ShanghaiTech CS177, 2024, pp. 1–6

doi: DOI HERE
Advance Access Publication Date: 23-6-2024
Paper

PAPER

OmniSynth: Cell-line Specific Synthetic Lethality Prediction Using Multi-Omics Data

Mutian Hong,[1] Zhihao Tang,[1] Siyu Tao[1] and Jie Zheng[1,*]

[1]School of Information Science and Technology, ShanghaiTech University, 1 Zhongke Road, 201210, Shanghai, China
[*]Corresponding author. zhengjie@shanghaitech.edu.cn
FOR PUBLISHER ONLY Received on Date Month Year; revised on Date Month Year; accepted on Date Month Year

Abstract

In biology, synthetic lethality refers to a condition where the combination of two gene mutations leads to cell death, whereas the deletion of any single gene alone does not cause death. This phenomenon is significant in cancer research because cancer cells often rely on specific gene combinations to survive and to escape from the immune system. By identifying these lethal combinations, researchers can develop more targeted cancer therapies that selectively kill cancer cells by regulating the specific gene without affecting normal cells. This study presents OmniSynth, a framework designed to predict cell-line specific synthetic lethality interactions. OmniSynth integrates multiple omics data sources, including biomedical knowledge graphs, protein data, gene expression, mutations, and other bioinformatics data, to predict synthetic lethality interactions in specific cell lines. The results demonstrate OmniSynth's high accuracy and generalization ability, particularly in cell lines A375, A549, and Jurkat, highlighting its broad applicability across different cell line contexts.

Key words: synthetic lethality, multi-omics data, machine learning, cancer research

Introduction

In biology, synthetic lethality refers to a condition where the combination of two gene mutations leads to cell death, whereas the deletion of any single gene alone does not cause death. This phenomenon is significant in cancer research because cancer cells often rely on specific gene combinations to survive and to escape from the immune system. By identifying these lethal combinations, researchers can develop more targeted cancer therapies that selectively kill cancer cells by regulating the specific gene without affecting normal cells.

As the fact that there are only few pairs of SL through the whole gene pairs in a cell, and the traditional wet-lab takes a long time and a high cost to find the useful SL pairs, so some methods that can reduce the wet-lab experiments is essential.

The significance of machine learning in predicting synthetic lethality lies in its ability to handle vast amounts of genetic data and identify potential lethal combinations. By training models, machine learning algorithms can detect patterns in complex biological data and predict which gene combinations might lead to synthetic lethality based on data and without the explicit prior hand-made knowledge. This approach not only increases research efficiency and reduces experimental time and costs but also uncovers gene combinations that traditional experimental methods might miss, providing new avenues for personalized medicine and precision therapy.

Context-specific synthetic lethality refers to situations where synthetic lethality occurs only under certain biological conditions or contexts, such as specific cell types, environmental conditions, or the presence of particular genetic backgrounds. This concept is important because it recognizes that the interactions between genes and their effects on cell viability can vary depending on the cellular environment and other factors.

Cell-line specific synthetic lethality focuses on synthetic lethal interactions that occur uniquely in particular cell lines. This specificity is crucial for cancer research, as different cancer cell lines can have distinct genetic and epigenetic landscapes. By studying cell-line specific synthetic lethal interactions, researchers can identify unique vulnerabilities in different cancer types and experimenter can also inspect the synthetic lethality in specific cell lines.

Additionally, cell-line specific synthetic lethality offers a significant advantage for experimental validation. In pan-cancer predictions, biologists often face challenges in selecting appropriate cell lines for testing, as the predicted interactions may not be relevant across different cell types. However, in context-specific scenarios, biologists can confidently choose a single, well-defined cell line for validation, ensuring that the experimental conditions are optimal for observing the predicted synthetic lethal interactions. This targeted approach not only streamlines the experimental process but also enhances the reliability and relevance of the findings.

Therefore, we propose OmniSynth, a framework designed to predict cell-line specific synthetic lethality interactions. OmniSynth integrates multiple omics data sources, including biomedical knowledge graphs, protein data, gene expression, mutations, and other bioinformatics data, to predict synthetic lethality interactions in specific cell lines. By combining

these diverse data sources, OmniSynth can capture a more comprehensive biological picture, thereby enhancing the accuracy and reliability of predictions.

The core advantage of OmniSynth lies in its ability to integrate multimodal data. Biomedical knowledge graphs provide known interactions between genes, protein data reveal protein functions and interactions, gene expression data show gene activity under various conditions, and mutation information offers insights into how specific genetic changes affect cell viability. By integrating these heterogeneous data types, OmniSynth can capture the biological mechanisms from multiple perspectives, leading to the identification of more biologically meaningful synthetic lethality interactions and can find the implicit relations that between these different datas.

We validated OmniSynth in three cell lines: A375, A549, and Jurkat. The results demonstrated that OmniSynth excels in predicting cell-line specific synthetic lethality interactions, showing high accuracy and generalization ability. Specifically, OmniSynth exhibited outstanding predictive performance in these cell lines, reliably identifying cell-specific synthetic lethality interactions. This outcome not only validates OmniSynth's predictive capabilities but also showcases its broad applicability across different cell line contexts, providing a solid foundation for subsequent experimental validation and potential clinical applications.

Methodology

Problem Description

In the context of machine learning, the task of predicting context-specific synthetic lethality (SL) interactions can be formalized as a binary classification problem. Given a pair of genes, the goal is to predict whether the combination of these two genes results in a synthetic lethal interaction under specific biological contexts, such as a particular cell line, genetic background, or environmental condition.

Formally, let $G = \{g_1, g_2, \ldots, g_n\}$ be the set of genes under consideration. The task involves evaluating pairs of genes (g_i, g_j), where $g_i, g_j \in G$ and $i \neq j$. Each gene pair (g_i, g_j) is associated with a feature vector $\mathbf{x}_{ij}$ that encodes relevant biological information, such as gene expression levels, mutation status, pathway involvement, and other omics data.

The SL prediction task can be described as:

$$y_{ij} = f(\mathbf{x}_{ij})$$

where $y_{ij} \in \{0, 1\}$ is the binary label indicating whether the gene pair (g_i, g_j) exhibits synthetic lethality (1) or not (0) in a specific context. The function f represents the machine learning model trained to perform the prediction.

Overview of OmniSynth

OmniSynth leverages a sophisticated architecture to predict cell-line specific synthetic lethality interactions by integrating various types of biological data. The framework consists of several key components, each responsible for processing different types of input data and generating embeddings that are subsequently used for prediction. The overall architecture of OmniSynth is illustrated in Figure 1 and can be described as follows:

ESM Embedding

The process begins with peptide sequences representing the genes of interest. The peptide sequences are fed into an ESM module to generate embeddings. ESM is used to capture the evolutionary and functional properties of proteins based on their sequences. The ESM module produces embeddings for both Gene A and Gene B, which are then passed through fully connected (fc) layers to further refine these embeddings.

HGT Embedding

Nodes in a biomedical knowledge graph, representing various biological entities and their relationships, are used as input. The knowledge graph nodes are processed by the HGT module to generate embeddings. HGT captures complex relationships between different types of biological entities within the graph. Similar to the ESM module, HGT produces embeddings for both Gene A and Gene B, which are also passed through fully connected layers for refinement.

Cell Line Embedding

Data specific to certain cell lines, including expression levels and copy number variations, are used as input. PCA is applied to reduce the high dimensionality of the cell line data, generating embeddings that summarize the key characteristics of the cell lines.

Gene Expression, Mutation, and Copy Number Variation Embedding

We directly concatenate the gene expression, mutation, and copy number variation data to form a comprehensive feature vector that represents the expression level, whether it is a mutation hotspot in the cell line, and the copy number variation data of Gene A and Gene B in a specific cell line context.

Integration and Prediction

1. **Feature Concatenation:** The refined embeddings from the ESM, HGT, and PCA modules are concatenated to form a comprehensive feature vector that represents the interaction between Gene A and Gene B in a specific cell line context.
2. **Fully Connected Layers:** The concatenated feature vector is passed through a series of fully connected (FC) layers. These layers learn complex interactions and dependencies within the integrated feature set.
3. **Prediction Output:** The final FC layer produces a prediction output, indicating the score of whether the gene pair exhibits synthetic lethality (1) or not (0).
4. **Loss Function:** The model is trained using a focal loss function, which is particularly effective in handling class imbalance by focusing more on hard-to-classify examples.

Data Collection

To train and validate the OmniSynth framework, we collected a diverse set of biological data from several well-established databases. This section details the sources and types of data used in our study.

Peptide Sequence Data

Peptide sequences corresponding to the genes of interest were obtained from the UniProt database. UniProt provides a comprehensive, high-quality, and freely accessible resource of protein sequence and functional information. These sequences

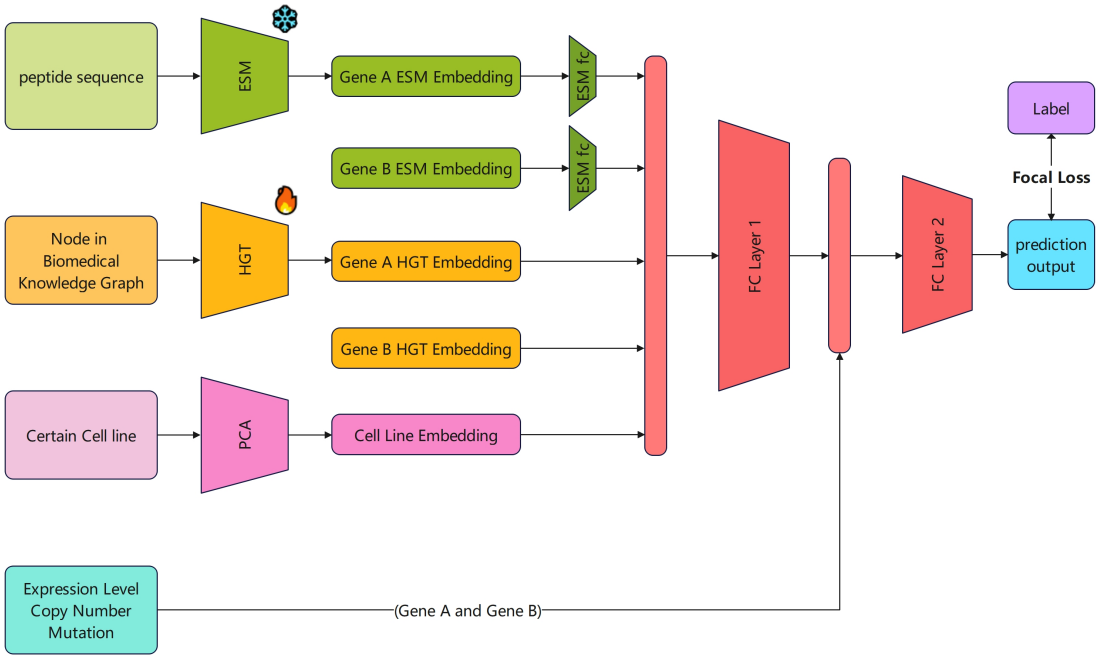

Fig. 1. Illustration of OmniSynth framework.

serve as the input for the ESM module, which generates evolutionary scale modeling embeddings to capture the evolutionary and functional properties of the proteins.

Biomedical Knowledge Graph

Biomedical knowledge graph data was sourced from PrimeKG. PrimeKG is an integrative knowledge graph that includes information on various biological entities and their interactions, encompassing a wide range of biomedical data sources. This resource allows us to leverage complex relationships between genes, proteins, and other biological entities to generate HGT embeddings for the genes of interest.

Cell Line Data

Cell line-specific data, including gene expression levels and copy number variations, were extracted from the Cancer Cell Line Encyclopedia (CCLE). The CCLE project provides detailed genetic, transcriptomic, and epigenetic characterization of a large panel of human cancer cell lines. This rich dataset enables us to generate PCA embeddings that summarize the key characteristics of specific cell lines.

Gene Expression, Mutation, and Copy Number Variation Data

We also utilized data from the CCLE for gene expression, mutation status, and copy number variation. The CCLE provides extensive information on the expression levels of genes across different cell lines, the mutation status of these genes, and their copy number variations. This data was directly concatenated to form comprehensive feature vectors representing the expression level, mutation hotspot status, and copy number variation data of Gene A and Gene B in specific cell line contexts.

Model Selection

ESM Module

For the ESM module, we employed the pretrained ESM2-650M model, which uses evolutionary scale modeling to capture the evolutionary and functional properties of proteins based on their sequences. This model generates high-dimensional embeddings (1280 dimensions) that encapsulate rich information about the protein sequences. To manage the dimensionality and prevent overshadowing other data inputs, we reduced these embeddings to 256 dimensions using a linear layer. This dimensionality reduction helps in maintaining a balance between retaining essential information and ensuring that the integration with other data types remains effective.

HGT Module

The HGT (Heterogeneous Graph Transformer) module was selected to process data from the biomedical knowledge graph. HGT is particularly suitable for this task as it can capture

complex relationships between diverse biological entities within the graph. We utilized LukePi's implementation of HGT, which is pre-trained on tasks such as node degree prediction and edge existence and type prediction. This pre-training helps the model to effectively encode the structural and relational information present in the knowledge graph.

Before feeding the data into the HGT module, we employed the HGTLoader for data sampling. The HGTLoader samples the graph data in a way that allows for efficient mini-batch training, ensuring that the model can handle large-scale graphs without compromising on the richness of the relationships captured. By using this pre-trained model and fine-tuning it on our specific tasks, we ensure that the embeddings generated are well-suited to capture the nuances of the gene interactions within the specific context of different cell lines.

PCA Module

For the PCA module, we processed data specific to various cell lines, including mutation counts and copy number variations. Principal Component Analysis (PCA) was employed to reduce the dimensionality of this data while retaining the most significant features. PCA is particularly advantageous because it minimizes the risk of overfitting and effectively handles missing data by setting them to zero.

Using PCA, we condensed the high-dimensional cell line data into a more manageable form, capturing the primary characteristics that differentiate the cell lines. This dimensionality reduction not only simplifies the subsequent processing steps but also enhances the model's ability to generalize across different cell lines by focusing on the most critical aspects of the data.

Data Preprocessing and Model Training

Data Preprocessing

We selected three cell lines for training: A375, A549, and Jurkat. The raw data for these cell lines is presented in Table 1.

The initial data showed that the Jurkat cell line had a significantly lower number of positive samples, making it difficult for the model to accurately classify positive samples during training. To address this imbalance, we performed resampling to increase the number of positive samples for the Jurkat cell line to let the positive rate similar to others. The resampled data is shown in Table 2.

BCE Loss is commonly used in binary classification problems. It calculates the loss by comparing the predicted probability with the actual class label (0 or 1). The formula for BCE Loss is:

$$\text{BCE Loss} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(p_i) + (1-y_i)\log(1-p_i)\right]$$

where y_i is the actual label, p_i is the predicted probability, and N is the number of samples.

Focal Loss is designed to address the class imbalance problem by down-weighting the loss assigned to well-classified examples. This focuses the model on learning from hard-to-classify examples. The formula for Focal Loss is:

$$\text{Focal Loss} = -\frac{1}{N}\sum_{i=1}^{N}\left[\alpha(1-p_i)^{\gamma} y_i \log(p_i) + (1-\alpha)p_i^{\gamma}(1-y_i)\log(1-p_i)\right]$$

where α is a balancing factor, γ is the focusing parameter, y_i is the actual label, p_i is the predicted probability, and N is the number of samples.

In the testing phase, we compared the performance of the model trained with BCE Loss and Focal Loss to determine the most effective loss function for handling the class imbalance in the dataset.

Data Splitting and Model Training

For model training, we split the data into training, validation, and test sets with the following proportions: 70% for training, 20% for validation, and 10% for testing. We trained the model with one NVIDIA® TESLA® P40 GPU, using the Adam Optimizer. The best model in validation according to AUC was selected for evaluation on the test set.

Model Evaluation

We evaluated the model using the best-performing version based on its performance on the validation set. The evaluation metrics selected for assessing the model's performance include:

- **AUC (Area Under the Curve)**: Measures the ability of the model to distinguish between positive and negative classes.
- **AUPR (Area Under the Precision-Recall Curve)**: Provides insight into the precision and recall trade-offs.
- **F1 Score**: The harmonic mean of precision and recall, providing a balance between the two.
- **BACC (Balanced Accuracy)**: Accounts for both sensitivity and specificity, offering a balanced view of performance across classes.
- **KAPPA**: Measures the agreement between predicted and actual labels, adjusting for chance agreement.

These metrics were chosen to provide a comprehensive evaluation of the model's performance, particularly in the context of imbalanced data. Each metric offers unique insights into different aspects of the model's classification capabilities, ensuring a robust assessment of its effectiveness in predicting synthetic lethality interactions.

Results and Discussion

Ablation Study Results

We conducted ablation experiments to assess the contributions of different components of the OmniSynth framework. The components considered in these experiments included the HGT module, Focal Loss, ESM embeddings, cell line representations, and gene expression data. The performance of the model was evaluated using several metrics: AUC (Area Under the Curve), AUPR (Area Under the Precision-Recall Curve), F1 Score, Balanced Accuracy (BACC), and Kappa statistic (KAPPA). The results for cell lines A375, A549, and Jurkat are presented in Table 3.

Analysis of Results

Impact of Focal Loss

From the results, we observe that the inclusion of Focal Loss generally improves the F1 score and Kappa statistic across the board, especially in the imbalanced Jurkat cell line, indicating better handling of the class imbalance. However, in the A375 and A549 cell lines, the improvement is less significant, possibly

Table 1. Raw Data for Cell Lines

Cell Line	Pair Number	Positive	Negative	Positive Rate	Gene Number
A375	515	22	493	0.0411	396
A549	4346	179	4167	0.0411	755
Jurkat	57291	295	56996	0.0051	339

Table 2. Resampled Data for Cell Lines

Cell Line	Pair Number	Positive	Negative	Positive Rate	Gene Number
A375	515	22	493	0.0411	396
A549	4346	179	4167	0.0411	755
Jurkat	6628	295	6333	0.0445	-

due to differences in the distribution of positive data across cell lines.

Contribution of ESM Embeddings

The addition of ESM embeddings significantly enhances the model's AUC and AUPR scores across all cell lines. For instance, in the A549 cell line, the AUC increased from 0.929 to 0.946 when ESM embeddings were included alongside HGT and Focal Loss. This demonstrates the importance of incorporating evolutionary scale modeling to capture the functional properties of proteins and the hidden factors causing SL among proteins.

Role of Cell Line Representations and Gene Expression Data

Adding the cell-line representation alone may cause a side-effect on all metrics, potentially due to the distant relationship between cell-line data and other data types, introducing noise and reducing performance. However, combining cell line-specific representations and gene expression, mutation, and copy number data shows some impact in the Jurkat cell line, improving the F1 score from 0.214 to 0.303 and Kappa from 0.195 to 0.282, although with slight decreases in AUC and AUPR, suggesting a trade-off between different aspects of model performance.

For A375 and A549, including these two pieces of information seems less beneficial, possibly because the HGT for KnowledgeGraph and ESM for protein already provided sufficient information.

Conclusion

OmniSynth is a robust framework for predicting cell-line specific synthetic lethality interactions by integrating multiple omics data sources. It leverages data from biomedical knowledge graphs, protein sequences, gene expression, mutations, and other bioinformatics data to provide accurate and reliable predictions.

Our validation in three cell lines: A375, A549, and Jurkat, demonstrated OmniSynth's high accuracy and generalization ability. The ablation studies highlighted the significance of integrating various data components, particularly the inclusion of Focal Loss, ESM embeddings, and cell line-specific representations, which improved the model's performance.

In summary, OmniSynth advances synthetic lethality prediction by effectively handling complex biological data, aiding the development of targeted cancer therapies, and contributing to personalized medicine and precision oncology.

Future Work

To further validate the performance of OmniSynth and explore its potential enhancements, several directions for future work are proposed.

First, we will conduct extensive validation of cell-line adaptation. This involves testing OmniSynth on a broader range of cell lines to assess its generalization capability. By examining the model's performance across diverse biological contexts, we can ensure its robustness and applicability to various cell lines.

Second, a comprehensive comparative analysis with other existing synthetic lethality prediction methods is necessary. By comparing OmniSynth with alternative approaches, we can highlight its advantages and identify any limitations. This comparative study will provide valuable insights into the relative performance of Omni Synth.

Third, we plan to integrate additional data modalities into OmniSynth. Incorporating diverse data types, such as gene textual descriptions encoded with models like BERT or GPT, will enhance the richness of the input features. Furthermore, we aim to use advanced models like GeneFormer for more sophisticated gene feature extraction, thereby improving the overall predictive accuracy.

Improving model efficiency is another crucial area of future work. The current performance of OmniSynth is constrained by the HGT sampler. To address this, we will explore more efficient sampling methods to enhance training speed and overall model performance. This optimization will enable more effective handling of large-scale data.

Additionally, we will focus on optimizing the model structure to support distributed training. The current architecture of OmniSynth does not facilitate data parallelism for distributed training. By modifying the model to accommodate distributed training, we can achieve better scalability and efficiency, especially when working with extensive datasets.

Lastly, we propose conducting extended ablation studies to evaluate the importance of each data modality. These additional experiments will provide deeper insights into the contributions of different data types to the overall model performance. By systematically examining the impact of various components, we can further refine and enhance OmniSynth.

Table 3. Ablation Study Results

Cell Line	HGT	Focal Loss	ESM	Cell Line Represent	Gene Expression	AUC	AUPR	F1	BACC	KAPPA
A375	✓	✗	✗	✗	✗	0.918	0.705	0.538	0.832	0.523
A375	✓	✓	✗	✗	✗	0.883	0.680	**0.554**	0.761	**0.542**
A375	✓	✓	✓	✗	✗	0.928	0.726	0.397	0.715	0.383
A375	✓	✓	✓	✓	✗	0.925	0.597	0.304	0.677	0.294
A375	✓	✓	✓	✓	✓	**0.955**	0.715	0.386	0.679	0.373
A549	✓	✗	✗	✗	✗	0.923	0.410	0.295	0.603	0.276
A549	✓	✓	✗	✗	✗	0.929	0.445	0.309	0.616	0.289
A549	✓	✓	✓	✗	✗	**0.946**	0.473	0.399	0.674	0.378
A549	✓	✓	✓	✓	✗	0.940	**0.492**	**0.453**	**0.694**	**0.432**
A549	✓	✓	✓	✓	✓	0.943	0.465	0.339	0.631	0.319
Jurkat	✓	✗	✗	✗	✗	0.796	0.234	0.013	0.503	0.012
Jurkat	✓	✓	✗	✗	✗	0.847	0.272	0.105	0.528	0.141
Jurkat	✓	✓	✓	✗	✗	0.870	0.340	0.255	0.594	0.238
Jurkat	✓	✓	✓	✓	✗	**0.881**	**0.384**	0.214	0.604	0.195
Jurkat	✓	✓	✓	✓	✓	0.878	0.368	**0.303**	**0.617**	**0.282**

References

1. The UniProt Consortium, "UniProt: the universal protein knowledgebase," Nucleic Acids Research, Volume 45, Issue D1: D158-D169, 2017.
2. Franklin W Huang et al., "Next-generation characterization of the Cancer Cell Line Encyclopedia," Nature, Volume 569, Number 7757: 503-508, 2019.
3. Shike Wang et al., "KG4SL: knowledge graph neural network for synthetic lethality prediction in human cancers," Bioinformatics, Year 2021.
4. Zeming Lin, Halil Akin, Roshan Rao, et al., "Language models of protein sequences at the scale of evolution enable accurate structure prediction," bioRxiv, Year 2022.
5. Payal Chandak et al., "Building a knowledge graph to enable precision medicine," Scientific Data, Year 2023, Volume(issue): page numbers, doi: 10.1038/s41597-023-01960-3.

Author contributions statement

Mutian Hong completed most of the work in this paper. This includes conceiving the experiment, conducting the experiment, analyzing the results, and writing and reviewing the manuscript. Specifically, Mutian Hong was responsible for:

- Designing and developing the OmniSynth framework
- Collecting and preprocessing the multi-omics data
- Analyzing and visulizing the data
- Implementing and training the machine learning models
- Conducting validation and testing on the A375, A549, and Jurkat cell lines
- Performing the ablation studies and evaluating the model's performance
- Writing the manuscript, including the abstract, methodology, results, discussion, and conclusion sections
- Preparing figures and tables for the paper
- Preparing the slides for the presentation
- Giving the presentation
- Reviewing and finalizing the manuscript for submission

Tang Zhihao tried to conceiving the experiment, and helped to examine the paper.

Siyu Tao guided the experiment and provided significant contributions to the improvement of the model scheme. Specifically, Siyu Tao:

- Provided technical assistance in the design and implementation of the experiments
- Assisted in troubleshooting and optimizing the machine learning models
- Contributed to the interpretation of the experimental results and their implications

Acknowledgments

This work received guidance from senior student Siyu Tao, senior student Ruoyin Zhang, and Professor Jie Zheng. Parts of the manuscript writing and code implementation were assisted by OpenAI's GPT-4o. Additionally, some parts of the code were written with the assistance of GitHub Copilot.