measurement, its sensitivity to the magnitude of vectors posed challenges in capturing the nuanced similarity between embeddings, leading to its lesser suitability for our context.

$$\text{Euclidean Similarity}(\mathbf{v}_1, \mathbf{v}_2) = \frac{1}{1 + \|\mathbf{v}_1 - \mathbf{v}_2\|} \tag{4}$$

Dot Product Similarity Dot product similarity was considered for its straightforward calculation of vector alignment. However, the lack of normalization means it can be influenced by the magnitude of the vectors, which can vary significantly in embedding spaces, thus affecting its effectiveness in our scenario.

$$\text{Dot Product Similarity}(\mathbf{v}_1, \mathbf{v}_2) = \mathbf{v}_1 \cdot \mathbf{v}_2 \tag{5}$$

After all these considerations, we concluded that cosine similarity was most aligned with our objectives, offering a balance between computational efficiency and the ability to accurately reflect the conceptual similarity between textual and molecular structures.

4 Results

4.1 Evaluation

The performance of the models was assessed using the Label Ranking Average Precision (LRAP) [1] score, which evaluates the quality of label rankings produced by the model for each sample. LRAP is defined as follows:

$$LRAP(\mathbf{y}, \mathbf{y}') = \frac{1}{\text{nsamples}} \sum_{i=0}^{\text{nsamples}-1} \frac{1}{||\mathbf{y}_i||_0} \sum_{j:y_{ij}\neq 0} \frac{1}{|L_{ij}| \cdot \text{rank}_{ij}}, \tag{6}$$

where $L_{ij} = \{k : y_{ik} = 1, y'_{ik} \geq y'_{ij}\}$, $\text{rank}_{ij} = |k : y'_{ik} \geq y'_{ij}|$, and $||\cdot||_0$ denotes the cardinality of the set.

This metric calculates the fraction of true labels ranked higher than a threshold, emphasizing the accuracy of the most relevant labels for each sample. In contexts where there is exactly one relevant label per sample, as in our case, LRAP is equivalent to the Mean Reciprocal Rank (MRR) [2], providing a direct measure of the model's ability to rank the correct label highest among all labels.

4.2 Model

Our model achieves its best performance by integrating SciBERT as the text encoder with a 3-layer RGAT-LSTM graph encoder, optimized through NT-Xent loss. This architecture ensures robust co-training, utilizing dot product similarity to evaluate the alignment between textual descriptions and molecular graph representations.

4.2.1 Text Encoder: SciBERT

We use SciBERT as our text encoder to leverage its pre-training on a vast corpus of scientific literature, enabling it to accurately capture context from molecular descriptions. To ensure compatibility with the graph encoder, its output is projected into 300 dimensions. This projection, followed by layer normalization, optimizes the embedding process and alignment between the two modalities.

Table 1: Comparison of Text Encoders with GCN and Contrastive Loss (5 epochs)

Text Encoder	Performance
DistilBERT	0.3480
BioBERT	0.4393
SciBERT	**0.4535**

4.2.2 Graph Encoder: RGAT-LSTM (3 Layers)

The graph encoder is built upon the RGAT-LSTM 1 architecture, incorporating three layers of Graph Attention Networks (GAT) with Residual Gated Graph LSTM units 2. This design effectively captures both local and global dependencies within molecular graphs. The addition of dropout between convolutional layers and layer normalization ensures robust representation learning and aligns the graph encoder's output with the text encoder's projected space.

4.2.3 Loss Function: NT-Xent

The Normalized Temperature-Scaled Cross-Entropy (NT-Xent) loss is used with a temperature of 0.1 for co-training, promoting proximity between similar text-molecule pairs and distancing dissimilar ones within the shared representation space. This loss function's stability and effectiveness are crucial for the generation of meaningful embeddings.

[1] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.label_ranking_average_precision_score.html
[2] https://en.wikipedia.org/wiki/Mean_reciprocal_rank

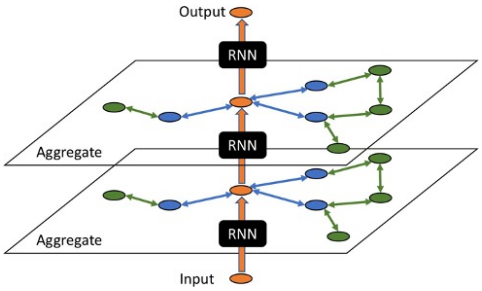

Figure 1: RGAT-LSTM Architecture

$$\hat{X}^{l+1} = \hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}H^{l}\Theta^{l}$$
$$I^{l+1} = \sigma(X^{l+1}W_i + H^{l}U_i + [b_i]_N)$$
$$F^{l+1} = \sigma(X^{l+1}W_f + H^{l}U_f + [b_f]_N)$$
$$O^{l+1} = \sigma(X^{l+1}W_o + H^{l}U_o + [b_o]_N)$$
$$\hat{C}^{l+1} = tanh(X^{l+1}W_c + H^{l}U_c + [b_c]_N)$$
$$C^{l+1} = F^{l+1} \circ C^{l} + I^{l+1} \circ \hat{C}^{l+1}$$
$$H^{l+1} = O^{l+1} \circ tanh(C^{l+1})$$

Figure 2: Mathematical Formulation

Table 2: Comparison of Graph Encoders with SciBERT and NT-Xent Loss (0.1 temperature, 10 epochs)

Graph Encoder	**Performance**
GAT with Residuals	0.8241
GAT with Gated Residuals	0.8286
RGAT-LSTM	**0.8502**

4.2.4 Similarity Metric: Cosine Similarity

Cosine similarity has proven to be highly effective in capturing semantic relationships between textual descriptions and molecular graph representations, contributing significantly to the model's accuracy in retrieving molecules based on natural language queries.

4.2.5 Training

The model was trained using the AdamW optimizer, adopting a fine-tuning learning rate of 3e-5 for the text encoder and 1e-4 for the graph encoder, to respect the distinct learning dynamics of each encoder. A linear annealing rate with 1,500 steps of warmup was applied to smoothly adjust the learning rates, enhancing training stability. The training, conducted over 60 epochs with a batch size of 32, took 8 hours on a single A100 GPU and achieved optimal validation performance at the 55th epoch, demonstrating the effectiveness of our architecture and training strategy.

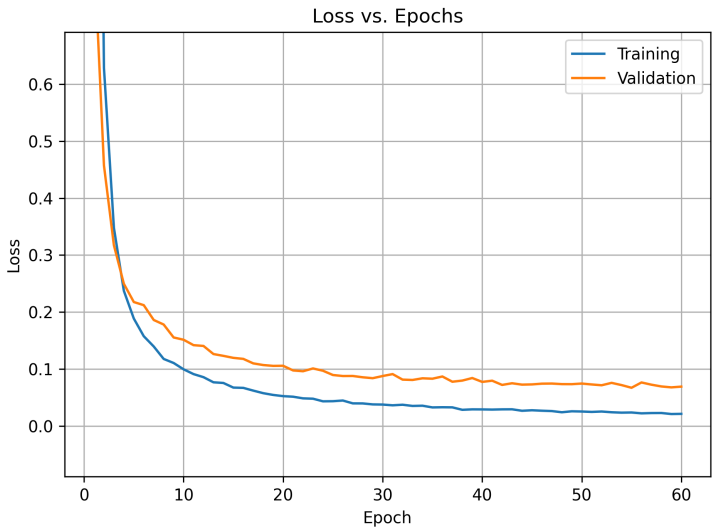

Figure 3: Loss Over Training Epochs

5 Conclusion and Future Work

In this study, we presented a comprehensive methodology for addressing the ALTEGRAD 2023 Data Challenge, leveraging state-of-the-art pre-trained text encoders and diverse graph neural network architectures. Our experiments, incorporating DistilBERT, BioBERT, and SciBERT as text encoders, along with various graph encoders and loss functions, demonstrated the efficacy of the proposed approach in bridging the semantic gap between natural language queries and molecular graph representations.

The results highlight the importance of choosing appropriate combinations of text and graph encoders, loss functions, and similarity metrics for achieving optimal performance. Our best model, combining SciBERT with RGAT-LSTM, NT-Xent loss, and cosine similarity, outperformed other configurations, showcasing the effectiveness of our chosen architecture.

Future research should focus on exploring new pre-trained text encoders and innovative graph encoder architectures to deepen molecular data comprehension. Investigating alternative loss functions and similarity metrics could unlock further improvements. Additionally, enhancing the model's scalability for larger datasets and its adaptability across various domains remains crucial. Collaboration across disciplines promises to refine and expand the model's applications, driving forward the integration of chemistry and artificial intelligence.

References

[1] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text, 2019.

[2] Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks?, 2022.

[3] Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks, 2020.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020.

[5] Kirill Degtyarenko, Paula de Matos, Marcus Ennis, Janna Hastings, Martin Zbinden, Alan McNaught, Rafael Alcántara, Michael Darsow, Mickaël Guedj, and Michael Ashburner. ChEBI: a database and ontology for chemical entities of biological interest. *Nucleic Acids Research*, 36(suppl_1):D344–D350, 10 2007.

[6] Carl Edwards, ChengXiang Zhai, and Heng Ji. Text2Mol: Cross-modal molecule retrieval with natural language queries. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 595–607, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[7] Janna Hastings, Gareth Owen, Adriano Dekker, Marcus Ennis, Namrata Kale, Venkatesh Muthukrishnan, Steve Turner, Neil Swainston, Pedro Mendes, and Christoph Steinbeck. ChEBI in 2016: Improved services and an expanding collection of metabolites. *Nucleic Acids Res*, 44(D1):D1214–9, October 2015.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[9] Binxuan Huang and Kathleen M. Carley. Residual or gate? towards deeper graph neural networks for inductive graph representation learning, 2019.

[10] Sunghwan Kim, Paul A Thiessen, Evan E Bolton, Jie Chen, Gang Fu, Asta Gindulyte, Lianyi Han, Jane He, Siqian He, Benjamin A Shoemaker, Jiyao Wang, Bo Yu, Jian Zhang, and Stephen H Bryant. PubChem substance and compound databases. *Nucleic Acids Res*, 44(D1):D1202–13, September 2015.

[11] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks, 2017.

[12] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, September 2019.

[13] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2020.

[14] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009.

[15] Yunsheng Shi, Zhengjie Huang, Shikun Feng, Hui Zhong, Wenjin Wang, and Yu Sun. Masked label prediction: Unified message passing model for semi-supervised classification, 2021.

[16] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[17] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2019.

[18] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks, 2018.

[19] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks?, 2019.

[20] Wenyu Zhao, Dong Zhou, Buqing Cao, Kai Zhang, and Jinjun Chen. Adversarial modality alignment network for cross-modal molecule retrieval. *IEEE Transactions on Artificial Intelligence*, 5(1):278–289, 2024.

[21] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications, 2021.