MultiVae: A Python package for Multimodal Variational Autoencoders on Partial Datasets

Agathe Senellart[1,2*¶] **and Stéphanie Allassonnière**[2*]

1 Université de Paris-Cité **2** Inria **3** Inserm **¶** Corresponding author ***** These authors contributed equally.

Summary

In recent years, there has been a major boom in the development of multimodal machine learning models. Among open topics, representation (fusion) and generation of multimodal data are very active fields of research. Recently, Multimodal Variational Autoencoders (VAEs) have been attracting growing interest for both tasks, thanks to their versatility, scalability, and interpretability as probabilistic latent variable models. They are also particularly interesting models in the *partially observed* setting, as most of them can be trained even with missing data. This last point makes them particularly interesting for research fields such as the medical field, where missing data are commonplace (Antelmi et al., 2019; Lawry Aguila et al., 2023).

We present MultiVae, an open-source Python library for bringing together unified implementations of multimodal VAEs. It has been designed for easy, customizable use of these models on fully or partially observed data. This library also facilitates the development and benchmarking of new algorithms by integrating several benchmark datasets, a variety of evaluation metrics and tools for monitoring and sharing models.

Multimodal Variational Autoencoders

In Multimodal Machine Learning, two goals are generally targeted: (1) Learn a shared representation from multiple modalities; (2) Learn to generate one missing modality given the ones that are available.

Multimodal Variational Autoencoders aim at solving both issues at the same time. These models learn a latent representation z of all modalities in a lower dimensional common space and learn to *decode* z to generate any modality (Suzuki & Matsuo, 2022).

Let $X = (x_1, x_2, ...x_M)$ contain M modalities. In the VAE setting, we suppose that the generative process behind the observed data is the following:

$$z \sim p(z) \qquad \text{and} \forall 1 \leq i \leq M, x_i|z \sim p_\theta(x_i|z)\,, \qquad (1)$$

where $p(z)$ is a prior distribution that is often fixed, and $p_\theta(x_i|z)$ are called *decoders* and are parameterized by neural network. Typically, $p_\theta(x_i|z) = \mathcal{N}(x_i; \mu_\theta(z), \sigma_\theta(z))$ where $\mu_\theta, \sigma_\theta$ are neural networks. We aim to learn these *decoders* that translate z into the high dimensional data x_i. At the same time, we aim to learn an *encoder* $q_\phi(z|X)$ that map observations to the latent space. $q_\phi(z|X)$ is also parameterized by a neural network. Derived from variational inference (Kingma & Welling, 2014), the VAE objective writes:

$$\mathcal{L}(X) = \mathbb{E}_{q_\phi(z|X)} \left(\sum_i \ln(p_\theta(x_i|z)) \right) - KL(q_\phi(z|X)|p(z))$$

.

The first term is a reconstruction loss and the second term can be seen as a regularization term that avoids overfitting. A typical training step of a multimodal VAE consists in encoding the

37 data with the encoder, reconstructing each modality with the decoders and taking a gradient
38 step to optimize the loss $\mathcal{L}(X)$.

39 Most multimodal VAEs differ in how they construct the encoder $q_\phi(z|X)$. In the figure below,
40 we summarize several approaches: *Aggregated models* (Shi et al., 2019; Sutter et al., 2021; Wu
41 & Goodman, 2018) use a mean or a product operation to aggregate the information coming
42 from all modalities, where *Joint models* (Senellart et al., 2023; Suzuki et al., 2016; Vedantam
43 et al., 2018) use a neural network taking all modalities as input. Finally *coordinated models*
44 (Tian & Engel, 2019; Wang et al., 2017a) use different latent spaces but add a constraint
45 term in the loss to force them to be similar.

Aggregated Models   Joint Models   Coordinated Models

⟷ : Similarity constraint
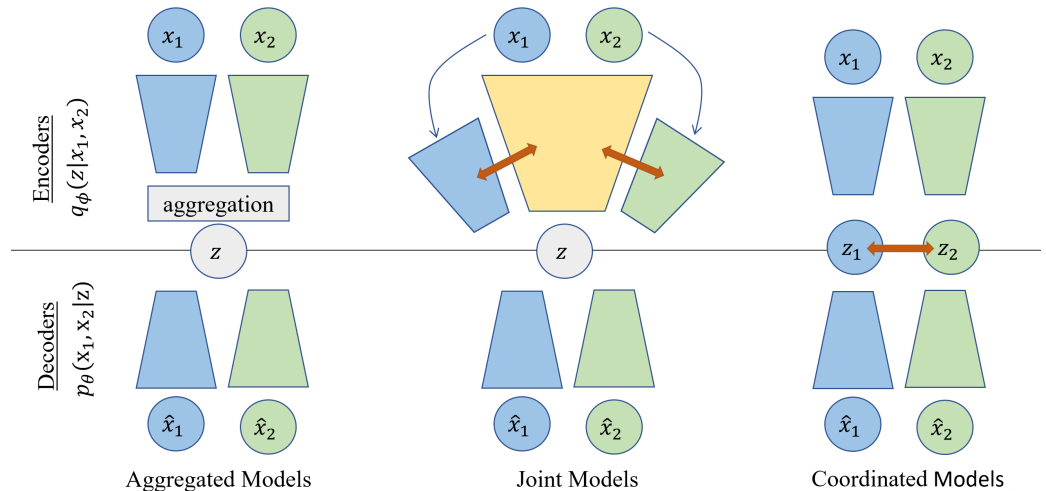
47 In our library, we implement all these approaches in an unified and modular way.

48 Aggregated models offer a natural way of learning on incomplete datasets: for an incomplete
49 sample X, we use only the available modalities to encode the data and compute the loss.
50 However, except in our library MultiVae, there does not exist an implementation of these
51 models that can be used on incomplete datasets in a straightforward manner.

Data Augmentation

53 Another application of these models is Data Augmentation (DA): from sampling latent codes z
54 and decoding them, *fully synthetic multimodal* samples can be generated to augment a dataset.
55 DA has been proven useful in many data-intensive deep learning applications (Chadebec et
56 al., 2023). In a dedicated module `multivae.samplers`, we propose different ways of sampling
57 latent codes z to further explore the generative abilities of these models.

Statement of need

59 Although multimodal VAEs have interesting applications in different fields, the lack of easy-to-
60 use and verified implementations might hinder applicative research. With MultiVae, we offer
61 unified implementations, designed to be easy to use by non-specialists. In order to propose
62 reliable implementations, we reproduced, whenever possible, a key result from the original paper.
63 Some works similar to ours have grouped together model implementations: the Multimodal
64 VAE Comparison Toolkit (Sejnova et al., 2024) includes 4 models and the Pixyz(Masahiro
65 Suzuki & Matsuo, 2023) library contains 2 multimodal models. The work closest to ours
66 and released while we were developing our library is `multi-view-ae` (Aguila et al., 2023),
67 which contains a dozen of models. We compare in a summarizing table below, the different
68 features of each work. Our library complements what already exists: our API is quite different

compared to previous work, the models implemented are not all the same, and for those we have in common, our implementation offers additional parameterization options. Indeed, for each model, we have made sure to offer great flexibility on parameters and to include all implementation details present in the original codes. Our library also offers additional features: **compatibility with incomplete data**, which we consider essential for real-life applications, and a range of tools dedicated to research and development of new algorithms: benchmark datasets, **metrics modules** and **samplers**, for testing and analyzing models.

List of models and features

In the Table below, we list available models and features, and compare to previous work. This symbol (✓*) indicates that the implementation include additional options.

Models/ Features	Ours	(Aguila et al., 2023)	(Sejnova et al., 2024)
JMVAE (Suzuki et al., 2016)	✓*	✓	
MVAE(Wu & Goodman, 2018)	✓*	✓	✓
MMVAE(Shi et al., 2019)	✓*	✓	✓
MoPoE(Sutter et al., 2021)	✓*	✓	✓
DMVAE(Lee & Pavlovic, 2021)	✓	✓*	✓
MVTCAE(Hwang et al., 2021)	✓	✓	
MMVAE+(Palumbo et al., 2023)	✓*	✓	
CMVAE(Palumbo et al., 2024)	✓		
Nexus(Vasco et al., 2022)	✓		
CVAE(Kingma & Welling, 2014)	✓		
MHVAE(Dorent et al., 2023)	✓		
TELBO(Vedantam et al., 2018)	✓		
JNF(Senellart et al., 2023)	✓		
CRMVAE(Suzuki & Matsuo, 2023)	✓		
MCVAE(Antelmi et al., 2019)		✓	
mAAE		✓	
DVCCA(Wang et al., 2017b)		✓	
mWAE		✓	
mmJSD(Sutter et al., 2020)		✓	
gPoE(Lawry Aguila et al., 2023)		✓	
Support of Incomplete datasets	✓		
GMM Sampler	✓		
MAF Sampler, IAF Sampler	✓		
Metrics: Likelihood, Coherences, FIDs, Reconstruction, Clustering	✓		
Ready-to-use Datasets	✓		✓
Model sharing via Hugging Face	✓		

An important difference in our user-interface, is that we handle all training and model parameters within python dataclasses while (Aguila et al., 2023; Sejnova et al., 2024) uses independant YAML configuration files.

Code quality and documentation

Our code is available on Github (https://github.com/AgatheSenellart/MultiVae) and Pypi and we provide a full online documentation at (https://multivae.readthedocs.io/). The main features are illustrated through **tutorials** made available either as notebooks or scripts allowing

86 users to get started easily. To further showcase how to use our library for research applications,
87 we provide detailed case studies in the documentation.

Acknowledgements

References

92 Aguila, A. L., Jayme, A., Montaña-Brown, N., Heuveline, V., & Altmann, A. (2023). Multi-
93   view-AE: A python package for multi-view autoencoder models. *Journal of Open Source*
94   *Software*, 8(85), 5093. https://doi.org/10.21105/joss.05093

95 Antelmi, L., Ayache, N., Robert, P., & Lorenzi, M. (2019). *Sparse multi-channel variational*
96   *autoencoder for the joint analysis of heterogeneous data*. 97, 302–311. https://proceedings.
97   mlr.press/v97/antelmi19a.html

98 Chadebec, C., Thibeau-Sutre, E., Burgos, N., & Allassonnière, S. (2023). Data augmentation
99   in high dimensional low sample size setting using a geometry-based variational autoencoder.
100   *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3), 2879–2896.
101   https://doi.org/10.1109/TPAMI.2022.3185773

102 Dorent, R., Haouchine, N., Kogl, F., Joutard, S., Juvekar, P., Torio, E., Golby, A. J., Ourselin,
103   S., Frisken, S., Vercauteren, T., Kapur, T., & Wells, W. M. (2023). Unified brain MR-
104   ultrasound synthesis using multi-modal hierarchical representations. In *Medical image*
105   *computing and computer assisted intervention – MICCAI 2023* (pp. 448–458). Springer
106   Nature Switzerland. https://doi.org/10.1007/978-3-031-43999-5_43

107 Hwang, H., Kim, G.-H., Hong, S., & Kim, K.-E. (2021). Multi-view representation learning
108   via total correlation objective. *Advances in Neural Information Processing Systems*, 34,
109   12194–12207.

110 Kingma, D. P., & Welling, M. (2014). *Auto-Encoding Variational Bayes*. arXiv. http:
111   //arxiv.org/abs/1312.6114

112 Lawry Aguila, A., Chapman, J., & Altmann, A. (2023). *Multi-modal variational autoencoders*
113   *for normative modelling across multiple imaging modalities*. 425–434. https://doi.
114   org/10.1007/978-3-031-43907-0_41

115 Lee, M., & Pavlovic, V. (2021). *Private-shared disentangled multimodal VAE for learning of*
116   *latent representations*. 1692–1700. https://doi.org/10.1109/CVPRW53098.2021.00185

117 Masahiro Suzuki, T. K., & Matsuo, Y. (2023). Pixyz: A python library for developing deep
118   generative models. In *Advanced Robotics* (No. 0; Vol. 0, pp. 1–16). Taylor & Francis.
119   https://doi.org/10.1080/01691864.2023.2244568

120 Palumbo, E., Daunhawer, I., & Vogt, J. E. (2023). *MMVAE+: ENHANCING THE GENERA-*
121   *TIVE QUALITY OF MULTIMODAL VAES WITHOUT COMPROMISES.*

122 Palumbo, E., Manduchi, L., Laguna, S., Chopard, D., & Vogt, J. E. (2024). *Deep generative*
123   *clustering with multimodal diffusion variational autoencoders*. https://openreview.net/
124   forum?id=k5THrhXDV3

125 Sejnova, G., Vavrecka, M., Stepanova, K., & Taniguchi, T. (2024). *Benchmarking multimodal*
126   *variational autoencoders: CdSprites+ dataset and toolkit*. https://arxiv.org/abs/2209.
127   03048

128 Senellart, A., Chadebec, C., & Allassonnière, S. (2023). Improving multimodal joint varia-

129 tional autoencoders through normalizing flows and correlation analysis. *arXiv Preprint*
130 *arXiv:2305.11832*.

131 Shi, Y., Siddharth, N., Paige, B., & Torr, P. H. S. (2019). Variational Mixture-of-Experts
132 Autoencoders for Multi-Modal Deep Generative Models. *arXiv:1911.03393 [Cs, Stat]*.
133 http://arxiv.org/abs/1911.03393

134 Sutter, T. M., Daunhawer, I., & Vogt, J. E. (2020). Multimodal generative learning utilizing
135 jensen-shannon-divergence. *CoRR*, *abs/2006.08242*. https://arxiv.org/abs/2006.08242

136 Sutter, T. M., Daunhawer, I., & Vogt, J. E. (2021). Generalized Multimodal ELBO. *ICLR*.

137 Suzuki, M., & Matsuo, Y. (2022). A survey of multimodal deep generative models. *Advanced*
138 *Robotics*, *36*(5-6), 261–278. https://doi.org/10.1080/01691864.2022.2035253

139 Suzuki, M., & Matsuo, Y. (2023). *Mitigating the limitations of multimodal VAEs with*
140 *coordination-based approach*. https://openreview.net/forum?id=Rn8u4MYgeNJ

141 Suzuki, M., Nakayama, K., & Matsuo, Y. (2016). Joint Multimodal Learning with Deep
142 Generative Models. *arXiv:1611.01891 [Cs, Stat]*. http://arxiv.org/abs/1611.01891

143 Tian, Y., & Engel, J. (2019). Latent Translation: Crossing Modalities by Bridging Generative
144 Models. *ArXiv*.

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

147 Vedantam, R., Fischer, I., Huang, J., & Murphy, K. (2018). Generative Models of Visually
148 Grounded Imagination. *arXiv:1705.10762 [Cs, Stat]*. http://arxiv.org/abs/1705.10762

149 Wang, W., Yan, X., Lee, H., & Livescu, K. (2017b). *Deep Variational Canonical Correlation*
150 *Analysis*. arXiv. https://doi.org/10.48550/arXiv.1610.03454

151 Wang, W., Yan, X., Lee, H., & Livescu, K. (2017a). *Deep Variational Canonical Correlation*
152 *Analysis*. arXiv. https://doi.org/10.48550/arXiv.1610.03454

153 Wu, M., & Goodman, N. (2018). Multimodal Generative Models for Scalable Weakly-Supervised
154 Learning. *Advances in Neural Information Processing Systems*, *31*. https://proceedings.
155 neurips.cc/paper/2018/hash/1102a326d5f7c9e04fc3c89d0ede88c9-Abstract.html