Hellinger Multimodal Variational Autoencoders

Huyen Vo [†,‡] **Isabel Valera** [†,‡]
[†] Department of Computer Science, Saarland University, DE
[‡] MPI-SWS, Saarland Informatics Campus, DE

Abstract

Multimodal variational autoencoders (VAEs) are widely used for weakly supervised generative learning with multiple modalities. Predominant methods aggregate unimodal inference distributions using either a product of experts (PoE), a mixture of experts (MoE), or their combinations to approximate the joint posterior. In this work, we revisit multimodal inference through the lens of probabilistic opinion pooling, an optimization-based approach. We start from Hölder pooling with $\alpha = 0.5$, which corresponds to the unique symmetric member of the α-divergence family, and derive a moment-matching approximation, termed Hellinger. We then leverage such an approximation to propose HELVAE, a multimodal VAE that avoids sub-sampling, yielding an efficient yet effective model that: (i) learns more expressive latent representations as additional modalities are observed; and (ii) empirically achieves better trade-offs between generative coherence and quality, outperforming state-of-the-art multimodal VAE models.

1 INTRODUCTION

Recently, frameworks for cross-generation, such as image-to-text (Li et al., 2021) and text-to-image (Radford et al., 2021; Nichol et al., 2022), have received attention (Li et al., 2019; Zhang et al., 2021; Nichol et al., 2022; Saharia et al., 2022; Gu et al., 2022). However, these advances focus on cross-modality learning through conditional dependencies from one modality to another, often neglecting shared semantic representations across modalities. In practice, multimodal datasets are weakly supervised, as collecting complete annotations is costly, and access to all modalities at inference time is rarely guaranteed. Multimodal VAEs address this by providing a probabilistic framework that learns joint latent representations capturing shared and modality-specific structures, while also enabling cross-generation, missing-modality inference, and weakly supervised learning.

To handle missing-modality inference, early multimodal VAEs (Suzuki et al., 2016; Vedantam et al., 2018) lacked scalability, as they required a separate encoder for each subset of modalities. More recent methods (Wu and Goodman, 2018; Shi et al., 2019; Sutter et al., 2021) introduce scalable inference mechanisms that efficiently learn from many modalities using a joint encoder factorized into unimodal components. These approaches primarily follow two paradigms: product of experts (PoE), such as MVAE (Wu and Goodman, 2018), and mixture of experts (MoE), such as MMVAE (Shi et al., 2019). Both, however, have flaws in estimating the joint posterior: PoE can be biased when expert precisions are miscalibrated or collapse if any expert assigns low density, while MoE cannot yield a posterior sharper than its components, limiting efficiency in high-dimensional spaces (Hinton, 2002). Their performance is typically assessed through two criteria: generative quality, which measures how well the model fits the data distribution, and generative coherence, which evaluates semantic consistency across modalities. Thus, an effective multimodal generative model should ideally achieve both. However, in Daunhawer et al. (2021), the authors show that existing approaches struggle to balance these objectives, limiting applicability to complex real-world settings. In particular, they note that mixture-based models such as MMVAE rely on sub-sampling,[1] which imposes an undesirable upper bound on the multimodal ELBO and limits generative quality (Daunhawer et al., 2021).

In this work, we study PoE and MoE aggregation

Proceedings of the 29[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300. Correspondence to huyenvo@mpi-sws.org.

[1] Sub-sampling refers to training multimodal VAEs by randomly selecting subsets of modalities instead of using all modalities at once.

through probabilistic opinion pooling, and more specifically, Hölder pooling, which corresponds to minimizing α-divergence (Amari, 1985; Zhu and Rohwer, 1995). In this perspective, PoE can be seen as log-linear pooling (Genest, 1984) when $\alpha \to 0$ and MoE as linear pooling (Stone, 1961) when $\alpha = 1$. We then leverage the unique symmetric case of the α-divergence family, $\alpha = 0.5$, proportional to the squared Hellinger distance. This distance is a bounded metric on probability measures, defined as the L^2 norm between square-root densities (Nikulin et al., 2001). Building on this, we apply moment matching to project the pooled distribution onto a single Gaussian, allowing multimodal VAEs avoid sub-sampling. We then leverage this approximation to propose the **HEL**linger multimodal **VAE** (HELVAE), an efficient yet effective model without sub-sampling during training.

In summary, our main contributions are three-fold: (i) We introduce a probabilistic opinion pooling perspective for multimodal VAEs, which generalizes common methods such as PoE and MoE as special cases of Hölder pooling with respective α; (ii) we develop a moment-matching approximation of Hölder pooling with $\alpha = 0.5$, termed Hellinger aggregation, which is particularly effective when combining different types of experts; and, building on this, (iii) we then propose HELVAE, a multimodal VAE that provides an efficient multimodal VAE that balances generative coherence and quality. Our empirical results on three benchmark datasets (PolyMNIST, CUB Image-Captions, and bimodal CelebA), show that HELVAE learns better latent representations and shows synergy [2] across modalities, leading to better trade-offs between generative coherence and quality than existing methods.

2 RELATED WORK

Extending VAEs (Kingma and Welling, 2013) to multiple data modalities, early multimodal VAEs (Suzuki et al., 2016; Vedantam et al., 2018) lacked scalability. More recent approaches address this using PoE (Wu and Goodman, 2018), MoE (Shi et al., 2019), or their combination, the mixture of product of experts (MoPoE) (Sutter et al., 2021). Building on this line, later work has introduced WBVAE (Qiu et al., 2025), which leverages Wasserstein barycenter aggregation, and CoDEVAE (Mancisidor et al., 2025), which defines joint distributions through a mixture of consensus among dependent experts (CoDE). Although CoDEVAE claims it does not require sub-sampling, since all loss terms for all subsets are trained rather than some as in mixture-based methods, CoDEVAE still incurs a relatively high computational cost of $\mathcal{O}(2^M - 1)$, similar to MoPoE and MWBVAE, where M is the number of modalities. Beyond these approaches, flexible aggregation methods based on permutation-invariant neural networks (Hirt et al., 2024) have also been proposed.

In this context, Daunhawer et al. (2021) show that multimodal VAEs struggle with a trade-off between generative coherence and quality. Javaloy et al. (2022) further identify modality collapse as a consequence of conflicting gradients during training. Since then, performance has been improved by regularization objectives such as mmJSD (Sutter et al., 2020), which aligns unimodal and joint posteriors via Jensen–Shannon divergence, and MVTCAE (Hwang et al., 2021), which captures cross-view dependencies via total correlation. In addition, hierarchical latent spaces have been leveraged to capture heterogeneity among modalities (Vasco et al., 2022; Wolff et al., 2022). Another line of work has focused on dynamic priors and posteriors, using unimodal posteriors as priors (Sutter et al., 2020; Joy et al., 2022), energy-based priors (Yuan et al., 2024), variational mixtures of posteriors (Vamp) priors (Sutter et al., 2024), score-based priors (Wesego and Rooshenas, 2024b), Markov random field (Oubari et al., 2024), and flow-based posteriors (Senellart and Allassonnière, 2025). More recently, diffusion models have been used in multimodal VAEs, either at the decoder or the latent level (Wesego and Rooshenas, 2024a; Bounoua et al., 2024; Palumbo et al., 2024). Finally, prior work has considered separating modality-specific latent subspaces alongside a shared subspace (Sutter et al., 2020; Lee and Pavlovic, 2021; Palumbo et al., 2023), with MMVAE+ (Palumbo et al., 2023) as the state-of-the-art model that introduces a new objective for mixture-based models.

While these approaches consider different aspects of multimodal representation learning, they generally use the two most common combination schemes and are orthogonal to this work, which specifically focuses on the modality aggregation approach. We compare our model with recent aggregation-based methods, as well as MMVAE+ with additional modality-specific latent variables, and show that it achieves comparable generative performance without requiring extra complexity.

3 PRELIMINARIES

3.1 Multimodal VAEs

We consider multimodal data $\mathbb{X}$, comprising M modalities $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_M$, which are assumed to be generated from a shared latent variable $\boldsymbol{z}$ using modality-specific parametric decoders $\{p_{\theta_j}(\boldsymbol{x}_j|\boldsymbol{z})\}_{j=1}^M$. Assum-

[2] Synergy (Shi et al., 2019) is the effect that learning from multiple modalities improves generative performance in each modality.

ing conditional independence between modalities given $\boldsymbol{z}$, the joint distribution factorizes as: $p_\theta(\mathbb{X}, \boldsymbol{z}) = p(\boldsymbol{z}) \prod_{j=1}^{M} p_{\theta_j}(\boldsymbol{x}_j|\boldsymbol{z})$, where $p(\boldsymbol{z})$ is the prior over the latent space. The objective of a multimodal VAE is to maximize the log-likelihood of data over M modalities:

$$\log p_\theta(\mathbb{X}) = \mathrm{KL}(q_\phi(\boldsymbol{z}|\mathbb{X}) \| p_\theta(\boldsymbol{z}|\mathbb{X})) + \mathcal{L}(\theta, \phi; \mathbb{X}),$$

$$\mathcal{L}(\theta, \phi; \mathbb{X}) = \mathbb{E}_{q_\phi(\boldsymbol{z}|\mathbb{X})}[\log p_\theta(\mathbb{X}|\boldsymbol{z})] - \mathrm{KL}(q_\phi(\boldsymbol{z}|\mathbb{X}) \| p(\boldsymbol{z})).$$

The distribution $q_\phi(\boldsymbol{z}|\mathbb{X})$ is a variational posterior approximation with learnable parameters ϕ, since the true posterior is intractable. From the non-negativity of the KL divergence, yielding $\log p_\theta(\mathbb{X}) \geq \mathcal{L}(\theta, \phi; \mathbb{X})$. The term $\mathcal{L}(\theta, \phi; \mathbb{X})$ is the evidence lower bound (ELBO) on the marginal log-likelihood, and we maximize this bound instead. Maximizing the ELBO requires approximating the joint posterior $q_\phi(\boldsymbol{z}|\mathbb{X})$ in a way that scales flexibly to many modalities. Two common approaches are PoE (Wu and Goodman, 2018) and MoE (Shi et al., 2019), where the joint distributions are approximated as $q_\phi(\boldsymbol{z}|\mathbb{X}) = c \prod_{j=1}^{M} q_{\phi_j}(\boldsymbol{z}|\boldsymbol{x}_j)$ and $q_\phi(\boldsymbol{z}|\mathbb{X}) = \frac{1}{M} \sum_{j=1}^{M} q_{\phi_j}(\boldsymbol{z}|\boldsymbol{x}_j)$, respectively, with c a normalization constant that guarantees the validity of the resulting probability measure.

3.2 Probabilistic opinion pooling: an optimization-based approach

Probabilistic opinion pooling is a principled framework for aggregating multiple probability density functions (pdfs) into a single consensus pdf. Let $\boldsymbol{z} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_D)^\top \in \mathbb{R}^D$ be a continuous random variable. Given a set of pdfs $\{q_j(\boldsymbol{z})\}_{j=1}^{M} \in \mathcal{P}^M$ with associated non-negative weights $\{\lambda_j\}_{j=1}^{M}$ satisfying $\sum_{j=1}^{M} \lambda_j = 1$, a pooled density is defined as a weighted aggregation of these pdfs. A popular pooling function is linear pooling (Stone, 1961), which computes a weighted arithmetic average: $q(\boldsymbol{z}) = \sum_{j=1}^{M} \lambda_j q_j(\boldsymbol{z})$. Another popular one is the log-linear pooling function (Genest, 1984), which takes a weighted geometric average: $q(\boldsymbol{z}) = c \prod_{j=1}^{M} (q_j(\boldsymbol{z}))^{\lambda_j}$ where c is a normalization factor.

In an optimization-based approach, the pooling function is obtained by minimizing a weighted average of some discrepancy measure between the aggregated pdf and the individual pdfs. One class of discrepancy measures that we consider in this paper is the family of f-divergences. Given a convex function $f : \mathbb{R}^+ \to \mathbb{R}$ satisfying $f(1) = 0$, the f-divergence between two pdfs $q_j(\boldsymbol{z})$ and $\varphi(\boldsymbol{z})$ over the domain $\mathbb{R}^D$ is defined as:

$$\mathcal{D}_f(q_j \| \varphi) = \int \varphi(\boldsymbol{z}) f\left(\frac{q_j(\boldsymbol{z})}{\varphi(\boldsymbol{z})}\right) d\boldsymbol{z}.$$

The aggregated pdf $q(\boldsymbol{z})$ is obtained by solving the following minimization problem:

$$q(\boldsymbol{z}) = \arg\min_{\varphi \in \mathcal{P}} \sum_{j=1}^{M} \lambda_j \mathcal{D}_f(q_j \| \varphi). \tag{1}$$

As stated in Abbas (2009); Qiu et al. (2025), log-linear pooling and linear pooling can be derived as solutions to the minimization problem in Equation 1, corresponding respectively to the reverse and forward KL divergences, which are both members of the f-divergence family: $f(x) = -\log x$ for the reverse KL and $f(x) = x \log x$ for the forward KL. We generalize these two special cases to an entire family of divergences and corresponding pooling functions, both parameterized by a real-valued parameter α. Specifically, we consider $f(x) = \frac{x^\alpha - 1}{\alpha(\alpha-1)}$ for $x > 0$ and $\alpha \in \mathbb{R} \backslash \{0, 1\}$. This yields the family of α-divergences defined as:

$$\mathcal{D}_\alpha(q_j \| \varphi) = \frac{1}{\alpha(\alpha-1)} \int \varphi(\boldsymbol{z}) \frac{(q_j(\boldsymbol{z}))^\alpha - (\varphi(\boldsymbol{z}))^\alpha}{(\varphi(\boldsymbol{z}))^\alpha} d\boldsymbol{z}.$$

The solution to the problem in Equation 1 for the α-divergences is an α-parameterized family of Hölder pooling functions, as mentioned in Garg et al. (2004); Koliander et al. (2022):

$$q(\boldsymbol{z}) = c \left(\sum_{j=1}^{M} \lambda_j (q_j(\boldsymbol{z}))^\alpha \right)^{1/\alpha},$$

where $c = 1 / \int \left(\sum_{j=1}^{M} \lambda_j (q_j(\boldsymbol{z}))^\alpha \right)^{1/\alpha} d\boldsymbol{z}$. In the limit $\alpha \to 0$, the Hölder pooling function reduces to the log-linear pooling function, and for $\alpha = 1$, it becomes the linear pooling function, consistent with the facts that $\lim_{\alpha \to 0} \mathcal{D}_\alpha(q_j \| \varphi) = \mathrm{KL}(\varphi \| q_j)$ (reverse KL) and $\lim_{\alpha \to 1} \mathcal{D}_\alpha(q_j \| \varphi) = \mathrm{KL}(q_j \| \varphi)$ (forward KL) (Minka et al., 2005). It is worth noting that the above choices of α yield asymmetric and unbounded discrepancy measures, which do not define a metric space for probability measures. In what follows, we consider the unique symmetric case of the α-divergence family, corresponding to $\alpha = 0.5$:

$$\begin{aligned} \mathcal{D}_{0.5}(q_j \| \varphi) &= 2 \int \left(\sqrt{q_j(\boldsymbol{z})} - \sqrt{\varphi(\boldsymbol{z})} \right)^2 d\boldsymbol{z} \\ &= 4\mathrm{Hel}^2(q_j \| \varphi), \end{aligned}$$

where $\mathrm{Hel}(q_j \| \varphi)$ is the Hellinger distance (Nikulin et al., 2001) between two distributions, symmetric and bounded between 0 and 1 (Tsybakov, 2009).

The use of α-divergence with $\alpha = 0.5$ was introduced in Black-Box α (BB-α) (Hernandez-Lobato et al., 2016), an approximate inference method based on the minimization of α-divergence. Their experiments

demonstrate that BB-α with non-standard settings of α, such as $\alpha = 0.5$, often yields better predictive performance compared to $\alpha \to 0$ variational Bayes (VB) or $\alpha = 1$ expectation propagation (EP). This motivates our use of α-divergence with $\alpha = 0.5$ in Hölder pooling, which is defined as:

$$q(\boldsymbol{z}) = c \left(\sum_{j=1}^{M} \lambda_j \sqrt{q_j(\boldsymbol{z})} \right)^2 \tag{2}$$
$$= c \left(\sum_{j=1}^{M} \lambda_j^2 q_j(\boldsymbol{z}) + 2 \sum_{i=1}^{M} \sum_{j>i}^{M} \lambda_i \lambda_j \sqrt{q_i(\boldsymbol{z}) q_j(\boldsymbol{z})} \right),$$

where $c = 1/\int \left(\sum_{j=1}^{M} \lambda_j \sqrt{q_j(\boldsymbol{z})} \right)^2 d\boldsymbol{z}$.

We note that $q(\boldsymbol{z})$ takes the form of a squared mixture model, closely related to recent squared subtractive mixture formulations, which have been shown to provide increased expressiveness for density estimation tasks (Loconte et al., 2024, 2025, 2026). This suggests that exploring negative pooling weights could be a promising direction for future work.

4 METHODOLOGY

By applying the probabilistic opinion pooling view to multimodal VAEs, we aim to design a pooling function that aggregates unimodal posteriors to approximate the true joint posterior. MVAE (Wu and Goodman, 2018) employs a PoE, corresponding to log-linear pooling, which yields sharp approximations but struggles to optimize the individual experts, as noted by the authors. In contrast, MMVAE (Shi et al., 2019) adopts a MoE, corresponding to linear pooling, which optimizes experts effectively but cannot produce a sharper posterior than its components. Thus, the choice of pooling function directly shapes the properties of the learned model. Moreover, as highlighted by Winkler (1981); Dietrich and List (2016); Mancisidor et al. (2025), both PoE and MoE depend on an overoptimistic independence assumption between experts for simplicity. In practice, this assumption rarely holds, as data modalities are different sources of information about the same object. By contrast, Hölder pooling with $\alpha = 0.5$ introduces cross terms such as $\sqrt{q_i(\boldsymbol{z}) q_j(\boldsymbol{z})}$, which naturally induce soft dependencies between experts by amplifying regions of mutual support.

To illustrate, we consider three Gaussian experts with diagonal covariance $\{\mathcal{N}(\boldsymbol{\mu}_j, \mathrm{diag}(\boldsymbol{\sigma}_j^2))\}_{j=1}^3$, expert 1 with $\boldsymbol{\mu}_1 = (0,0)^\top, \boldsymbol{\sigma}_1^2 = (0.5, 0.5)^\top$, and expert 2 with $\boldsymbol{\mu}_2 = (1.0, 0.2)^\top, \boldsymbol{\sigma}_2^2 = (0.6, 0.6)^\top$, close to each other, while expert 3 with $\boldsymbol{\mu}_3 = (4,0)^\top, \boldsymbol{\sigma}_3^2 = (0.2, 0.2)^\top$, is sharp and far away. As shown in Figure 1, PoE shifts toward the sharpest expert, while MoE spreads mass

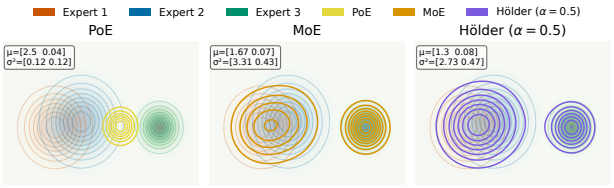


Figure 1: PoE, MoE, and Hölder pooling ($\alpha = 0.5$) with two agreeing experts and one disagreeing sharp expert.

across all experts and fails to capture the local agreement between experts 1 and 2. In contrast, Hölder pooling with $\alpha = 0.5$ overcomes this by placing more mass around the first two experts, yielding a sharper distribution with reduced variance compared to MoE. This shows how Hölder pooling better reflects multimodal structure than either PoE or MoE.

In the multimodal VAE setting, each unimodal posterior (expert) is modeled as a multivariate Gaussian with diagonal covariance matrix, which we propose to aggregate using the Hellinger function. This function is derived from Hölder pooling with $\alpha = 0.5$ and uses moment matching (Bowman and Shenton, 2004) to project the pooled density onto a diagonal Gaussian. In the next part, we derive this aggregation function and then use it in the proposed HELVAE.

Hellinger aggregation. Let unimodal inference distributions $\{q_{\phi_j}\}_{j=1}^M$ be explicitly given as Gaussian probability densities on $\mathbb{R}^D$ with diagonal covariance matrices, $\{\mathcal{N}(\boldsymbol{\mu}_j, \mathrm{diag}(\boldsymbol{\sigma}_j^2))\}_{j=1}^M$, where $\boldsymbol{\mu}_j = (\mu_{j,1}, \mu_{j,2}, \ldots, \mu_{j,D})^\top \in \mathbb{R}^D$ and $\boldsymbol{\sigma}_j^2 = (\sigma_{j,1}^2, \sigma_{j,2}^2, \ldots, \sigma_{j,D}^2)^\top \in \mathbb{R}^D$. For each pair of modalities (i,j) with $1 \le i < j \le M$, we then define the auxiliary parameters $\boldsymbol{\mu}_{ij}, \boldsymbol{\sigma}_{ij}^2 \in \mathbb{R}^D$ by specifying their corresponding dimension $d \in \{1, 2, \ldots, D\}$ as follows:

$$\mu_{ij,d} = \frac{\mu_{i,d}\, \sigma_{j,d}^2 + \mu_{j,d}\, \sigma_{i,d}^2}{\sigma_{i,d}^2 + \sigma_{j,d}^2}, \quad \sigma_{ij,d}^2 = \frac{2\sigma_{i,d}^2 \sigma_{j,d}^2}{\sigma_{i,d}^2 + \sigma_{j,d}^2}.$$

The Bhattacharyya coefficient [3] between q_{ϕ_i} and q_{ϕ_j} is given by

$$S_{ij} = \prod_{d=1}^{D} \sqrt{\frac{2\sigma_{i,d}\sigma_{j,d}}{\sigma_{i,d}^2 + \sigma_{j,d}^2}} \exp\left(-\frac{(\mu_{i,d} - \mu_{j,d})^2}{4(\sigma_{i,d}^2 + \sigma_{j,d}^2)} \right),$$

which allows us to compute the the normalizing constant as:

$$c = M + 2 \sum_{i=1}^{M} \sum_{j>i}^{M} S_{ij}.$$

Finally, we approximate the joint posterior resulting from Hölder pooling with $\alpha = 0.5$ in Equation 2 via

[3] The Bhattacharyya coefficient measures the overlap between two probability distributions, taking values in $[0, 1]$, with 1 indicating identical distributions.

moment matching as a D-dimensional Gaussian distribution, $q_\phi(\boldsymbol{z}|\mathbb{X}) \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\Sigma}})$, with mean $\tilde{\boldsymbol{\mu}} \in \mathbb{R}^D$ and covariance matrix $\tilde{\boldsymbol{\Sigma}} \in \mathbb{R}^{D\times D}$, corresponding to the first moment and the second central moment of the Hölder-pooled density, i.e.,

$$\tilde{\boldsymbol{\mu}} = \int \boldsymbol{z} q(\boldsymbol{z}) d\boldsymbol{z}, \quad \tilde{\boldsymbol{\Sigma}} = \int \boldsymbol{z}\boldsymbol{z}^\top q(\boldsymbol{z}) d\boldsymbol{z} - \tilde{\boldsymbol{\mu}}\tilde{\boldsymbol{\mu}}^\top.$$

In VAE models, the latent variable is typically parameterized as a multivariate Gaussian with diagonal covariance (Kingma and Welling, 2013). Thus, we simplify $\tilde{\boldsymbol{\Sigma}}$ by removing off-diagonal terms, yielding $\tilde{\boldsymbol{\Sigma}} \approx \text{diag}(\tilde{\sigma}_1^2, \tilde{\sigma}_2^2, \ldots, \tilde{\sigma}_D^2)$. As a result, we compute the mean and the variance of each dimension $d \in \{1, 2, \ldots, D\}$ of the latent vector $\mathbf{z}$ as follows:

$$\tilde{\mu}_d = \frac{1}{c}\left(\sum_{j=1}^{M} \mu_{j,d} + 2\sum_{i=1}^{M}\sum_{j>i}^{M} \mu_{ij,d} S_{ij}\right),$$

$$\tilde{\sigma}_d^2 = \frac{1}{c}\Bigg(\sum_{j=1}^{M}(\mu_{j,d}^2 + \sigma_{j,d}^2) + 2\sum_{i=1}^{M}\sum_{j>i}^{M}(\mu_{ij,d}^2 + \sigma_{ij,d}^2) S_{ij}\Bigg) - (\tilde{\mu}_d)^2.$$

The derivation above presents the moment-matching approximation of Hölder pooling with $\alpha = 0.5$ for a set of multivariate Gaussians with diagonal covariance. The complete derivation is given in Appendix A, where it is derived under the assumption of subset weights. Since determining these weights in multimodal VAEs is generally nontrivial, we follow PoE and MoE and set them *uniformly*. *Hellinger aggregation defines a novel multimodal VAE, called HELVAE.* An important property of HELVAE is that it does not rely on subsampling, which negatively affects likelihood estimation and generative quality in multimodal VAEs. As shown by our experiments in Section 5, HELVAE leads to better trade-offs between generation quality and coherence than MVAE and MMVAE, yielding competitive or even better results than more complex approaches such as the state-of-the-art, MMVAE+.

Illustrative example. We construct a synthetic setup to evaluate PoE, MoE, Hölder ($\alpha = 0.5$), and Hellinger in Figure 2. The set of experts is defined as:

$$q_i(\boldsymbol{z}) = \begin{cases} \mathcal{N}(\boldsymbol{z} \mid (0,0)^\top, 0.5\boldsymbol{I}), & i \in \mathcal{G} \text{ (good experts)}, \\ \mathcal{N}(\boldsymbol{z} \mid (4,4)^\top, 0.2\boldsymbol{I}), & i \in \mathcal{B} \text{ (bad experts)}, \end{cases}$$

and the true distribution is $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. Evaluation metrics include (i) expected negative log-likelihood (NLL), estimated by Monte Carlo samples from the true distribution; (ii) the Bhattacharyya coefficient with respect to the true distribution; and (iii) sharpness, measured as the trace of the covariance of the Gaussian

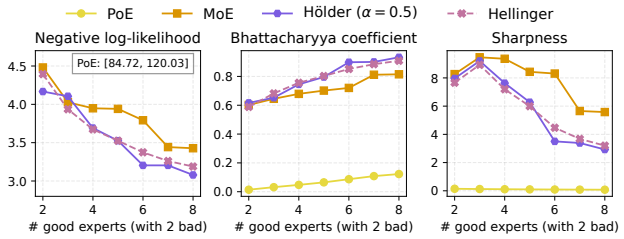


Figure 2: Performance of PoE, MoE, Hölder pooling ($\alpha = 0.5$), and Hellinger aggregators as a function of the number of good experts (with two bad experts fixed). Evaluation is based on negative log-likelihood (↓), the Bhattacharyya coefficient (↑), and sharpness (↓). We show the minimum and maximum NLL values for PoE.

approximation. Since MoE and Hölder ($\alpha = 0.5$) do not yield Gaussian densities in closed form, all metrics are approximated by sampling. The results show that PoE performs poorly, with high NLL and low Bhattacharyya coefficient, as it is biased toward the sharper bad expert. Other models maintain low NLL. Compared to MoE, Hölder ($\alpha = 0.5$) and Hellinger yield lower NLL, sharper distributions and stronger distributional overlap with the true distribution as the number of good experts increases, demonstrating their ability to capture dependencies among good experts.

Mixture of HELVAEs. Following MoPoE (Sutter et al., 2021), we introduce a variant of HELVAE that constructs a mixture of Hölder poolings with $\alpha = 0.5$, termed MoHELVAE. Specifically, we consider the powerset of M modalities $\mathcal{P}(\mathbb{X})$, consisting of all $2^M - 1$ non-empty subsets. For each subset $\mathbb{X}_k \in \mathcal{P}(\mathbb{X})$, we define its posterior approximation as:

$$\tilde{q}_\phi(\boldsymbol{z}|\mathbb{X}_k) = \text{HEL}(\{q_{\phi_j}(\boldsymbol{z}|\boldsymbol{x}_j), \forall \boldsymbol{x}_j \in \mathbb{X}_k\}),$$

then the joint posterior of MoHELVAE is given by

$$q_\phi(\boldsymbol{z}|\mathbb{X}) = \frac{1}{2^M - 1}\sum_{\mathbb{X}_k \in \mathcal{P}(\mathbb{X})} \tilde{q}_\phi(\boldsymbol{z}|\mathbb{X}_k).$$

The proposed MoHELVAE model outperforms state-of-the-art approaches in terms of latent representation and generative coherence on the challenging CelebA dataset, as demonstrated in Ablation study 6.1.

5 EXPERIMENTAL RESULTS

Following prior work in this domain (Sutter et al., 2021; Daunhawer et al., 2021; Palumbo et al., 2023), we evaluate our approach on three benchmark datasets: PolyMNIST (Sutter et al., 2021), Caltech Birds (CUB) Image-Captions (Wah et al., 2011), and bimodal CelebA (Liu et al., 2015). PolyMNIST is a synthetic dataset with five modalities, where each sample consists of MNIST digits of the same label patched

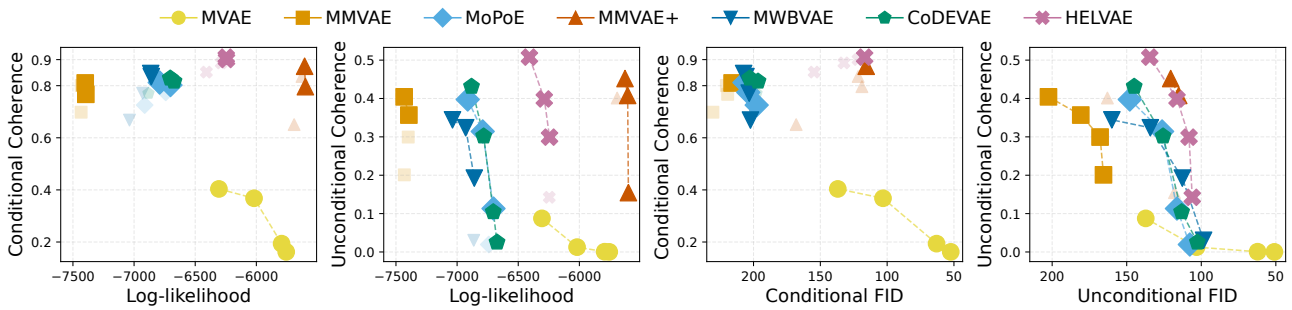


Figure 3: Trade-offs on the PolyMNIST dataset between generative coherence (↑) and log-likelihood estimation (↑), as well as between generative coherence and generative quality (↓), for $\beta \in \{1, 2.5, 5, 10\}$. For each model, the Pareto front (dashed line) connects the non-dominated points that achieve the best trade-offs.

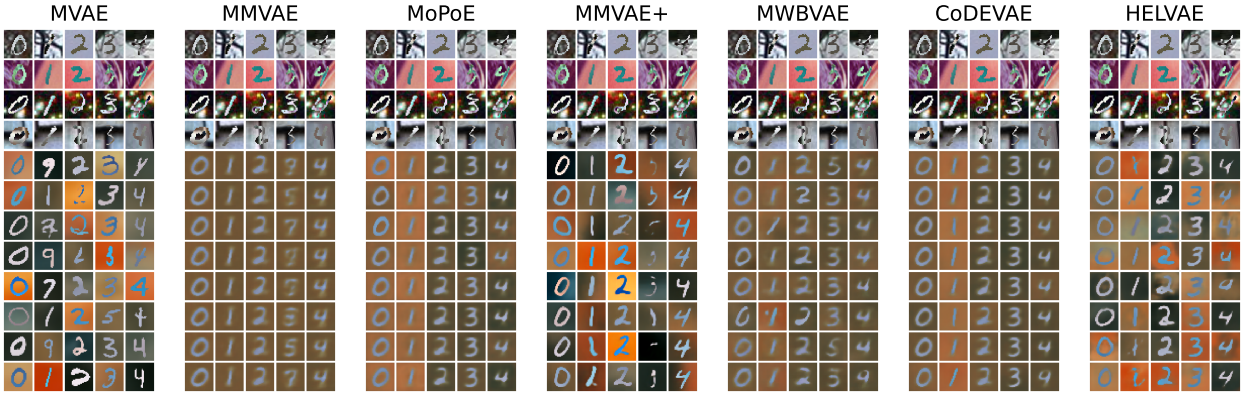


Figure 4: Conditionally generated samples of the second modality (fifth to last rows), given the corresponding test examples from the other four modalities (first four rows) on PolyMNIST, with each model evaluated at its best $\beta \in \{1, 2.5, 5, 10\}$.

onto random crops from five distinct background images. CUB Image-Captions consists of bird images paired with textual descriptions. The bimodal CelebA dataset consists of human face images paired with text describing facial attributes.

We compare the performance of HELVAE with MVAE (Wu and Goodman, 2018), MMVAE (Shi et al., 2019), MoPoE (Sutter et al., 2021), MMVAE+ (Palumbo et al., 2023), MWBVAE (Mixture of WBVAE) (Qiu et al., 2025), and CoDEVAE (Mancisidor et al., 2025). Among these, MMVAE+ is the only model that incorporates both modality-specific and shared latent variables, while the others focus on aggregating unimodal posterior distributions.

We evaluate performance based on generative coherence and generative quality (measured using FID (Heusel et al., 2017)), using samples generated from both the joint posterior and the prior. To assess the approximation quality of the joint posterior, we estimate the log-likelihood, which provides a tightness lower bound. We further evaluate the quality of the joint latent representation $\boldsymbol{z}$ using a linear classifier. All experiments report the average performance over three different seeds. Technical details and further results are provided in Appendix B.

5.1 Trade-offs between generative coherence and generative quality

PolyMNIST. Figure 3 shows the trade-offs between generative coherence and log-likelihood estimation, as well as between generative coherence and quality across all β values [4]. MVAE, based on PoE, yields tight log-likelihood estimates but performs poorly in coherence. In contrast, mixture-based models such as MMVAE, MoPoE, MWBVAE, and CoDEVAE achieve relatively high coherence, though not the best overall. Meanwhile, MMVAE+ and HELVAE both achieve better trade-offs: HELVAE attains higher coherence in both conditional and unconditional settings; while MMVAE+ provides a tighter log-likelihood bound, which is a direct result of the modality-specific variables. While generative coherence is important, the generated modalities should also be of high quality. Following the same pattern, MVAE generates high-quality images but lacks coherence. HELVAE achieves generative quality comparable to MMVAE+ while attaining better coherence. The other models yield lower-quality images due to sub-sampling of modalities during training. Qualitative results in Figure 4

[4] β-VAE (Higgins et al., 2017) scales the KL term to balance reconstruction and disentanglement.

Table 1: Generative coherence and quality on the bimodal CelebA dataset, with each model evaluated at its best $\beta \in \{1, 2.5, 5\}$, and rankings are reported accordingly.

	Conditional		Unconditional	
	Coherence (↑)	FID (↓)	Coherence (↑)	FID (↓)
MVAE	0.315 ± 0.013 (7)	**75.62 ± 3.01 (1)**	0.199 ± 0.021 (6)	**78.64 ± 2.73 (1)**
MMVAE	0.370 ± 0.007 (6)	95.10 ± 11.68 (6)	0.209 ± 0.011 (4)	91.04 ± 5.82 (6)
MoPoE	0.376 ± 0.005 (4)	81.74 ± 2.85 (3)	0.219 ± 0.006 (2)	82.50 ± 2.45 (3)
MMVAE+	0.382 ± 0.013 (2)	99.72 ± 5.23 (7)	**0.258 ± 0.024 (1)**	101.85 ± 5.13 (7)
MWBVAE	0.376 ± 0.003 (4)	88.28 ± 1.58 (4)	0.197 ± 0.006 (7)	88.74 ± 1.09 (5)
CoDEVAE	0.378 ± 0.004 (3)	91.24 ± 11.27 (5)	0.219 ± 0.020 (2)	88.15 ± 6.68 (4)
HELVAE	**0.386 ± 0.003 (1)**	80.17 ± 1.52 (2)	0.204 ± 0.013 (5)	81.89 ± 1.56 (2)

Table 2: Caption-to-image conditional generative coherence and quality on the CUB Image-Captions dataset, with each model evaluated at its best $\beta \in \{1, 2.5, 5\}$, and rankings for each metric are reported accordingly.

	Conditional	
	Coherence (↑)	FID (↓)
MVAE	0.233 ± 0.038 (7)	**148.69 ± 1.97 (1)**
MMVAE	0.688 ± 0.054 (5)	225.77 ± 1.66 (7)
MoPoE	0.675 ± 0.084 (6)	202.19 ± 4.68 (6)
MMVAE+	0.720 ± 0.090 (3)	164.94 ± 1.50 (3)
MWBVAE	0.704 ± 0.079 (4)	196.42 ± 5.58 (5)
CoDEVAE	**0.750 ± 0.050 (1)**	175.97 ± 0.30 (4)
HELVAE	**0.750 ± 0.105 (1)**	157.56 ± 0.95 (2)

Table 3: Classification accuracy based on the latent representation learned from the bimodal CelebA dataset, with each model evaluated at its best $\beta \in \{1, 2.5, 5\}$. We report the mean average precision over all attributes (I: Image; T: Text; Joint: I and T), and rankings accordingly.

	Latent Representation (↑)		
	I	T	Joint
MVAE	0.326 (6)	0.327 (7)	0.349 (6)
MMVAE	0.360 (5)	0.410 (4)	0.369 (5)
MoPoE	0.362 (3)	0.398 (6)	0.398 (4)
MMVAE+	0.314 (7)	**0.461 (1)**	0.346 (7)
MWBVAE	0.361 (4)	0.415 (3)	**0.410 (1)**
CoDEVAE	**0.365 (1)**	0.418 (2)	0.399 (3)
HELVAE	**0.365 (1)**	0.400 (5)	0.408 (2)

show that HELVAE achieves a better balance between coherence and quality compared to the other methods.

CUB Image-Captions. Table 2 reports quantitative results for caption-to-image generation, where conditional coherence is computed following Palumbo et al. (2023), since we do not have annotations for the shared content between modalities. Results from MVAE, MMVAE, and MoPoE highlight the difficulty of the task with real images: MVAE produces reasonable image quality but lacks coherence, while MMVAE and MoPoE suffer from modality collapse in mixture-based models. In contrast, our model achieves the highest conditional coherence, which is the same as CoDEVAE but with lower conditional FID than other models. HELVAE ranks second in FID and outperforms MMVAE+. These results confirm the effectiveness of our approach in real-world multimodal settings with substantial modality-specific information and significant heterogeneity across modalities.

Figure 5 shows qualitative results for caption-to-image generation. As we can see, although MVAE achieves high-quality images, these images do not follow the captions well and are noisy, resulting in low coherence. MMVAE, MWBVAE, and CoDEVAE often collapse modality-specific features to average values, yielding blurred images. In contrast, HELVAE and MMVAE+ avoid this collapse, producing images that are both high quality and coherent with the captions. Although HELVAE images are less diverse due to the lack of a modality-specific latent space, its generated images are well-shaped and align closely with the captions.

Bimodal CelebA. As shown in Table 1, HELVAE achieves the highest conditional coherence while maintaining generative quality (second-best FID). By contrast, MMVAE+ attains the highest unconditional coherence but suffers from high FID. CoDEVAE and MoPoE achieve the second-best unconditional coherence; however, CoDEVAE has high FID, and MoPoE underperforms HELVAE on other metrics. Consistent with the previous dataset, MVAE has the best generative quality but at high degradation in coherence.

5.2 Latent representation quality

The quality of the learned representations serves as a proxy for their usefulness in downstream tasks. From Table 3, when evaluating latent representation accuracy on the CelebA dataset, HELVAE shares the highest score on the image modality with CoDEVAE. Although it ranks fifth on the text modality, it achieves the second-best performance in the joint setting, show-

(a) MVAE (b) MMVAE (c) MoPoE (d) MMVAE+ (e) MWBVAE (f) CoDEVAE (g) HELVAE

Figure 5: Qualitative results for caption-to-image generation on the CUB Image-Captions dataset, with each model evaluated at its best $\beta \in \{1, 2.5, 5\}$, where the input caption is used to conditionally generate four images per model.

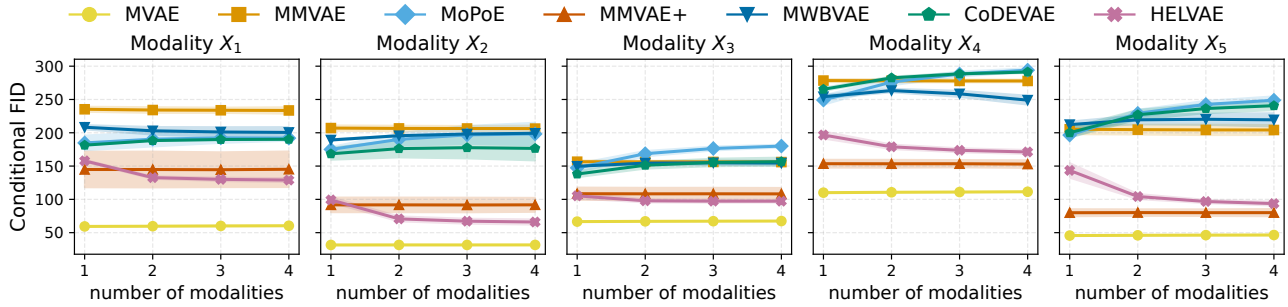


Figure 6: Conditional generative quality (↓) on the PolyMNIST dataset for target modalities $X_1 \dots X_5$ as a function of the number of input modalities, averaged across subsets of a given size excluding the target. Markers indicate means, and shaded regions represent standard deviations.

Table 4: Model performance of HELVAE and MoHELVAE on the bimodal CelebA dataset. Evaluation is based on generative coherence (Coh ↑), generative quality (FID ↓), and latent representation accuracy (LR ↑).

		HELVAE	MoHELVAE
Coh	Conditional	0.386 ± 0.003	**0.394 ± 0.011**
	Unconditional	0.204 ± 0.013	**0.229 ± 0.020**
FID	Conditional	**80.17 ± 1.52**	82.28 ± 4.72
	Unconditional	**81.89 ± 1.56**	83.14 ± 4.87
LR	Image	0.365 ± 0.004	**0.379 ± 0.013**
	Text	0.400 ± 0.004	**0.441 ± 0.019**
	Image & Text	0.408 ± 0.005	**0.430 ± 0.018**

ing a gain over its single-modality performance. In contrast, MWBVAE, CoDEVAE, and MMVAE+ do not benefit from additional modalities in the same way. This is an important criterion in multimodal generative models, known as synergy (Shi et al., 2019).

6 ABLATION STUDIES

6.1 Effectiveness of Mixture of HELVAEs

From Table 4, MoHELVAE surpasses HELVAE in generative coherence (conditional and unconditional) and latent representation accuracy. However, its FID is slightly higher than that of HELVAE, and representation accuracy drops when combining image and text, a pattern also observed in other mixture-based models on the bimodal CelebA dataset. Compared with Tables 1 and 3, MoHELVAE outperforms other models and, in particular, achieves a better balance between coherence and quality than MMVAE+, highlighting the efficiency of incorporating the joint posterior across all subsets of modalities.

6.2 Effect of the number of input modalities on model performance

Figure 6 shows PolyMNIST performance as a function of the number of observed input modalities at $\beta = 2.5$, the setting recommended for MMVAE+ (Palumbo et al., 2023). As the number of observed modalities increases, MoPoE and CoDEVAE exhibit decreased generative quality, while MMVAE+ maintains competitive FID scores but shows only limited benefit from additional inputs. In contrast, HELVAE exhibits clear multimodal synergy: its FID steadily decreases as additional modalities are provided, suggesting that it can effectively exploit complementary information across views. This trend is also consistent with the qualitative results: for examples showing that HELVAE generates more coherent samples when conditioned on four modalities rather than one, see Appendix B.

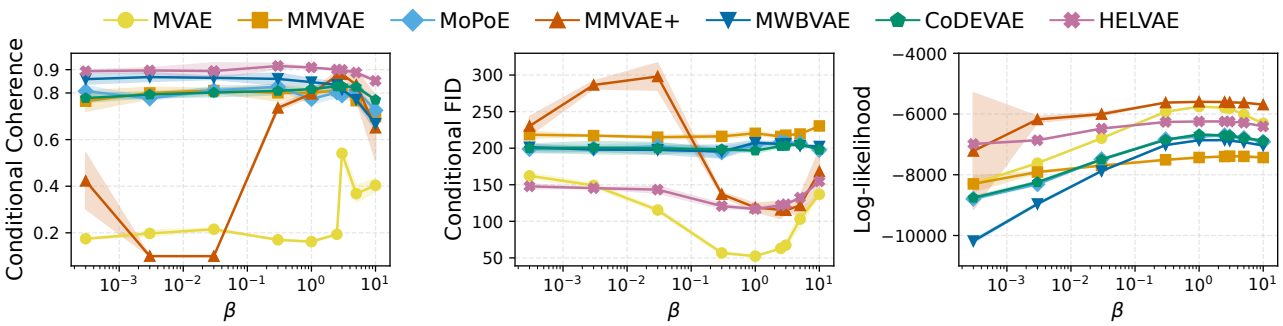


Figure 7: Performance on the PolyMNIST dataset for different values of $\beta \in \{3e^{-4}, 3e^{-3}, 3e^{-2}, 3e^{-1}, 1, 2.5, 3, 5, 10\}$. Metrics include conditional generative coherence (↑), conditional generative quality (↓), and log-likelihood estimation (↑). Markers denote means, while shaded regions represent standard deviations.

Table 5: Average training time per batch, measured in seconds, on the three benchmark datasets.

	PolyMNIST $M = 5$	CUB $M = 2$	CelebA $M = 2$
MVAE	0.1887	0.2184	1.0294
MMVAE	0.2362	0.1876	0.5764
MoPoE	0.1216	0.2269	0.6011
MMVAE+	0.2884	0.2270	0.9916
MWBVAE	0.1134	0.1408	0.5712
CoDEVAE	0.1614	0.0950	0.5817
HELVAE	**0.0895**	**0.0863**	**0.5272**
MoHELVAE	—	—	0.5918

6.3 Effect of β on model performance

Figure 7 reports performance across different β values on the PolyMNIST dataset, chosen following Daunhawer et al. (2021). While MMVAE+ reaches the highest log-likelihood, its coherence drops and FID increases sharply at both small and large β. MMVAE, MoPoE, MWBVAE, and CoDEVAE remain stable but yield lower coherence and higher FID than HELVAE. Overall, HELVAE delivers the best performance. It achieves the highest coherence, maintains competitive FID, and attains high log-likelihood across a broad range of β values, indicating that its performance is robust to the choice of this hyperparameter.

6.4 Computational cost analysis

Table 5 reports the average batch training time on the three benchmark datasets. HELVAE consistently has the shortest batch training time across all three. For mixture-based models with $\mathcal{O}(2^M)$ cost for the aggregation, such as MoPoE, MWBVAE, CoDEVAE, and MoHELVAE, training times on CelebA are comparable to HELVAE, but the gaps are larger on PolyMNIST and CUB. By contrast, MVAE, MMVAE, and MMVAE+ are the most time-consuming models: for MVAE, this is mainly due to its training scheme rather than PoE aggregation, while for MMVAE and MMVAE+, the higher cost comes from the multi-sample estimator. Overall, HELVAE is the most computationally efficient baseline while remaining effective.

Besides, *when the number of modalities is small*, such as in CelebA with two modalities, MoHELVAE remains computationally efficient, with batch training time comparable to HELVAE and MoPoE, while also outperforming other approaches in terms of latent representation quality and generative coherence.

7 CONCLUSION

This work introduced HELVAE, a multimodal VAE based on Hellinger aggregation and Hölder pooling, that improves semantic coherence without sacrificing generative quality. However, the gains in conditional coherence come with a slight trade-off in unconditional coherence and sample diversity, especially compared to MMVAE+, which uses modality-specific private latents. A natural promising extension is therefore to incorporate private latent variables into HELVAE. Moreover, we plan to study HELVAE with more expressive encoders, such as CLIP-style encoders (Radford et al., 2021). In addition, learnable pooling weights or the use of negative weights, as proposed in the squared subtractive mixture model (Loconte et al., 2024, 2025, 2026), may further improve performance. We also aim to extend Hellinger aggregation beyond VAEs, including to Gaussian process experts (Cohen et al., 2020) and other broader settings that aggregate information from expert distributions (Liusie et al., 2024; Zhang et al., 2025). In the future, we want to explore more multimodal learning frameworks with other powerful generative models, such as diffusion (Chen et al., 2024; Rojas et al., 2025) and flows (Campbell et al., 2024). The Pytorch code of our paper can be found at https://github.com/vothuckhanhhuyen/hellinger-multimodal-variational-autoencoders.

Acknowledgements

We thank the anonymous reviewers at AISTATS 2026 for their helpful and constructive comments. We also thank María Martínez García and Batuhan Koyuncu from the Probabilistic ML group at Saarland University for valuable discussions. This work has been supported by the project *"Society-Aware Machine Learning: The paradigm shift demanded by society to trust machine learning"*, funded by the European Union and led by Isabel Valera (ERC-2021-STG, SAML, 101040177). Huyen Vo also acknowledges her membership in the CS@Max Planck doctoral program. Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the aforementioned funding agencies. Neither of the aforementioned parties can be held responsible for them.

References

Abbas, A. E. (2009). A kullback-leibler view of linear and log-linear pools. *Decis. Anal.*, 6:25–37.

Amari, S. (1985). Differential-geometrical methods in statistics. *Lecture Notes in Statistics*.

Bounoua, M., Franzese, G., and Michiardi, P. (2024). Multi-modal latent diffusion. *Entropy*, 26(4):320.

Bowman, K. O. and Shenton, L. (2004). Estimation: Method of moments. *Encyclopedia of Statistical Sciences*, 3.

Campbell, A., Yim, J., Barzilay, R., Rainforth, T., and Jaakkola, T. (2024). Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In *International Conference on Machine Learning*, pages 5453–5512. PMLR.

Chen, C., Ding, H., Sisman, B., Xu, Y., Xie, O., Yao, B. Z., Tran, S. D., and Zeng, B. (2024). Diffusion models for multi-task generative modeling. In *The Twelfth International Conference on Learning Representations*.

Cohen, S., Mbuvha, R., Marwala, T., and Deisenroth, M. (2020). Healing products of Gaussian process experts. In III, H. D. and Singh, A., editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 2068–2077. PMLR.

Daunhawer, I., Sutter, T. M., Chin-Cheong, K., Palumbo, E., and Vogt, J. E. (2021). On the limitations of multimodal vaes. In *International Conference on Learning Representations*.

Dietrich, F. and List, C. (2016). Probabilistic opinion pooling.

Garg, A., Jayram, T. S., Vaithyanathan, S., and Zhu, H. (2004). Generalized opinion pooling. *Annals of Mathematics and Artificial Intelligence*.

Genest, C. (1984). A characterization theorem for externally bayesian groups. *The Annals of Statistics*, pages 1100–1105.

Gu, S., Chen, D., Bao, J., Wen, F., Zhang, B., Chen, D., Yuan, L., and Guo, B. (2022). Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10696–10706.

Hernandez-Lobato, J., Li, Y., Rowland, M., Bui, T., Hernández-Lobato, D., and Turner, R. (2016). Black-box alpha divergence minimization. In *International Conference on Machine Learning*, pages 1511–1520. PMLR.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Processing Systems*, 30.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2017). beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*.

Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800.

Hirt, M., Campolo, D., Leong, V., and Ortega, J.-P. (2024). Learning multi-modal generative models with permutation-invariant encoders and tighter variational objectives. *Transactions on Machine Learning Research*.

Hwang, H., Kim, G.-H., Hong, S., and Kim, K.-E. (2021). Multi-view representation learning via total correlation objective. *Advances in Neural Information Processing Systems*, 34:12194–12207.

Javaloy, A., Meghdadi, M., and Valera, I. (2022). Mitigating modality collapse in multimodal vaes via impartial optimization. In *International Conference on Machine Learning*, pages 9938–9964. PMLR.

Joy, T., Shi, Y., Torr, P., Rainforth, T., Schmon, S. M., et al. (2022). Learning multimodal vaes through mutual supervision. In *International Conference on Learning Representations*.

Kingma, D. P. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Koliander, G., El-Laham, Y., Djurić, P. M., and Hlawatsch, F. (2022). Fusion of probability density functions. *Proceedings of the IEEE*, 110(4):404–453.

Lee, M. and Pavlovic, V. (2021). Private-shared disentangled multimodal vae for learning of latent representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1692–1700.

Li, B., Qi, X., Lukasiewicz, T., and Torr, P. (2019). Controllable text-to-image generation. *Advances in Neural Information Processing Systems*, 32.

Li, J., Selvaraju, R., Gotmare, A., Joty, S., Xiong, C., and Hoi, S. C. H. (2021). Align before fuse: Vision and language representation learning with momentum distillation. *Advances in Neural Information Processing Systems*, 34:9694–9705.

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738.

Liusie, A., Raina, V., Fathullah, Y., and Gales, M. (2024). Efficient llm comparative assessment: a product of experts framework for pairwise comparisons. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6835–6855.

Loconte, L., Javaloy, A., and Vergari, A. (2026). How to square tensor networks and circuits without squaring them. In *The Fourteenth International Conference on Learning Representations*.

Loconte, L., Mengel, S., and Vergari, A. (2025). Sum of squares circuits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 19077–19085.

Loconte, L., Sladek, A. M., Mengel, S., Trapp, M., Solin, A., Gillis, N., and Vergari, A. (2024). Subtractive mixture models via squaring: Representation and learning. In *The Twelfth International Conference on Learning Representations*.

Mancisidor, R. A., Jenssen, R., Yu, S., and Kampffmeyer, M. (2025). Aggregation of dependent expert distributions in multimodal variational autoencoders. In *Forty-second International Conference on Machine Learning*.

Minka, T. et al. (2005). Divergence measures and message passing.

Nichol, A. Q., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., Mcgrew, B., Sutskever, I., and Chen, M. (2022). Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, pages 16784–16804. PMLR.

Nikulin, M. S. et al. (2001). Hellinger distance. *Encyclopedia of mathematics*, 78.

Oubari, F., Baha, M. E., Meunier, R., Décatoire, R., and Mougeot, M. (2024). A markov random field multi-modal variational autoencoder. *arXiv preprint arXiv:2408.09576*.

Palumbo, E., Daunhawer, I., and Vogt, J. E. (2023). Mmvae+: Enhancing the generative quality of multimodal vaes without compromises. In *The Eleventh International Conference on Learning Representations*. OpenReview.

Palumbo, E., Manduchi, L., Laguna Cillero, S., Chopard, D., and Vogt, J. E. (2024). Deep generative clustering with multimodal diffusion variational autoencoders. In *The Twelfth International Conference on Learning Representations (ICLR 2024)*. OpenReview.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *the Journal of Machine Learning Research*, 12:2825–2830.

Qiu, P., Zhu, W., Kumar, S., Chen, X., Yang, J., Sun, X., Razi, A., Wang, Y., and Sotiras, A. (2025). Multimodal variational autoencoder: A barycentric view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 20060–20068.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PmLR.

Rojas, K., Zhu, Y., Zhu, S., Ye, F. X.-F., and Tao, M. (2025). Diffuse everything: Multimodal diffusion models on arbitrary state spaces. In *Forty-second International Conference on Machine Learning*.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. (2022). Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494.

Senellart, A. and Allassonnière, S. (2025). Bridging the inference gap in mutimodal variational autoencoders. *arXiv preprint arXiv:2502.03952*.

Senellart, A., Chadebec, C., and Allassonnière, S. (2025). Multivae: A python package for multimodal variational autoencoders on partial datasets. *Journal of Open Source Software*, 10(110):7996.

Shi, Y., Paige, B., Torr, P., et al. (2019). Variational mixture-of-experts autoencoders for multi-

modal deep generative models. *Advances in Neural Information Processing Systems*, 32.

Stone, M. (1961). The opinion pool. *The Annals of Mathematical Statistics*, pages 1339–1342.

Sutter, T., Daunhawer, I., and Vogt, J. (2020). Multimodal generative learning utilizing jensen-shannon-divergence. *Advances in Neural Information Processing Systems*, 33:6100–6110.

Sutter, T., Meng, Y., Agostini, A., Chopard, D., Fortin, N., Vogt, J., Shahbaba, B., and Mandt, S. (2024). Unity by diversity: Improved representation learning for multimodal vaes. *Advances in Neural Information Processing Systems*, 37:74262–74297.

Sutter, T. M., Daunhawer, I., and Vogt, J. E. (2021). Generalized multimodal elbo. In *International Conference on Learning Representations*.

Suzuki, M., Nakayama, K., and Matsuo, Y. (2016). Joint multimodal learning with deep generative models. *arXiv preprint arXiv:1611.01891*.

Tsybakov, A. (2009). Introduction to nonparametric estimation. springer series in statistics. springer, new york.

Vasco, M., Yin, H., Melo, F. S., and Paiva, A. (2022). Leveraging hierarchy in multimodal generative models for effective cross-modality inference. *Neural Networks*, 146:238–255.

Vedantam, R., Fischer, I., Huang, J., and Murphy, K. (2018). Generative models of visually grounded imagination. In *International Conference on Learning Representations*.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. (2011). The caltech-ucsd birds-200-2011 dataset.

Wesego, D. and Rooshenas, A. (2024a). Revising multimodal vaes with diffusion decoders. *arXiv preprint arXiv:2408.16883*.

Wesego, D. and Rooshenas, P. (2024b). Score-based multimodal autoencoder. *Transactions on Machine Learning Research*.

Winkler, R. L. (1981). Combining probability distributions from dependent information sources. *Management Science*, 27(4):479–488.

Wolff, J., Krishnan, R. G., Ruff, L., Morshuis, J. N., Klein, T., Nakajima, S., and Nabi, M. (2022). Hierarchical multimodal variational autoencoders.

Wu, M. and Goodman, N. (2018). Multimodal generative models for scalable weakly-supervised learning. *Advances in Neural Information Processing Systems*, 31.

Yuan, S., Cui, J., Li, H., and Han, T. (2024). Learning multimodal latent generative models with energy-based prior. In *European Conference on Computer Vision*, pages 86–100. Springer.

Zhang, H., Koh, J. Y., Baldridge, J., Lee, H., and Yang, Y. (2021). Cross-modal contrastive learning for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 833–842.

Zhang, Y., Murtuza-Lanier, C., Li, Z., Du, Y., and Wu, J. (2025). Product of experts for visual generation. *arXiv preprint arXiv:2506.08894*.

Zhu, H. and Rohwer, R. (1995). Information geometric measurements of generalisation.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. Yes
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. Yes
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. Yes. See Appendix

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. Yes
   (b) Complete proofs of all theoretical results. Yes. See Appendix
   (c) Clear explanations of any assumptions. Yes

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). Yes. See Appendix
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). Yes. See Appendix
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). Yes
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). Yes. See Appendix

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. Yes
   (b) The license information of the assets, if applicable. Yes
   (c) New assets either in the supplemental material or as a URL, if applicable. No
   (d) Information about consent from data providers/curators. Not Applicable
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. Not Applicable

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. Not Applicable
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. Not Applicable
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. Not Applicable

Supplementary Materials
Hellinger Multimodal Variational Autoencoders

Table of Contents

The supplementary material is organized as follows: In Section A, we provide the derivation of Hellinger aggregation and ablation studies on different types of experts across multiple aggregation methods. Section B provides details on the dataset, evaluation criteria, as well as the experimental setup, including architecture and hyperparameters. Additional qualitative and quantitative results on three benchmark datasets are also included in Section B, which could not be included in the main paper.

A AGGREGATION METHODS

A.1 Derivation of Hellinger aggregation

Consider the Hölder pooling function with $\alpha = 0.5$, applied to the set of unimodal posteriors $\{q_{\phi_j}(\boldsymbol{z}|\boldsymbol{x}_j)\}_{j=1}^M$. Each posterior is a multivariate Gaussian distributions with diagonal covariance matrix, $q_{\phi_j}(\boldsymbol{z}|\boldsymbol{x}_j) = \mathcal{N}\left(\boldsymbol{\mu}_j, \text{diag}(\boldsymbol{\sigma}_j^2)\right)$, where $\boldsymbol{z} = (z_1, z_2, \ldots, z_D)^\top \in \mathbb{R}^D$ denotes the latent variable, $\boldsymbol{\mu}_j = (\mu_{j,1}, \mu_{j,2}, \ldots, \mu_{j,D})^\top \in \mathbb{R}^D$ the mean vector, and $\text{diag}(\boldsymbol{\sigma}_j^2) = \text{diag}(\sigma_{j,1}^2, \sigma_{j,2}^2, \ldots, \sigma_{j,D}^2)^\top \in \mathbb{R}^{D\times D}$ the covariance matrix. For simplicity, we denote the unimodal posterior by $q_j(\boldsymbol{z}) = q_{\phi_j}(\boldsymbol{z}|\boldsymbol{x}_j)$, and the approximate joint posterior by $q(\boldsymbol{z}) = q_\phi(\boldsymbol{z}|\mathbb{X})$. The pooled density is then defined as:

$$q(\boldsymbol{z}) = c\left(\sum_{j=1}^{M} \lambda_j \sqrt{q_j(\boldsymbol{z})}\right)^2, \tag{3}$$

where $c = 1/\int \left(\sum_{j=1}^{M} \lambda_j \sqrt{q_j(\boldsymbol{z})}\right)^2 d\boldsymbol{z}$. As the exact pooling function is not Gaussian, we will project the optimal $q(\boldsymbol{z})$ in Equation 3 back onto Gaussian family by moment matching, with mean and variance are defined as follow:

$$\tilde{\boldsymbol{\mu}} = \int \boldsymbol{z} q(\boldsymbol{z}) d\boldsymbol{z} = \frac{\int \boldsymbol{z} \left(\sum_{j=1}^{M} \lambda_j \sqrt{q_j(\boldsymbol{z})}\right)^2 d\boldsymbol{z}}{\int \left(\sum_{j=1}^{M} \lambda_j \sqrt{q_j(\boldsymbol{z})}\right)^2 d\boldsymbol{z}}, \tag{4}$$

$$\tilde{\boldsymbol{\Sigma}} = \int \boldsymbol{z}\boldsymbol{z}^\top q(\boldsymbol{z}) d\boldsymbol{z} - \tilde{\boldsymbol{\mu}}\tilde{\boldsymbol{\mu}}^\top = \frac{\int \boldsymbol{z}\boldsymbol{z}^\top \left(\sum_{j=1}^{M} \lambda_j \sqrt{q_j(\boldsymbol{z})}\right)^2 d\boldsymbol{z}}{\int \left(\sum_{j=1}^{M} \lambda_j \sqrt{q_j(\boldsymbol{z})}\right)^2 d\boldsymbol{z}} - \tilde{\boldsymbol{\mu}}\tilde{\boldsymbol{\mu}}^\top. \tag{5}$$

Step 1. We calculate the denominator term in both Equations 4, and 5 for mean and variance:

$$\begin{aligned}\int \left(\sum_{j=1}^{M} \lambda_j \sqrt{q_j(\boldsymbol{z})}\right)^2 d\boldsymbol{z} &= \sum_{j=1}^{M} \lambda_j^2 \int q_j(\boldsymbol{z}) d\boldsymbol{z} + 2\sum_{i=1}^{M}\sum_{j>i}^{M} \lambda_i\lambda_j \int \sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})} d\boldsymbol{z} \\ &= \sum_{j=1}^{M} \lambda_j^2 + 2\sum_{i=1}^{M}\sum_{j>i}^{M} \lambda_i\lambda_j S_{ij},\end{aligned} \tag{6}$$

as $\int q_j(\boldsymbol{z}) d\boldsymbol{z} = 1$ and $S_{ij} = \int \sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})} d\boldsymbol{z}$ is a Hellinger affinity between two Gaussians. As $q_i(\boldsymbol{z})$ and $q_j(\boldsymbol{z})$ are two multivariate Gaussian distributions with diagonal covariance matrix, they can be factorized across dimensions as follows:

$$q_i(\boldsymbol{z}) = \prod_{d=1}^{D} q_{i,d}(z_d), \quad q_j(\boldsymbol{z}) = \prod_{d=1}^{D} q_{j,d}(z_d),$$

where each $q_{i,d}(z_d)$ and $q_{j,d}(z_d)$ denotes a one-dimensional Gaussian distribution along the coordinate z_d. Hence, we can express S_{ij} as:

$$S_{ij} = \int \sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})} d\boldsymbol{z} = \int \sqrt{\prod_{d=1}^{D} q_{i,d}(z_d)q_{j,d}(z_d)} d\boldsymbol{z} = \prod_{d=1}^{D} \int \sqrt{q_{i,d}(z_d)q_{j,d}(z_d)} dz_d. \tag{7}$$

To derive further, we need to know the density of $q_{i,d}(z_d)$ and $q_{j,d}(z_d)$:

$$q_{i,d}(z_d) = \frac{1}{\sqrt{2\pi}\sigma_{i,d}} \exp\left(-\frac{(z_d - \mu_{i,d})^2}{2\sigma_{i,d}^2}\right), \quad q_{j,d}(z_d) = \frac{1}{\sqrt{2\pi}\sigma_{j,d}} \exp\left(-\frac{(z_d - \mu_{j,d})^2}{2\sigma_{j,d}^2}\right).$$

Then, $\int \sqrt{q_{i,d}(z_d)q_{j,d}(z_d)} dz_d$ becomes:

$$\int \sqrt{q_{i,d}(z_d)q_{j,d}(z_d)} dz_d = \int \frac{1}{\sqrt{2\pi\sigma_{i,d}\sigma_{j,d}}} \exp\left(-\frac{1}{4}\left[\frac{(z_d - \mu_{i,d})^2}{\sigma_{i,d}^2} + \frac{(z_d - \mu_{j,d})^2}{\sigma_{j,d}^2}\right]\right) dz_d.$$

Next, we expand the expression inside the exponential term:

$$\begin{aligned}-\frac{1}{4}\left[\frac{(z_d - \mu_{i,d})^2}{\sigma_{i,d}^2} + \frac{(z_d - \mu_{j,d})^2}{\sigma_{j,d}^2}\right] &= -\frac{1}{2}A_d z_d^2 + B_d z_d + C_d \\ &= -\frac{1}{2}A_d\left[z_d^2 - 2\frac{B_d}{A_d}z_d + \left(\frac{B_d}{A_d}\right)^2\right] + \frac{B_d^2}{2A_d} + C_d \\ &= -\frac{1}{2}A_d\left(z_d - \frac{B_d}{A_d}\right)^2 + \frac{B_d^2}{2A_d} + C_d,\end{aligned}$$

where

$$A_d = \frac{1}{2}\left(\frac{1}{\sigma_{i,d}^2} + \frac{1}{\sigma_{j,d}^2}\right), \quad B_d = \frac{1}{2}\left(\frac{\mu_{i,d}}{\sigma_{i,d}^2} + \frac{\mu_{j,d}}{\sigma_{j,d}^2}\right), \quad C_d = -\frac{1}{4}\left(\frac{\mu_{i,d}^2}{\sigma_{i,d}^2} + \frac{\mu_{j,d}^2}{\sigma_{j,d}^2}\right). \tag{8}$$

Combining all the above terms, we obtain an expression proportional to a Gaussian in z_d. Consequently, we can write $\int \sqrt{q_{i,d}(z_d)q_{j,d}(z_d)}dz_d$ as:

$$\begin{aligned}\int \sqrt{q_{i,d}(z_d)q_{j,d}(z_d)}dz_d &= \int \frac{1}{\sqrt{2\pi\sigma_{i,d}\sigma_{j,d}}} \exp\left(-\frac{1}{2}A_d\left(z_d - \frac{B_d}{A_d}\right)^2 + \frac{B_d^2}{2A_d} + C_d\right)dz_d \\ &= \int \frac{1}{\sqrt{2\pi\sigma_{i,d}\sigma_{j,d}}} \exp\left(\frac{B_d^2}{2A_d} + C_d\right)\exp\left(-\frac{(z_d - \frac{B_d}{A_d})^2}{2A_d^{-1}}\right)dz_d \\ &= \frac{A_d^{-1/2}}{\sqrt{\sigma_{i,d}\sigma_{j,d}}} \exp\left(\frac{B_d^2}{2A_d} + C_d\right). \end{aligned} \tag{9}$$

The first term in the product is given by:

$$\frac{A_d^{-1/2}}{\sqrt{\sigma_{i,d}\sigma_{j,d}}} = \frac{1}{\sqrt{\sigma_{i,d}\sigma_{j,d}}}\left(\frac{\sigma_{i,d}^2 + \sigma_{j,d}^2}{2\sigma_{i,d}^2\sigma_{j,d}^2}\right)^{-1/2} = \sqrt{\frac{2\sigma_{i,d}\sigma_{j,d}}{\sigma_{i,d}^2 + \sigma_{j,d}^2}}.$$

The second term in the product is given by:

$$\exp\left(\frac{B_d^2}{2A_d} + C_d\right) = \exp\left(-\frac{1}{4}\frac{(\mu_{i,d} - \mu_{j,d})^2}{\sigma_{i,d}^2 + \sigma_{j,d}^2}\right).$$

Then, we obtain the final expression for $\int \sqrt{q_{i,d}(z_d)q_{j,d}(z_d)}dz_d$:

$$\int \sqrt{q_{i,d}(z_d)q_{j,d}(z_d)}dz_d = \sqrt{\frac{2\sigma_{i,d}\sigma_{j,d}}{\sigma_{i,d}^2 + \sigma_{j,d}^2}} \exp\left(-\frac{1}{4}\frac{(\mu_{i,d} - \mu_{j,d})^2}{\sigma_{i,d}^2 + \sigma_{j,d}^2}\right). \tag{10}$$

For each dimension d, this term corresponds to the Hellinger affinity between the one-dimensional Gaussians $q_{i,d}(z_d)$ and $q_{j,d}(z_d)$. Hence, S_{ij} in Equation 7 can be rewritten as:

$$S_{ij} = \prod_{d=1}^{D} \sqrt{\frac{2\sigma_{i,d}\sigma_{j,d}}{\sigma_{i,d}^2 + \sigma_{j,d}^2}} \exp\left(-\frac{1}{4}\frac{(\mu_{i,d} - \mu_{j,d})^2}{\sigma_{i,d}^2 + \sigma_{j,d}^2}\right). \tag{11}$$

Step 2. We compute the numerator in Equation 4 for the mean:

$$\begin{aligned}\int \boldsymbol{z}\left(\sum_{j=1}^{M}\lambda_j\sqrt{q_j(\boldsymbol{z})}\right)^2 d\boldsymbol{z} &= \sum_{j=1}^{M}\lambda_j^2\int \boldsymbol{z}q_j(\boldsymbol{z})d\boldsymbol{z} + 2\sum_{i=1}^{M}\sum_{j>i}^{M}\lambda_i\lambda_j\int \boldsymbol{z}\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}d\boldsymbol{z} \\ &= \sum_{j=1}^{M}\lambda_j^2\boldsymbol{\mu}_j + 2\sum_{i=1}^{M}\sum_{j>i}^{M}\lambda_i\lambda_j\boldsymbol{M}_{ij}, \end{aligned} \tag{12}$$

as $\int \boldsymbol{z}q_j(\boldsymbol{z})d\boldsymbol{z} = \boldsymbol{\mu}_j$ and $\boldsymbol{M}_{ij} = \int \boldsymbol{z}\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}d\boldsymbol{z}$. Vector $\boldsymbol{M}_{ij} \in \mathbb{R}^D$ represents the first moment of $\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}$, it can be computed analogously to S_{ij} by factorizing across dimensions. Denoting $\boldsymbol{M}_{ij} = (M_{ij,1}, M_{ij,2}, \ldots, M_{ij,D})^\top$, we obtain $M_{ij,d}$ for each coordinate d:

$$\begin{aligned}M_{ij,d} &= \int z_d\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}d\boldsymbol{z} \\ &= \left[\int z_d\sqrt{q_{i,d}(z_d)q_{j,d}(z_d)}dz_d\right]\prod_{m\neq d}^{D}\left[\int \sqrt{q_{i,m}(z_m)q_{j,m}(z_m)}dz_m\right]. \end{aligned}$$

The first term in the product is computed in the same way as in Equations 9 and 10:

$$\begin{aligned}\int z_d\sqrt{q_{i,d}(z_d), q_{j,d}(z_d)}dz_d &= \int \frac{1}{\sqrt{2\pi\sigma_{i,d}\sigma_{j,d}}}\exp\left(\frac{B_d^2}{2A_d}+C_d\right)z_d\exp\left(-\frac{(z_d-\frac{B_d}{A_d})^2}{2A_d^{-1}}\right)dz_d\\ &= \frac{A_d^{-1/2}}{\sqrt{\sigma_{i,d}\sigma_{j,d}}}\exp\left(\frac{B_d^2}{2A_d}+C_d\right)\frac{B_d}{A_d}\\ &= \sqrt{\frac{2\sigma_{i,d}\sigma_{j,d}}{\sigma_{i,d}^2+\sigma_{j,d}^2}}\exp\left(-\frac{1}{4}\frac{(\mu_{i,d}-\mu_{j,d})^2}{\sigma_{i,d}^2+\sigma_{j,d}^2}\right)\mu_{ij,d},\end{aligned} \tag{13}$$

where

$$\mu_{ij,d} = \frac{B_d}{A_d} = \frac{\mu_{i,d}\sigma_{j,d}^2+\mu_{j,d}\sigma_{i,d}^2}{\sigma_{i,d}^2+\sigma_{j,d}^2}, \tag{14}$$

be derived from Equation 8. The second term in the product can be computed from Equation 10:

$$\prod_{m\neq d}^{D}\int\sqrt{q_{i,m}(z_m)q_{j,m}(z_m)}dz_m = \prod_{m\neq d}^{D}\sqrt{\frac{2\sigma_{i,m}\sigma_{j,m}}{\sigma_{i,m}^2+\sigma_{j,m}^2}}\exp\left(-\frac{1}{4}\frac{(\mu_{i,m}-\mu_{j,m})^2}{\sigma_{i,m}^2+\sigma_{j,m}^2}\right)$$

Then, we obtain the final expression for $M_{ij,d}$:

$$M_{ij,d} = \mu_{ij,d}\prod_{d=1}^{D}\sqrt{\frac{2\sigma_{i,d}\sigma_{j,d}}{\sigma_{i,d}^2+\sigma_{j,d}^2}}\exp\left(-\frac{1}{4}\frac{(\mu_{i,d}-\mu_{j,d})^2}{\sigma_{i,d}^2+\sigma_{j,d}^2}\right) = \mu_{ij,d}S_{ij}. \tag{15}$$

Therefore, $\boldsymbol{M}_{ij} = (\mu_{ij,1}, \mu_{ij,2}, \ldots, \mu_{ij,D})^\top S_{ij}$.

Step 3. The mean $\tilde{\boldsymbol{\mu}}$ is obtained by combining Equations 6, 12, and 15, yielding:

$$\tilde{\boldsymbol{\mu}} = \frac{\sum_{j=1}^{M}\lambda_j^2\boldsymbol{\mu}_j + 2\sum_{i=1}^{M}\sum_{j>i}^{M}\lambda_i\lambda_j\boldsymbol{M}_{ij}}{\sum_{j=1}^{M}\lambda_j^2 + 2\sum_{i=1}^{M}\sum_{j>i}^{M}\lambda_i\lambda_j S_{ij}}.$$

For each dimension d, the component $\tilde{\mu}_d$ of $\tilde{\boldsymbol{\mu}} = (\tilde{\mu}_1, \tilde{\mu}_2, \ldots, \tilde{\mu}_D) \in \mathbb{R}^D$ can be expressed as:

$$\tilde{\mu}_d = \frac{\sum_{j=1}^{M}\lambda_j^2\mu_{j,d} + 2\sum_{i=1}^{M}\sum_{j>i}^{M}\lambda_i\lambda_j M_{ij,d}}{\sum_{j=1}^{M}\lambda_j^2 + 2\sum_{i=1}^{M}\sum_{j>i}^{M}\lambda_i\lambda_j S_{ij}} = \frac{\sum_{j=1}^{M}\lambda_j^2\mu_{j,d} + 2\sum_{i=1}^{M}\sum_{j>i}^{M}\lambda_i\lambda_j\mu_{ij,d}S_{ij}}{\sum_{j=1}^{M}\lambda_j^2 + 2\sum_{i=1}^{M}\sum_{j>i}^{M}\lambda_i\lambda_j S_{ij}}. \tag{16}$$

Step 4. We compute the numerator in Equation 5 for the variance:

$$\begin{aligned}\int_z \boldsymbol{z}\boldsymbol{z}^\top\left(\sum_{j=1}^{M}\lambda_j\sqrt{q_j(\boldsymbol{z})}\right)^2 d\boldsymbol{z} &= \sum_{j=1}^{M}\lambda_j^2\int_z \boldsymbol{z}\boldsymbol{z}^\top q_j(\boldsymbol{z})d\boldsymbol{z} + 2\sum_{i=1}^{M}\sum_{j>i}^{M}\lambda_i\lambda_j\int_z \boldsymbol{z}\boldsymbol{z}^\top\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}d\boldsymbol{z}\\ &= \sum_{j=1}^{M}\lambda_j^2(\boldsymbol{\mu}_j\boldsymbol{\mu}_j^\top + \mathrm{diag}(\boldsymbol{\sigma}_j^2)) + 2\sum_{i=1}^{M}\sum_{j>i}^{M}\lambda_i\lambda_j\boldsymbol{V}_{ij},\end{aligned} \tag{17}$$

as $\int_z \boldsymbol{z}\boldsymbol{z}^\top q_j(\boldsymbol{z})d\boldsymbol{z} = \boldsymbol{\mu}_j\boldsymbol{\mu}_j^\top + \mathrm{diag}(\boldsymbol{\sigma}_j^2)$ and $\boldsymbol{V}_{ij} = \int \boldsymbol{z}\boldsymbol{z}^\top\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}d\boldsymbol{z}$. Matrix $\boldsymbol{V}_{ij}$ represents the second moment of $\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}$, it can be computed analogously to S_{ij} by factorizing across dimensions. For the off diagonal term of $\boldsymbol{V}_{ij}$ (when $d \neq e$), the expression factorizes as:

$$\begin{aligned}V_{ij,d,e} &= \int z_d z_e\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}d\boldsymbol{z}\\ &= \left[\int z_d\sqrt{q_{i,d}(z_d)q_{j,d}(z_d)}dz_d\right]\left[\int z_e\sqrt{q_{i,e}(z_e)q_{j,e}(z_e)}dz_e\right]\prod_{m\neq d,e}^{D}\left[\int\sqrt{q_{i,m}(z_m)q_{j,m}(z_m)}dz_m\right]\\ &= \mu_{ij,d}\mu_{ij,e}\prod_{d=1}^{D}\sqrt{\frac{2\sigma_{i,d}\sigma_{j,d}}{\sigma_{i,d}^2+\sigma_{j,d}^2}}\exp\left(-\frac{1}{4}\frac{(\mu_{i,d}-\mu_{j,d})^2}{\sigma_{i,d}^2+\sigma_{j,d}^2}\right)\\ &= \mu_{ij,d}\mu_{ij,e}S_{ij},\end{aligned}$$

be derived from Equations 10 and 13. For the diagonal term of $\boldsymbol{V}_{ij}$ (when $d = e$), the expression factorizes as:

$$\begin{aligned} V_{ij,d,d} &= \int z_d^2 \sqrt{q_i(\boldsymbol{z}) q_j(\boldsymbol{z})} d\boldsymbol{z} \\ &= \left[\int z_d^2 \sqrt{q_{i,d}(z_d) q_{j,d}(z_d)} dz_d \right] \prod_{m \neq d}^{D} \left[\int \sqrt{q_{i,m}(z_m) q_{j,m}(z_m)} dz_m \right]. \end{aligned}$$

The first term in the product is computed in the same way as in Equations 9 and 10:

$$\begin{aligned} \int z_d^2 \sqrt{q_{i,d}(z_d), q_{j,d}(z_d)} dz_d &= \int \frac{1}{\sqrt{2\pi\sigma_{i,d}\sigma_{j,d}}} \exp\left(\frac{B_d^2}{2A_d} + C_d \right) z_d^2 \exp\left(-\frac{(z_d - \frac{B_d}{A_d})^2}{2A_d^{-1}} \right) dz_d \\ &= \frac{A_d^{-1/2}}{\sqrt{\sigma_{i,d}\sigma_{j,d}}} \exp\left(\frac{B_d^2}{2A_d} + C_d \right) \left(\frac{B_d^2}{A_d^2} + A_d^{-1} \right) \\ &= \sqrt{\frac{2\sigma_{i,d}\sigma_{j,d}}{\sigma_{i,d}^2 + \sigma_{j,d}^2}} \exp\left(-\frac{1}{4} \frac{(\mu_{i,d} - \mu_{j,d})^2}{\sigma_{i,d}^2 + \sigma_{j,d}^2} \right) (\mu_{ij,d}^2 + \sigma_{ij,d}^2), \end{aligned}$$

where

$$\sigma_{ij,d}^2 = A_d^{-1} = \frac{2\sigma_{i,d}^2 \sigma_{j,d}^2}{\sigma_{i,d}^2 + \sigma_{j,d}^2}, \tag{18}$$

be derived from Equation 8. The second term in the product follows from Equation 10. Thus, $V_{ij,d,d}$ becomes:

$$V_{ij,d,d} = (\mu_{ij,d}^2 + \sigma_{ij,d}^2) \prod_{d=1}^{D} \sqrt{\frac{2\sigma_{i,d}\sigma_{j,d}}{\sigma_{i,d}^2 + \sigma_{j,d}^2}} \exp\left(-\frac{1}{4} \frac{(\mu_{i,d} - \mu_{j,d})^2}{\sigma_{i,d}^2 + \sigma_{j,d}^2} \right) = (\mu_{ij,d}^2 + \sigma_{ij,d}^2) S_{ij}. \tag{19}$$

Therefore, we can express matrix $\boldsymbol{V}_{ij}$ by using off diagonal terms $V_{ij,d,e} = \mu_{ij,d}\mu_{ij,e}S_{ij}$ when $d \neq e$ and diagonal term $V_{ij,d,d} = (\mu_{ij,d}^2 + \sigma_{ij,d}^2)S_{ij}$ when $d = e$.

Step 5. The variance $\tilde{\boldsymbol{\Sigma}}$ is obtained by combining Equations 6, 17, and 19, yielding:

$$\tilde{\boldsymbol{\Sigma}} = \frac{\sum_{j=1}^{M} \lambda_j^2 (\boldsymbol{\mu}_j \boldsymbol{\mu}_j^\top + \boldsymbol{\sigma}_j^2 \boldsymbol{I}_D) + 2\sum_{i=1}^{M}\sum_{j>i}^{M} \lambda_i \lambda_j \boldsymbol{V}_{ij}}{\sum_{j=1}^{M} \lambda_j^2 + 2\sum_{i=1}^{M}\sum_{j>i}^{M} \lambda_i \lambda_j S_{ij}} - \tilde{\boldsymbol{\mu}}\tilde{\boldsymbol{\mu}}^\top.$$

By approximating $q(\boldsymbol{z})$ with a diagonal covariance matrix, we simplify $\tilde{\boldsymbol{\Sigma}}$ by removing the off-diagonal terms, yielding $\tilde{\boldsymbol{\Sigma}} \approx \mathrm{diag}(\tilde{\sigma}_1^2, \tilde{\sigma}_2^2, \ldots, \tilde{\sigma}_D^2)$, where each $\tilde{\sigma}_d^2$ corresponds to dimension d and can be expressed as:

$$\begin{aligned} \tilde{\sigma}_d^2 &= \frac{\sum_{j=1}^{M} \lambda_j^2 (\mu_{j,d}^2 + \sigma_{j,d}^2) + 2\sum_{i=1}^{M}\sum_{j>i}^{M} \lambda_i \lambda_j V_{ij,d,d}}{\sum_{j=1}^{M} \lambda_j^2 + 2\sum_{i=1}^{M}\sum_{j>i}^{M} \lambda_i \lambda_j S_{ij}} - (\tilde{\mu}_d)^2 \\ &= \frac{\sum_{j=1}^{M} \lambda_j^2 (\mu_{j,d}^2 + \sigma_{j,d}^2) + 2\sum_{i=1}^{M}\sum_{j>i}^{M} \lambda_i \lambda_j (\mu_{ij,d}^2 + \sigma_{ij,d}^2) S_{ij}}{\sum_{j=1}^{M} \lambda_j^2 + 2\sum_{i=1}^{M}\sum_{j>i}^{M} \lambda_i \lambda_j S_{ij}} - (\tilde{\mu}_d)^2. \end{aligned} \tag{20}$$

Step 6. Assuming $\lambda_j = 1/M$ for all j, we approximate $q(\boldsymbol{z}) \sim \tilde{q}(\boldsymbol{z}) = \mathcal{N}(\tilde{\boldsymbol{\mu}}, \mathrm{diag}(\tilde{\boldsymbol{\sigma}}^2))$, with $\tilde{\boldsymbol{\mu}} = (\tilde{\mu}_1, \tilde{\mu}_2 \ldots, \tilde{\mu}_D)^\top \in \mathbb{R}^D$ and $\tilde{\boldsymbol{\sigma}}^2 = (\tilde{\sigma}_1^2, \tilde{\sigma}_2^2 \ldots, \tilde{\sigma}_D^2)^\top \in \mathbb{R}^D$. For each dimension d, Equations 16 and 20 give:

$$\tilde{\mu}_d = \frac{\sum_{j=1}^{M} \mu_{j,d} + 2\sum_{i=1}^{M}\sum_{j>i}^{M} \mu_{ij,d} S_{ij}}{M + 2\sum_{i=1}^{M}\sum_{j>i}^{M} S_{ij}},$$

$$\tilde{\sigma}_d^2 = \frac{\sum_{j} (\mu_{j,d}^2 + \sigma_{j,d}^2) + 2\sum_{i=1}^{M}\sum_{j>i}^{M} (\mu_{ij,d}^2 + \sigma_{ij,d}^2) S_{ij}}{M + 2\sum_{i=1}^{M}\sum_{j>i}^{M} S_{ij}} - (\tilde{\mu}_d)^2,$$

with $\mu_{ij,d}$, $\sigma_{ij,d}$, and S_{ij} as defined in Equations 14, 18, and 11. □

A.2 Hellinger aggregation beyond Gaussians: the exponential family

Our proposed Hellinger aggregation is not restricted to Gaussians: it can be adapted to many distributions in the exponential family for which the closed-form expressions of their Hellinger distances are known. Assume each modality posterior $q_j(\boldsymbol{z})$ belongs to the same exponential family with a common base measure $h(\boldsymbol{z})$, a fixed sufficient statistics $T(\boldsymbol{z})$, and common support $\mathcal{Z}$: $q_\eta(\boldsymbol{z}) = h(\boldsymbol{z})\exp(\eta^\top T(\boldsymbol{z}) - A(\eta)), \boldsymbol{z} \in \mathcal{Z}$. Let modality j have natural parameter η_j for all $j \in \{1, \dots, M\}$, we then expand the cross-term $\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}$.

$$\begin{aligned}
\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})} &= \sqrt{q_{\eta_i}(\boldsymbol{z})q_{\eta_j}(\boldsymbol{z})} \\
&= \sqrt{h(\boldsymbol{z})\exp(\eta_i^\top T(\boldsymbol{z}) - A(\eta_i))h(\boldsymbol{z})\exp(\eta_j^\top T(\boldsymbol{z}) - A(\eta_i))} \\
&= h(\boldsymbol{z})\exp\left(\frac{\eta_i^\top + \eta_j^\top}{2}T(\boldsymbol{z}) - \frac{A(\eta_i) + A(\eta_j)}{2}\right) \\
&= h(\boldsymbol{z})\exp\left(\eta_{ij}^\top T(\boldsymbol{z}) - \frac{A(\eta_i) + A(\eta_j)}{2}\right), \quad \eta_{ij} = \frac{\eta_i + \eta_j}{2}.
\end{aligned}$$

Hence, we can expand the Hellinger affinity $S_{ij} = \int \sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}d\boldsymbol{z}$ as:

$$\begin{aligned}
S_{ij} &= \int h(\boldsymbol{z})\exp\left(\eta_{ij}^\top T(\boldsymbol{z}) - \frac{A(\eta_i) + A(\eta_j)}{2}\right)d\boldsymbol{z} \\
&= \exp\left(-\frac{A(\eta_i) + A(\eta_j)}{2}\right)\int h(\boldsymbol{z})\exp(\eta_{ij}^\top T(\boldsymbol{z}))d\boldsymbol{z} \\
&= \exp\left(-\frac{A(\eta_i) + A(\eta_j)}{2}\right)\exp(A(\eta_{ij})) \\
&= \exp\left(A(\eta_{ij}) - \frac{1}{2}A(\eta_i) - \frac{1}{2}A(\eta_j)\right).
\end{aligned}$$

Next, we will derive the first moment of $\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}$, namely $\boldsymbol{M}_{ij} = \int \boldsymbol{z}\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}d\boldsymbol{z}$.

$$\begin{aligned}
\boldsymbol{M}_{ij} &= \int \boldsymbol{z}h(\boldsymbol{z})\exp\left(\eta_{ij}^\top T(\boldsymbol{z}) - \frac{A(\eta_i) + A(\eta_j)}{2}\right)d\boldsymbol{z} \\
&= \exp\left(-\frac{A(\eta_i) + A(\eta_j)}{2}\right)\int \boldsymbol{z}h(\boldsymbol{z})\exp(\eta_{ij}^\top T(\boldsymbol{z}))d\boldsymbol{z} \\
&= \exp\left(A(\eta_{ij}) - \frac{1}{2}A(\eta_i) - \frac{1}{2}A(\eta_j)\right)\int \boldsymbol{z}h(\boldsymbol{z})\exp(\eta_{ij}^\top T(\boldsymbol{z}) - A(\eta_{ij}))d\boldsymbol{z} \\
&= S_{ij}\int z q_{\eta_{ij}}(\boldsymbol{z})d\boldsymbol{z} \\
&= S_{ij}\mathbb{E}_{q_{\eta_{ij}}}[\boldsymbol{z}].
\end{aligned}$$

Similarly, for the second moment of $\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}$, we can derive $\boldsymbol{V}_{ij} = \int \boldsymbol{z}\boldsymbol{z}^\top\sqrt{q_i(\boldsymbol{z})q_j(\boldsymbol{z})}d\boldsymbol{z} = S_{ij}\mathbb{E}_{q_{\eta_{ij}}}[\boldsymbol{z}\boldsymbol{z}^\top]$.

Since exponential-family distributions satisfy $\mathbb{E}_{q_\eta}[T(\boldsymbol{z})] = \nabla A(\eta)$ and $\mathrm{Cov}_{q_\eta}[T(\boldsymbol{z})] = \nabla^2 A(\eta)$, if $\boldsymbol{z}$ and $\boldsymbol{z}\boldsymbol{z}^\top$ are included in $T(\boldsymbol{z})$ (e.g., for a Gaussian), this yields $\mathbb{E}_{q_{\eta_{ij}}}[\boldsymbol{z}]$ and $\mathbb{E}_{q_{\eta_{ij}}}[\boldsymbol{z}\boldsymbol{z}^\top]$ immediately; otherwise these moments can be obtained from the known closed-form moments of the chosen family evaluated at η_{ij}.

Example: Exponential distribution. Let each unimodal distribution be $q_j(\boldsymbol{z}) = \mathrm{Exp}(\alpha_j) = \alpha_j e^{-\alpha_j \boldsymbol{z}}$ on $\boldsymbol{z} \geq 0$. The corresponding exponential-family components are:

$$T(\boldsymbol{z}) = \boldsymbol{z}, \quad h(\boldsymbol{z}) = \mathbf{1}\{\boldsymbol{z} \geq 0\}, \quad \eta_j = -\alpha_j, \quad A(\eta_j) = -\log(-\eta_j).$$

Using these, we can calculate the Hellinger affinity S_{ij} as:

$$\eta_{ij} = -\frac{\alpha_i + \alpha_j}{2}, \quad S_{ij} = \exp\left(-\log\left(\frac{\alpha_i + \alpha_j}{2}\right) + \frac{1}{2}\log\alpha_i + \frac{1}{2}\log\alpha_j\right) = \frac{2\sqrt{\alpha_i\alpha_j}}{\alpha_i + \alpha_j}.$$

Since $q_{\eta_{ij}}(\boldsymbol{z}) = \text{Exp}\left(\frac{\alpha_i+\alpha_j}{2}\right)$, we have

$$\mathbb{E}_{q_{\eta_{ij}}}[\boldsymbol{z}] = \frac{2}{\alpha_i+\alpha_j}, \quad \mathbb{E}_{q_{\eta_{ij}}}[\boldsymbol{z}\boldsymbol{z}^\top] = \frac{8}{(\alpha_i+\alpha_j)^2}.$$

Therefore, we can obtain M_{ij} and V_{ij}.

A.3 Effect of the number of good and bad experts on multiple aggregation methods

We extend the setting in Figure 2 by comparing with the Wasserstein barycenter (WB) from Qiu et al. (2025), which uses the 2-Wasserstein distance. As shown in Figure 8, PoE suffers from overconfidence, resulting in significantly higher NLL and lower Bhattacharyya coefficients. WB is more stable than PoE, achieving a sharp distribution and reasonable overlap with the true distribution, though its NLL increases noticeably with more bad experts. Without bad experts, MoE performs slightly better than Hölder ($\alpha = 0.5$) and Hellinger, but with more bad experts, it yields higher NLL, lower Bhattacharyya coefficients, and less sharpness. This is because MoE and WB do not model dependencies between experts, which prevents them from distinguishing bad experts from good ones as their number increases. In contrast, for Hölder ($\alpha = 0.5$) and Hellinger, the overlap term S_{ij} between good and bad experts is small, which lowers the contribution of bad experts to the joint posterior. As a result, they achieve the best balance across all metrics, showing strong robustness to different expert types.

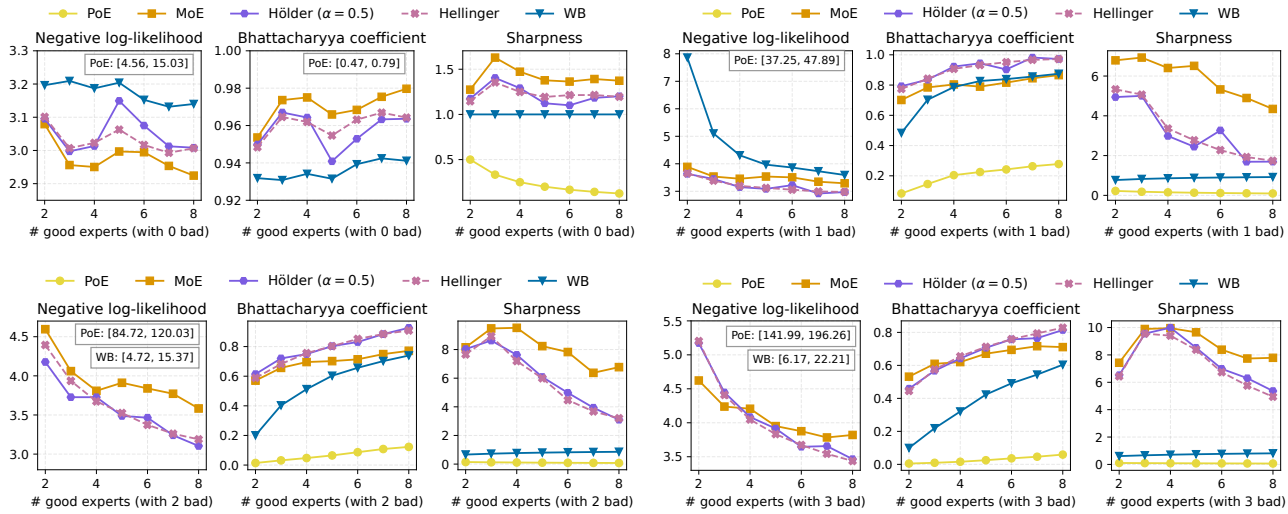


Figure 8: Performance of PoE, MoE, Hölder pooling ($\alpha = 0.5$), Hellinger, and WB aggregators as a function of the number of good experts (with 0, 1, 2, and 3 bad experts fixed). Evaluation is based on negative log-likelihood (↓), Bhattacharyya coefficient (↑), and sharpness (↓). Minimum and maximum NLL and BC values for PoE and WB are shown.

A.4 Effect of learnable pooling weights on model performance

Since we use uniform pooling weights, we also trained HELVAE with learnable pooling weights to examine the difference. In Table 6, on PolyMNIST at $\beta = 5$ (the best trade-off setting), uniform and learned pooling weights give comparable results, and the learned weights converge to nearly uniform values across modalities. In contrast, on CUB and CelebA, learning pooling weights degrades generative coherence and quality, as the optimization downweights noisier modalities in the aggregated posterior, allowing a few modalities to dominate. We therefore keep uniform pooling weights, as in PoE and MoE, since they provide a robust trade-off across modalities.

B EXPERIMENTAL DETAILS

B.1 Datasets

PolyMNIST. The dataset was introduced by Sutter et al. (2021) as an extension of MNIST with varied backgrounds. A random 28×28 crop from five images serves as the background, forming a five-modality dataset.

Table 6: Generative quality and generative coherence on the PolyMNIST dataset. The table shows the results for HELVAE with uniform vs. learned pooling weights at $\beta = 5$.

	Conditional		Unconditional	
	Coherence (↑)	FID (↓)	Coherence (↑)	FID (↓)
HELVAE (uniform)	0.887 ± 0.006	132.35 ± 0.92	0.399 ± 0.006	116.27 ± 1.19
HELVAE (learned)	0.886 ± 0.002	130.25 ± 1.84	0.421 ± 0.028	115.20 ± 1.04

Shared information is the digit label, while background and handwriting style represent modality-specific features. The training and test sets contain 60,000 and 10,000 images, respectively.

CUB Image-Captions. The Caltech Birds dataset consists of 11,788 natural images of birds, each paired with 10 fine-grained captions describing appearance characteristics. The training and testing sets contain 88,550 and 29,330 samples, respectively. It has been widely used in multimodal VAE research (Shi et al., 2019; Daunhawer et al., 2021; Palumbo et al., 2023; Mancisidor et al., 2025). However, Shi et al. (2019) employed a simplified version in which bird images were replaced with ResNet embeddings.

Bimodal CelebA. Each CelebA image (Liu et al., 2015) is annotated with 40 facial attributes. The bimodal CelebA dataset, introduced by Sutter et al. (2020), extends the original dataset with a text modality describing each face using these attributes. In this setting, negative attributes are omitted rather than explicitly negated, making learning more challenging since attributes appear in varying positions within the text, introducing additional variability. The training and test sets contain 162,770 and 19,867 samples, respectively.

Figure 9: Illustrative samples from PolyMNIST (5 modalities), CUB Image-Captions (2 modalities) and bimodal CelebA (2 modalities), respectively.

B.2 Evaluation criteria

Generative coherence (PolyMNIST and CelebA). The coherence of conditionally generated samples reflects how well the imputed modalities align with the available ones in shared information. To compute generative coherence, we first train classifier networks on the training samples of each individual modality. Unconditional coherence is evaluated by classifying modalities generated from the prior and measuring the proportion assigned the same label. Conditional coherence is measured by classifying modalities generated from non-empty subsets that exclude the modality being evaluated. The reported generative coherence values are averaged over all generated images conditioned on all corresponding subsets and modalities.

Generative coherence (CUB). We follow the approach of Palumbo et al. (2023) to calculate conditional coherence. We construct captions of the form *this bird is completely* [*color*], where [*color*] is one of {*white, yellow, red, blue, green, gray, brown, black*}. For each caption, we generate ten images (eighty in total). We then count the number of pixels within the HSV ranges specified in Table 2 in Palumbo et al. (2023). An image is considered coherent if the caption color appears among the two dominant color classes (highest pixel counts). Conditional coherence is then reported as the fraction of coherent images over the total number generated.

Generative quality. To evaluate generative quality, we use the Fréchet Inception Distance (FID) score (Heusel et al., 2017), a state-of-the-art metric for quantifying the visual quality of samples produced by generative models

in image domains. FID measures the distance between the feature distributions of real and generated images, computed using activations of a pretrained Inception network, and has been shown to correlate well with human judgment. Lower FID values indicate closer alignment between generated and real data distributions.

Latent representation. The quality of the learned representations serves as a proxy for their usefulness in downstream tasks beyond the training objective. High-quality representations of modality subsets are essential for conditionally generating coherent samples. We evaluate latent representations using logistic regression in scikit-learn (Pedregosa et al., 2011). The classifier is trained on 500 encoded samples from the training set and evaluated on the test set. Reported results correspond to average performance across all test batches.

B.3 Experimental setups

We train all models using the reparameterization trick (Kingma and Welling, 2013) and the Adam optimizer (Kingma, 2014) on NVIDIA A100-PCIE-40GB GPUs with 64 CPU cores. Following prior work, KL terms in Equation 3.1 are weighted by a coefficient β (Higgins et al., 2017), i.e. $\beta\mathrm{KL}(q_\phi(\boldsymbol{z}|\mathbb{X})\|p_\theta(\boldsymbol{z}))$, selected via cross-validation from $\{1, 2.5, 5, 10\}$, with the best β for each dataset reported in Table 7. For all datasets, the prior is assumed to be an isotropic Gaussian, while unimodal posterior distributions are modeled as multivariate Gaussians with diagonal covariance. For all experiments, modality likelihoods are weighted by relative dimensionality, where the dominant modality is fixed to 1.0 and the others are scaled proportionally to their data dimensions (Shi et al., 2019; Sutter et al., 2021; Javaloy et al., 2022). We trained HELVAE with mixed precision. Results are reported as the mean and standard deviation across three independent runs. For benchmark models, we rely on the original implementations: MVAE, MMVAE, MoPoE, MMVAE+, CoDEVAE and open source library MultiVAE (Senellart et al., 2025).

Table 7: Optimal β for all models on the PolyMNIST, CUB and CelebA datasets.

	MVAE	MMVAE	MoPoE	MMVAE+	MWBVAE	CoDEVAE	HELVAE
PolyMNIST	2.5	2.5	5	2.5	2.5	5	5
CUB	2.5	2.5	5	2.5	2.5	5	2.5
CelebA	1	1	1	1	1	1	1

PolyMNIST. The encoder and decoder architectures follow Daunhawer et al. (2021); Palumbo et al. (2023), which employ ResNet backbones for all image modalities. We assume Laplace likelihoods for all modalities with Gaussian priors and posteriors, except for MMVAE and MMVAE+, where Laplace priors and posteriors are used. All models were trained for 500 epochs with an initial learning rate of $5e^{-4}$ and a latent space of size 512, except for MMVAE and MMVAE+, which were trained for 150 epochs. For MMVAE, the latent space size was set to 160, while for MMVAE+ the shared and modality-specific latent subspaces were both set to 32 dimensions, as suggested in the original implementations. The batch size was set to 256.

CUB Image-Captions. The encoder and decoder architectures follow Palumbo et al. (2023), using ResNet networks for image modality and CNNs for text modality. We assume Laplace and one-hot categorical likelihood distributions for images and captions, respectively, with Gaussian priors and posteriors. All models were trained with an initial learning rate of $5e^{-4}$ and a latent space of 64 dimensions. For MMVAE+, the modality-specific latent space was set to 16 dimensions and the shared latent space to 48 dimensions. We trained MVAE, MoPoE, MWBVAE, CoDEVAE, and HELVAE for 150 epochs, while MMVAE and MMVAE+ were trained for 50 epochs with $K = 10$, using the DReG estimator in the objective. The batch size for MMVAE and MMVAE+ was set to 32, while for the other models it was 128.

Bimodal CelebA. The encoder and decoder architectures follow Sutter et al. (2021), using residual blocks for both modalities. We assume Laplace and one-hot categorical likelihood distributions for images and captions, respectively, with Gaussian priors and posteriors. All models were trained with an initial learning rate of $1e^{-4}$ and a latent space of 32 dimensions. Following Sutter et al. (2021), an additional modality-specific latent space of 32 dimensions was added to each modality, resulting in a total latent dimension of 64 per modality. The batch size was set to 256, and models were trained for 200 epochs.

Note. *For MMVAE+ on the CUB dataset, we report results from the original source code and paper, as we were unable to reproduce the results in our experimental setting despite extensive efforts.*

B.4 Additional results: PolyMNIST

In Table 8, we report the numerical values corresponding to the quantitative comparison in Figure 3. The results show that only HELVAE and MMVAE+ consistently achieve both high generative quality and high generative coherence across different β values, for both conditional and unconditional generation. In contrast, alternative models either underperform on one of the two criteria or reach competitive scores only for conditional or unconditional generation (e.g. CoDEVAE). Furthermore, while MMVAE+ achieves its best performance at $\beta = 2.5$, HELVAE outperforms MMVAE+ at both smaller $\beta = 1$ and larger $\beta = 10$ values. Figure 10 show qualitative results of conditionally generated samples of the second modality given corresponding test examples from one, two, three, and four modalities. Each column shows distinct samples drawn from the approximate joint posterior, ideally preserving digit identity while capturing stylistic variations. HELVAE generates more coherent samples when conditioned on four modalities compared to a single one, whereas MMVAE+ shows decreased coherence under the same setting, indicating that HELVAE effectively benefits from additional modalities.

B.5 Additional results: CUB Image-Captions

Figure 11 shows the generated samples used to assess generative coherence across models on this dataset. The qualitative results of HELVAE are consistent with the quantitative results: the generated images are coherent with the captions and exhibit high visual quality. Figure 12 shows caption-to-image generations. For MVAE, which is a product-based approach, we observe substantial variation in the generated images but low coherence. MMVAE, MWBVAE, and CoDEVAE often collapse modality-specific features to average values, yielding blurred images. In contrast, HELVAE and MMVAE+ avoids this collapse, producing images that are both high in quality and coherent with the captions. Moreover, in Figure 12, HELVAE generates samples where colors align well with the captions, and the faces and beaks are more structured, yielding coherent bird shapes.

B.6 Additional results: Bimodal CelebA

We show the distribution of evaluations for each attribute in Figure 14 (latent representation) and Figure 15 (generation coherence). HELVAE achieves good performance on most large attributes, though smaller ones remain more challenging. Across all input modalities, latent representation quality and generative coherence remain consistent, indicating that the model can generate effectively even when a modality is missing. Figure 13 shows that global attributes such as gender and smiling are captured well, consistent with the latent representations.

Table 8: Generative quality and generative coherence on the PolyMNIST dataset. Each table shows the results for the compared models for a different β values, and rankings for each metric are reported accordingly.

	Conditional		Unconditional	
$\beta = 1$	Coherence (↑)	FID (↓)	Coherence (↑)	FID (↓)
MVAE	0.176 ± 0.007 (7)	**52.58 ± 0.42 (1)**	0.000 ± 0.000 (7)	**51.08 ± 0.33 (1)**
MMVAE	0.793 ± 0.015 (5)	219.21 ± 3.38 (7)	**0.197 ± 0.004 (1)**	165.66 ± 4.42 (7)
MoPoE	0.774 ± 0.023 (6)	203.92 ± 4.58 (5)	0.019 ± 0.002 (6)	107.69 ± 1.51 (4)
MMVAE+	0.796 ± 0.012 (4)	118.89 ± 5.40 (3)	0.155 ± 0.011 (2)	117.97 ± 4.05 (6)
MWBVAE	0.850 ± 0.029 (2)	206.88 ± 4.86 (6)	0.046 ± 0.002 (4)	111.45 ± 4.79 (5)
CoDEVAE	0.817 ± 0.010 (3)	196.64 ± 3.63 (4)	0.025 ± 0.001 (5)	102.61 ± 2.41 (2)
HELVAE	**0.910 ± 0.001 (1)**	116.83 ± 0.62 (2)	0.142 ± 0.011 (3)	106.05 ± 0.25 (3)

	Conditional		Unconditional	
$\beta = 2.5$	Coherence (↑)	FID (↓)	Coherence (↑)	FID (↓)
MVAE	0.475 ± 0.053 (7)	**64.01 ± 1.45 (1)**	0.019 ± 0.004 (7)	**63.87 ± 1.63 (1)**
MMVAE	0.798 ± 0.008 (6)	218.62 ± 1.72 (7)	0.199 ± 0.011 (3)	165.71 ± 1.83 (7)
MoPoE	0.802 ± 0.020 (5)	208.48 ± 1.66 (5)	0.113 ± 0.026 (5)	116.34 ± 1.28 (5)
MMVAE+	0.875 ± 0.018 (2)	115.67 ± 11.83 (2)	**0.408 ± 0.016 (1)**	115.13 ± 9.78 (4)
MWBVAE	0.832 ± 0.008 (3)	209.51 ± 2.20 (6)	0.145 ± 0.020 (4)	118.61 ± 4.46 (6)
CoDEVAE	0.828 ± 0.011 (4)	202.81 ± 1.86 (4)	0.104 ± 0.012 (6)	113.20 ± 2.61 (3)
HELVAE	**0.900 ± 0.004 (1)**	121.78 ± 3.21 (3)	0.299 ± 0.017 (2)	108.21 ± 2.44 (2)

	Conditional		Unconditional	
$\beta = 5$	Coherence (↑)	FID (↓)	Coherence (↑)	FID (↓)
MVAE	0.410 ± 0.033 (7)	**96.56 ± 0.67 (1)**	0.019 ± 0.057 (7)	**96.75 ± 0.73 (1)**
MMVAE	0.792 ± 0.011 (5)	218.32 ± 0.77 (7)	0.197 ± 0.014 (6)	166.00 ± 1.50 (7)
MoPoE	0.812 ± 0.010 (4)	206.79 ± 1.59 (5)	0.314 ± 0.011 (3)	126.27 ± 2.67 (5)
MMVAE+	0.835 ± 0.063 (2)	121.68 ± 13.01 (2)	**0.452 ± 0.095 (1)**	120.52 ± 11.16 (3)
MWBVAE	0.790 ± 0.022 (6)	211.23 ± 2.37 (6)	0.248 ± 0.012 (5)	131.39 ± 3.58 (6)
CoDEVAE	0.826 ± 0.010 (3)	206.04 ± 1.90 (4)	0.302 ± 0.032 (4)	125.63 ± 1.60 (4)
HELVAE	**0.887 ± 0.006 (1)**	132.35 ± 0.92 (3)	0.399 ± 0.006 (2)	116.27 ± 1.19 (2)

	Conditional		Unconditional	
$\beta = 10$	Coherence (↑)	FID (↓)	Coherence (↑)	FID (↓)
MVAE	0.372 ± 0.006 (7)	**137.33 ± 0.06 (1)**	0.056 ± 0.010 (7)	137.67 ± 0.41 (2)
MMVAE	0.803 ± 0.011 (2)	217.84 ± 3.90 (7)	0.185 ± 0.013 (6)	164.66 ± 2.62 (7)
MoPoE	0.726 ± 0.009 (4)	197.64 ± 1.57 (4)	0.397 ± 0.017 (4)	147.95 ± 1.35 (4)
MMVAE+	0.651 ± 0.138 (6)	168.08 ± 29.47 (3)	0.401 ± 0.143 (3)	162.94 ± 27.75 (6)
MWBVAE	0.686 ± 0.008 (5)	211.89 ± 1.51 (6)	0.297 ± 0.002 (5)	152.39 ± 1.50 (5)
CoDEVAE	0.771 ± 0.001 (3)	197.76 ± 0.93 (5)	0.431 ± 0.006 (2)	144.99 ± 1.32 (3)
HELVAE	**0.852 ± 0.008 (1)**	154.63 ± 0.73 (2)	**0.508 ± 0.019 (1)**	**134.54 ± 1.10 (1)**

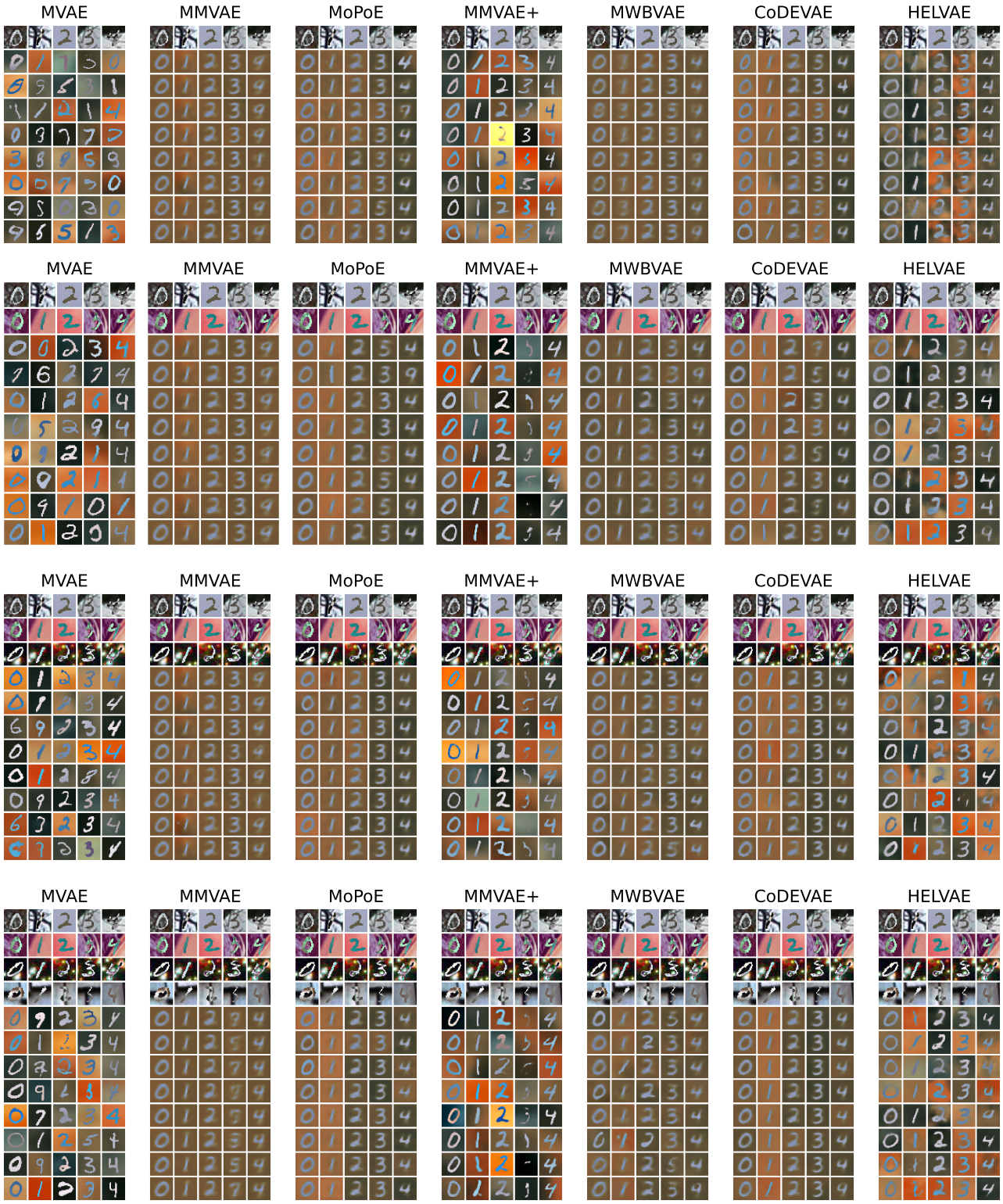
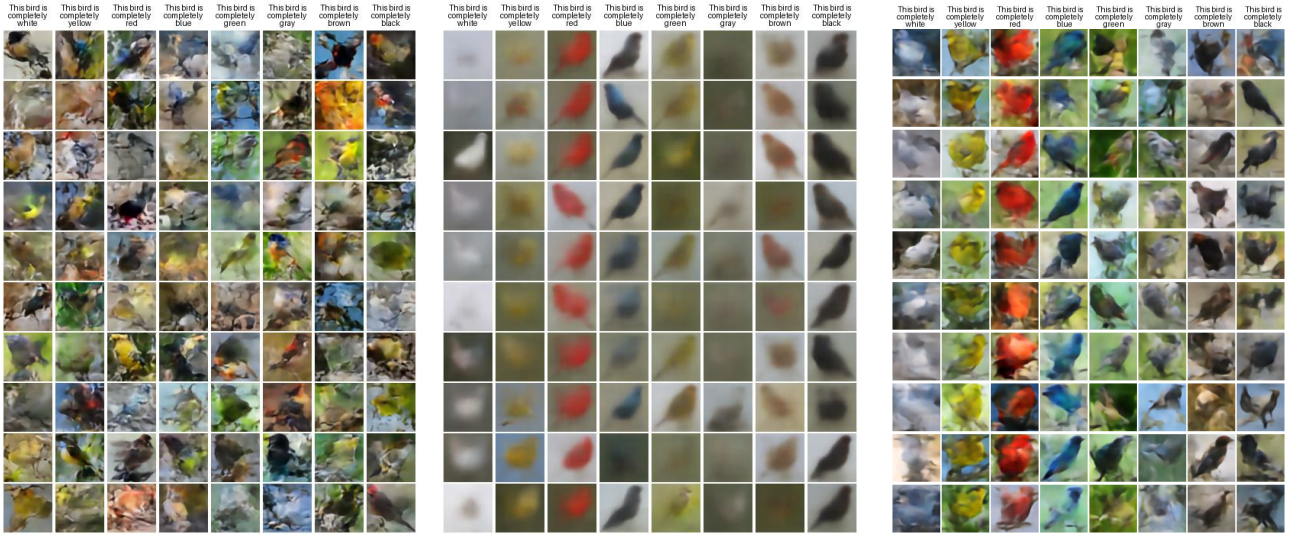


Figure 10: Conditionally generated samples of the second modality on the PolyMNIST dataset, with each model evaluated at its best $\beta \in \{1, 2.5, 5, 10\}$. The four panels show generations conditioned on (i) the first modality, (ii) the first and third, (iii) the first, third, and fourth, and (iv) the first, third, fourth, and fifth modalities. In each panel, the top rows show the conditioning inputs and the subsequent rows display the generated samples.



(a) MVAE (b) MMVAE (c) MMVAE+

(d) MWBVAE (e) CoDEVAE (f) HELVAE

Figure 11: Qualitative results of caption-to-image generation on the CUB Image-Captions dataset, used to assess generative coherence across models, with each model evaluated at its best $\beta \in \{1, 2.5, 5\}$.

(a) MVAE (b) MMVAE (c) MMVAE+

(d) MWBVAE (e) CoDEVAE (f) HELVAE

Figure 12: Qualitative results of caption-to-image generation for the CUB Image-Captions dataset, with each model evaluated at its best $\beta \in \{1, 2.5, 5\}$.

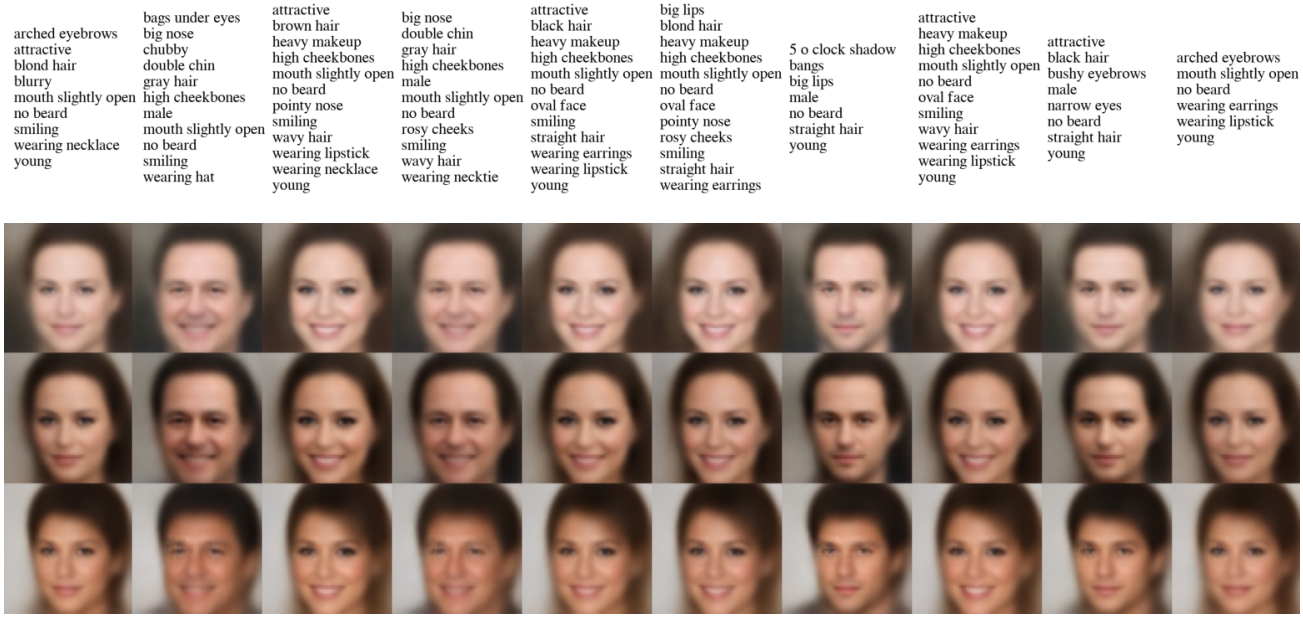


Figure 13: Qualitative results on the bimodal CelebA dataset using MoHELVAE, with $\beta = 1$. Each column shows images conditionally generated from the text shown at the top.

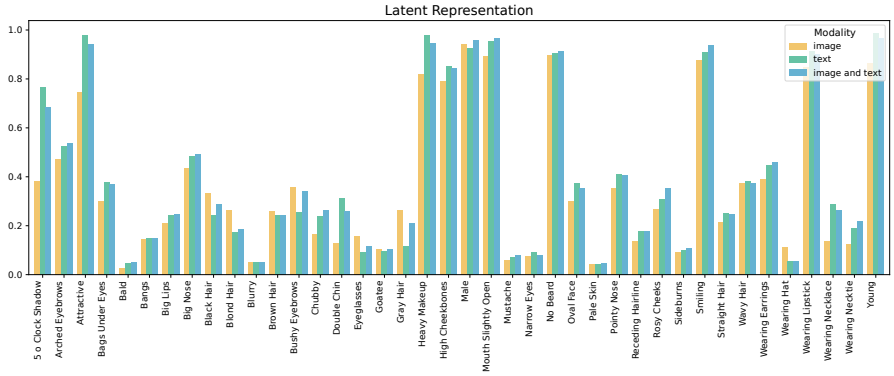
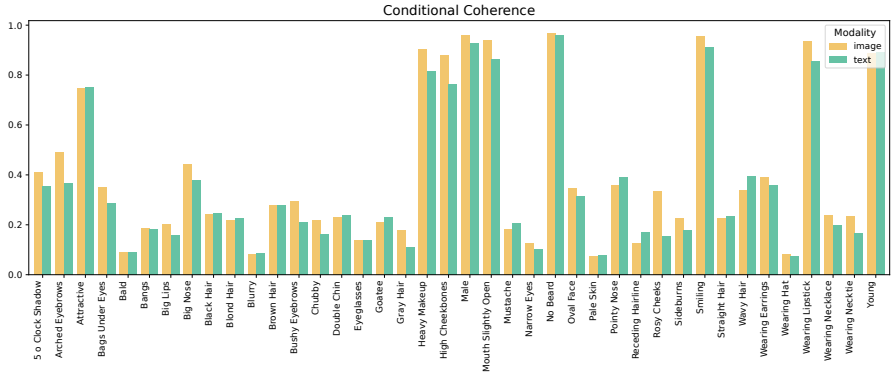
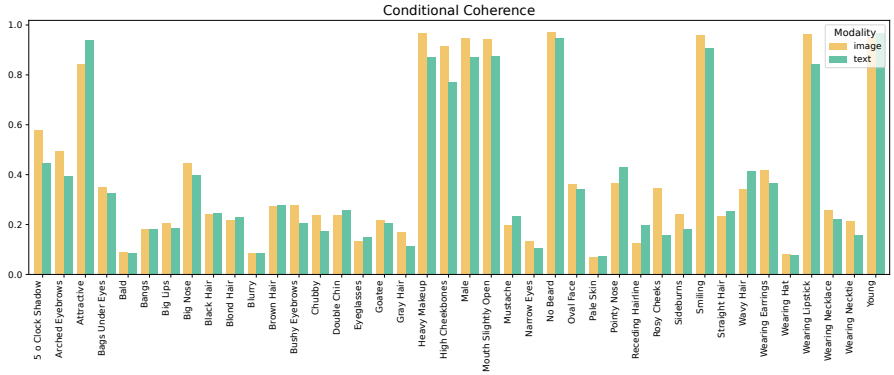


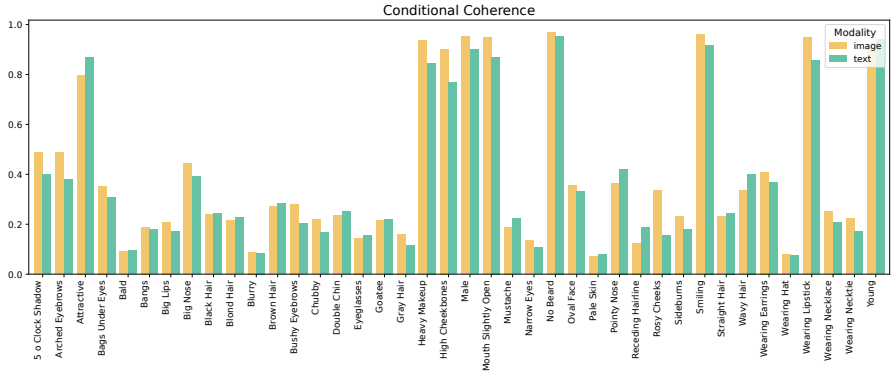
Figure 14: Latent representation (↑) of the bimodal CelebA dataset using the HELVAE model, with $\beta = 1$.



(a) Input modality: image



(b) Input modality: text



(c) Input modality: image and text

Figure 15: Conditional coherence (↑) on the bimodal CelebA dataset using the HELVAE model, with $\beta = 1$. In each subplot, images and texts are generated conditionally from the modality or subset of modalities indicated in the caption.