A Annotation Guidelines

This section describes the manual annotation protocol used to label drug-drug interaction (DDI) pairs in the MUDI dataset. Our objective is to create a standardized, clinically meaningful categorization of pharmacodynamic interactions into three classes: `Synergism`, `Antagonism`, and `New Effect`.

A.1 Drug Name Masking Policy

To ensure consistent pattern recognition and reduce annotator bias toward specific drug identities, all drug mentions in the original DrugBank interaction descriptions are replaced with abstract placeholders before annotation. This masking step is essential for allowing models and human annotators to focus on the nature of the pharmacodynamic interaction rather than the specific lexical forms of drug names.

Standardized Placeholders. The two interacting drugs in each sentence are masked using `[DRUG1]` and `[DRUG2]`. Any additional drug names appearing in the same sentence are replaced with `[DRUGOTHER]`. This abstraction is applied to both brand names and generic names, as well as chemical synonyms.

Drug Mention Alignment. Since DrugBank does not provide token-level alignment between drug entities and their textual positions, we employ a flexible string-matching algorithm to detect mentions of each drug and its synonyms. This process uses:

- Canonical names and aliases from DrugBank metadata.
- Case-insensitive matching.
- Partial overlap resolution (to disambiguate e.g., “Promazine” vs. “Acepromazine”).

Manual Review and Correction. In some cases, automatic matching resulted in incomplete or incorrect spans – especially when drugs shared lexical substrings or when spacing/punctuation was irregular. To address this, a team of three linguistics students manually reviewed and corrected all unique masked sentence templates. The original corpus of 244,921 raw DDI descriptions was thereby reduced to 287 distinct masked templates, from which 241 final sentence types were retained after quality filtering. Inter-annotator agreement during this manual review phase was near-perfect, with disagreements resolved by discussion and unanimous consensus.

Impact on Annotation Quality. This masking strategy eliminates the risk of model or annotator bias due to prior familiarity with drug names or brand-specific expectations. By enforcing a uniform abstracted representation across the dataset, we enable more consistent labeling of pharmacodynamic effects and allow models to generalize beyond known drug pairs.

A.2 Labeling Rules

Interaction descriptions in MUDI are categorized into one of three pharmacodynamic classes based on a set of carefully constructed lexical heuristics. These heuristics are grounded in recurring sentence structures observed in DrugBank and designed to reflect pharmacological theory [8, 23].

Synergism. Labeled when `[DRUG1]` enhances the pharmacological effect, bioavailability, or systemic concentration of `[DRUG2]`. Typical patterns include increased absorption, inhibited excretion, or elevated therapeutic efficacy.

Rule Templates for `Synergism`:

- `[DRUG1]` can cause an increase in the absorption of `[DRUG2]` resulting in an increased serum concentration and potentially a worsening of adverse effects.
- `[DRUG1]` may decrease the excretion rate of `[DRUG2]` which could result in a higher serum level.
- `[DRUG1]` may increase the `[activities names]` activities of `[DRUG2]`.
- The bioavailability of `[DRUG1]` can be increased when combined with `[DRUG2]`.
- The excretion of `[DRUG1]` can be decreased when combined with `[DRUG2]`.
- The metabolism of `[DRUG1]` can be decreased when combined with `[DRUG2]`.
- The protein binding of `[DRUG1]` can be decreased when combined with `[DRUG2]`.
- The serum concentration of `[DRUG1]` can be increased when it is combined with `[DRUG2]`.
- The serum concentration of `[metabolite name]`, an active metabolite of `[DRUG1]`, can be increased when used in combination with `[DRUG2]`.
- The therapeutic efficacy of `[DRUG1]` can be increased when used in combination with `[DRUG2]`.

Antagonism. An interaction is labeled as `Antagonism` if the description suggests that `[DRUG1]` inhibits, reduces, or interferes with the pharmacodynamic action of `[DRUG2]`, including diminished absorption, faster metabolism, or decreased efficacy.

Rule Templates for `Antagonism`:

- `[DRUG1]` can cause a decrease in the absorption of `[DRUG2]` resulting in a reduced serum concentration and potentially a decrease in efficacy.
- `[DRUG1]` may decrease effectiveness of `[DRUG2]` as a diagnostic agent.
- `[DRUG1]` may decrease the `[activities names]` activities of `[DRUG2]`.
- `[DRUG1]` may increase the excretion rate of `[DRUG2]` which could result in a lower serum level and potentially a reduction in efficacy.
- The absorption of `[DRUG1]` can be decreased when combined with `[DRUG2]`.
- The bioavailability of `[DRUG1]` can be decreased when combined with `[DRUG2]`.
- The excretion of `[DRUG1]` can be increased when combined with `[DRUG2]`.
- The metabolism of `[DRUG1]` can be increased when combined with `[DRUG2]`.
- The risk or severity of `[adverse effects]` can be decreased when `[DRUG1]` is combined with `[DRUG2]`.
- The serum concentration of `[DRUG1]` can be decreased when it is combined with `[DRUG2]`.
- The serum concentration of `[metabolite name]`, an active metabolite of `[DRUG1]`, can be decreased when used in combination with `[DRUG2]`.

- The therapeutic efficacy of [DRUG1] can be decreased when used in combination with [DRUG2].

New Effect. An interaction is labeled as `New Effect` when the interaction leads to a novel adverse effect not known to be associated with either [DRUG1] or [DRUG2] independently. Initially, an interaction assigned the `New Effect` label when the sentence contains biomedical event terms (typically adverse effects) that cannot be clearly attributed to either `Synergism` or `Antagonism` patterns. These are typically masked sentences that do not indicate enhancement or suppression but instead describe the emergence of a distinct pharmacological effect.

To validate whether the reported effect is indeed novel to the drug combination, we perform a comparison between the extracted biomedical term and the known side effect profiles of both drugs. This process includes:

(1) Automatically identifying the biomedical effect term in the masked sentence.
(2) Matching the term against the adverse event metadata of each drug using stemming and synonym expansion [18, 22].
(3) Assigning the `New Effect` label only if the effect is absent from both drug profiles.
(4) Reassigning the sentence to `Synergism` if the effect is already associated with one of the drugs.

All candidate `New Effect` annotations are subsequently reviewed by domain experts to ensure biomedical validity. This step ensures that no pharmacologically implausible or redundant labels remain in the final dataset.

No or Unclear Interaction. Interaction descriptions that do not exhibit clear pharmacodynamic effects – such as therapeutic enhancement, attenuation, or novel adverse outcomes – are not assigned a positive label. This includes cases with vague, incomplete, or purely pharmacokinetic information lacking clinical consequence. In line with established pharmacological theory [8, 23], such drug pairs are considered to exhibit *no or unclear interaction*, indicating insufficient evidence to support classification into one of the defined pharmacodynamic categories.

A.3 Annotation Quality Control

To ensure the biomedical validity and consistency of interaction labels in MUDI, we implemented a two-phase expert curation protocol. After automated annotation using lexical heuristics (Section 3.2), each of the 241 distinct masked interaction templates was reviewed by two physicians with domain expertise in clinical pharmacology and drug safety.

Each expert independently examined the interaction context, checked consistency with known drug properties, and validated the correctness of the assigned label with respect to both pharmacodynamic semantics and biomedical relevance. In cases where the experts initially disagreed, they conducted a focused discussion to reach a consensus. This adjudication phase ensured the removal of residual annotation noise introduced by the automated pipeline, particularly in borderline or semantically ambiguous cases.

This dual-review and consensus-based process ensures that MUDI maintains clinically credible labels and supports reliable downstream model development.

B Dataset Access and Organization

This appendix provides practical information for obtaining and using the MUDI dataset, including access instructions, licensing terms, and directory structure.

B.1 Access and Licensing

The MUDI dataset is derived from DrugBank, a publicly accessible biomedical database. According to DrugBank's data usage policy[3], academic and non-commercial use of DrugBank content is permitted under its custom license for research purposes. In full compliance with this policy, MUDI builds on openly accessible DrugBank fields and restricts redistribution to non-commercial academic use only.

We release the MUDI dataset under the Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0)[4]. This license enables researchers to copy, distribute, and adapt the dataset for academic purposes, provided that:

- Proper credit is given to both the MUDI project and the original DrugBank resource.
- Any derived works or models are clearly marked as adaptations.
- No commercial use is made without explicit written permission.

Users may download the dataset, documentation, preprocessing scripts, and baseline code from the following links:

- **Zenodo**: https://zenodo.org/records/15544551 – the dataset archive with DOI for stable citation.
- **GitHub**: https://github.com/hoangbros03/MUDI – the codebase repository for preprocessing, baseline models, and future updates.

A permanent DOI link[5] is assigned via Zenodo to ensure stable referencing and citation.

By downloading and using MUDI, users agree to comply with the license terms and responsible usage guidelines outlined in Appendix C.

B.2 Dataset Structure

The MUDI dataset is organized into a clear and modular directory layout to facilitate ease of use, reproducibility, and multimodal experimentation:

[3] https://go.drugbank.com/legal
[4] https://creativecommons.org/licenses/by-nc/4.0/
[5] https://doi.org/10.5281/zenodo.15544551

```
MUDI/
├── dataset/
│   ├── train.csv
│   └── test.csv
├── molecules/
│   ├── images/
│   │   └── [drug_id].png
│   └── graphs/
│       └── [drug_id].graphml
├── preprocess/
│   ├── formula_handler.py
│   ├── graph_handler.py
│   ├── image_handler.py
│   └── text_handler.py
├── drug_info.json
├── README.md
└── LICENSE.txt
```

dataset/train.csv *and* test.csv. Each row in these CSV files corresponds to a labeled drug-drug interaction (DDI) instance and contains the following fields:

- DRUG1: Unique identifier for the first drug.
- Interaction: One of the pharmacodynamic classes (Synergism, Antagonism, or New Effect).
- DRUG2: Unique identifier for the second drug.

All drug identifiers are keys in drug_info.json for retrieving textual, structural, and molecular representations.

molecules/images/. Each PNG image represents the 2D chemical structure of a drug generated from its SMILES string using RDKit. All images are:

- Named as [drug_id].png.
- Stored at a standardized resolution of 1000×800 pixels.
- Ready for direct use with image encoders such as Vision Transformer.

molecules/graphs/. Each file is a GraphML-encoded molecular graph where:

- Nodes represent atoms.
- Edges represent bonds (e.g., single, double, aromatic).

The files follow the standard GraphML format, with recommended compatibility via the NetworkX Python library.

drug_info.json. This JSON file consolidates metadata for each drug. Each entry contains:

- name: Human-readable drug name, e.g., "Amitriptyline".
- description: A dictionary containing pharmacological text fields used for textual modeling:
  - summary: A concise overview of the drug's identity, primary use, and general characteristics.
  - indication: Approved medical conditions or diseases that the drug is prescribed to treat.
  - metabolism: Description of the drug's metabolic pathway, including hepatic enzymes involved (e.g., CYP450 family).
  - pharmacodynamics: Explanation of the biological effects, mechanism of drug action, and dose-response relationships.
  - moa (mechanism of action): Detailed molecular-level explanation of how the drug achieves its intended effect, such as receptor binding or enzyme inhibition.
- formula: The molecular formula representing the elemental composition of the drug (e.g., C20H25N3O).
- smiles: The canonical SMILES (Simplified Molecular Input Line Entry System) string encoding the molecular structure in a compact, text-based format.

These fields are used to build modality-specific representations for textual, formula-based, and structural modeling.

B.3 File Standards and Preprocessing Code

File Formats.

- All interaction data is UTF-8 encoded CSV.
- Molecular graphs follow the GraphML standard.
- Chemical structure diagrams are stored as PNG images.
- Metadata is provided as structured JSON.

Preprocessing Scripts. The preprocess/ directory includes modular Python scripts for converting raw inputs into model-ready formats:

- text_handler.py: Create a JSON object holding textual information, including name, description, SMILES, and formula.
- formula_handler.py: Improve the representation of molecule elements within chemical formulas before they are passed to the text handler.
- image_handler.py: Load the images from the dataset and convert them into tensors.
- graph_handler.py: Load and create graph objects.

Key dependencies include RDKit, networkx, and transformers.

B.4 Getting Started

To quickly begin using MUDI:

- Refer to README.md for installation, tutorials, and citation information.
- Use the drug_info.json file to retrieve all relevant metadata for each drug.
- Apply the preprocessing scripts to regenerate modality-specific features as needed.
- Evaluate models using the provided train/test splits and compute metrics such as precision, recall, micro-F1, and macro-F1.

The provided setup is fully reproducible and extensible for future multimodal biomedical research.

C Responsible Usage and Ethical Guidelines

This appendix details the ethical foundations, responsible usage requirements, and recommended best practices for working with the MUDI dataset. These principles aim to promote transparency, safety, and compliance in biomedical AI research.

C.1 Data Provenance and Privacy

The MUDI dataset is built entirely from non-sensitive, publicly accessible data obtained from the DrugBank database [30], a reputable biomedical resource. All data sources are governed by DrugBank's academic use policy[6], which permits reuse for non-commercial research.

Importantly, MUDI does not contain any protected health information (PHI), patient-level records, or personally identifiable information (PII). No data originates from clinical trials, electronic medical records, or real-world hospital systems. As such, the dataset does not fall under the scope of human subjects research and does not require ethical approval from an institutional review board (IRB). It is also exempt from compliance obligations under HIPAA, GDPR, or related data privacy regulations.

C.2 Intended Use

MUDI is intended exclusively for academic research, education, and non-commercial purposes. Acceptable use cases include, but are not limited to:

- Development, benchmarking, and publication of multimodal learning algorithms for biomedical knowledge discovery.
- Research in drug-drug interaction (DDI) prediction, representation learning, cross-modal retrieval, and zero-shot biomedical reasoning.
- Classroom use in university-level courses or technical workshops on machine learning, drug discovery, or bioinformatics.

Any commercial use of the dataset is prohibited under the terms of the CC BY-NC 4.0 license without explicit written permission from the authors.

C.3 Known Limitations and Usage Caveats

Despite thorough curation and validation, MUDI remains a research-focused dataset and carries certain limitations:

- **No Clinical Validation:** Pharmacodynamic interaction labels are generated through lexical rules and expert curation, but not independently verified in wet-lab or clinical settings. The dataset is not intended for clinical use or decision support.
- **Potential Label Ambiguity:** The source descriptions from DrugBank are natural language statements, which may contain implicit or ambiguous interaction signals. While the annotation pipeline includes validation steps, some residual label noise is inevitable.
- **Pharmacodynamic Scope Only:** MUDI exclusively targets pharmacodynamic interactions. It does not cover pharmacokinetic DDIs such as those involving absorption, distribution, metabolism, or excretion (ADME) pathways.
- **Bias Toward Common Drugs:** Interaction labels are inherently more complete for well-studied drugs. This may bias model performance toward drugs with richer metadata and documented histories.

Researchers should exercise caution when interpreting results for clinical decision support or downstream biomedical applications.

C.4 Responsible Research Practices

We encourage users of MUDI to adopt the following practices to uphold ethical standards and maximize the scientific value of their work:

- Clearly cite the MUDI dataset and its associated publication in all derivative research.
- Disclose all modeling assumptions, training data subsets, and evaluation procedures to support reproducibility.
- Publicly release code and model checkpoints when possible, subject to the same licensing terms.
- Transparently communicate dataset limitations, especially when proposing real-world or clinical applications.
- Avoid deploying or advertising models trained on MUDI for direct use in patient care without formal clinical validation and regulatory approval.

D Baseline Model Configurations

This appendix provides detailed descriptions of the baseline models used to benchmark the MUDI dataset, including architecture choices, modality-specific preprocessing, and mathematical formulations.

D.1 Single-Modality Baselines

D.1.1 Text-only Baseline (BioMedBERT).

The text-only model uses BioMedBERT [10] to encode concatenated pharmacological fields: summary, indication, mechanism of action, pharmacodynamics, and metabolism. The fields are joined into a single input sequence, separated by special tokens.

Given an input sequence $\mathbf{x}^{\text{text}}$, the model computes hidden representations $\mathbf{h}_i$ for each token:

$$\mathbf{h}_i = \mathcal{E}_{\text{BioMedBERT}}(\mathbf{x}^{\text{text}})_i,$$

where i indexes the tokens.

We extract the embedding corresponding to the `[CLS]` token, $\mathbf{h}_{\texttt{[CLS]}}$, and apply a linear classification layer:

$$\hat{\mathbf{y}}_{\text{text}} = \text{softmax}(\mathbf{h}_{\texttt{[CLS]}}\mathbf{W}_t + \mathbf{b}_t),$$

where $\mathbf{W}_t \in \mathbb{R}^{d\times C}$ and $\mathbf{b}_t \in \mathbb{R}^C$ are learnable parameters, d is the embedding dimension, C is the number of output classes, and $\hat{\mathbf{y}}_{\text{text}} \in \mathbb{R}^C$ is the predicted class distribution.

D.1.2 Graph-only Baseline (GCN).

The graph-only model uses a two-layer Graph Convolutional Network (GCN) [16] to process the molecular structure graphs generated from SMILES strings.

Each molecule graph is represented as an adjacency matrix $\mathbf{A}$ and a feature matrix $\mathbf{X}$ containing atom features. The GCN updates node features as:

$$\mathbf{H}^{(1)} = \sigma\left(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{X}\mathbf{W}^{(0)}\right),$$
$$\mathbf{H}^{(2)} = \sigma\left(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(1)}\mathbf{W}^{(1)}\right),$$

where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ is the adjacency matrix with self-loops added, $\tilde{\mathbf{D}}$ is the corresponding degree matrix, and σ denotes the ReLU activation function. The learnable parameters $\mathbf{W}^{(0)} \in \mathbb{R}^{d_{\text{in}}\times d_{\text{hidden}}}$ and $\mathbf{W}^{(1)} \in \mathbb{R}^{d_{\text{hidden}}\times d_{\text{out}}}$ are weight matrices for the first and second GCN layers, respectively, where d_{in} is the input node feature dimension, d_{hidden} is the hidden dimension, and d_{out} is the output node feature dimension.

After the second GCN layer, we obtain node-level embeddings $\mathbf{H}^{(2)} = [\mathbf{h}_1^{(2)}, \mathbf{h}_2^{(2)}, \dots, \mathbf{h}_n^{(2)}]$, where $\mathbf{h}_i^{(2)} \in \mathbb{R}^d$ is the feature vector

[6] https://go.drugbank.com/legal

of the i-th node and n is the number of nodes in the molecular graph.

We apply global max pooling across nodes to produce a graph-level embedding:

$$\mathbf{z}_{\text{graph}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{h}_i^{(2)}.$$

The pooled graph representation $\mathbf{z}_{\text{graph}}$ is then passed through a linear classifier:

$$\hat{\mathbf{y}}_{\text{graph}} = \text{softmax}(\mathbf{z}_{\text{graph}}\mathbf{W}_g + \mathbf{b}_g),$$

where $\mathbf{W}_g \in \mathbb{R}^{d\times C}$ and $\mathbf{b}_g \in \mathbb{R}^C$ are the classification weights and bias, and C is the number of interaction classes.

D.1.3 Image-only Baseline (ViT). The image-only model employs a Vision Transformer (ViT) [7] to encode 2D chemical structure images.

Given an input image $\mathbf{x}^{\text{img}} \in \mathbb{R}^{H\times W\times 3}$, where H and W denote the height and width, the image is divided into N non-overlapping patches, each of size $P \times P$ pixels.

Each patch is flattened into a vector and linearly projected into a d-dimensional embedding space via a learnable matrix $\mathbf{W}_p \in \mathbb{R}^{(P^2\times 3)\times d}$:

$$\mathbf{z}_i = \mathbf{x}_i^{\text{patch}}\mathbf{W}_p + \mathbf{b}_p, \quad \forall i = 1, \dots, N$$

where $\mathbf{x}_i^{\text{patch}}$ is the flattened pixel vector of the i-th patch.

The sequence of patch embeddings $\{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_N\}$ is prepended with a learnable `[CLS]` token embedding $\mathbf{z}_{\texttt{[CLS]}} \in \mathbb{R}^d$, and positional encodings are added to preserve spatial information.

The resulting sequence is input into a standard Transformer encoder:

$$\mathbf{H} = \mathcal{E}_{\text{ViT}}\left(\left[\mathbf{z}_{\texttt{[CLS]}}, \mathbf{z}_1, \dots, \mathbf{z}_N\right]\right),$$

where $\mathcal{E}_{\text{ViT}}$ denotes the stack of transformer layers.

We extract the output corresponding to the `[CLS]` token, denoted as $\mathbf{h}_{\texttt{[CLS]}} \in \mathbb{R}^d$, and apply a linear classifier:

$$\hat{\mathbf{y}}_{\text{image}} = \text{softmax}(\mathbf{h}_{\texttt{[CLS]}}\mathbf{W}_v + \mathbf{b}_v),$$

where $\mathbf{W}_v \in \mathbb{R}^{d\times C}$ and $\mathbf{b}_v \in \mathbb{R}^C$ are learnable parameters, d is the hidden dimension, and C is the number of output classes.

D.2 Multimodal Baselines

D.2.1 Late Fusion Baseline. The late fusion baseline combines predictions from six independent single-modality classifiers, each trained on a distinct representation of drug information. Let $\mathcal{M}$ denote the set of modalities:

$$\mathcal{M} = \{\text{name}, \text{description}, \text{SMILES}, \text{formula}, \text{graph}, \text{image}\}.$$

Each modality is processed by a dedicated model:

- **Name, Description, SMILES, Formula:** Each field is input into a separate BioMedBERT encoder to produce four independent textual predictions. For `formula`, the chemical formula is translated into a sequence of full element names (e.g., `C20H25N3O` → `carbon 20 hydrogen 25 nitrogen 3 oxygen`) to align with the input of language models.
- **Graph:** The molecular structure graph is encoded using a two-layer GCN.
- **Image:** The 2D chemical structure is processed by a Vision Transformer.

Given a drug pair, each model $m \in \mathcal{M}$ produces a predicted label $\hat{y}_m \in C$, where $C = \{\texttt{Synergism}, \texttt{Antagonism}, \texttt{New Effect}\}$ is the set of pharmacodynamic interaction classes.

The final prediction $\hat{y}_{\text{late}}$ is obtained through majority voting across modalities:

$$\hat{y}_{\text{late}} = \arg\max_{c\in C} \sum_{m\in\mathcal{M}} \mathbb{I}(\hat{y}_m = c),$$

where $\mathbb{I}(\cdot)$ is the indicator function.

In the case of a tie (i.e., multiple classes receiving equal votes), we apply a deterministic rule that prioritizes modalities based on their average F1 performance on the MUDI dataset, in the following order: `graph` → `name` → `image` → `SMILES` → `formula` → `description`. This ordering is based on the empirical performance of the individual models on the development set.

D.2.2 Intermediate Fusion Baseline. The intermediate fusion baseline constructs a joint representation by integrating six modality-specific embeddings for each drug in the input pair. For a given drug pair (d_1, d_2), we extract embeddings from the following modalities: name, description, SMILES (text), formula, molecular graph, and chemical image.

Each modality-specific encoder independently processes both drugs:

$$\begin{aligned} \mathbf{z}_m^{(1)} &= \mathcal{E}_m(d_1), \\ \mathbf{z}_m^{(2)} &= \mathcal{E}_m(d_2), \end{aligned} \quad \text{for each modality } m \in \mathcal{M},$$

where $\mathcal{M}$ is the set of modalities, and $\mathbf{z}_m^{(i)} \in \mathbb{R}^{d_m}$ denotes the embedding of drug d_i in modality m.

We concatenate the two drug embeddings for each modality:

$$\tilde{\mathbf{z}}_m = \left[\mathbf{z}_m^{(1)};\ \mathbf{z}_m^{(2)}\right] \in \mathbb{R}^{2d_m},$$

and subsequently form the full multimodal representation by concatenating across all modalities:

$$\mathbf{z}_{\text{fused}} = \left[\tilde{\mathbf{z}}_{\text{name}};\ \tilde{\mathbf{z}}_{\text{desc}};\ \tilde{\mathbf{z}}_{\text{smiles}};\ \tilde{\mathbf{z}}_{\text{formula}};\ \tilde{\mathbf{z}}_{\text{graph}};\ \tilde{\mathbf{z}}_{\text{image}}\right] \in \mathbb{R}^{d_{\text{fused}}},$$

where $d_{\text{fused}} = 2(d_n + d_d + d_s + d_f + d_g + d_i)$.

This joint embedding is passed through a two-layer multilayer perceptron (MLP) with ReLU activation:

$$\begin{aligned} \mathbf{h}_1 &= \sigma(\mathbf{z}_{\text{fused}}\mathbf{W}_1 + \mathbf{b}_1), \\ \hat{\mathbf{y}}_{\text{inter}} &= \text{softmax}(\mathbf{h}_1\mathbf{W}_2 + \mathbf{b}_2), \end{aligned}$$

where:

- $\mathbf{W}_1 \in \mathbb{R}^{d_{\text{fused}}\times d_{\text{hidden}}}$ and $\mathbf{b}_1 \in \mathbb{R}^{d_{\text{hidden}}}$ are parameters of the first MLP layer,
- $\mathbf{W}_2 \in \mathbb{R}^{d_{\text{hidden}}\times C}$ and $\mathbf{b}_2 \in \mathbb{R}^C$ are parameters of the classification head,
- C is the number of pharmacodynamic interaction classes.

This fusion strategy enables the model to learn pairwise dependencies and cross-modal interactions between drugs in a unified and expressive representation space.

D.3 Classification Task Definition

The central task in MUDI is formulated as a multi-class classification problem over pharmacodynamic drug-drug interactions. Given a pair of drugs (d_1, d_2), the goal is to predict a single interaction label $y \in C$ based on their multimodal features. The label set C includes four possible classes:

- `Synergism` – drug d_1 enhances the effect of d_2.
- `Antagonism` – drug d_1 reduces or nullifies the effect of d_2.
- `New Effect` – the combination produces a novel outcome not present in individual use.
- `No Interaction` — no significant pharmacodynamic interaction is known or observed.

Each sample is annotated with one of the four mutually exclusive labels, with directional semantics included for `Synergism` and `Antagonism`. Specifically, (d_1, d_2) and (d_2, d_1) may correspond to different labels or directions unless symmetry is explicitly annotated (e.g., in `New Effect` cases).

To align with clinical interest in detecting meaningful drug interactions, our evaluation protocol concentrates on the three **positive interaction classes** – `Synergism`, `Antagonism`, and `New Effect`. The `No Interaction` label is used during training to simulate realistic class imbalance and improve discrimination, but is excluded from performance metric computation during test-time evaluation, following best practices in biomedical literature [21].

D.4 Prediction Thresholds

All models produce a probability distribution over the four interaction classes via a softmax output layer. Final class predictions are made using a maximum likelihood decision rule:

$$\hat{y} = \arg\max_{c \in C} \; p(c \mid \mathbf{x}),$$

where $C = \{$`Synergism`, `Antagonism`, `New Effect`, `No Interaction`$\}$, and $p(c \mid \mathbf{x})$ is the predicted probability for class c given multimodal input $\mathbf{x}$.

No additional confidence thresholding is applied. This choice ensures fair and consistent comparison across models, particularly under class imbalance conditions.

D.5 Reproducibility

To promote transparency and facilitate fair comparison, we standardize all experimental procedures as follows:

- **Randomness control:** All experiments are conducted with fixed random seeds for `PyTorch`, `NumPy`, and system-level generators to ensure consistent results across runs.
- **Dataset splits:** We use the same predefined training and test sets for all baseline models and fusion strategies.
- **Evaluation consistency:** All models are evaluated using a unified set of metrics and evaluation scripts, ensuring consistent treatment of prediction outputs under both direction-aware and direction-agnostic settings.

The full evaluation pipeline, including scoring functions and matching logic, is publicly available in the official repository (see Appendix B). This setup enables full replication of our results and supports future benchmarking efforts on the MUDI dataset.

E Evaluation Protocols and Settings

E.1 Evaluation Metrics

To evaluate model performance on clinically meaningful interaction types, we report standard classification metrics computed over the three positive classes: `Synergism`, `Antagonism`, and `New Effect`.

Precision (P). For each class, Precision is the proportion of correctly predicted instances among all instances assigned to that class:

$$\text{Precision}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c},$$

where TP_c and FP_c denote true positives and false positives for class c.

Recall (R). Recall is the proportion of correctly predicted instances among all actual instances of the class:

$$\text{Recall}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c},$$

where FN_c is the number of false negatives for class c.

F1 Score (F1). The F1 score is the harmonic mean of Precision and Recall:

$$\text{F1}_c = \frac{2 \cdot \text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c}.$$

Micro-averaged Metrics. Micro-averaging aggregates true positives, false positives, and false negatives across all positive classes before computing Precision, Recall, and F1:

$$\text{Precision}_{\text{micro}} = \frac{\sum_c \text{TP}_c}{\sum_c (\text{TP}_c + \text{FP}_c)}, \quad \text{Recall}_{\text{micro}} = \frac{\sum_c \text{TP}_c}{\sum_c (\text{TP}_c + \text{FN}_c)},$$

$$\text{Micro-F1} = \frac{2 \cdot \text{Precision}_{\text{micro}} \cdot \text{Recall}_{\text{micro}}}{\text{Precision}_{\text{micro}} + \text{Recall}_{\text{micro}}}.$$

Macro-averaged Metrics. Macro-averaging computes the unweighted mean of the per-class metrics. We first compute macro-averaged Precision and Recall:

$$\text{Precision}_{\text{macro}} = \frac{1}{3} \sum_{c=1}^{3} \text{Precision}_c, \quad \text{Recall}_{\text{macro}} = \frac{1}{3} \sum_{c=1}^{3} \text{Recall}_c.$$

Then, we define the Macro-F1 score as the harmonic mean of these macro-averaged values:

$$\text{Macro-F1} = \frac{2 \cdot \text{Precision}_{\text{macro}} \cdot \text{Recall}_{\text{macro}}}{\text{Precision}_{\text{macro}} + \text{Recall}_{\text{macro}}}.$$

Treatment of Negative Class (`No Interaction`). Although the negative class is included during training to enhance model calibration and decision boundaries, it is excluded from all evaluation metrics. This decision reflects established practice in biomedical relation extraction [21], which emphasizes performance on clinically actionable positive interactions.

E.2 Evaluation Settings

We evaluate model performance under two distinct matching settings to reflect different use scenarios:

- **Direction-aware Matching.** Drug pairs are treated as ordered tuples; that is, (`DRUG1`, `DRUG2`) and (`DRUG2`, `DRUG1`) are considered distinct. This setting requires models to not only detect the correct interaction type but also capture the directionality – i.e., which drug initiates or modulates the effect.

- **Direction-agnostic Matching.** Drug pairs are treated as unordered sets. A prediction is considered correct if it matches the ground-truth interaction type for either (`DRUG1`, `DRUG2`) or its reversed pair (`DRUG2`, `DRUG1`). This relaxed setting reflects clinical cases where directionality is either symmetric or not explicitly defined.

Unless otherwise noted, all results reported in the main text follow the stricter direction-aware setting, which better aligns with real-world pharmacodynamic modeling.

F Training Environment and Hyperparameter Configurations

This appendix details the computational environment, software stack, and hyperparameter configurations used for training and evaluating all baseline models.

F.1 Hardware and Software Environment

Experiments are conducted on a Linux server with the following specifications:

- CPU: Intel(R) Xeon(R) CPU (2.2 GHz, 2 cores)
- GPU: 2× NVIDIA T4 GPUs (16GB VRAM each)
- RAM: 32GB DDR4 Memory
- Storage: 128GB NVMe SSD

The software environment is standardized as follows:

- Operating System: Ubuntu 22.04 LTS
- Python: 3.10
- PyTorch: 2.0.1
- CUDA: 11.8
- Transformers Library (HuggingFace): 4.31
- DGL (Deep Graph Library): 1.1.1
- scikit-learn: 1.2.2
- RDKit: 2022.09.5
- Additional packages: NumPy 1.24, SciPy 1.10, Matplotlib 3.7

F.2 General Training Settings

Unless otherwise specified, the following settings are shared across all baseline models:

- Optimizer: Adam [15]
- Initial learning rate: 5×10^{-5}
- Batch size: 32
- Learning rate scheduler: linear decay with warm-up (10% of total steps)
- Weight decay: 1×10^{-2}
- Dropout rate: 0.1 (applied after embeddings and in MLPs)
- Number of epochs: 100
- Gradient clipping: maximum norm of 1.0

Early stopping is applied based on validation loss with a patience of 5 epochs.

F.3 Model-Specific Hyper-parameters

F.3.1 Text-only Baseline (BioMedBERT).

- Pretrained checkpoint: 'BioMedBERT-Base (uncased)'
- Maximum sequence length: 512 tokens
- Hidden size: 768
- Number of transformer layers: 12
- Number of attention heads: 12
- Fine-tuned end-to-end on MUDI dataset

F.3.2 Graph-only Baseline (GCN).

- Number of GCN layers: 2
- Hidden dimension (d_{hidden}): 768
- Input features: 37 atom features (one-hot encoded)
- Activation: ReLU
- Readout: global max pooling

F.3.3 Image-only Baseline (ViT).

- Pretrained checkpoint: 'ViT-B/16'
- Image size: 1000×800 pixels
- Patch size: 16×16
- Hidden size: 768
- Number of transformer layers: 12
- Number of attention heads: 12
- MLP head dimension: 3072
- Fine-tuned end-to-end on MUDI dataset

F.3.4 Late Fusion Baseline.

- No additional training is performed.
- Predictions are aggregated from six single-modality models: Name, Description, SMILES, Formula, Graph, and Image.
- Tie-breaking priority: `graph` → `name` → `image` → `SMILES` → `formula` → `description`.

F.3.5 Intermediate Fusion Baseline.

- Embedding dimension for each modality: $d = 768$.
- Concatenated embedding dimension: $d_{\text{fusion}} = 6 \times 768 = 4608$.
- Fusion MLP: Two fully connected layers.
- Hidden dimension: 1024.
- Activation: ReLU.
- Dropout: 0.1 after each layer.
- Output: 4-way softmax classification.

F.4 Training Time

On average, training our baseline model takes:

- **Single-modality model**: 3.5–4 hours per modality (BioMedBERT, GCN, ViT, etc.).
- **Intermediate Fusion**: 4–4.5 hours, including preloading all modality-specific encoders and training the fusion MLP.
- **Late Fusion**: No additional training time, as predictions are directly aggregated from pretrained single-modality models.

All timing estimates are based on dual-GPU training using two NVIDIA T4 GPUs with 16GB memory on each GPU.

G Additional Results

G.1 Single-Modality Analysis

Table 6 presents the classification performance of each individual modality on the MUDI dataset under both direction-aware and direction-agnostic matching. Among all modalities, the molecular **graph**-based model performs best, achieving a Micro-F1 of 65.44% and Macro-F1 of 57.36% in the direction-agnostic setting.

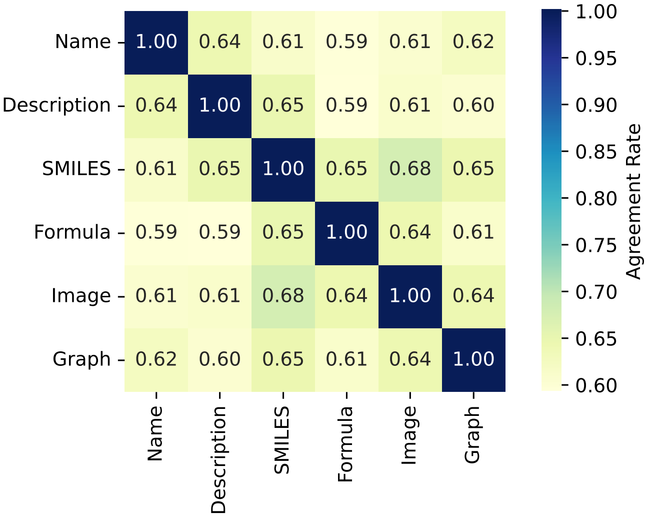

Figure 4: Agreement rate between different modalities.

This demonstrates the effectiveness of topological molecular representations in capturing pharmacodynamic interactions. The **name**-based model also yields surprisingly strong results, suggesting that drug identity alone encodes useful priors, especially when paired drugs have known interaction profiles.

In contrast, modalities like **SMILES**, **formula**, and **description** show limited standalone performance, particularly for the *New Effect* class. This result is consistent with the difficulty of extracting discriminative features from raw strings or sparse chemical formulas, and the noisiness of unstructured textual fields. Direction-aware results are consistently lower across all modalities, highlighting the increased challenge when models must account for interaction asymmetry.

To further understand how different modalities contribute to predictions, we visualize their agreement in Figure 4. The heatmap shows that while all modality pairs exhibit moderate correlation (typically between 0.59 and 0.68), **SMILES** and **image** achieve the highest agreement at 0.68, likely due to shared molecular-level information. This finding motivates future work on modality selection, weighted ensembling, or modality-specific gating to optimize fusion strategies.

G.2 Additional Ablation Studies

To further understand the contribution of modalities, we conduct several ablation experiments, each removing a modality to exhibit its impact on performance. Figure 5 illustrates the reductions in macro and micro F1 scores for Intermediate Fusion. Additionally, Figure 6 shows the decrease in the macro-averaged F1 metric across two fusion strategies.

The ablation study results in Figure 5 reveal a significant impact from Molecular Graph and demonstrate its enormous impact, as the scores are reduced by around 9%. Conversely, the performance when excluding Image and SMILE channels slightly improves the micro-averaged F1 but declines the macro-averaged F1, indicating potential conflict when adding these input sources. All other modalities contribute positively to prediction accuracy, however their individual impact is comparatively small.

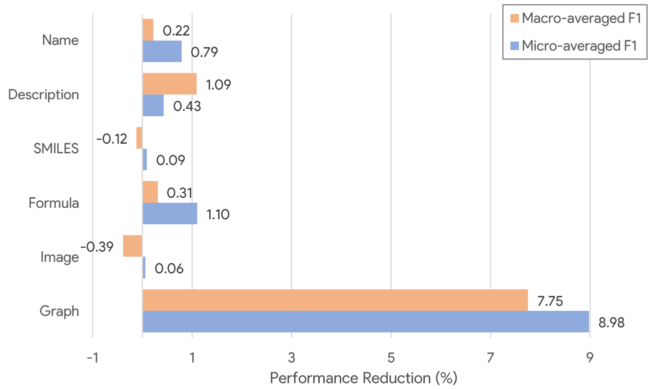

Figure 5: Macro and Micro F1 reduction with Intermediate Fusion.

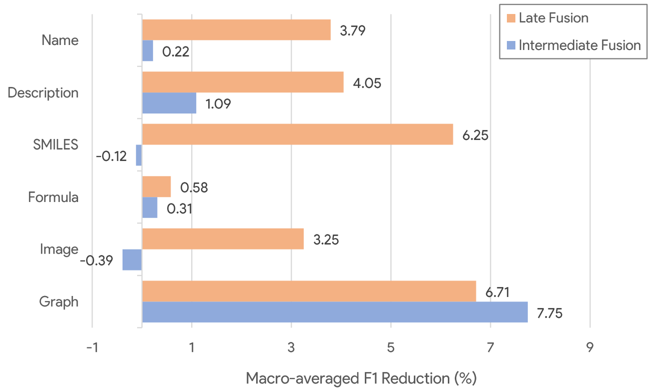

Figure 6: Macro F1 reduction between two fusion strategies.

As shown in Figure 6, Late Fusion generally shows a greater performance reduction than Intermediate Fusion when a specific modality is ablated. While removing SMILES in Intermediate Fusion shows a minor increase, its exclusion in Late Fusion results in a substantial 6.25% reduction, which underscores the inherent importance of this modality. This highlights the need for advanced fusion methods that can effectively integrate heterogeneous information from diverse modalities, as current baseline models may not fully leverage their complementary contributions.

Received 30 May 2025; revised xx xxx 2025; accepted xx xxx 2025

Table 6: Results of Single-Modality baselines on our MUDI dataset.

		Direction-aware Matching					**Direction-agnostic Matching**				
Modality	Metric	Synergism	Antagonism	New Effect	Macro-averaged	Micro-averaged	Synergism	Antagonism	New Effect	Macro-averaged	Micro-averaged
Name	Precision	38.73	41.13	33.84	37.90	38.81	62.92	59.21	53.74	58.62	62.25
	Recall	60.57	36.03	18.31	38.30	53.53	68.21	36.03	35.1	46.45	61.18
	F1	47.25	38.41	23.76	38.10	44.99	65.46	44.8	42.46	51.83	61.71
Description	Precision	41.86	0.00	0.00	13.95	41.86	65.78	0.00	0.00	21.93	65.78
	Recall	65.33	0.00	0.00	21.78	50.03	71.92	0.00	0.00	23.97	56.31
	F1	51.02	0.00	0.00	17.01	45.58	68.72	0.00	0.00	22.90	60.68
SMILES	Precision	32.35	30.64	0.00	21.00	32.17	54.32	47.95	0.00	34.09	53.58
	Recall	62.27	36.76	0.00	33.01	53.33	68.93	36.76	0.00	35.23	60.24
	F1	42.58	33.42	0.00	25.67	40.13	60.76	41.62	0.00	34.65	56.72
Formula	Precision	29.75	21.11	24.1	24.99	28.56	50.68	35.78	36.94	41.13	48.51
	Recall	56.55	31.57	8.07	32.06	48.98	63.99	31.57	15.84	37.13	56.22
	F1	38.99	25.3	12.09	28.09	36.08	56.56	33.54	22.17	39.03	52.08
Image	Precision	32.37	28.77	29.54	30.23	31.86	54.45	45.15	47.94	49.18	53.05
	Recall	54.82	37.94	12.76	35.17	48.96	62.09	37.94	25.01	41.68	56.25
	F1	40.71	32.72	17.82	32.51	38.6	58.02	41.23	32.87	45.12	54.6
Graph	Precision	53.09	61.27	52.30	55.55	54.05	77.23	74.56	71.24	74.34	76.69
	Recall	53.96	46.09	17.79	39.28	49.91	61.25	43.88	34.97	46.70	57.07
	F1	53.52	52.61	26.55	46.02	51.90	68.32	55.25	46.92	57.36	65.44