Cross-modal Graph Contrastive Learning with Cellular Images

Shuangjia Zheng[1,2*†], Jiahua Rao[1†], Jixian Zhang[2], Lianyu Zhou[2,3], Ethan Cohen[4], Wei Lu[2], Chengtao Li[2] and Yuedong Yang[1*]

[1]Department of Computer Science, Sun Yat-sen University.
[2]Galixir Technologies.
[3] Department of Computer Science, Xiamen University.
[4] IBENS, Ecole Normale Supérieure, PSL Research Institute.

*Corresponding author(s). E-mail(s): zhengshj9@mail2.sysu.edu.cn; yangyd25@mail.sysu.edu.cn;
†These authors contributed equally to this work.

Contents

A CIL-750K details

The original CIL dataset includes 919,265 five-channel fields of view containing 30,616 test compounds. It also includes metadata files which record morphological features for each cell in each image, both at the single cell level and at the population average level (i.e. per well); a workflow for image analysis to generate morphological features is also provided. Quality control indicators are provided as metadata, indicating fields of view that are out of focus or contain highly fluorescent material or debris. Chemical annotations are also provided for the application of compound processing. Figure 1 shows the molecular data distribution and the number of view per molecule in CIL dataset.

In CIL, each molecular intervention is imaged from multiple views in an experimental well and the experiment was repeated several times, resulting in an average of 30 views for each molecule. In order to keep the data balanced, we restricted each molecule to a maximum of 30 images, resulting in a cross-modal graph-image benchmark containing 750K views. Each view has a resolution of 692×520 pixels and 5 channels. These images were imaged with the ImageXpress Micro XLS automated microscope at 20×magnification. We resize the images to 128×128 without any cropping to fit the CNN models' input format. Figure 2 shows examples of molecules and corresponding images from the CIL dataset and Figure 3 shows the multiple views of a random selected molecule.

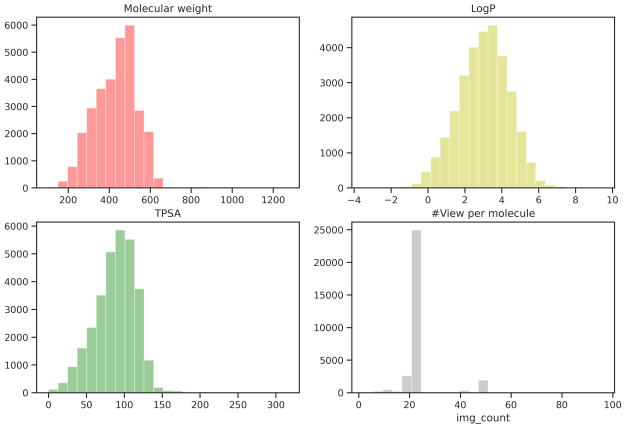

Fig. 1: Data Distribution of CIL.

B Implementation details and hyperparameters.

Here we describe the implementation details for the pre-training and fine-tuning stages.

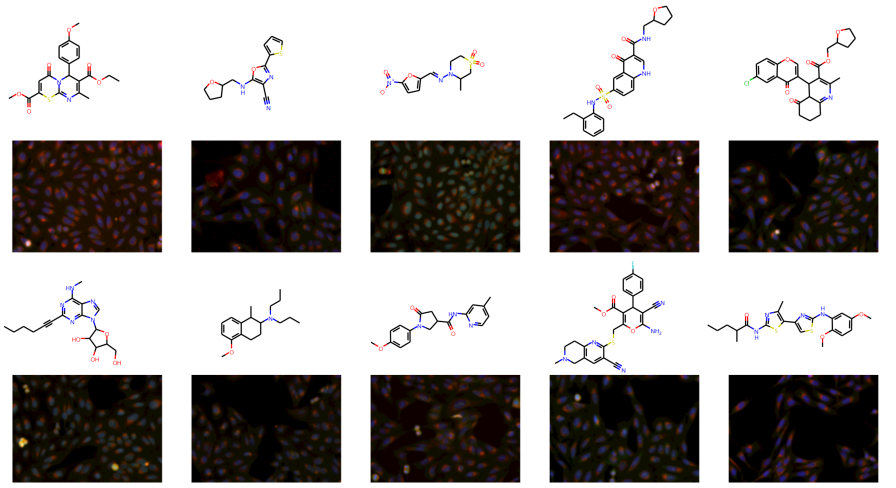

Fig. 2: A random selection of 10 molecules and corresponding cellular images (1 view).

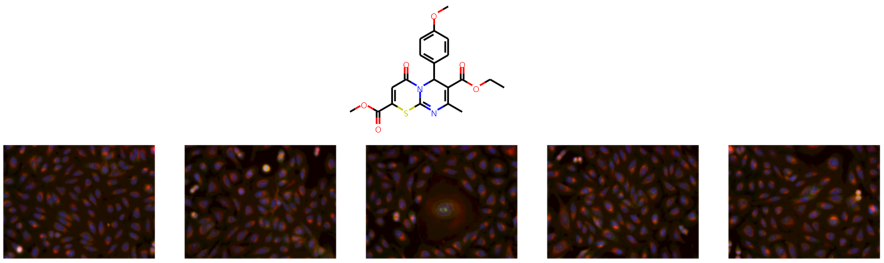

Fig. 3: Five different views on the same molecule.

B.1 Generative Graph-image matching.

We employ Variational Auto-Encoders (VAE) as generative agents, which are asked to recover the representation of one modality given the parallel representation from the other modality. For example, we need to model the conditional likelihood $p(z_I \mid z_G)$ when generating the cellular image from their corresponding molecular graph. The reparameterized variable could be defined as $z_G = \mu_G + \sigma_G \cdot \zeta$ with mean μ_G, covariance σ_G, and $\zeta \sim \mathcal{N}(0, 1)$. Therefore, we have the following lower bound:

$$logp(z_I \mid z_G) \geq \mathbb{E}_{q(z_G|z_I)}[\log p(z_I \mid z_G)] - \mathcal{D}_{KL}(q(z_G \mid z_I) \| p(z_G)) \qquad (1)$$

Similarly, when generating the molecular graph from their corresponding cellular image, we have:

$$logp(z_G \mid z_I) \geq \mathbb{E}_{q(z_I|z_G)}[\log p(z_G \mid z_I)] - \mathcal{D}_{KL}(q(z_I \mid z_G) \| p(z_I)) \tag{2}$$

Both the above objectives are composed of a conditional log-likelihood and a KL-divergence.

Following the variation representation reconstruction (VRR) of [1], we use the mean-squared error (MSE) for reconstruction on the representation space:

$$\mathbb{E}_{q(z_G|z_I)}[\log p(z_I \mid z_G)] = \mathbb{E}_{q(z_G|z_I)}[\left\| z_G - q(z_G^{'} \mid z_I) \right\|_2^2] + C \tag{3}$$

$$\mathbb{E}_{q(z_I|z_G)}[\log p(z_G \mid z_I)] = \mathbb{E}_{q(z_I|z_G)}[\left\| z_I - q(z_I^{'} \mid z_G) \right\|_2^2] + C \tag{4}$$

Thus, combining both two regularizers mentioned above, the final GM loss function can be formulated as:

$$\begin{aligned} \mathcal{L}_{GM} = & -\frac{\lambda_{kl}}{2}(\mathcal{D}_{KL}(q_\phi(z_I \mid z_G) \| p(z_I)) + \mathcal{D}_{KL}(q_\phi(z_G \mid z_I) \| p(z_G))) \\ & + \frac{1}{2}(\mathbb{E}_{q(z_I|z_G)}[\left\| z_I - q(z_I^{'} \mid z_G) \right\|_2^2] + \mathbb{E}_{q(z_G|z_I)}[\left\| z_G - q(z_G^{'} \mid z_I) \right\|_2^2]) \end{aligned} \tag{5}$$

B.2 Clinical Outcome Prediction

Dataset

To standardize the clinical-trial-outcome predictions, we use the Trial Outcome Prediction (TOP) benchmark constructed by HINT, which incorporate rich data components including drug molecule information, disease information, trial eligibility criteria and trial outcome information. Herein, we consider phase-level evaluation on the trial outcome, where we predict the outcome of a single-phase study. Since each phase has different goals (e.g., phase I is for safety, whereas phases II and III are for efficacy), we evaluate phases I, II, and III separately. We follow the data splitting proposed by HINT and data statistics are shown in Table 5

Phase-level	Molecule	Successes	Failures
Phase I	944	564	380
Phase II	2865	1396	1469
Phase III	1752	1203	549

Table 1: Statistics of Clinical Outcome Datasets.

Baselines

We first include three machine learning-based methods (RF, LR, XGBoost) and a knowledge-aware GNN model HINT as our baseline. **Random Forest (RF)** is

a bagging algorithm for classification or regression problems, which obtains the prediction by voting or averaging of each base learner (decision tree). **Logistic regression (LR)** is a simple, parallelizable classification method that uses maximum likelihood estimation for parameter estimation. **XGBoost**, also called an extreme gradient boosting tree, uses CART regression tree or linear classifier as a base learner to ensemble model predictions. These machine learning baselines utilize 1024-dimensional Morgan fingerprint features for trial outcome prediction. **HINT** is a hierarchical interaction network designed for clinical-trial-outcome predictions. It uses (1) 1024-dimensional Morgan fingerprint features, (2) a pre-trained BERT model to encode eligibility criteria into sentence embedding and (3) a graph-based attention model GRAM to encode disease information. Furthermore, we also include the self-supervised learning methods to constitute our baselines, including **ContextPred**, **GraphLoG**, **GROVER**, **GraphCL** and **JOAO**. For this downstream task, we use the molecule encoders over input molecule graphs for the fine-tuning of clinical outcome prediction.

Fune-tuning hyperparameter

For fine-tuning, an extra linear classifier is appended to the pre-trained GNN. We fine-tune the model for 100 epochs using a batch size of 32 with a dropout rate of 50%. We use the Adam optimizer with an initial learning rate of 1e-3. Experiments are performed for 5 times, with mean and standard deviation of ROC-AUC and PR-AUC are reported.

B.3 Molecular property Prediction

Dataset

BBBP: The Blood-brain barrier penetration dataset includes binary labels for 2035 compounds on their permeability properties.**Tox21**: The Tox21 dataset was created in the Tox21 data challenge, which contains qualitative toxicity measurements for 7821 compounds on 12 different targets, including nuclear receptors and stress response pathways. **HIV**: 41K compounds with binary labels for HIV virus replication inhibition. **ToxCast** includes 8576 drug compounds with binary labels of toxicity experiment outcomes with 617 targets. **ESOL**: The ESOL is a small dataset consisting of water solubility data for 1128 compounds.**Lipophilicity**: Experimental data for the octanol/water distribution coefficient of 4200 molecules.

Dataset	Tasks	Type	Molecule	Metric
BBBP	1	GC	2,035	ROC-AUC
Tox21	12	GC	7,821	ROC-AUC
HIV	1	GC	41K	ROC-AUC
ToxCast	617	GC	8,576	ROC-AUC
ESOL	1	GR	1,128	RMSE
Lipophilicity	1	GR	4,198	RMSE

Table 2: Statistics of datasets. GC for Graph Classification, GR for Graph Regression.

Features	Size	Description
Atom type	101	type of atom (e.g C,N,O)
Hybridization	6	sp, sp2, sp3, sp3d, sp3d2 or unknown
Number of H	1	number of bond hydrogen atoms
Degrees	1	number of neighbor atoms
Formal Charges	1	number of formal charge
Valences	1	number of valences

Table 3: Atom features

Features	Size	Description
Bond type	4	single, double, triple, aromatic
Stereo	6	none, any, E/Z or cis/trans
In ring	1	whether the bond is part of a ring
conjugated	1	whether the bond is conjugated

Table 4: Bond features

Featurization Extraction

The feature extraction contains three parts: 1) Node feature extraction. 2) Bond feature extraction. 3) Topology connection matrix. We use RDKit to extract all features as the input of GNN. Table 3 and Table 4 show the atom and bond features we used in MIGA.

Fune-tuning hyperparameter

For fine-tuning, we followed the GraphCL's [2] settings. An extra linear layer is appended to the pre-trained GNN to perform classification and regression, respectively. We fine-tune the model for 100 epochs using a batch size of 32 with a dropout rate of 50%. We use the Adam optimizer with an initial learning rate of 1e-3. Experiments are performed 5 times, with means and standard deviations of AUC and RMSE are reported.

B.4 Dataset of cDNA-induced graph retrieval studies

We collected cellular images that were overexpressed with cDNA open reading frames for 6 genes by [3], including BRCA1, HIF1A, JUN, STAT3, TP53 and HSPA5. We used ExCAPEDB database [4] to retrieve gene-specific agonists and non-functional molecules that have not been observed in the training set.

Gene	Functional cpds	Non-functional cpds	cDNA-induced Images
BRCA1	6937	9269	6
JUN	21	826	12
HIF1A	52	205	6
HSPA5	656	2498	12
TP53	3285	9364	12
STAT3	87	4179	18

Table 5: Statistics of cDNA-induced graph retrieval datasets.

C Case study for graph retrieval

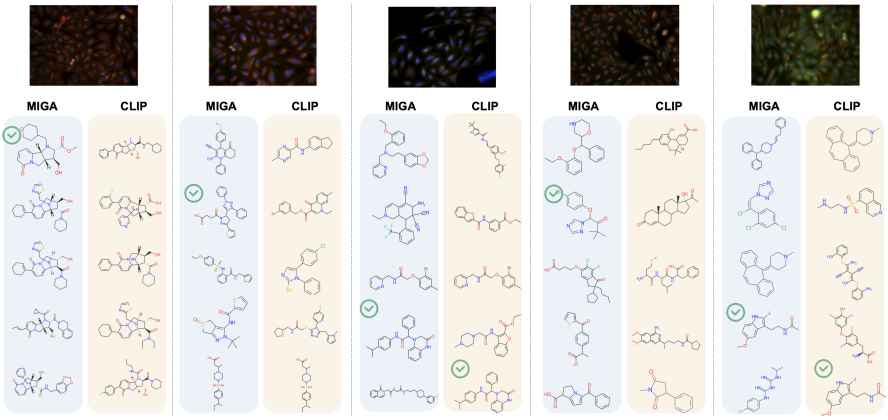

Fig. 4: Additional examples for graph retrieval task. The top-ranked molecules retrieved by our method and baseline are shown. Molecules that hit the ground truth are flagged.

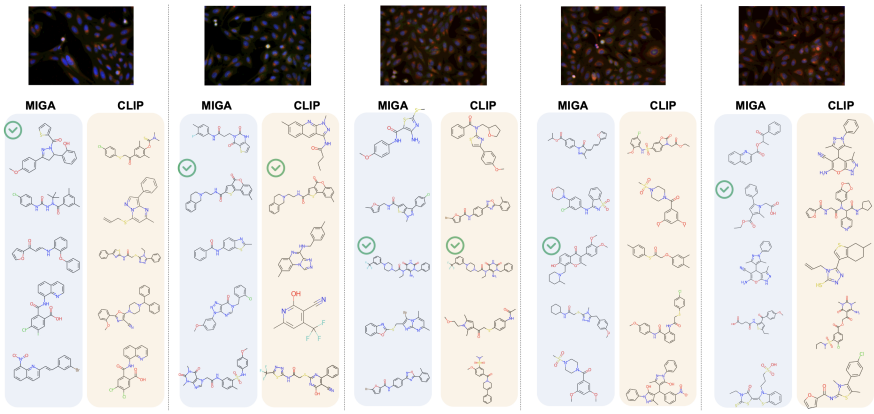

Fig. 5: Additional examples for graph retrieval task. The top-ranked molecules retrieved by our method and baseline are shown. Molecules that hit the ground truth are flagged.

D Case study for image retrieval

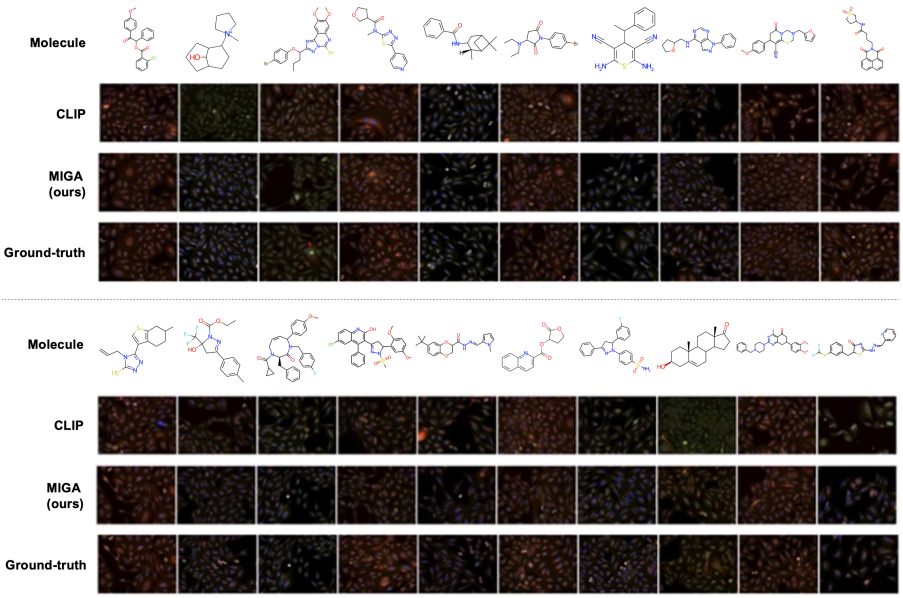

Fig. 6: Additional examples for image retrieval task. The images retrieved by our method and baseline are shown.

E Case study for zero-shot graph retrieval

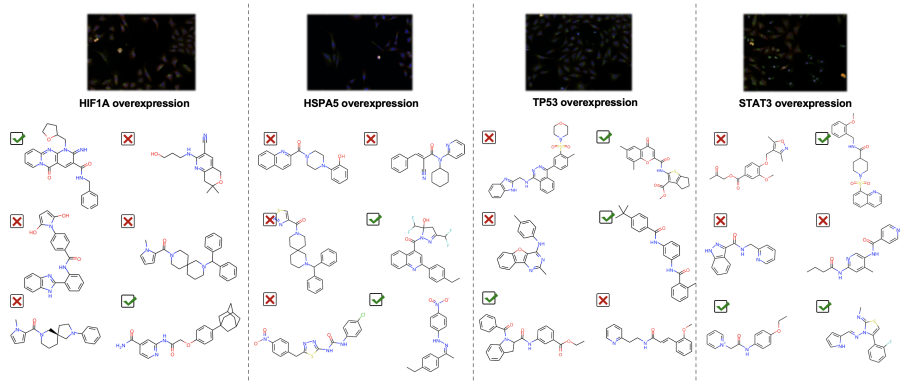

Fig. 7: Additional examples for zero-shot graph retrieval task. The figure shows the cells induced by the cDNA interventions for specific genes (HIF1A, HSPA5, TP53, STAT3) and our model can identify diverse molecules that have similar functions to these cDNA interventions (ticked).

F Ablation study

Figure 4 (a) shows the effect of the number of view per molecule perform on graph retrieval task. We notice that with less than 10 views, the more views involve in pre-training the better the results would be, but using all views (on average 25 views per molecule) does not achieve more gains. We attribute this phenomenon to the batch effect of the cellular images, where images from different wells can vary considerably because of the experimental errors and thus mislead the model.

Figure 4 (b) studies the impact of CNN architecture choices on the graph retrieval task. Due to the relatively small amount of data, we use small CNN variants (ResNet34[5], EfficientNet-B1[6], DenseNet121[7] and ViT_tiny[8]) for evaluation. We note that small CNN models such as ResNet, EfficientNet and DenseNet achieve comparable performance while bigger models like ViT do not show superiority. We assume this is because these models are pre-trained on non-cellular images, so heavier models do not necessarily lead to gains on cellular tasks.

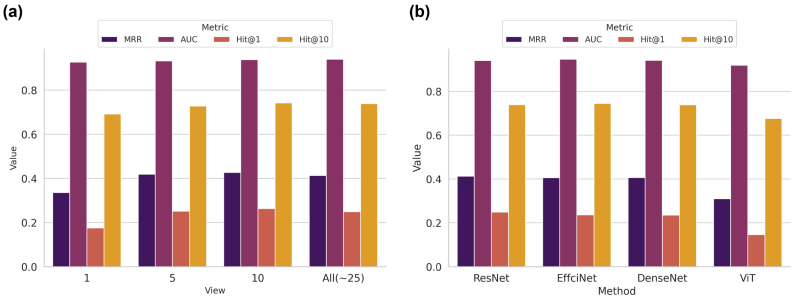

Fig. 8: (a) Effect of number of view per molecule and (b) effect of CNN architecture.

References

[1] Liu, S., Wang, H., Liu, W., Lasenby, J., Guo, H., Tang, J.: Pre-training molecular graph representation with 3D geometry. In: International Conference on Learning Representations (2022)

[2] You, Y., Chen, T., Sui, Y., Chen, T., Wang, Z., Shen, Y.: Graph contrastive learning with augmentations. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.-F., Lin, H.-T. (eds.) Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS (2020)

[3] Rohban, M.H., Singh, S., Wu, X., Berthet, J.B., Bray, M.-A., Shrestha, Y., Varelas, X., Boehm, J.S., Carpenter, A.E.: Systematic morphological profiling of human gene and allele function via cell painting. Elife **6**, 24060 (2017)

[4] Sun, J., Jeliazkova, N., Chupakhin, V., Golib-Dzib, J.-F., Engkvist, O., Carlsson, L., Wegner, J., Ceulemans, H., Georgiev, I., Jeliazkov, V., *et al.*: Excape-db: an integrated large scale dataset facilitating big data analysis in chemogenomics. Journal of cheminformatics **9**(1), 1–9 (2017)

[5] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)

[6] Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning, pp. 6105–6114 (2019). PMLR

[7] Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708 (2017)

[8] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)