Contrastive MIM: A Contrastive Mutual Information Framework for Unified Generative and Discriminative Representation Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

We present *Contrastive Mutual Information Machine* (cMIM), a probabilistic framework that adds a contrastive objective to the Mutual Information Machine (MIM) Livne et al. [2019], yielding representations that are effective for both discriminative and generative tasks. Unlike conventional contrastive learning van den Oord et al. [2018], Chen et al. [2020], He et al. [2020], cMIM does not require positive data augmentations and exhibits reduced sensitivity to batch size. We further introduce *informative embeddings*, a generic training-free method to extract enriched features from decoder hidden states of encoder–decoder models.

We evaluate cMIM on life-science tasks, including molecular property prediction on ZINC-15 Sterling and Irwin [2015], Weininger [1988] (ESOL, FreeSolv, Lipophilicity) and biomedical image classification (MedMNIST Yang et al. [2021]). cMIM consistently improves downstream accuracy over MIM and InfoNCE baselines while maintaining comparable reconstruction quality. These results indicate cMIM is a promising foundation-style representation learner for biomolecular and biomedical applications and is readily extendable to multi-modal settings (e.g., molecules + omics + imaging).

1 Introduction

Learning representations that support both unknown downstream prediction tasks and generative use cases is central to life-science machine learning, where data span molecules, sequences, imaging, and multi-omics. Contrastive losses such as InfoNCE van den Oord et al. [2018], Chen et al. [2020] are effective but often hinge on meaningful augmentations and large batches. Probabilistic auto-encoders maximize information about inputs, yet their latent geometry can be suboptimal for discriminative tasks Livne et al. [2019], Reidenbach et al. [2023].

We propose **cMIM**, a simple contrastive extension of MIM that (i) introduces a discriminator over pair-similarity without requiring positive augmentations, (ii) aligns local clustering with global angular separation, and (iii) remains robust to batch-size via Monte Carlo expectations Hoeffding [1963]. We also propose **informative embeddings**, obtained from decoder hidden states, that substantially improve downstream performance *without extra training*.

We evaluate cMIM on molecular property prediction on *ZINC-15* Sterling and Irwin [2015], Weininger [1988] (ESOL, FreeSolv, Lipophilicity) and biomedical imaging with *MedMNIST* Yang et al. [2021]. Experiments show consistent gains over MIM and InfoNCE with reduced batch-size sensitivity.

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

Algorithm 1 Learning parameters θ of cMIM

Require: Samples from dataset $\mathcal{P}(\boldsymbol{x})$

while not converged **do**

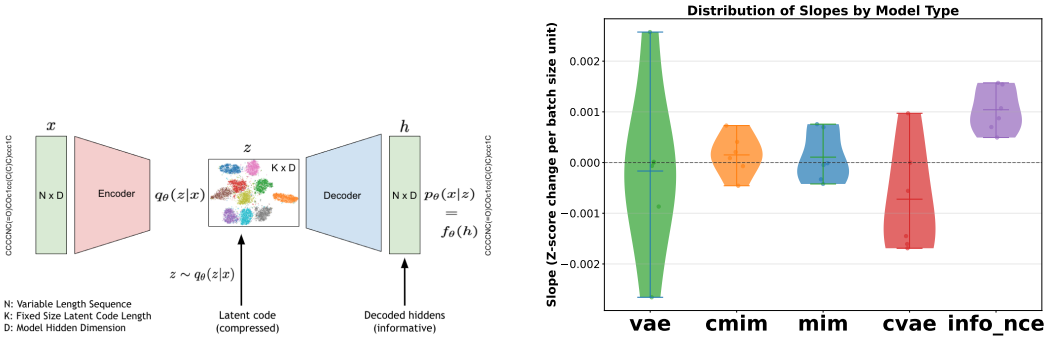
  $\mathcal{D} \leftarrow \{\boldsymbol{x}_j, \boldsymbol{z}_j \sim q_{\boldsymbol{\theta}}(\boldsymbol{z} \mid \boldsymbol{x})\, \mathcal{P}(\boldsymbol{x})\}_{j=1}^{B}$ {Sample a batch}

  $\hat{\mathcal{L}}_{\text{A-MIM}} = -\frac{1}{B} \sum_{i=1}^{B} \left(\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i \mid \boldsymbol{z}_i) + \log p_{k=1}(\boldsymbol{x}_i, \boldsymbol{z}_i) + \frac{1}{2}(\log q_{\boldsymbol{\theta}}(\boldsymbol{z}_i \mid \boldsymbol{x}_i) + \log p(\boldsymbol{z}_i)) \right)$

  $\theta \leftarrow \theta - \eta \nabla_\theta \hat{\mathcal{L}}_{\text{A-MIM}}$ {Reparameterized gradients}

end while

Figure 1: Training algorithm for cMIM.



(a) (b)

Figure 2: Left: **(a)** Informative embeddings $\boldsymbol{h}$ are taken from decoder hidden states before mapping to $p_{\boldsymbol{\theta}}(\boldsymbol{x} \mid \boldsymbol{z})$. For autoregressive decoders we use teacher forcing. Right: **(b)** Distribution of slopes from linear fits of accuracy vs. batch-size for different models. Each point corresponds to the average z-score of a model trained on MNIST-like datasets. Both MIM and cMIM are not sensitive to batch-size.

2 Formulation

Preliminaries. Let $\boldsymbol{x}$ denote the observation and $\boldsymbol{z}$ the latent code. MIM is a probabilistic auto-encoder that maximizes mutual information between $\boldsymbol{x}$ and $\boldsymbol{z}$ while encouraging clustered latents via entropy minimization.

Contrastive variable and objective. We introduce a binary variable k that indicates whether $(\boldsymbol{x}, \boldsymbol{z})$ is a matched pair (k=1) or a mismatched pair (k=0). A matched pair is defined as $\boldsymbol{z}_i \sim q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}_i)$. We define encoder/decoder factorizations with a discriminator over k:

$$q_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z}, k) = q_{\boldsymbol{\theta}}(k|\boldsymbol{x}, \boldsymbol{z})\, q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})\, q_{\boldsymbol{\theta}}(\boldsymbol{x}), \quad p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z}, k) = p_{\boldsymbol{\theta}}(k|\boldsymbol{x}, \boldsymbol{z})\, p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\, p_{\boldsymbol{\theta}}(\boldsymbol{z}). \tag{1}$$

Let $\boldsymbol{z}_i \sim q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}_i)$. Using a temperature-scaled cosine similarity $s(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau$ with $g_{ij} \equiv g(\boldsymbol{z}_i, \boldsymbol{z}_j) = \exp(s(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)$, we set

$$p_{k=1}(\boldsymbol{x}_i, \boldsymbol{z}_i) \;=\; \frac{g_{ii}}{g(\boldsymbol{z}_i, \boldsymbol{z}_i) + \mathbb{E}_{\boldsymbol{x}' \sim P(\boldsymbol{x}),\, \boldsymbol{z}' \sim q_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}')}[g(\boldsymbol{z}_i, \boldsymbol{z}')]} \;\approx\; \frac{g_{ii}}{g_{ii} + \frac{1}{B-1}\sum_{j \neq i} g_{ij}}. \tag{2}$$

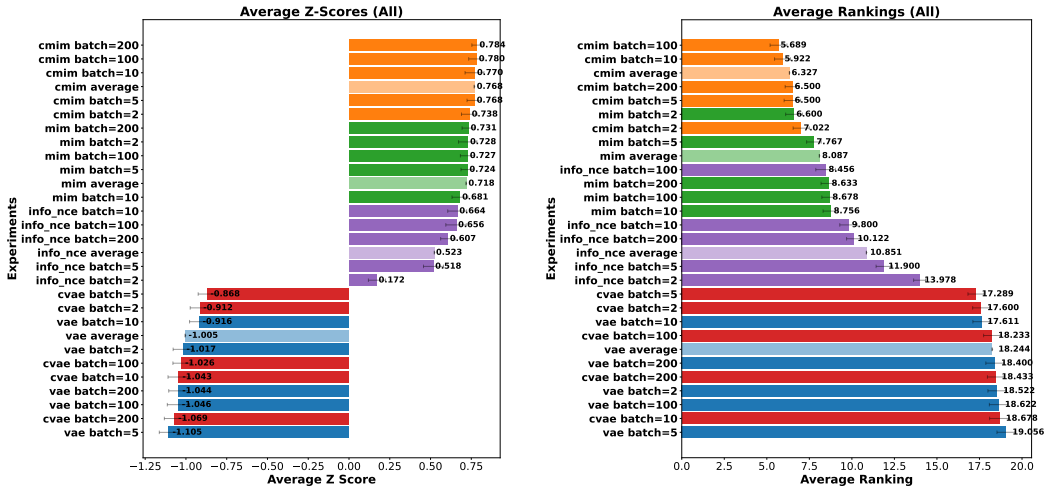
Training samples always satisfy k=1 since $\boldsymbol{z}_i$ is drawn conditionally on $\boldsymbol{x}_i$. Replacing the denominator expectation by an in-batch Monte Carlo estimate yields a simple, augmentation-free contrastive term with concentration improving in B.

Relation to InfoNCE. Defining s_{ij}=$s(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau$ makes Eq. (2) equivalent to an InfoNCE softmax where the positive logit is offset by $\log(B-1)$. This calibrates $p_{k=1}$ to $1/2$ when logits are equal (independent of B), reducing batch-size sensitivity. See Appendix for more details.

Objective. We adopt the A-MIM upper bound [Livne et al., 2020] with the added contrastive term:

$$\widehat{\mathcal{L}}_{\text{A-MIM}}(\theta; \mathcal{D}) = -\frac{1}{N}\sum_{i=1}^{N} \Big[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i|\boldsymbol{z}_i) + \log p_{k=1}(\boldsymbol{x}_i, \boldsymbol{z}_i) + \frac{1}{2}\Big(\log q_{\boldsymbol{\theta}}(\boldsymbol{z}_i|\boldsymbol{x}_i) + \log P(\boldsymbol{z}_i) \Big) \Big], \tag{3}$$

where $P(\boldsymbol{z})$ is a Normal anchor prior. The first term preserves generative fidelity; the second encourages global angular separation.



(a) Z-scores with error bars ↑ **(b)** Rankings with error bars ↓

Figure 3: Classification accuracy across datasets and classifiers. Colors indicate model families: cMIM (orange), MIM (green), InfoNCE (purple), VAE (blue), cVAE (red). Light shades denote model averages. cMIM consistently outperforms all baselines across batch-sizes and metrics.

Model (Latent K × D)	ESOL		FreeSolv		Lipophilicity		Recon.
	SVM	MLP	SVM	MLP	SVM	MLP	
MIM (1 × 512) [reproduction]	0.65	0.34	2.23	1.82	0.663	0.61	100%
cMIM (1 × 512)	0.47	0.19	2.32	1.67	0.546	0.38	100%
MIM (1 × 512) info emb	0.21	0.29	1.55	1.4	0.234	0.28	100%
cMIM (1 × 512) info emb	0.21	0.24	1.74	1.35	0.24	0.23	100%
CDDD (512)	**0.33**		**0.94**		**0.4**		
†Seq2seq (N × 512)	0.37	0.43	1.24	1.4	0.46	0.61	100%
†Perceiver (4 × 512)	0.4	0.36	1.22	1.05	0.48	0.47	100%
†VAE (4 × 512)	0.55	0.49	1.65	3.3	0.63	0.55	46%
MIM (1 × 512)	0.58	0.54	1.95	1.9	0.66	0.62	100%
Morgan fingerprints (512)	1.52	1.26	5.09	3.94	0.63	0.61	

Table 1: Comparison of models on ESOL, FreeSolv, and Lipophilicity using SVM and MLP regressors, with reconstruction accuracy. Top: our results. Bottom: results from Reidenbach et al. [2023]. For †models, sequence representations were averaged to 512 dimensions. Bold: best non-MIM results. Highlighted: best among MIM-based models. Note that CDDD training included these classification tasks.

Informative embeddings. Instead of using $\boldsymbol{z}$ directly, we take the decoder hidden states $\boldsymbol{h}$ *before* mapping to $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ parameters, and pool them (mean over sequence/space) to obtain $\boldsymbol{h}_i$. For autoregressive decoders we use teacher forcing, i.e., $\boldsymbol{h}_i = \text{Decoder}(\boldsymbol{x}_i, \boldsymbol{z}_i)$. These *informative embeddings* enrich $\boldsymbol{z}$ with a context of the decoder distribution at no extra training cost. See Figure 2a.

3 Experiments

Setup. We report biomolecular and biomedical classification results.

Molecular property prediction. Following Reidenbach et al. [2023], we train on a large tranche of ZINC-15 Sterling and Irwin [2015] with SMILES tokenization Weininger [1988] and evaluate ESOL, FreeSolv, and Lipophilicity on held-out splits. We compare MIM Livne et al. [2019], cMIM, VAE Kingma and Ba [2015], and an InfoNCE encoder van den Oord et al. [2018]. Embeddings are

either (i) the mean encoder code z, or (ii) *informative embeddings* h from the decoder. Downstream regressors are SVM/MLP (Scikit-learn defaults Pedregosa et al. [2011]). See Table 1.

Biomedical imaging. We evaluate on MedMNIST Yang et al. [2021] as a light-weight benchmark for learning transferable biomedical image representations. All models share the same Perceiver-style encoder/decoder Jaegle et al. [2021]; InfoNCE uses the encoder only. We report accuracy across datasets and summarize with average ranks/z-scores. See Figure 3.

Key Results. (i) *Biomolecules.* cMIM consistently improves ESOL/FreeSolv/Lipophilicity errors relative to MIM and VAE when using informative embeddings, and is competitive with chemical baselines (e.g., CDDD Winter et al. [2019]) while retaining perfect reconstruction.
(ii) *Biomedical imaging.* Across MedMNIST tasks, cMIM dominates average rankings over MIM/InfoNCE/VAE across batch-sizes.
(iii) *Robustness.* Slopes from accuracy vs. batch-size fits cluster near zero for cMIM and MIM, while InfoNCE increases with batch-size, confirming reduced sensitivity. cMIM, like MIM, remains stable even with very small batch-sizes, unlike InfoNCE which degrades as batch-size decreases.
(iv) *Generative fidelity.* cMIM matches MIM reconstruction on molecules and shows slight improvements on biomedical images (Appendix Figure 6), suggesting a benign regularization effect.

4 Related Work

Contrastive learning. CPC/InfoNCE van den Oord et al. [2018], SimCLR Chen et al. [2020], and MoCo He et al. [2020] demonstrated strong representation learning but rely on augmentations and many negatives. Augmentation-free variants such as BYOL Grill et al. [2020] or SimSiam Chen and He [2021] introduce architectural asymmetries/predictors.

Mutual-information auto-encoding. MIM Livne et al. [2019] maximizes mutual information while clustering latents; related MI-regularized VAEs/auto-encoders optimize information-theoretic surrogates but often require intricate weighting. Our contribution integrates a calibrated contrastive term into a probabilistic MIM, improving global discriminative geometry with minimal complexity.

Life-science foundation modeling. Sequence-to-sequence and continuous descriptor models (e.g., CDDD Winter et al. [2019]) support molecular property prediction; MedMNIST Yang et al. [2021] popularizes light-weight biomedical imaging benchmarks. Our results show cMIM's unified generative+discriminative modeling is competitive across both, and the method is directly extensible to multi-modal use cases.

5 Limitations

While cMIM improves discriminative performance and batch-size robustness, limitations remain. Generative evaluation is restricted to reconstruction, leaving open questions on sample quality and controllability. Scalability to larger models and modalities (e.g., video, long-context text) is untested. Performance may still depend on the similarity function, temperature τ, and the number of negatives, which adds computational cost. Future work should address scaling, broader modalities, and generative analysis.

6 Conclusions

We introduced cMIM, a simple augmentation-free contrastive extension to probabilistic representation learning that improves discriminative performance while maintaining generative fidelity. On molecular properties Sterling and Irwin [2015], Weininger [1988], Winter et al. [2019] and biomedical imaging Yang et al. [2021], cMIM (especially with informative embeddings) outperforms MIM/InfoNCE/VAEs and is robust to batch-size.

Specifically, cMIM offers a practical backbone for life-science foundation models: its factorization and decoder-based embeddings extend naturally to multi-modal molecular+omics or molecular+imaging setups. Future work: (i) multi-modal joint training, (ii) scaling to larger architectures Jaegle et al. [2021] and datasets, and (iii) systematic studies of geometry and calibration Wang and Isola [2020].

References

Alexander Alemi, Ben Poole, Ian Fischer, Joshua Dillon, Rif A. Saurous, and Kevin Murphy. Fixing a broken elbo. In *International Conference on Machine Learning (ICML)*, pages 159–168, 2018.

Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc., 2009.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, 2021.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: an extension of MNIST to handwritten letters. *CoRR*, abs/1702.05373, 2017. URL `http://arxiv.org/abs/1702.05373`.

Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012. doi: 10.1109/MSP.2012.2211477.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9729–9738, 2020.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *International Conference on Learning Representations (ICLR)*, 2017.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, March 1963. URL `http://www.jstor.org/stable/2282952`.

Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, et al. Minicpm: Unveiling the potential of small language models with scalable training strategies. *arXiv preprint arXiv:2404.06395*, 2024.

Ross Irwin, Spyridon Dimitriadis, Jiazhen He, and Esben Jannik Bjerrum. Chemformer: a pre-trained transformer for computational chemistry. *Machine Learning: Science and Technology*, 3(1):015022, 2022. doi: 10.1088/2632-2153/ac3ffb. URL `https://doi.org/10.1088/2632-2153/ac3ffb`.

Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pages 4651–4664. PMLR, 2021.

Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A. Shoemaker, Paul A. Thiessen, Bo Yu, Leonid Zaslavsky, Jian Zhang, and Evan E. Bolton. Pubchem 2019 update: improved access to chemical data. *Nucleic Acids Research*, 47(D1):D1102–D1109, 2018. doi: 10.1093/nar/gky1033. URL `https://doi.org/10.1093/nar/gky1033`.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Oleksii Kuchaiev, Jason Li, Huyen Nguyen, Oleksii Hrinchuk, Ryan Leary, Boris Ginsburg, Samuel Kriman, Stanislav Beliaev, Vitaly Lavrukhin, Jack Cook, et al. Nemo: a toolkit for building ai applications using neural modules. *arXiv preprint arXiv:1909.09577*, 2019.

Micha Livne, Kevin Swersky, and David J. Fleet. MIM: Mutual Information Machine. *arXiv preprint arXiv:1910.03175*, 2019.

Micha Livne, Kevin Swersky, and David J Fleet. Sentencemim: A latent variable language model. *arXiv preprint arXiv:2003.02645*, 2020.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011. URL http://jmlr.org/papers/v12/pedregosa11a.html.

Danny Reidenbach, Micha Livne, Rajesh K. Ilango, Michelle Gill, and Johnny Israeli. Improving small molecule generation using mutual information machine. *arXiv preprint arXiv:2208.09016*, 2023.

Teague Sterling and John J. Irwin. Zinc 15 – ligand discovery for everyone. *Journal of Chemical Information and Modeling*, 55(11):2324–2337, 2015. doi: 10.1021/acs.jcim.5b00559. URL https://doi.org/10.1021/acs.jcim.5b00559.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pages 9929–9939. PMLR, 2020.

David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, 1988. doi: 10.1021/ci00057a005. URL https://doi.org/10.1021/ci00057a005.

Robin Winter, Floriane Montanari, Frank Noé, and Djork-Arné Clevert. Learning continuous and data-driven molecular descriptors by translating equivalent chemical representations. *Chemical Science*, 10:1692–1701, 2019. doi: 10.1039/C8SC04175J. URL http://dx.doi.org/10.1039/C8SC04175J.

Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. Medmnist v2: A large-scale lightweight benchmark for 2d and 3d biomedical image classification. *CoRR*, abs/2110.14795, 2021. URL https://arxiv.org/abs/2110.14795.

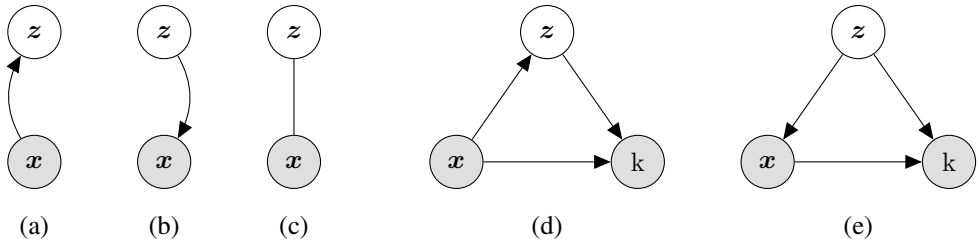


Figure 4: (Left) A MIM model learns two factorizations of a joint distribution (encoding/decoding) and an undirected joint. (Right) cMIM extends MIM with a binary variable k to encourage global discriminative structure while preserving local clustering.

A Extended Formulation

Background: Contrastive Learning

Intuition. Contrastive learning pulls positives together and pushes negatives apart using a similarity score. With cosine similarity $\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j) = \frac{\boldsymbol{z}_i \cdot \boldsymbol{z}_j}{\|\boldsymbol{z}_i\| \, \|\boldsymbol{z}_j\|}$ and temperature τ, define $g(\boldsymbol{z}_i, \boldsymbol{z}_j) = \exp(\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)$. The InfoNCE loss per sample is

$$\text{InfoNCE}(\boldsymbol{x}_i, \boldsymbol{x}_i^+) = -\log\left(\frac{g(\boldsymbol{z}_i, \boldsymbol{z}_i^+)}{\sum_{j=1}^{B} g(\boldsymbol{z}_i, \boldsymbol{z}_j)}\right). \tag{4}$$

cMIM without Data Augmentation

Intuition. We add a latent Bernoulli k that judges whether $(\boldsymbol{x}, \boldsymbol{z})$ is a matched pair. Its calibrated probability yields an augmentation-free contrastive term. We extend the MIM graphical model with k and define the joint factorizations

$$q_\theta(\boldsymbol{x}, \boldsymbol{z}, \mathrm{k}) = q_\theta(\mathrm{k} \mid \boldsymbol{x}, \boldsymbol{z})\, q_\theta(\boldsymbol{z} \mid \boldsymbol{x})\, q_\theta(\boldsymbol{x}), \qquad p_\theta(\boldsymbol{x}, \boldsymbol{z}, \mathrm{k}) = p_\theta(\mathrm{k} \mid \boldsymbol{x}, \boldsymbol{z})\, p_\theta(\boldsymbol{x} \mid \boldsymbol{z})\, p_\theta(\boldsymbol{z}). \tag{5}$$

With $\boldsymbol{z}_i \sim q_\theta(\boldsymbol{z} \mid \boldsymbol{x}_i)$, the discriminator over k shares parameters in both paths and is Bernoulli with success probability

$$p_{k=1}(\boldsymbol{x}_i, \boldsymbol{z}_i) = \frac{g(\boldsymbol{z}_i, \boldsymbol{z}_i)}{g(\boldsymbol{z}_i, \boldsymbol{z}_i) + \mathbb{E}_{\boldsymbol{x}' \sim \mathcal{P}(\boldsymbol{x}),\ \boldsymbol{z}' \sim q_\theta(\boldsymbol{z} \mid \boldsymbol{x}')}[g(\boldsymbol{z}_i, \boldsymbol{z}')]} \approx \frac{g(\boldsymbol{z}_i, \boldsymbol{z}_i)}{g(\boldsymbol{z}_i, \boldsymbol{z}_i) + \frac{1}{B-1}\sum_{\substack{j=1 \\ j \neq i}}^{B} g(\boldsymbol{z}_i, \boldsymbol{z}_j)}. \tag{6}$$

During training, k=1 because $\boldsymbol{z}_i$ is sampled given $\boldsymbol{x}_i$; the expectation is approximated in-batch.

Concentration. *Intuition.* The in-batch estimate of the negative mean is well-behaved for moderate B. Since cosine similarity lies in $[-1, 1]$, $g \in [\mathrm{e}^{-1/\tau}, \mathrm{e}^{1/\tau}]$. Hoeffding's inequality implies the in-batch Monte Carlo estimate of the negative mean concentrates around its expectation:

$$\Pr\left(\left|\frac{1}{B-1}\sum_{j \neq i} g(\boldsymbol{z}_i, \boldsymbol{z}_j) - \mu\right| \geq \epsilon\right) \leq 2\exp\left(-\frac{2(B-1)\epsilon^2}{(\mathrm{e}^{1/\tau} - \mathrm{e}^{-1/\tau})^2}\right). \tag{7}$$

Relation to InfoNCE

Intuition. $-\log p_{k=1}$ is an InfoNCE-like cross-entropy with a *calibrated* positive logit that removes batch-size dependence at logit parity. Let $s_{ij} = \mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau$ so $g(\boldsymbol{z}_i, \boldsymbol{z}_j) = \exp(s_{ij})$. Then from Eq. (6),

$$p_{k=1} = \frac{\exp(s_{ii})}{\exp(s_{ii}) + \frac{1}{B-1}\sum_{j \neq i}\exp(s_{ij})} = \frac{\exp\big(s_{ii} + \log(B-1)\big)}{\exp\big(s_{ii} + \log(B-1)\big) + \sum_{j \neq i}\exp(s_{ij})}. \tag{8}$$

Thus $-\log p_{k=1}$ equals an InfoNCE cross-entropy where the positive logit is shifted by $\log(B-1)$.
Calibration. If all logits are equal, $p_{k=1} = 1/2$ (independent of B) versus $1/B$ in standard InfoNCE. With cosine similarity $s_{ii} = 1/\tau$ is constant; attraction comes from the MIM term.

cMIM Training Objective

Intuition. We optimize an A-MIM upper bound with an added contrastive term, preserving reconstruction while shaping global angular geometry. Define the mixture model

$$\mathcal{M}_\theta(\boldsymbol{x}, \boldsymbol{z}, \mathrm{k}) = \tfrac{1}{2}\big(p_\theta(\mathrm{k} \mid \boldsymbol{z}, \boldsymbol{x})\, p_\theta(\boldsymbol{x} \mid \boldsymbol{z})\, p_\theta(\boldsymbol{z}) + q_\theta(\mathrm{k} \mid \boldsymbol{z}, \boldsymbol{x})\, q_\theta(\boldsymbol{z} \mid \boldsymbol{x})\, q_\theta(\boldsymbol{x})\big), \tag{9}$$

with sampling distribution $\mathcal{M}_S(\boldsymbol{x}, \boldsymbol{z}, \mathrm{k})$ as in MIM . The learning objective upper-bounds the negative mixture entropy:

$$\mathcal{L}_{\text{MIM}}(\theta) = \tfrac{1}{2}\Big(\mathrm{CE}(\mathcal{M}_S, q_\theta) + \mathrm{CE}(\mathcal{M}_S, p_\theta)\Big) \geq H_{\mathcal{M}_S}(\boldsymbol{x}, \mathrm{k}) + H_{\mathcal{M}_S}(\boldsymbol{z}) - I_{\mathcal{M}_S}(\boldsymbol{x}, \mathrm{k}; \boldsymbol{z}). \tag{10}$$

For A-MIM (sampling along the encoding path),

$$\begin{aligned}\mathcal{L}_{\text{A-MIM}}(\theta) = -\tfrac{1}{2}\, \mathbb{E}_{\boldsymbol{x}\sim\mathcal{P}(\boldsymbol{x}),\ \boldsymbol{z}\sim q_\theta(\boldsymbol{z}|\boldsymbol{x}),\ \mathrm{k}=1}\Big[&\log p_\theta(\mathrm{k} \mid \boldsymbol{z}, \boldsymbol{x}) + \log p_\theta(\boldsymbol{x} \mid \boldsymbol{z}) + \log p_\theta(\boldsymbol{z}) \\ &+ \log q_\theta(\mathrm{k} \mid \boldsymbol{z}, \boldsymbol{x}) + \log q_\theta(\boldsymbol{z} \mid \boldsymbol{x}) + \log q_\theta(\boldsymbol{x})\Big].\end{aligned} \tag{11}$$

The empirical loss with anchor prior $p(\boldsymbol{z}) = \mathcal{N}(0, \mathbf{I})$ is

$$\hat{\mathcal{L}}_{\text{A-MIM}}(\theta; \mathcal{D}) = -\tfrac{1}{N}\sum_{i=1}^{N}\Big(\log p_\theta(\boldsymbol{x}_i \mid \boldsymbol{z}_i) + \log p_{k=1}(\boldsymbol{x}_i, \boldsymbol{z}_i) + \tfrac{1}{2}(\log q_\theta(\boldsymbol{z}_i \mid \boldsymbol{x}_i) + \log p(\boldsymbol{z}_i))\Big). \tag{12}$$

B Experiment and Training Details

To evaluate cMIM, we conduct experiments on a 2D toy example, MNIST-like images, and molecular property prediction (MolMIM by Reidenbach et al. [2023]). The toy example isolates the geometric effect of the contrastive term. For images, we examine classification, batch-size sensitivity, and reconstruction. For molecules, we assess reconstruction and downstream regression.

B.1 Experiment Details and Datasets

#	Dataset	Train Samples	Test Samples	Categories	Description
1	MNIST	60,000	10,000	10	Handwritten digits
2	Fashion MNIST	60,000	10,000	10	Clothing images
3	EMNIST Letters	88,800	14,800	27	Handwritten letters
4	EMNIST Digits	240,000	40,000	10	Handwritten digits
5	PathMNIST	89,996	7,180	9	Colon tissue histology
6	DermaMNIST	7,007	2,003	7	Skin lesion images
7	OCTMNIST	97,477	8,646	4	Retinal OCT images
8	PneumoniaMNIST	9,728	2,433	2	Pneumonia chest X-rays
9	RetinaMNIST	1,600	400	5	Retinal fundus images
10	BreastMNIST	7,000	2,000	2	Breast tumor ultrasound
11	BloodMNIST	11,959	3,432	8	Blood cell microscopy
12	TissueMNIST	165,466	47,711	8	Kidney tissue cells
13	OrganAMNIST	34,581	8,336	11	Abdominal organ CT scans
14	OrganCMNIST	13,000	3,239	11	Organ CT, central slices
15	OrganSMNIST	23,000	5,749	11	Organ CT, sagittal slices

Table 2: **Image classification datasets.** MNIST/EMNIST (rows 1–4) are handwriting; rows 5–15 are MedMNIST biomedical imaging tasks spanning pathology, retina, chest X-ray, ultrasound, and CT.

All models are trained unsupervised. We select the checkpoint with lowest validation loss (no peeking at test accuracy). For downstream tasks, we freeze the encoder–decoder and train lightweight classifiers on learned representations. Accuracy is not monitored during pretraining to avoid selection bias; training runs to convergence for fairness across models.

2D Toy Example. A synthetic dataset of 1000 points in 2D is initialized in the first quadrant. We visualize how the contrastive term in Eq. (6) shapes latent geometry.

Image Classification on MNIST-like Datasets. We train MIM, cMIM, VAE, cVAE (VAE + contrastive term), and InfoNCE to convergence on MNIST Deng [2012], EMNIST (letters, digits) Cohen et al. [2017], and MedMNIST Yang et al. [2021]. Images are resized to 28×28 and binarized when needed. We use $\tau = 0.1$ (as in InfoNCE) after a small sweep $\tau \in \{0.1, 1\}$. Encoder: Perceiver Jaegle et al. [2021] with 1 cross-attention, 4 self-attention layers, hidden size 16; 784 pixels $\to$ 400 steps $\to$ 64-dim latent. Decoder mirrors the encoder. Training: 500k steps; batch-sizes $\{2, 5, 10, 100, 200\}$; Adam lr 10^{-3}; WSD scheduler Hu et al. [2024]. Classifiers: KNN (k=5; cosine & Euclidean) and one-hidden-layer MLP (width 400, Adam 10^{-3}, 1000 steps).

Molecular Property Prediction. We use ZINC-15 Sterling and Irwin [2015] with SMILES Weininger [1988], following Reidenbach et al. [2023]. Targets: ESOL, FreeSolv, Lipophilicity. Here $\tau = 1$; both MIM and cMIM train for 250k steps. We evaluate SVM/MLP regressors with/without informative embeddings and compare to CDDD Winter et al. [2019]. Additional architectural details are below.

B.2 Image Classification

Architecture.

- Encoder: flatten $\to$ linear to $(784, 16)$ $\to$ Perceiver to $(400, 16)$ $\to$ linear to 1 $\to$ layer norm $\to$ linear to 64.
- $q_\theta(\boldsymbol{z} \mid \boldsymbol{x})$: Gaussian with mean/variance from linear heads on encoder output.
- Decoder: linear $64 \to (64, 16)$ $\to$ Perceiver $(400, 16)$ $\to$ linear to 1 $\to$ layer norm $\to$ linear to 784 $\to$ reshape to 28×28.
- $p_\theta(\boldsymbol{x} \mid \boldsymbol{z})$: Bernoulli with logits predicted from decoder output.
- Prior: standard Normal.

All models use Adam lr $1\mathrm{e}{-3}$ with WSD (10% warmup/decay) for 500k steps, independent of batch-size.

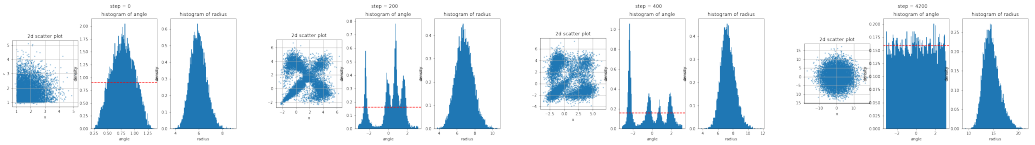
B.3 Molecular Property Prediction

Dataset. We train on a tranche of ZINC-15 [Sterling and Irwin, 2015], reactive & annotated, with molecular weight ≤ 500Da and logP ≤ 5. We select 730M molecules and split into train/val/test (723M train). To isolate framework effects, we hold architecture and hyperparameters fixed across models. For context, Chemformer used 100M molecules [Sterling and Irwin, 2015] (about 20$\times$ CDDD's 72M from ZINC-15+PubChem [Kim et al., 2018]); MoLFormer-XL used $\sim$1.1B molecules.

Data augmentation. Following Irwin et al. [2022], we use masking and SMILES enumeration [Weininger, 1988]. Masking (10%) is used only for MegaMolBART. MegaMolBART/PerBART/MolVAE apply SMILES enumeration with different encoder/decoder permutations; MolMIM improves when encoder and decoder see the *same* permutation, simplifying training.

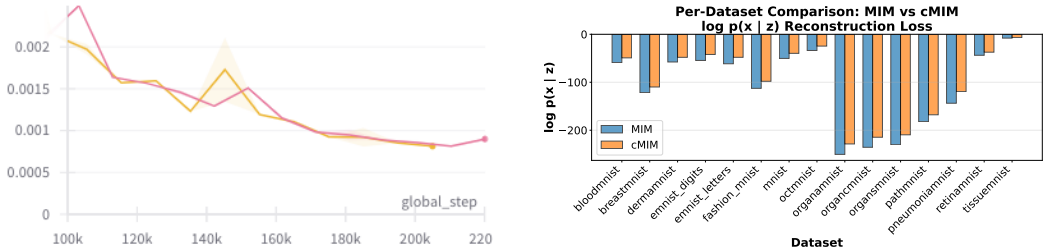
Model details. Implemented with NeMo Megatron [Kuchaiev et al., 2019]. RegEx tokenizer with 523 tokens [Bird et al., 2009]. Encoders/decoders: 6 layers, hidden size 512, 8 heads, FFN 2048. Perceiver-based models define K (hidden length) with $H = K \times D$ total hidden size (cf. Fig. 2a). Params: MegaMolBART 58.9M; PerBART 64.6M; MolVAE/MolMIM 65.2M. MolVAE uses β-VAE [Higgins et al., 2017] with $\beta = 1/D$.

Optimization. Adam [Kingma and Ba, 2015] with learning rate 1.0, betas $(0.9, 0.999)$, $\epsilon = 10^{-8}$, weight decay 0. Noam scheduler [Vaswani et al., 2017] (warm-up ratio 0.008; min lr $1\mathrm{e}{-5}$). Max sequence length 512; dropout 0.1; local batch 256; global batch 16384. Training: 1,000,000 steps, fp16, 4$\times$ nodes, 16$\times$ V100 32GB/node. MolVAE also reported with $\beta = 1/H$; we found this balances rate/distortion [Alemi et al., 2018]. MolMIM does not require β tuning.



(a) Step 0 **(b)** Step 200 **(c)** Step 400 **(d)** Step 4200

Figure 5: **Contrastive term induces angular uniformity.** Effect of Eq. (6) on a 2D toy example. Each panel shows latent space (left), angle histogram (middle), and radius histogram (right). From **(a)** initialization to **(d)** 4200 steps, cMIM distributes points uniformly in angle while allowing radii to vary, complementing MIM 's clustering and improving separability.



(a) ZINC15 validation loss **(b)** MNIST-like test log-likelihood

Figure 6: **Generative fidelity is preserved or improved.** **(a)** Molecular data: cMIM (yellow) and MIM (pink) exhibit comparable validation reconstruction loss. **(b)** Images: cMIM achieves better average per-example Bernoulli log-likelihood (-96.25) than MIM (-109.64), a 12.2% relative improvement, suggesting a benign regularization effect. (All likelihoods are averaged per example; for images we report mean Bernoulli log-likelihood over pixels.)

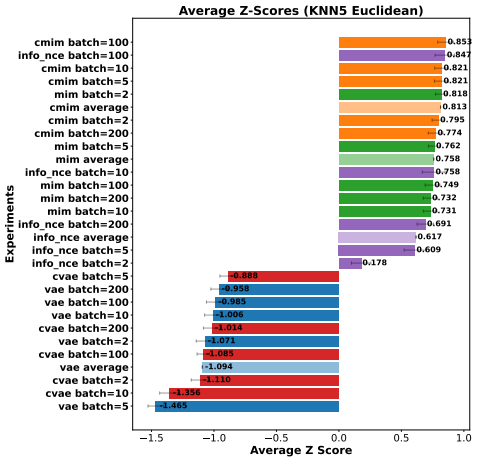
C Additional Results

C.1 Effects of cMIM Loss on 2D Toy Example

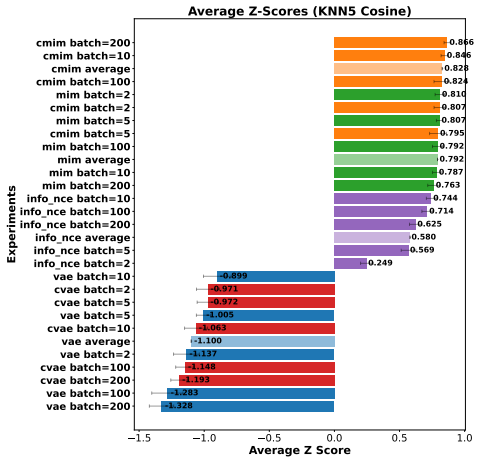
We minimize the negative log-likelihood associated with Eq. (6) using $\tau{=}1$. As expected Wang and Isola [2020], angular uniformity emerges without collapsing radii, indicating that the contrastive term integrates smoothly with the MIM objective.

C.2 Reconstruction

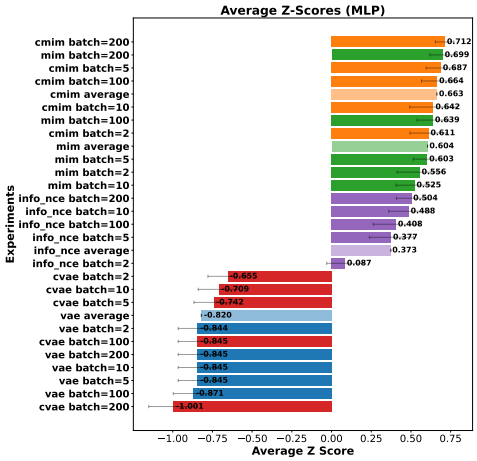
C.3 MNIST-like Image Classification



(a) KNN5 (Euclidean)

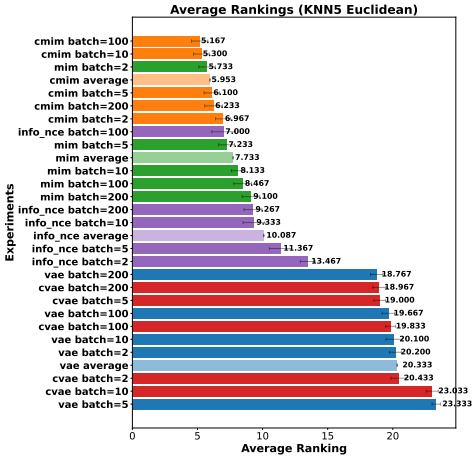


(b) KNN5 (Cosine)

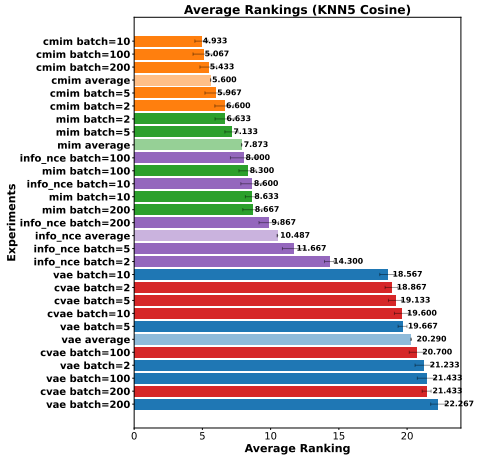


(c) MLP

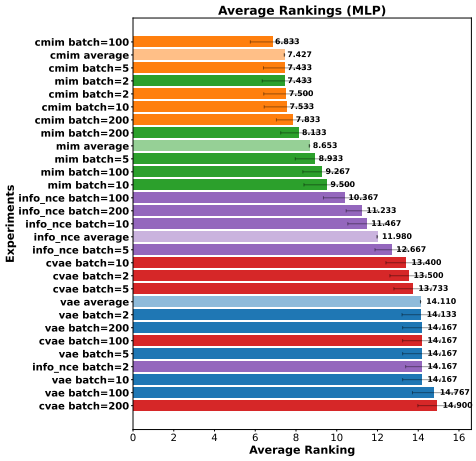
Figure 7: **Z-scores across evaluators and batch-sizes.** cMIM attains higher average z-scores than MIM/InfoNCE/VAE across KNN and MLP evaluators, with reduced variance across batch-sizes.



(a) KNN5 (Euclidean)



(b) KNN5 (Cosine)



(c) MLP

Figure 8: **Average rankings (lower is better).** Across evaluators and batch-sizes, cMIM consistently attains top rankings with narrow error bars, while InfoNCE varies markedly with batch-size.