Demystifying Generative AI: A Pedagogical Approach

Emanuele Ballarin[1,†], Francesco Giacomarra[1,†], Andrea Mecchina[1,*,†] and Nicholas Andrea Pearson[1,†]

[1]Department of Mathematics, Informatics, and Geosciences, University of Trieste, Via Valerio 12/1, 34127 Trieste, Italy

Abstract

Artificial Intelligence (AI) has become ubiquitous in contemporary society, increasingly assuming a central role across different fields and not limited to scientific research, but including healthcare, education, communication, and the creative industry. Specifically, Generative AI (GenAI), the branch of AI which typically produces synthetic content in response to a user prompt, is increasingly playing a fundamental role in these fields. In this perspective, it is crucial to train individuals with the skills and understanding needed to effectively leverage AI as a transformative tool which is efficiently integrated within the society. In this context, school education plays a key role in introducing students to the subject, especially considering that they are already regular users of GenAI-based technologies. We devised a 2-hour lecture, tailored to high-school students aged between 14 and 19, to introduce and familiarize them with these concepts. The lecture is designed to be accessible not only to final-year students or those following science-oriented curricula, but to a broader audience. This contribution presents the lecture materials, including both theoretical content and hands-on notebooks, discusses them, and makes them available to the wider public, while also sharing the insights from the participants.

Keywords

Generative AI, Variational Autoencoders, AI Literacy, Education

1. Introduction

The rapid transformation of contemporary society by Artificial Intelligence (AI) marks a true revolution [1], reshaping the way in which people work and act. Its impact extends well beyond researchers and developers, reaching traditional firms, public institutions, and individuals across all sectors. Among the various types of AI, Generative AI (GenAI) has achieved impressive results across different tasks, ranging from the generation of text [2, 3] to images [4], code [5], and even music [6].

Given their growing influence, the responsible use of such technologies has become of paramount relevance [7]. We believe that providing end users with a basic understanding of the inner workings of these methods can help them use these tools effectively and with a clear awareness of both their strengths and limitations. To this end, we suggest these topics to be addressed in school education as well, given that students are already acquainted users of AI tools on a daily basis [8].

Variational autoencoders (VAEs) [9, 10] are a simple yet powerful GenAI architecture, enabling the introduction of advanced concepts with minimal to no prior knowledge. As a matter of fact, the idea of introducing VAEs to young students has already been successfully tested [11]. We also adopted these models to design a 2-hour introductory lecture for high-school students, delivering 4 sessions to institutes from Friuli-Venezia Giulia, ranging from classical to technical curricula. Each session involved 20 to 50 students, sometimes with multiple classes simultaneously.

Students are required to have access to an internet-connected device, as the proposed activity also includes a practical, guided session. Therefore, about 15 minutes at the beginning of each lecture have been allocated for setup purposes. The remaining scheduled timeline for each lecture is divided into two segments, which mirror the structure of this work:

Ital-IA 2025: 5th National Conference on Artificial Intelligence, organized by CINI, June 23–24, 2025, Trieste, Italy

*Corresponding author.

†These authors contributed equally.

emanuele@ballarin.cc (E. Ballarin); francesco.giacomarra@phd.units.it (F. Giacomarra); andrea.mecchina@phd.units.it (A. Mecchina); nicholasandrea.pearson@phd.units.it (N. A. Pearson)

0000-0003-3673-0665 (E. Ballarin); 0000-0001-6687-442X (F. Giacomarra); 0009-0000-8863-6377 (A. Mecchina); 0009-0008-4285-6831 (N. A. Pearson)

- Section 2, taking approximately 1 hour of lecture time, introduces the fundamental concepts and mathematical tools necessary to understand the contents of the subsequent practical segment;
- Section 3, covering the remaining 45 minutes of the lecture, is dedicated to practica, which allows students to engage in hands-on coding activities until the end of the session.

Finally, Sect. 4 shares insights and feedback gathered from the interactions with the students during the lectures, reviews the key findings and recommendations, and concludes.

2. Key Concepts and Foundations

The initial segment of the lecture offers a basic conceptual understanding of GenAI. First, a concise introduction to neural networks is given, outlining their key features in a black-box fashion before describing their potential applications. Then, some insights on computer vision are given, addressing the way in which images are represented in computer science. Finally, a high level description of the architecture of VAEs is presented, simplifying their inner process in a conceptually-intuitive way.

During the first segment, common errors which typically appear in AI-generated images are shown, in order to raise awareness and sensibility of the audience to the risks related to the topic. The following subsections discuss the materials and strategies adopted to make mathematically and computationally challenging concepts accessible to a high-school audience.

2.1. Overview of Neural Networks

The lecture begins with a presentation of biological neurons, a topic which most high-school students are familiar with, in some form. This first level of abstraction has the objective of describing the parallelism [12] between the biological neuron and its artificial counterpart: both receive input, usually from other neurons, process it through simple operations, and transmit a signal to connected neurons.

The ability of collections of neurons to complete complex tasks when operating together exposes the idea behind neural networks. Intuitive and easily understandable examples, such as the relationship between students height and other physical attributes, help reinforce the concepts of model input, output, and parameters. Finally, practical applications of neural networks in domains students are likely familiar with, such as computer vision and natural language processing, are presented.

2.2. A Primer on Computer Vision

A brief introduction to the field of computer vision, including how images are stored and processed in computer science, is necessary since the practica segment largely involves generating visual data. This particular data type is chosen because the quality of the output of a GenAI algorithm in this scenario can be easily appreciated even by a non-expert audience.

Students are introduced to how greyscale images are encoded by pixel matrices, whose values range between 0 and 255. Building upon this [13], the concept is extended to the 3 Red, Green and Blue (RGB) channels, allowing the represent color images. The parallelism with natural human vision and its links with computational image manipulation are described. An intuitive overview on how images are generated with AI based models if given through demonstrative examples[1].

2.3. VAEs Made Simple

A high-level abstraction of the architecture behind VAEs is given. VAEs [14] are generative models consisting of two functional modules, the encoder and the decoder. The encoder learns to compress the input data into a lower dimensional representation, governed by a probabilistic distribution. From this representation in the latent space, the decoder attempts to reconstruct the original input or to generate new data similar to that on which it was trained on.

[1] E.g., https://thispersondoesnotexist.com, as accessed on May 27, 2025.

Although the probability distribution governing the latent space plays a key role in VAEs, such concept is simplified in order to provide a explanation tailored for the target audience. The encoder and decoder blocks are considered as black-box neural networks and the main focus of the lecture is on a simplified version of the latent space and its relation to the input and output of a VAE model.

An effective illustration of how the original data space is represented in the latent space is offered by a simple comparison between the concept of a city and the various types of maps it can be represented with. Depending on the users needs, a different map can be used to represent different information of the same city. Probably the most intuitive example is provided by the underground map which in itself is a complete representation of the city, while being greatly simplified and more compact.

Such comparison is particularly useful when introducing the critical concept of conditioning on the latent space: by incorporating additional information, the generation process of the model is guided, enabling controllable output representations. This class of methods, known as conditional VAEs (cVAEs) [15, 16], are fundamental in the subsequent practica segment.

3. Practica: Learning by Literate Programming

The second segment of our activity is devoted to a guided practical session, to be performed by the students under the supervision of the teacher. Specifically, literate programming [17] in the form of Jupyter Notebooks [18] is employed, interleaving written explanatory text, code, pictographic content, and the output of code execution. The use of the Google Colaboratory platform [19] allows to access an interactive notebook environment, with optional GPU-based acceleration, fully online and without the need for device preparation. PyTorch [20, 21] is used as the deep learning framework of choice.

Such practical activity is further subdivided into two self-contained units, for which an in-depth overview is provided in the subsections that follow.

3.1. VAEs Recap and Review

The first unit reviews the architecture of a VAE, its core operating principles, and the concept of latent space. It uses the generation of small greyscale images of handwritten digits from the MNIST dataset [22] as a working example. The encoder and decoder of a simple VAE are instantiated, with a latent space of dimension two as to allow for an intuitive visualization. The essential training loop for gradient-based parameter learning is also described.

With training yet to happen, the model is instantiated with random weights, and it is observed that the generative process is only able to produce random greyscale images. Afterwards, training is performed for a handful of epochs. As training progresses, a suitable reconstruction loss is printed on-screen as it decreases driven by the training process. Once the training has ended, the output of the generative process is finally shown, hopefully resembling the images originally part of the dataset, and yet never replicating them perfectly.

The idea of distributionally-similar data generation, crucial in the context of GenAI, is recalled. It is important to note that the goal of such phase is not that of obtaining a well-performing model in absolute terms, but that of conducting a real-time, interactive training of a VAE.

Subsequently, the focus shifts to the latent space. Using the encoder, some original input data are compressed to the latent space which, being two-dimensional, can be shown in its true form. By coloring the resulting points according to the original label of each input data, some considerations can be made on the inter-relations among input data and data categories within the latent space induced by the learned VAE model.

Finally, interpolation in latent space is experimented upon. Firstly, a uniform-grid sample of latent space points is decoded. Then, two original data samples are mapped to latent space, and the latent-space interpolation across them, along a given fixed-form trajectory, is decoded. Different variations of such tasks can be illustrated to further reinforce understanding and familiarity with the latent space concept, which is often found to be the most challenging aspect for young students.

During the laboratory, students can also be introduced to the concept of GPU-accelerated execution of deep learning models, which can significantly speed up the training process, enabling more complex models to be run efficiently.

3.2. Case Study: Realistic Human Faces Generation

The second unit is devoted to open-ended experimentation by teacher and students, using as an example the task of realistic human face image generation, eventually conditional on the presence or absence of given attributes in the result. To that end, the concept of cVAE is recalled.

Due to time and resource constraints, a pre-trained model is used in order to allow students to experiment with the task at hand. Conceptually, the training process is identical to that performed in the previous unit. However, increased image size, complexity, and model size for improved realism often requires a much longer training time or on-premises dedicated infrastructure. In our case, we trained our model on the CelebA dataset with facial features [23], which resulted in a training time of 5 hours and 53 minutes on one Nvidia A100 GPU. By balancing among output realism, resource intensity, and training time, a suitable configuration can be obtained for any resource constraint.

When training the model, a choice can be made about which potential attributes to include. In our case, to avoid the emergence of unforeseen or controversial biases, potentially sensitive attributes contained in the original dataset have been dropped. Once the training is over, it is useful to check whether some conditional parameters are better kept fixed, e.g. those determining the blurriness of the image, and to determine a suitable range for the others, so to avoid out-of-distribution unrealistic outputs being generated.

Initially, the architecture of a convolutional cVAE model [24] is defined. In this case, contrary to the previous unit, the latent space is endowed of an additional tensor of conditional variables, coding for the presence or absence of the given features, and their expression intensity. Upon such coding, the stochasticity of latent space sampling is composed. Although biologically simplistic, a link can be made to genotype/phenotype mapping as an intuitive example.

After the description of the selected attributes, experimentation can begin. Initially, all feature parameters are kept at their default values; for variable parameters, this is often the midpoint of the respective allowed interval. In such case, the generation happens as during the first unit, and the user has no significant control over the appearance of generated faces, beyond resampling a new batch. Afterwards, interpolation of some selected feature parameters is introduced. In such way, many different faces can be produced, each sequentially bearing more or less expression of the attributes selected.

Drawing to a close, students are engaged in a discussion about the quality of the results obtained in their experiments. Limitations of the adopted model are addressed as well, for example regarding what happens in case some features are deliberately made exceed their designated value constraints.

4. Conclusion

We found the second segment of the lecture critical in stimulating interest and increasing participation of the class. At the end of each lecture, some minutes are reserved to open discussion with both students and their teacher, to gather feedback and address AI-related questions.

Many students reported using GenAI tools regularly, yet lacked even a basic understanding of how these systems function. While students from scientific or technical backgrounds were generally better equipped to grasp the proposed content, those from classical curricula also showed strong interest, often engaging in thoughtful discussions on how AI could intersect with humanities, literature, and music, as well as the challenges of such integration.

Despite the limited number of lectures, the positive impact of the such initiative strongly supports the idea that AI literacy is fundamental, even at the high school level, to equip students with the knowledge needed to understand and use emerging AI tools more consciously and effectively.

Acknowledgments

The author(s) acknowledge the support of the Science Center Scientific Imaginary Trieste (https://www.immaginarioscientifico.it/en/), which assisted in coordinating the events, and the University of Trieste, which provided the venues for the lectures.

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT-4 from OpenAI in order to: Paraphrase and reword and Grammar and spelling check. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] S. Makridakis, The forthcoming artificial intelligence (ai) revolution: Its impact on society and firms, Futures 90 (2017) 46–60.
[2] A. Bahrini, M. Khamoshifar, H. Abbasimehr, R. J. Riggs, M. Esmaeili, R. M. Majdabadkohne, M. Pasehvar, Chatgpt: Applications, opportunities, and threats, in: 2023 Systems and Information Engineering Design Symposium (SIEDS), IEEE, 2023, pp. 274–279.
[3] D. Guo, Q. Zhu, D. Yang, Z. Xie, K. Dong, W. Zhang, G. Chen, X. Bi, Y. Wu, Y. Li, et al., Deepseek-coder: When the large language model meets programming–the rise of code intelligence, arXiv preprint arXiv:2401.14196 (2024).
[4] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, I. Sutskever, Zero-shot text-to-image generation, in: International conference on machine learning, Pmlr, 2021, pp. 8821–8831.
[5] A. M. Dakhel, V. Majdinasab, A. Nikanjam, F. Khomh, M. C. Desmarais, Z. M. J. Jiang, Github copilot ai pair programmer: Asset or liability?, Journal of Systems and Software 203 (2023) 111734.
[6] Y. Y. T. Nugroho, P. P. M. D. Manggala, The use of ai in creating music compositions: A case study on suno application, in: 7th Celt International Conference (CIC 2024), Atlantis Press, 2024, pp. 177–189.
[7] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, et al., On the opportunities and risks of foundation models, arXiv preprint arXiv:2108.07258 (2021).
[8] X. Zhai, X. Chu, C. S. Chai, M. S. Y. Jong, A. Istenic, M. Spector, J.-B. Liu, J. Yuan, Y. Li, A review of artificial intelligence (ai) in education from 2010 to 2020, Complexity 2021 (2021) 8812542.
[9] D. P. Kingma, M. Welling, et al., Auto-encoding variational bayes, 2013.
[10] D. J. Rezende, S. Mohamed, D. Wierstra, Stochastic backpropagation and approximate inference in deep generative models, in: International conference on machine learning, PMLR, 2014, pp. 1278–1286.
[11] Z. Lyu, S. Ali, C. Breazeal, Introducing variational autoencoders to high school students, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 36, 2022, pp. 12801–12809.
[12] W. S. McCulloch, W. Pitts, A logical calculus of the ideas immanent in nervous activity, The bulletin of mathematical biophysics 5 (1943) 115–133.
[13] B. Jähne, Digital image processing, Springer Science & Business Media, 2005.
[14] D. P. Kingma, M. Welling, et al., An introduction to variational autoencoders, Foundations and Trends® in Machine Learning 12 (2019) 307–392.
[15] K. Sohn, H. Lee, X. Yan, Learning structured output representation using deep conditional generative models, Advances in neural information processing systems 28 (2015).
[16] X. Yan, J. Yang, K. Sohn, H. Lee, Attribute2image: Conditional image generation from visual attributes, in: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14, Springer, 2016, pp. 776–791.

[17] D. E. Knuth, Literate programming, The computer journal 27 (1984) 97–111.
[18] E. Schulte, D. Davison, T. Dye, C. Dominik, A multi-language computing environment for literate programming and reproducible research, Journal of Statistical Software 46 (2012) 1–24.
[19] E. Bisong, Google colaboratory, in: Building machine learning and deep learning models on google cloud platform: a comprehensive guide for beginners, Springer, 2019, pp. 59–64.
[20] A. Paszke, Pytorch: An imperative style, high-performance deep learning library, arXiv preprint arXiv:1912.01703 (2019).
[21] J. Ansel, E. Yang, H. He, N. Gimelshein, A. Jain, M. Voznesensky, B. Bao, P. Bell, D. Berard, E. Burovski, et al., Pytorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation, in: Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2, 2024, pp. 929–947.
[22] Y. LeCun, C. Cortes, MNIST handwritten digit database (2010). URL: http://yann.lecun.com/exdb/mnist/.
[23] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 3730–3738.
[24] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (2002) 2278–2324.

A. Open Access Lecture Materials

Lecture slides, literate programming notebooks, Python code, data, and pre-trained models are considered an integral part of this work and are publicly disclosed accordingly. In particular, the following resources are provided to the reader:

- Lecture slides for the materials covered in Sect. 2 for the theoretical, frontal segment, available in the Italian language: https://url.ballarin.cc/italia_genai_slides;
- Jupyter notebooks for the practica segment covered in Sect. 3, provided via the Google Colab platform, available in the Italian language: https://url.ballarin.cc/italia_genai_notebooks;
- Code required to train and evaluate a VAE on the Celeb-A dataset for realistic face image generation: https://github.com/emaballarin/celeba_steerable_cvae;
- Pre-trained model for realistic face generation, used as part of the practica segment: https://huggingface.co/emaballarin/celeba_steerable_cvae/tree/main.