UNIVERSITÀ DELLA CALABRIA

Dipartimento di Matematica e Informatica
Dottorato di Ricerca in Matematica e Informatica
XXXVI ciclo

Tesi di Dottorato

Expanding the Frontiers in GenAI and XAI: Innovative Architectures and Applications

Settore Scientifico Disciplinare INF/01 – INFORMATICA

Candidato: Carlo Adornetto
Supervisore: Ch.mo Prof. Gianluigi Greco
Coordinatore: Ch.mo Prof. Giorgio Terracina

Sommario

L'Intelligenza Artificiale Generativa (GenAI) e l'Intelligenza Artificiale Spiegabile (XAI) hanno attirato un notevole interesse negli ultimi anni per il loro potenziale e la loro capacità di stimolare e ispirare il progresso della ricerca scientifica. Questa tesi esplora le nuove frontiere di queste discipline, presentando una raccolta di architetture e applicazioni innovative.
Nel campo della GenAI, questo lavoro introduce GIDnets, una rete neurale generativa progettata per risolvere problemi di inverse design tramite l'esplorazione dello spazio latente, mostrando accuratezza maggiore rispetto ai metodi esistenti. Inoltre, verrà discussa l'applicazione di transformers e spazio latente condizionato nel contesto della generazione automatica di referti medici. In aggiunta, la tesi indaga il ruolo degli agenti generativi, basati su modelli di linguaggio di grandi dimensioni (LLMs), nel contesto dei modelli basati su agenti, offrendo approfondimenti sulla loro validazione e le sfide emergenti. Tra queste, una delle sfide rilevanti è quella della modellazione della diffusione di opinioni in contesti sociali, la quale verrà discussa evidenziando il suo potenziale come promettente scenario applicativo per gli agenti generativi.
Nel dominio della XAI, questa tesi illustra l'impatto dei metodi computazionali sull'interpretazione dei dati, e in particolare nei casi in cui la Data Science e il Deep Learning (DL) sono impiegati per ottenere approfondimenti in campo biomedico. Nonostante i progressi, interpretare i modelli di DL rimane una questione dibattuta. SHAP (SHapley Additive exPlanations) si dimostra uno strumento potente per estrarre approfondimenti da questi modelli black-box. In questa tesi, la sua applicazione verrà discussa nei contesti della previsione di bancarotta aziendale e della predizione dei disastri naturali. Inoltre, viene proposto un nuovo algoritmo di DL basato su XAI per la selezione delle geni nel contesto della genomica funzionale. Questo algoritmo utilizza una nuova metrica ispirata da SHAP per identificare e quantificare l'impatto dei geni, migliorando significativamente la precisione delle predizioni nella prognosi di leucemia linfatica cronica.
Gli approcci innovativi presentati in questa tesi avanzano lo stato dell'arte nella GenAI e nella XAI, mostrando il potenziale di queste tecnologie per consentire la progettazione di soluzioni pratiche in vari domini.

Abstract

Generative Artificial Intelligence (GenAI) and Explainable Artificial Intelligence (XAI) have attracted significant interest in recent years due to their potential and their capacity to drive and inspire further research. This thesis explores new frontiers in these fields by presenting a collection of innovative architectures and applications.

In the realm of GenAI, this work introduces GIDnets, a generative neural network aimed at solving inverse design problems through latent space exploration, showcasing improvements over existing methods. Furthermore, the research explores the application of latent space conditioning and transformers for automatic medical report generation. The thesis also investigates the role of generative agents, based on Large Language Models (LLMs), in agent-based modeling, offering insights into their validation and the emerging challenges. One of the notable challenges addressed is the complexity of opinion diffusion in social environments, highlighting its potential as a promising application scenario for generative agents.

In the domain of XAI, this thesis illustrates the impact of computational methods on data interpretation, particularly when data science and Deep Learning (DL) are employed to gain insights in the biomedical field. Despite advancements, explaining DL models remains a debated issue. SHAP (SHapley Additive exPlanations) is demonstrated as a powerful tool for extracting insights from these black-box models and its application in bankruptcy prediction and natural disaster event scenarios will be discussed. Additionally, a new deep learning algorithm based on XAI is proposed for feature selection in genomics. This algorithm utilizes a new SHAP-inspired metric to identify and quantify the impact of genes, significantly enhancing the prediction accuracy for chronic lymphocytic leukemia.

The innovative approaches presented in this thesis advance the state-of-the-art in GenAI and XAI, showcasing the potential of these technologies to enable the design of practical solutions across various domains.

Contents

List of Figures

List of Tables

Introduction

As the research in Artificial Intelligence (AI) advances, it is becoming increasingly difficult to find a scientific or industrial field unaffected by its applications. Indeed, AI has the potential to address a wide range of diverse problems, offering significant solutions across various domains. Deep Learning (DL), in particular, has attracted significant interest as a subfield of AI that allows to identify statistical regularities and patterns hidden in large datasets, to be used for predictions or analysis [4].

The novel trend, which finds its roots in DL, is the development of Generative Artificial intelligence (GenAI) models. In general, GenAI can be identified as a dynamic and rapidly evolving discipline of AI, focused on creating models capable of generating new, synthetic data that resembles the characteristics of the input data used for training. For the purpose, GenAI models learn the underlying patterns and structures of the input datasets during the training phase and then use this knowledge to produce new, similar data. Their generative capability spans multiple domains, including text [5], images [6], audio, and even video generation. At the core of generative AI are sophisticated DL models such as Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and transformers.
The birth of the modern DL-based genAI was marked by the introduction of GANs, proposed by Ian Goodfellow and his colleagues in 2014 [7]. GANs utilize a unique game-theoretic approach where two neural networks—the generator and the discriminator—compete against each other. The generator creates synthetic data, while the discriminator attempts to distinguish between real and generated data. This adversarial process continues until the generator produces highly realistic data that the discriminator can no longer differentiate from real data.
In parallel, VAEs have provided another robust framework for data generation[5, 8]. Unlike GANs, VAEs work by encoding the input data into a latent space and then decoding it back to generate new data. This probabilistic approach enables VAEs to capture the underlying distribution of the data and generate diverse and meaningful samples.
Even if GANs and VAEs are considered foundational architectures in generative AI, recent years have witnessed a revolutionary shift with the introduction of transformers-based models. In particular, the advent of large language models (LLM) with hundreds of billion parameters (e.g. GPT-3), extended genAI applications beyond text generation to include translation, summarization, and even creative writing, underscoring the versatility and power of transformer architecture. Indeed, LLMs leverage transformers to generate human-like text based on a given prompt, requiring minimal fine-tuning for specialized tasks [9]. Their advent unlocked a plethora of possibilities for new applications and further research. However, the rise of generative AI also brings forth ethical concerns. The ability to generate realistic synthetic data can lead to the creation of deepfakes, raising issues related to misinformation and digital security. As such, the development and deployment of generative AI technologies

necessitate careful consideration of ethical guidelines and regulatory frameworks to mitigate potential risks.

Although powerful and disruptive, DL and GenAI bring with themselves the problem of explainability. DL models are usually too complex to be interpreted, and for this reason they are referred to as "black boxes". This led to the birth of Explainable Artificial Intelligence (XAI) as a critical area of research in the broader field of AI, whose aim is to explain the mechanisms behind the decision of DL models[10]. The lack of transparency poses significant challenges in fields where understanding the rationale behind AI decisions is crucial, such as healthcare, finance, and autonomous driving. In healthcare, for instance, clinicians require clear explanations of AI-driven diagnoses or treatment recommendations to ensure patient safety and make informed decisions. Similarly, in finance, regulators and stakeholders need to understand AI's decision-making processes to ensure compliance with legal standards and maintain trust. XAI seeks to bridge this gap by developing methods and tools that make AI systems more interpretable, allowing users to understand, trust, and effectively manage these systems. Research in XAI includes a variety of approaches, such as model interpretability techniques, which aim to make models inherently more understandable, and post-hoc explainability methods, which attempt to provide explanations for the outputs of otherwise opaque models [11]. Model interpretability techniques often involve designing models that are simpler and more transparent, such as decision trees or linear models, which are inherently easier to interpret than complex neural networks. On the other hand, post-hoc explainability methods include techniques like LIME (Local Interpretable Model-agnostic Explanations) [12] and SHAP (SHapley Additive exPlanations) [13], which can be applied to any model to explain individual predictions.
As the demand for ethical and accountable AI continues to grow, XAI represents a pivotal step towards integrating AI systems responsibly into society. Ensuring that AI systems are interpretable not only enhances user trust but also aligns with regulatory requirements and ethical guidelines, promoting transparency, fairness, and accountability in AI applications. Recent advancements in XAI have also focused on interactive and visual explanation tools, which allow users to explore and understand model behavior through intuitive interfaces. These tools are particularly valuable in operational settings, where real-time decision-making is essential, and users need to quickly grasp the reasons behind AI outputs. Moreover, the interdisciplinary nature of XAI research, which intersects computer science, cognitive psychology, and human-computer interaction, underscores its complexity and the need for a collaborative approach. As highlighted in recent literature, incorporating user feedback into the design of explainable systems is crucial for developing explanations that are not only technically accurate but also meaningful and useful to end-users [14].
In a world where AI is increasingly present in our daily lives and influences our choices, from the simplest to the most critical, XAI systems will play a fundamental role in ensuring that new AI technologies are both effective and trustworthy.

In this thesis, we will discuss our contributions to the aforementioned research domains, walking through a collection of original scientific papers. With this purpose, this work is organized into two parts.

Part I: New Frontiers of Generative AI Architectures. As a primary contribution, this work introduces GIDnet, a generative neural network designed to address inverse design

problems by exploring latent space. Inverse design problems often suffer from the non-uniqueness of solutions and typically fail to consider feasibility constraints in the final design. Generative AI holds promise for solving these issues by generating diverse solutions and encoding feasibility constraints within a well-structured latent space. To achieve this, mechanisms for creating and exploring latent space are essential. This research provides a comprehensive review of existing methods and conducts a thought-experiment campaign to demonstrate how, in this context, GIDNET advances the state-of-the-art in general inverse design problems. The evaluation includes testing on a diverse set of benchmarks and real-world material design challenges [15].
Well-structured latent spaces and encodings play a crucial role in GenAI, especially in DL. Indeed efficient encoding and decoding of complex data are essential for solving intricate problems in various domains, including biomedical applications. This thesis will discuss a new framework for automatic report generation, utilizing a context-conditioned latent space designed through a VAE. Such VAE is conditioned and jointly trained with a context predictor, and combined with an LLM to generate medical reports from chest X-ray images [16].
As generative models for language, LLMs are revolutionizing the way we face problems in many field, but also providing new opportunities in contexts where the solutions have been limited in the past. Indeed, they bring novelties to the study of social interactions and simulations in social environments. In this scenario, it is possible to develop agent-based simulations where the agents are equipped with social interaction capabilities that happen through natural language. This work will show an overview of the so called *generative agents* existing in the literature with a focus on the methods for verifying the alignment between their behavior and human-like behavior. The generative agent-based modeling framework can be used for studying the dynamics of opinion diffusion in social networks[17]. An example of traditional simulation in this context will be provided as a promising scenario of application for LLMs-based agents[18].

Part II: XAI Applications and Innovative Approaches. This thesis discusses the importance and the impact of computational methods on data interpretation, especially when data science and DL are utilized to gain insights in the biomedical field. The first reported example is the analysis of longitudinal data to correlate the adherence to the Mediterranean diet to physical activity and study their impact on inflammatory biomarker levels in healthy adolescents [19].
In the context of DL, another work will introduces a novel architecture for anatomical part segmentation in 3D micro-CT images [20]. While not explicitly framed in XAI, this is a significant example of how DL can reveal hidden information, enabling human interpretation that would otherwise be difficult to achieve. The innovative μ-Net uncovers the internal morphology of anatomical parts of organs, facilitating, on the one hand, the interpretation of images coming from highly sophisticated image-capturing techniques.
On the other hand, the need for innovative methods to explain the predictions of DL models remains a key area of interest. SHAP (SHapley Additive exPlanations) [13] is demonstrated as a powerful model-agnostic XAI method for extracting insights from black-box models by measuring the contribution of each independent variable to the model's predictions. The flexibility provided by SHAP and its model-agnostic nature makes it a versatile method that can be applied across various application scenarios. This thesis discusses two examples of its application. The first example uses Recurrent Neural Networks (RNNs) for bankruptcy prediction by analyzing temporal sequences of companies' financial indexes. The second

example quantifies the contribution of different variables to earthquake lateral spreading prediction using Graph Neural Networks (GNNs) and geospatial data [21].
While XAI enhances the interpretation of results by quantifying the relationship between input and output data, it also facilitates the development of more effective approaches in domains where data is challenging to manage. For example, in the field of functional genomics, gene expression profiles (GEP) datasets are usually highly dimensional and noisy. This thesis will introduce a new DL algorithm based on XAI for feature selection in genomics which employs a novel SHAP-inspired metric to select the most impactful and meaningful genes in the prediction of a disease [22]. Additionally, this work will report a study in which a preliminary version of the algorithm is being applied to prognosis prediction for patients affected from chronic lymphocytic leukemia [23].

Published papers. The contributions described in the thesis have been subject to scientific publications.

- Carlo Adornetto and Gianluigi Greco (2023). *GIDnets: generative neural networks for solving inverse design problems via latent space exploration.* In Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI (pp. 3404-3413).

- Carlo Adornetto, Valeria Fionda, and Gianluigi Greco (2023). *On the Effectiveness of Compact Strategies for Opinion Diffusion in Social Environments.* In ECAI 2023 (pp. 11-18). IOS Press.

- Carlo Adornetto, Antonella Guzzo, and Andrea Vasile (2023). *Automatic Medical Report Generation via Latent Space Conditioning and Transformers.* In 2023 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (pp. 0428-0435). IEEE.

- Catia Morelli, Ennio Avolio, Angelo Galluccio, Giovanna Caparello, Emanuele Manes, Simona Ferraro, Antonella Caruso, Daniela De Rose1, Ines Barone, Carlo Adornetto, Gianluigi Greco, Stefania Catalano, Sebastiano Andò, Diego Sisci, Cinzia Giordano, and Daniela Bonofiglio, (2021). *Nutrition education program and physical activity improve the adherence to the Mediterranean diet: impact on inflammatory biomarker levels in healthy adolescents from the DIMENU longitudinal study.* Frontiers in Nutrition, 8, 685247.

- Carlo Adornetto, Francesco Fasano, Iliess Zahid, Luigi Montaleone, Maurizui La Rocca, Gianluigi Greco, Alfio Cariola, (2024) *The Dilemma of Accuracy in Bankruptcy Prediction. A New Approach Using Explainable AI Techniques to Predict Corporate Crises.* SinergieSIMA2024

- Fortunato Morabito, Carlo Adornetto, Paola Monti, Adriana Amaro, Francesco Reggiani, Monica Colombo, Yissel Rodriguez-Aldana, Giovanni Tripepi, Graziella D'Arrigo, Claudia Vener, Federica Torricelli, Teresa Rossi, Antonino Neri, Manlio Ferrarini, Giovanna Cutrona, Massimo Gentile, and Gianluigi Greco, (2023). *Genes selection using deep learning and explainable artificial intelligence for chronic lymphocytic leukemia predicting the need and time to therapy.* Frontiers in Oncology, 13.

- Maria Giovanna Durante, Giovanni Terremoto, Carlo Adornetto, Gianluigi Greco, and Elle M. Rathje, (2024). *A new Graph Neural Network (GNN) based model for the evaluation of lateral spreading displacement in New Zealand.* Japanese Geotechnical Society Special Publication, 10(21), 776-780.

Preprint and Others:

- Carlo Adornetto and Gianluigi Greco (2023). *A New Deep Learning and XAI-Based Algorithm for Features Selection in Genomics.* arXiv preprint arXiv:2303.16914.
- Carlo Adornetto, Adrian Mora, Kai Hu, Leticia Izquierdo Garcia, Parfait Atchade Adelomou, Gianluigi Greco, Luis Alberto Alonso Pastor, and Kent Larson (2024). The Advent of Generative Agents in Agent-Based Modeling: Overview, Validation and Emerging Challenges.
- Pierangela Bruno, Edoardo De Rose, Carlo Adornetto, Francesco Calimeri, Sandro Donato, Raffaele Giuseppe Agostino, Daniela Amelio, Riccardo Barberi, Maria Carmela Cerra, Maria Caterina Crocco, Mariacristina Filice, Raffaele Filosa, Gianluigi Greco, Sandra Imbrogno, and Vincenzo Formoso (2024). *μ-Net: A Deep Learning-Based Architecture for μ-CT Segmentation.* arXiv preprint arXiv:2406.16724.

Structure of the thesis. The remaining part of this thesis is organized as follows:

- Part I is focused on the design of new GenAI architecture for real-world problem and the use of LLMs to simulate human behavior and social dynamics.
  - Chapter 1 introduces GIDNET, a novel DL generative architecture for solving inverse design problems.
  - Chapter 2 proposes a new framework for automatic generation of medical report starting from patient images, based on Variational Autoencoders (VAE) and LLMs.
  - Chapter 3 and 4, framed in the field of Agent-Based Modeling, provide an overview of the advent of LLMs to simulate human behavior in simulations. Moreover a new framework for opinion diffusion in social network is presented as a promising application scenario for the new generation of generative agents.
- Part II highlights the importance of interpretable AI models by going from the insight provided in data science and imaging applications to the need and the use of explainability methods for DL models in different scenarios. Finally, a new XAI-based approach for feature selection in the medical field is presented.
  - Chapter 5 reports a longitudinal study on the impact of a nutritional education program on adolescents' dietary habits and health.
  - Chapter 6 presents μ-Net, an innovative imaging framework for studying and the internal morphology of living beings' organs, by extracting information from 3D micro-tomography.

  - Chapters 7 and 8 will discuss two applications of DeepSHAP [13] for explaining DL model decisions, respectively in the fields of bankruptcy prediction and natural disaster prediction.
  - Chapters 9 and 10 propose a new algorithm for feature selection in genomics where an ad-hoc XAI-based score is defined. In the latter chapter, a preliminary version of the algorithm is applied to the prognosis prediction of chronic lymphocytic leukemia.

- Conclusion sums up the main contributions of this thesis and proposes future directions on this line of research.

Part I: Generative AI

Chapter 1

GIDnets: Generative Neural Networks for Solving Inverse Design Problems via Latent Space Exploration

In a number of different fields, including Engeneering, Chemistry and Physics, the design of technological tools and device structures is increasingly supported by deep-learning based methods, which provide suggestions on crucial architectural choices based on the properties that these tools and structures should exhibit. The paper proposes a novel architecture, named GIDNET, to address this *inverse design* problem, which is based on exploring a suitably defined latent space associated with the possible designs. Among its distinguishing features, GIDNET is capable of identifying the most appropriate starting point for the exploration and of likely converging into a point corresponding to a design that is a feasible one. Results of a thorough experimental activity evidence that GIDNET outperforms earlier approaches in the literature.

1.1 Introduction

The availability of increasingly effective AI techniques is paving the way to conceive novel approaches for supporting production processes over a wide spectrum of domains (e.g., [24, 25, 26, 27, 28]). In particular, in fields such as Engineering, Chemistry and Physics, the design of technological tools and device structures is progressively supported by *inverse design* (deep learning) methods, providing suggestions on crucial architectural choices based on the properties that these tools and devices should exhibit [29, 30, 31, 32, 33].

In order to illustrate the idea behind inverse design, let us consider a *photonic* application scenario where the approach is rapidly increasing its popularity [34, 35]. In this context, the *forward* problem is easily solved: a number of electromagnetic simulators are already available in the literature which are able to accurately predict the electromagnetic response of any given photonic device structure, such as a *metamaterial* or a *metasurface*—see the top part of Figure 1.1. In practice, however, physicians and optical engineers have to face the inverse problem. Indeed, they would like to design novel structures exhibiting some specific and desired electromagnetic responses; and, to this end, traditionally they have to explore the design space by considering several candidate structures before one enjoying the desired properties is found, possibly guiding the search with their knowledge about previous designs.

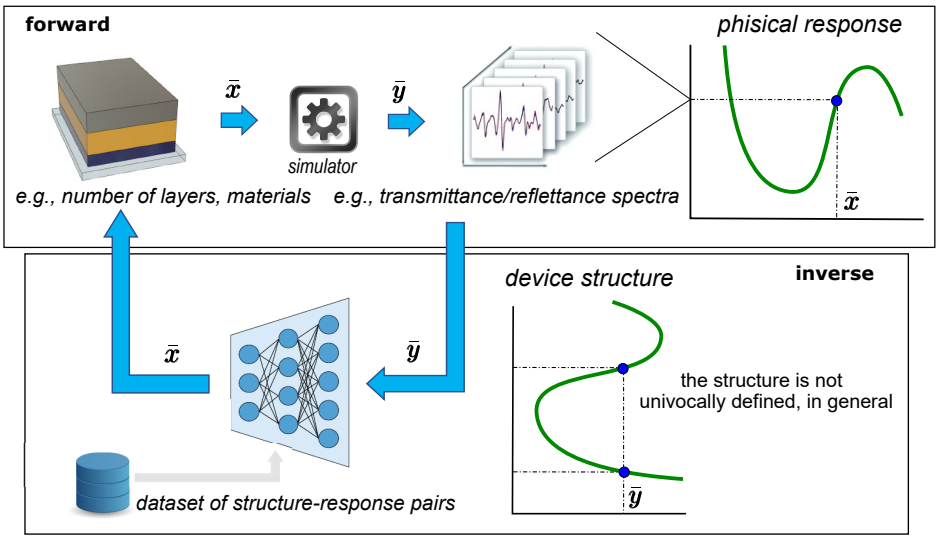


Figure 1.1: Forward computation and inverse design.

More formally, let $S \subseteq \mathbb{R}^n$ be the set of the *feasible* "input" values (e.g., encodings of the device structures that are allowed). Then, for a given forward function $\mathtt{F} : S \to \mathbb{R}^m$ (e.g., the physical simulator) and desired "output" value $\bar{\boldsymbol{y}} \in \mathbb{R}^m$ (e.g., the desired spectral response), the inverse design problem consists of computing a value $\bar{\boldsymbol{x}} \in S$ (e.g., the encoding of a device structure) such that $\bar{\boldsymbol{y}} = \mathtt{F}(\bar{\boldsymbol{x}})$. In particular, to accomplish the task, we can exploit the availability of a dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \{1,...,q\}}$ of (structure-response) pairs such that, for each $i \in \{1, ..., q\}$, $\boldsymbol{x}_i \in S$ and $\boldsymbol{y}_i = \mathtt{F}(\boldsymbol{x}_i)$ hold; indeed, $\mathcal{D}$ can be built via the simulator or it records results of experiments carried out on real structures.

Clearly enough, deep learning methods can help designers to avoid their time-consuming exploration over S, by using in a systematic way the given dataset $\mathcal{D}$. In fact, we might be tempted to view inverse design as a regression task, where we are asked to end up with a model that is trained to predict $\boldsymbol{x}_i$ based on $\boldsymbol{y}_i$, for each $i \in \{1, ..., q\}$. However, directly learning a model of this kind can be rather challenging in practice, because an "inverse function" is typically not well-defined at all: just think that $\mathcal{D}$ often contains several different structures with similar or identical responses (as in Figure 1.1), so that we are in charge of training a model with contrasting information. In addition, the output of inverse design typically consists of device structures that are one-hot encoded, hence requiring special care to deal with the feasibility of the solutions being produced. Combined with the fact that output spaces are likely high-dimensional, these characteristics pose specific challenges that already elicited the proposal of a number of ad-hoc solution approaches.

The starting point of our work is precisely a review of the most prominent deep-learning based methods proposed in the literature for inverse design. Despite their specific technical differences, most of existing methods share the idea of looking for the solution $\bar{\boldsymbol{x}}$ by directly working at the level of the space $\mathbb{R}^n$; indeed, they have been mainly conceived to deal with applications where $\mathbb{R}^n$ is a low-dimensional space. By departing from these approaches, a few works in the literature have already advocated the benefits of mapping the input space into a continuous latent space [36, 37]. This perspective is taken in the paper, by proposing a neural network architecture, named GIDNET, where the latent space is additionally constrained to the feasible region S and an exploration algorithm is used to end up with more accurate solutions. The benefits of these two novel ingredients will be eventually evidenced by the

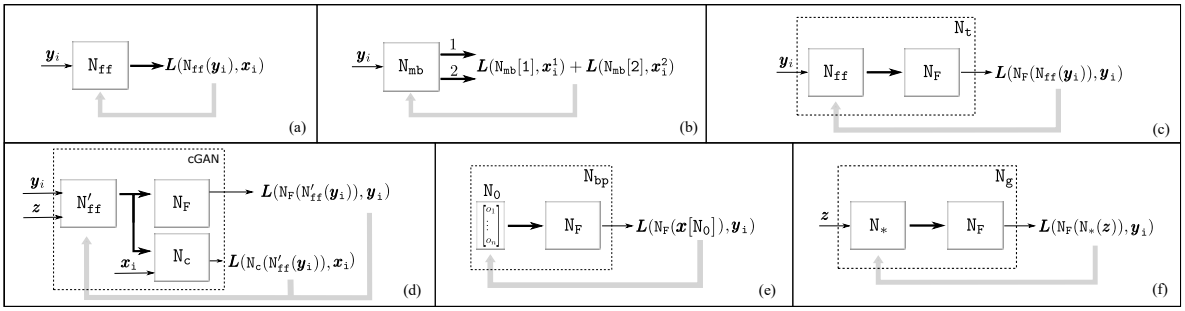


Figure 1.2: Earlier architectures in the literature. Bold arrows indicate the values $\bar{x}$, such that $\bar{y} = \mathtt{F}(\bar{x})$.

results of a thorough experimental activity conducted over several state-of-the-art benchmark datasets arising in different contexts (going beyond photonic applications).[1]

1.2 Related Works

We next classify inverse design methods in two main groups, which we name as *output-independent* and *output-dependent*.

Output-independent Methods. Methods in this group are aimed at building an *inverse* function $\mathtt{I} : \mathbb{R}^m \to S$ such that, *for each* $\boldsymbol{y} \in \mathbb{R}^m$, $\mathtt{F}(\mathtt{I}(\boldsymbol{y})) = \boldsymbol{y}$. So, they work on the entire output space $\mathbb{R}^m$, and do not require any kind of fine-tuning when some specific output value $\bar{\boldsymbol{y}} \in \mathbb{R}^m$ is given to hand and we look for the corresponding input $\bar{\boldsymbol{x}}$ such that $\mathtt{F}(\bar{\boldsymbol{x}}) = \bar{\boldsymbol{y}}$.

The basic method within this group models the inverse function via a network $\mathtt{N_{ff}}$, which is trained over $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \{1,\dots,q\}}$ by receiving as input the values $\boldsymbol{y}_i$ and returning as output the estimated inverse $\mathtt{N_{ff}}(\boldsymbol{y}_i)$, with the loss $\boldsymbol{L}$ to be optimized being its distance from $\boldsymbol{x}_i$—Figure 1.2(a). In fact, $\mathtt{N_{ff}}$ tends to remain improperly trained whenever $\mathtt{F}$ is not invertible because several input values can be associated with the same output [35].

To overcome this problem, a multi-branched architecture $\mathtt{N_{mb}}$ has been proposed by Zhang et al. ([38]). The network receives again $\boldsymbol{y}_i$, by however producing $h > 0$ outputs, being all possible different inverse values. The weights of $\mathtt{N_{mb}}$ are trained by optimizing the sum of the loss functions over the various outputs—see Figure 1.2(b), for an illustration with 2 branches. Unfortunately, however, this approach is often unviable in practice [34], since it requires a datasets storing, for each output value, all corresponding input values (which is an information that we typically lack).

Other approaches use a *tandem* network $\mathtt{N_t}$ [39, 40]. The idea is to pre-train a network $\mathtt{N_F}$ approximating the forward function $\mathtt{F}$. Then, $\mathtt{N_F}$ is coupled with $\mathtt{N_{ff}}$—see Figure 1.2(c). As a result, the tandem network acts as an autoencoder; however, the loss function (minimizing the distance between the input and the output of $\mathtt{N_t}$) is optimized via backpropagation over the weights of $\mathtt{N_{ff}}$ only; indeed, the pre-trained weights of $\mathtt{N_F}$ are freezed. After the training, for any value $\bar{\boldsymbol{y}}$, the solution is given by $\mathtt{N_{ff}}(\bar{\boldsymbol{y}})$.

A different approach is to modify $\mathtt{N_{ff}}$ into a network $\mathtt{N'_{ff}}$ receiving as input $\boldsymbol{y}_i$ plus some random noise $\boldsymbol{z}$, and producing $\boldsymbol{x}_i$ as output—see Figure 1.2(d). In fact, the given output

[1] A more detailed discussion of the experimental settings and results is available in the Supplementary Material.

value $\bar{\boldsymbol{y}}$ can be feed to the network many times, each time together with a different noise value, thereby producing several input values $\bar{\boldsymbol{x}}$ associated with $\bar{\boldsymbol{y}}$—so that we can pick the best one. This method also includes a critic/discriminative network $\mathtt{N}_\mathtt{c}$ enforcing that, on the training input $\boldsymbol{y}_i$ and whatever noise $\boldsymbol{z}$ is added, $\mathtt{N}'_\mathtt{ff}(\boldsymbol{y}_i, \boldsymbol{z})$ converges to $\boldsymbol{x}_i$ [41, 42]. Indeed, the networks $\mathtt{N}_\mathtt{c}$ and $\mathtt{N}'_\mathtt{ff}$ are eventually trained according to the *conditional* GAN framework [43, 34].

Output-dependent Methods. Output-dependent methods exploit the knowledge of $\bar{\boldsymbol{y}}$, thereby typically producing results of better quality for their ability to fine tune (on $\bar{\boldsymbol{y}}$) some pre-trained architecture. Noticeable examples are the backpropagation-based methods by [44, 45, 46]—see Figure 1.2(e). The idea is to use the pre-trained network $\mathtt{N}_\mathtt{F}$ coupled with a network $\mathtt{N}_\mathtt{0}$ with no input, and with one layer only consisting of n nodes whose weights are initialized at random. The network $\mathtt{N}_\mathtt{0}$ essentially encodes a value in $\mathbb{R}^n$, say $\boldsymbol{x}[\mathtt{N}_\mathtt{0}]$, which is passed to $\mathtt{N}_\mathtt{F}$. The loss function for the resulting network $\mathtt{N}_\mathtt{bp}$ is now meant to minimize the distance between $\bar{\boldsymbol{y}}$ and $\mathtt{N}_\mathtt{F}(\boldsymbol{x}[\mathtt{N}_\mathtt{0}])$, and it is optimized by backpropagation over the weighs of $\mathtt{N}_\mathtt{0}$ only (that is, $\mathtt{N}_\mathtt{F}$ is freezed as usual). Eventually, the desired input value $\bar{\boldsymbol{x}}$ (such that $\bar{\boldsymbol{y}} = \mathtt{F}(\bar{\boldsymbol{x}})$) is given by $\boldsymbol{x}[\mathtt{N}_\mathtt{0}]$ after the updates have been performed. A drawback is that the space of the weights is rather narrow, so that they often end up with undesired local optima [35]. In fact, a more general architecture can be obtained by replacing $\mathtt{N}_\mathtt{0}$ with the network $\mathtt{N}_\mathtt{ff}$ [47]. This leads to an output-dependent version of the tandem network in Figure 1.2(c)—which we use in the experiments.
A different approach to face that drawback is given by the Neural Adjoint (NA) method [48]—which is a variant of the method in Figure 1.2(e), hereinafter referred to as (e*). For a desired $\bar{\boldsymbol{y}}$, the method repeats T times the optimization of $\boldsymbol{x}[\mathtt{N}_\mathtt{0}]$, by starting from different random initialization of the $\mathtt{N}_\mathtt{0}$ weights; in addition, it introduces a *boundary loss* to constrain the final design $\boldsymbol{x}[\mathtt{N}_\mathtt{0}]$ to be a normally distributed variable. Notably, over real-valued inverse design benchmarks, it has been shown that the method achieves comparable or better performances than earlier methods [48], including methods based on (cVAE) *variational autoencoder* architectures whose latent space is conditioned by the knowledge of the output [37].

A different generative approach has been proposed by Jiang and Fang ([49]). Their architecture (call it $\mathtt{N}_\mathtt{g}$) can be seen as a generalization of $\mathtt{N}_\mathtt{bp}$, where in place of $\mathtt{N}_\mathtt{0}$ a more complex network $\mathtt{N}_*$ is used—Figure 1.2(f). In particular, $\mathtt{N}_*$ takes as input some random noise $\boldsymbol{z}$ and produces a value in $\mathbb{R}^n$, say $\boldsymbol{x}[\mathtt{N}_*(\boldsymbol{z})]$. For the given output value $\bar{\boldsymbol{y}}$, backpropagation is used to optimize the weights of $\mathtt{N}_*$, by considering as input for the network different samples of a uniformly distributed random variable. The result $\bar{\boldsymbol{x}}$ is taken as the value $\boldsymbol{x}[\mathtt{N}_*(\boldsymbol{z})]$ computed during the training and for which the distance between $\mathtt{F}(\boldsymbol{x}[\mathtt{N}_*(\boldsymbol{z})])$ and $\bar{\boldsymbol{y}}$ is minimized.

Finally, we mention that a VAE-based variant of the architecture in Figure 1.2(f) has been proposed too, where $\mathtt{N}^*$ is pre-trained as a decoder of a variational autoencoder and $\boldsymbol{z}$ is sampled from the latent space conditioned on the output and is optimized given $\bar{\boldsymbol{y}}$. Notably, this approach has been specifically designed [36] to deal with scenarios where the feasible region $S \subseteq \mathbb{R}^n$ is associated with a proper one-hot encoding of some categorical features.

Our works shares with [36] the idea of dealing with an embedding of $\mathbb{R}^n$ into a latent space. However, while that work relies on a plain optimization of a random point, we next introduce a method that explores large portions of the latent space to find better solutions, by eventually starting from a meaningful initialization.

1.3 Description of the GIDNET Approach

To tackle the inverse design problem on a generic function $\mathtt{F} : S \to \mathbb{R}^m$, with $S \subseteq \mathbb{R}^n$, we propose an output-dependent method where, for any $\bar{\boldsymbol{y}} \in \mathbb{R}^m$, the desired value $\bar{\boldsymbol{x}} \in \mathbb{R}^n$ such that $\bar{\boldsymbol{y}} = \mathtt{F}(\bar{\boldsymbol{x}})$ is built by means of a generative architecture. Our method founds on the following three ideas:

(1) First, we embed the space $\mathbb{R}^n$ into a suitably-defined latent space $\mathbb{R}^h$, which is relevant to deal with input spaces associated with complex representations going beyond plain numerical values (such as, for instance, with one-hot encodings of physical structures).

(2) Second, we pick a point in the latent space (corresponding to the solution in the input space) via a guided exploration starting from a convenient initial configuration. In particular, we first identify the k pairs $(\boldsymbol{x}_{j_\ell}, \boldsymbol{y}_{j_\ell}) \in \mathcal{D}$ for which the distance between $\boldsymbol{y}_{j_\ell}$ and $\bar{\boldsymbol{y}}$ is minimized, i.e., we look for the nearest neighbors of $\bar{\boldsymbol{y}}$. Then, we provide the input parts of such pairs (namely, $\boldsymbol{x}_{j_\ell}$) as starting *seeds*, and we design an architecture that selects an initial configuration[2] which comes as a linear combination of that seeds. Eventually, the generator is feed with random noise to implement a mechanism that explores the latent space all around that initial configuration.

(3) Finally, our architecture takes care of dealing with the feasibility constraints that restrict the space $\mathbb{R}^n$ to some subset $S \subseteq \mathbb{R}^n$ associated with a proper one-hot encoding of the categorical features (e.g., the materials of the device structure). Our solution choice is to deal with that constraints at the embedding, by using an autoencoder architecture that forces each value in the latent space—even though associated with an input that has been never seen in the training data—to be mapped into a valid encoding in S. This guarantees that all adjustments to the weights needed to optimize the loss function correspond to a *meaningful* exploration of the latent space.

In the following, we first detail the neural architecture and then discuss the algorithm to compute $\bar{\boldsymbol{x}}$, given $\bar{\boldsymbol{y}}$.

1.3.1 Neural Network Architecture

The Generative Inverse Design network (short: GIDNET) we propose to address inverse design is illustrated in Figure 1.3.

There, we can first notice an *encoder* E and a *decoder* D, which allow GIDNET to work on an embedding of $\mathbb{R}^n$ into a suitable latent space $\mathbb{R}^h$. In fact, E and D define an autoencoder that is pre-trained over the input-components of $\mathcal{D}$, namely on $\{\boldsymbol{x}_i\}_{i \in \{1,\ldots,q\}}$. The autoencoder is a standard one, except for how it handles the categorical features. Indeed, let $\boldsymbol{x}_{i,1}, \ldots, \boldsymbol{x}_{i,\ell}$ be the components of $\boldsymbol{x}_i$ that one-hot encode some given categorical feature over ℓ alternatives, i.e., such that $\sum_{j=1}^{\ell} \boldsymbol{x}_{i,j} = 1$ and $\boldsymbol{x}_{i,j} \in \{0, 1\}$, for each $j \in \{1, \ldots, \ell\}$. Let $\boldsymbol{x}'_{i,1} = \mathtt{D}(\mathtt{E}(\boldsymbol{x}_{i,1})), \ldots, \boldsymbol{x}'_{i,\ell} = \mathtt{D}(\mathtt{E}(\boldsymbol{x}_{i,\ell}))$ be the corresponding values produced by the autoencoder. Then, the last layer of D is defined to be a *softmax*, so that $\sum_{j=1}^{\ell} \boldsymbol{x}'_{i,j} = 1$ and $0 \leq \boldsymbol{x}'_{i,j} \leq 1$, for each $j \in \{1, \ldots, \ell\}$. In fact, such values might be arbitrary far from Boolean values; for instance, the autoencoder can well produce an output where each value

[2]Different methods (i.e., not necessarily based on the nearest neighbors) can select the starting seeds, leading to different designs that can be further evaluated and selected by the user.

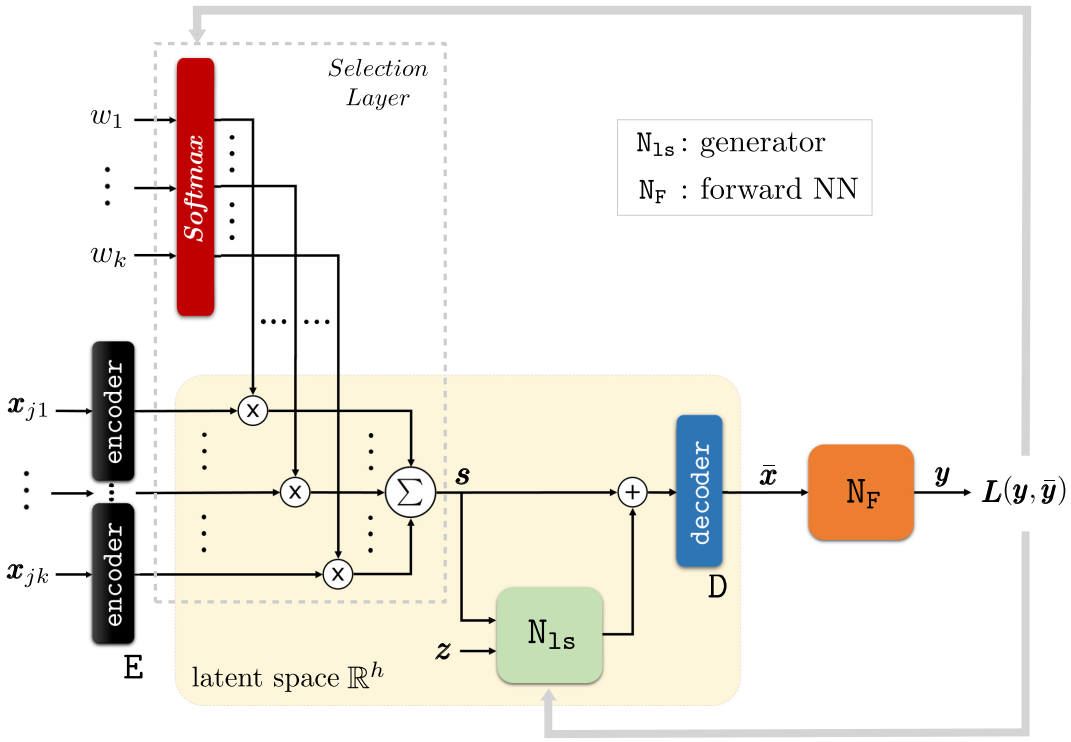


Figure 1.3: GIDNET architecture.

is close to $1/\ell$. To avoid these circumstances, we use the following loss function to train the autoencoder:

$$\sum_{i=1}^{q} ||\mathtt{D}(\mathtt{E}(\boldsymbol{x}_i)) - \boldsymbol{x}_i||_2 + \lambda_0 \cdot \Gamma(\boldsymbol{x}'_{i,1}, ..., \boldsymbol{x}'_{i,\ell}), \tag{1.1}$$

where $\Gamma(\boldsymbol{x}'_{i,1}, ..., \boldsymbol{x}'_{i,\ell}) = -((\boldsymbol{x}'_{i,1})^2 + \cdots + (\boldsymbol{x}'_{i,\ell})^2)$. In fact, the former term is the reconstruction error, while the latter (whose impact can be tuned via the factor $\lambda_0 \geq 0$) is a regularization term leading to configurations where precisely one of the components in $\{\boldsymbol{x}'_{i,1}, ..., \boldsymbol{x}'_{i,\ell}\}$ approaches to 1. The second block that emerges from Figure 1.3 is a network $\mathtt{N_F}$ simulating the forward function $\mathtt{F}$. Its weights are trained on $\mathcal{D}$ by considering the loss $\sum_{i=1}^{q} ||\mathtt{N_F}(\boldsymbol{x}_i) - \boldsymbol{y}_i||_2$. That is, we would like that the network is capable of predicting the output $\boldsymbol{y}_i$ given the input $\boldsymbol{x}_i$. In fact, the use of this network is reminiscent of some of the approaches in Figure 1.2.

After having discussed $\mathtt{E}$, $\mathtt{D}$ and $\mathtt{N_F}$, we are now in the position of appreciating the architecture of GIDNET. Note that it receives as input k seeds, say $\boldsymbol{x}_{j_1}, ..., \boldsymbol{x}_{j_k}$. These seeds are then embedded into the h-dimensional latent space via the encoder $\mathtt{E} : \mathbb{R}^n \rightarrow \mathbb{R}^h$. Then, a selection layer is used to provide GIDNET with the freedom to pick one of these seeds (or a combination of them); this is achieved by introducing the weights $w_1, ..., w_k$, and simply defining:

$$\boldsymbol{s} = \sum_{\ell=1}^{k} \mathtt{E}(\boldsymbol{x}_{j_\ell}) \times \frac{e^{w_\ell}}{e^{w_1} + \cdots + e^{w_k}}. \tag{1.2}$$

Intuitively, $\boldsymbol{s}$ defines a good point from which starting the exploration. Indeed, GIDNET includes a conditional generator $\mathtt{N_{ls}}$ which takes care of the exploration of the latent space

(in the inverse computation phase described below); from the architectural viewpoint, we just note here that it receives as input $\boldsymbol{s}$ plus some random noise $\boldsymbol{z}$ and produces a value $\mathtt{N}_{\mathtt{ls}}(\boldsymbol{s}, \boldsymbol{z}) \in \mathbb{R}^h$. This is used to define the output value that will be returned after decoding via $\mathtt{D} : \mathbb{R}^h \rightarrow \mathbb{R}^n$:

$$\bar{\boldsymbol{x}} = \mathtt{D}(\boldsymbol{s} + \mathtt{N}_{\mathtt{ls}}(\boldsymbol{s}, \boldsymbol{z})). \tag{1.3}$$

Eventually, to check the quality of $\bar{\boldsymbol{x}}$, the architecture uses $\mathtt{N}_{\mathtt{F}}$ to simulate the given forward function $\mathtt{F}$ (i.e., $\boldsymbol{y} = \mathtt{N}_{\mathtt{F}}(\bar{\boldsymbol{x}})$).

1.3.2 Inverse Computation

After $\mathtt{E}$, $\mathtt{D}$, and $\mathtt{N}_{\mathtt{F}}$ have been trained over $\mathcal{D}$, we freeze their weights. Hence, in the inverse computation phase, the trainable weights are $w_1, .., w_k$ plus all weights of the conditional generator $\mathtt{N}_{\mathtt{ls}}$. These weights are re-initialized and re-trained each time an output value $\bar{\boldsymbol{y}}$ is given and we look for an input $\bar{\boldsymbol{x}}$ such that $\bar{\boldsymbol{y}} = \mathtt{F}(\bar{\boldsymbol{x}})$. The algorithm is next discussed.

First, we have to compute the k seeds $\boldsymbol{x}_{j_1}, ..., \boldsymbol{x}_{j_k}$. To this end, we define $\boldsymbol{x}_{j_1}$ as $\mathbf{0} \in \mathbb{R}^n$ (that is, we allow GIDNET to neutralize the seeds), and we take $\boldsymbol{x}_{j_2}, ..., \boldsymbol{x}_{j_k}$ from the pairs in $\mathcal{D}$, i.e., $\{(\boldsymbol{x}_{j_2}, \boldsymbol{y}_{j_2}), ..., (\boldsymbol{x}_{j_k}, \boldsymbol{y}_{j_k})\} \subseteq \mathcal{D}$, in a way that there is no pair $(\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathcal{D}$ such that $||\boldsymbol{y}_i - \bar{\boldsymbol{y}}||_2 < \max_{i \in \{2,...k\}} ||\boldsymbol{y}_{j_k} - \bar{\boldsymbol{y}}||_2$; in words, we take the k-1 points whose corresponding forward values are the closest to $\bar{\boldsymbol{y}}$ over all pairs in $\mathcal{D}$—ties are resolved at random.

In fact, the seeds are just constant values that initialize the starting point $\boldsymbol{s}$ according to Equation (1.2). Concerning the trainable weights, instead, we initialize $\mathtt{N}_{\mathtt{ls}}$ at random while we take $w_i = 1/k$, for each $i \in \{1, ...k\}$, thereby starting the exploration from the *centroid* of the k seeds.

Given this initial configuration, we then train GIDNET via a number of training steps where $\bar{\boldsymbol{y}}$ is fixed and where different samples $\boldsymbol{z}$ from a uniform distribution are given as input to the conditional generator $\mathtt{N}_{\mathtt{ls}}$. Intuitively, by starting from the seed $\boldsymbol{s}$, each time a different value $\boldsymbol{z}$ is feed to the network, we aim at adjusting the weights of the conditional generator to end up with an output value $\boldsymbol{y}$ that gets even closer to $\bar{\boldsymbol{y}}$. Therefore, the sequence of the training steps corresponds to a path in the latent space leading to a point that is a good solution. In fact, we define the loss function $\boldsymbol{L}(\boldsymbol{y}, \bar{\boldsymbol{y}})$:

$$||\boldsymbol{y} - \bar{\boldsymbol{y}}||_2 + \lambda_1 \cdot \Gamma(w_1, ..., w_k) + \lambda_2 \cdot \varepsilon(\boldsymbol{x}),$$

where $\Gamma(w_1, ..., w_k) = -(w_1^2 + \cdots + w_k^2)$ is the same regularization function used for training $\mathtt{E}$ and $\mathtt{D}$, and where the term $\varepsilon(\boldsymbol{x}) = ||\mathtt{D}(\mathtt{E}(\boldsymbol{x})) - \boldsymbol{x}||_2$ is aimed at minimizing the reconstruction error of the autoencoder on input $\boldsymbol{x}$.

Eventually, $w_1, ..., w_k$ and the weights in $\mathtt{N}_{\mathtt{ls}}$ are updated by minimizing the loss $\boldsymbol{L}(\boldsymbol{y}, \bar{\boldsymbol{y}})$. Hence, we explore the latent space while allowing GIDNET to select a different initial configuration $\boldsymbol{s}$. The value $\boldsymbol{x}$ that is returned as output by GIDNET is the one computed via Equation (1.3) during the training phase, and for which the distance between $\mathtt{N}_{\mathtt{F}}(\boldsymbol{x})$ and $\bar{\boldsymbol{y}}$ is minimized. Furthermore, given that the blocks $\mathtt{E}$, $\mathtt{D}$, and $\mathtt{N}_{\mathtt{F}}$ are trained independently on the number k of seeds, it makes sense to re-execute the inverse computation step over different subsets of seeds by taking the best result obtained over the various configurations. By doing so, we enlarge the portion of the latent space that is explored. Similarly, we re-execute the inverse computation step by changing the learning rates and taking the best results derived over them.

	Name	$Dim(\boldsymbol{x})$	$Dim(\boldsymbol{y})$	Source
$\mathcal{D}_i$	f_i	3	2	[*here*]
$\mathcal{D}_5$	Ballistics	4	1	[50]
$\mathcal{D}_6$	Robotic arm	4	2	[50]
$\mathcal{D}_7$	Sine Wave	2	1	[48]
$\mathcal{D}_8$	Multilayer Stacks	5	256	[51]

Table 1.1: Real-Valued Datasets.

		Other methods				
		(a)	**(c)**	**(d)**	**(e*)**	**(f)**
$\mathcal{D}_1$	L2	0.030 (±0.022)	0.155 (±0.570)	0.018 (±0.012)	**0.004** (±0.0256)	0.112 (±0.286)
	MAE	0.019 (±0.015)	0.092 (±0.343)	0.012 (±0.009)	**0.002** (±0.0166)	0.068 (±0.172)
	RMSE	0.037	0.590	0.022	0.0259	0.307
$\mathcal{D}_2$	L2	0.009 (±0.011)	0.007 (±0.012)	0.017 (±0.022)	0.007 (±0.0392)	0.009 (±0.014)
	MAE	0.006 (±0.008)	0.005 (±0.009)	0.011 (±0.015)	0.004 (±0.0224)	0.006 (±0.009)
	RMSE	0.014	0.014	0.028	0.040	0.016
$\mathcal{D}_3$	L2	0.010 (±0.021)	0.026 (±0.07)	0.009 (±0.023)	0.016 (±0.073)	0.022 (±0.051)
	MAE	0.006 (±0.012)	0.015 (±0.04)	0.005 (±0.013)	0.0094 (±0.0417)	0.013 (±0.029)
	RMSE	0.023	0.075	0.025	0.0745	0.056
$\mathcal{D}_4$	L2	0.012 (±0.013)	0.006 (±0.015)	0.024 (±0.029)	0.0012 (±0.0057)	0.007 (±0.015)
	MAE	0.008 (±0.008)	0.004 (±0.011)	0.016 (±0.020)	0.0007 (±0.0032)	0.005 (±0.010)
	RMSE	0.017	0.016	0.038	**0.0059**	0.016

		GIDNET					
		k=12	**k=7**	**k=5**	**k=3**	**k=1**	**no** $\mathtt{N_{1s}}$
$\mathcal{D}_1$	L2	0.005 (±0.005)	0.005 (±0.007)	**0.004** (±0.004)	0.005 (±0.006)	0.083 (±0.304)	0.040 (±0.021)
	MAE	0.003 (±0.003)	0.003 (±0.005)	0.003 (±0.003)	0.003 (±0.004)	0.055 (±0.198)	0.026 (±0.014)
	RMSE	**0.006**	0.009	**0.006**	0.007	0.315	0.045
$\mathcal{D}_2$	L2	**0.005** (±0.003)	0.006 (±0.005)	0.006 (±0.005)	0.006 (±0.008)	0.029 (±0.059)	0.054 (±0.023)
	MAE	**0.003** (±0.002)	0.004 (±0.003)	0.004 (±0.004)	0.004 (±0.005)	0.020 (±0.039)	0.035 (±0.015)
	RMSE	**0.006**	0.008	0.008	0.010	0.066	0.059
$\mathcal{D}_3$	L2	0.006 (±0.020)	**0.005** (±0.019)	**0.005** (±0.019)	**0.005** (±0.019)	0.033 (±0.116)	0.029 (±0.032)
	MAE	**0.003** (±0.011)	**0.003** (±0.011)	**0.003** (±0.011)	**0.003** (±0.011)	0.022 (±0.079)	0.018 (±0.018)
	RMSE	0.020	0.019	0.019	0.019	0.121	0.043
$\mathcal{D}_4$	L2	**0.005** (±0.014)	0.013 (±0.047)	0.012 (±0.049)	0.021 (±0.064)	0.586 (±0.688)	0.035 (±0.043)
	MAE	**0.003** (±0.009)	0.008 (±0.031)	0.008 (±0.032)	0.014 (±0.043)	0.395 (±0.482)	0.023 (±0.030)
	RMSE	0.014	0.048	0.051	0.067	0.903	0.055

Table 1.2: Results on $\mathcal{D}_1$,...,$\mathcal{D}_4$—the best values are in bold. Since the mean L2-norm of the output values over each dataset is bounded by 2, such performances range in a relative scale from 59% to about 1%. Ablation on $\mathtt{N_{1s}}$ is with $k = 12$.

1.4 Experiments

To shed lights on the behaviour of GIDNET, we next discuss the results of a thorough experimental activity, where we considered all methods discussed in Section 1.2, except the multi-branched approach which—as already mentioned—is often unviable in practice. All network architectures we implemented have been instantiated by means of a model-selection phase, based on a gridsearch approach over the hyperparameter space of the network topologies, the learning rates in the stochastic gradient descent algorithm, the initialization strategies, and the kernel constraints. For GIDNET as well as for the other output-dependent methods, in the inverse computation phase, given the value $\bar{\boldsymbol{y}}$, we considered different learning rates in $\{0.01, 0.05, 0.1, 0.5\}$ to compute $\bar{\boldsymbol{x}}$ such that $\bar{\boldsymbol{y}} = \mathtt{F}(\bar{\boldsymbol{x}})$, by taking the best result over them.

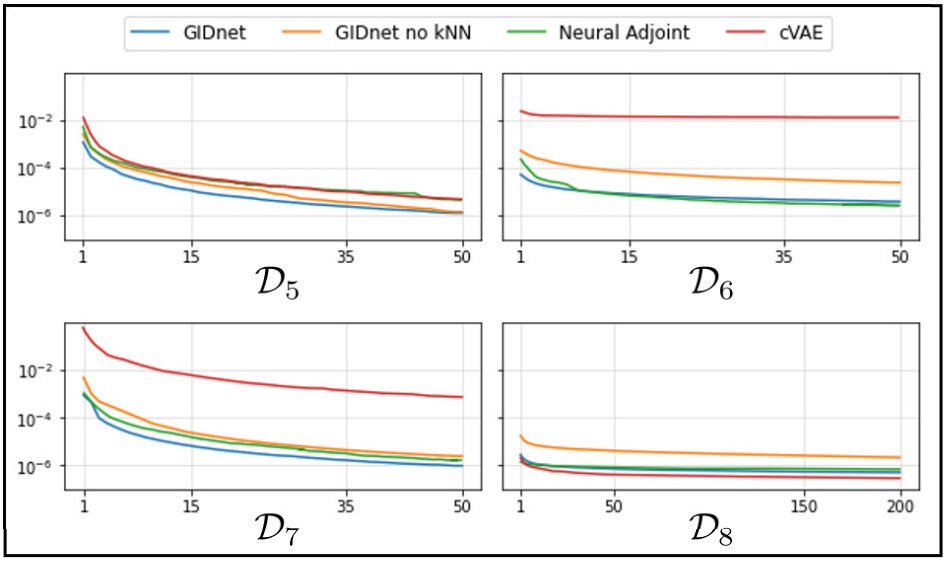


Figure 1.4: Results on $\mathcal{D}_5$,...,$\mathcal{D}_8$: mse by varying k on GIDNET, and T on Neural Adjoint and cVAE (implemented with restarting too).

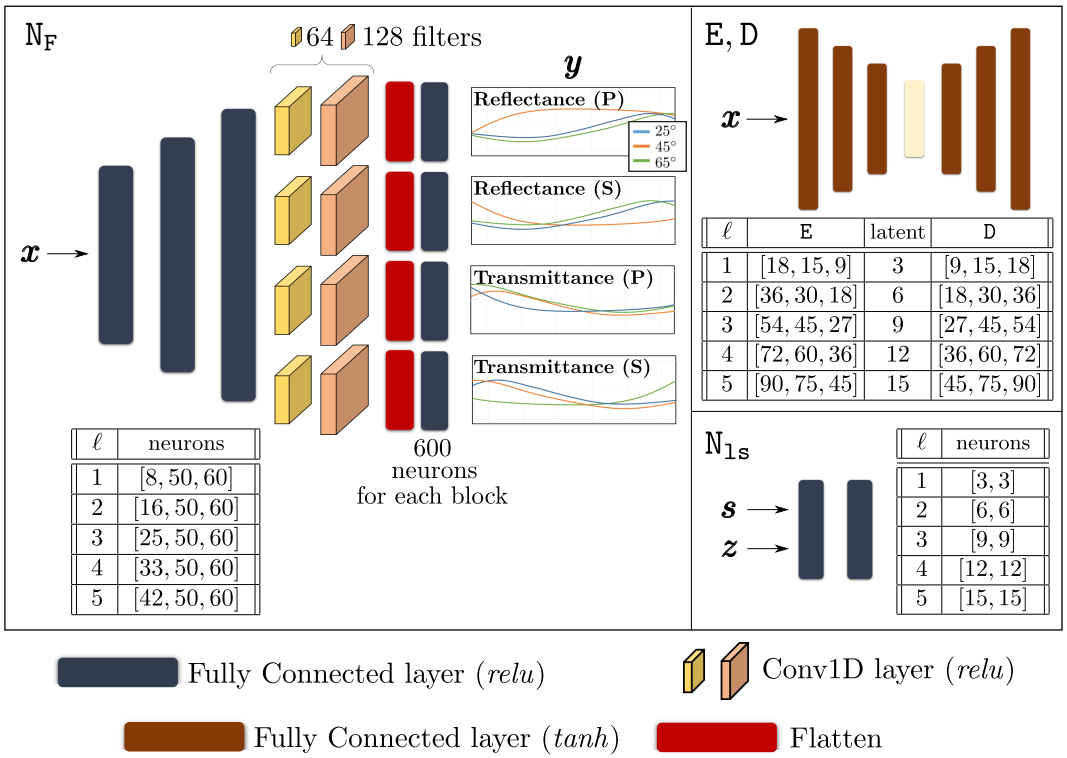
1.4.1 GIDNET on Real-Valued Functions

We first discuss the results on the datasets listed in Table 1.1, which are all defined on real-valued input spaces.

Datasets $\mathcal{D}_1, ..., \mathcal{D}_4$ are novel in the literature: each one comprises 10.000 pairs $(\boldsymbol{x}_i \in \mathbb{R}^3, \boldsymbol{y}_i \in \mathbb{R}^2)$ defined via classical functions emerging in physical systems and stressing the absence of a univocally-defined inverse value. For each dataset, we used 7.000 samples for training and 2.900 for validation during training; seed computation for GIDNET was carried out over these 9.900 samples. The remaining 100 samples were, instead, used for assessing the value of the metrics after the training phase has been completed. In particular, we consider L2-norm, MAE and RMSE.

A summary of our findings over $\mathcal{D}_1, ..., \mathcal{D}_4$ is reported in Table 1.2, from which the improvements provided by GIDNET over earlier methods clearly emerge. The figure also evidences the impact of varying the number k of seeds and of disabling the module $\mathtt{N_{ls}}$ implementing the conditional generation approach. The ablation study on k confirms the intuition that it make sense to enlarge the latent space to be explored by considering the best results over different number of seeds; in fact, it also emerges a trend of improvement at the growing of k. Moreover, it emerges that $\mathtt{N_{ls}}$ has a strong positive impact on the quality of the results; in fact, it is responsible of the exploration in the latent space (cf. Equation 1.3). And, finally, it is interesting to observe that GIDNET is rather robust w.r.t. statistical fluctuations over the different samples.

Datasets $\mathcal{D}_5, ..., \mathcal{D}_8$ instead appeared in earlier works and have been extensively used to benchmark earlier methods. Over them, the Neural Adjoint method (e*) emerged as the state of the art, together (in some cases) with the cVAE approach [48, 52, 50]. Accordingly, our experiments have been focused on assessing the performances GIDNET in comparison with them (by precisely adopting their experimental setting). Moreover, we also conducted an ablation study by selecting random initializations k seeds, rather than by taking the k nearest neighbors. Results are illustrated in Figure 1.4, which evidences the good performances of GIDNET even on these settings where the use of the latent space is trivialized by the low



ℓ	neurons
1	[8, 50, 60]
2	[16, 50, 60]
3	[25, 50, 60]
4	[33, 50, 60]
5	[42, 50, 60]

ℓ	E	latent	D
1	[18, 15, 9]	3	[9, 15, 18]
2	[36, 30, 18]	6	[18, 30, 36]
3	[54, 45, 27]	9	[27, 45, 54]
4	[72, 60, 36]	12	[36, 60, 72]
5	[90, 75, 45]	15	[45, 75, 90]

ℓ	neurons
1	[3, 3]
2	[6, 6]
3	[9, 9]
4	[12, 12]
5	[15, 15]

Figure 1.5: GIDNET on the photonic scenario, with varying ℓ. The architecture of $\mathtt{N_F}$ consists of 3 fully connected layers, leading to 4 parallel branches each one consisting of 2 convolutional layers; each branch is used to predict a specific waveform for the design. The architecture of the autoencoder consists of 3 encoding (fully connected) layers and of 3 decoding (fully connected) layers. The architecture of $\mathtt{N_{ls}}$ consists of 2 fully connected layers. For each layer and for each ℓ, the figure also reports—in tabular form—the corresponding number of neurons.

dimensionality of the input spaces; moreover, the ablation study further confirms the benefits of appropriately picking the k seeds.

1.4.2 GIDNET on Categorical Attributes

To assess the behaviour of GIDNET on a function involving categorical attributes, we considered a photonic scenario proposed by Lininger et al. ([3]), where the goal is to build a thin-film structure that gives rise to some desired reflectance and transmittance spectra.

Dataset Description. Each structure is made of up to 5 layers, each with thickness within the range $[1, 60]$nm and whose material can be Ag, Al2O3, ITO, Ni, or TiO2. The input space is therefore $\mathbb{R}^{\ell\times(1+5)}$, with ℓ being the number of layers—indeed, for each layer, we have to represent its thickness plus the material as a one-hot encoding over 5 alternatives. Each structure is associated with reflectance and transmittance spectra, obtained via the transfer matrix method [53], for two polarizations, at the incident angles of 25, 45, and 65 degrees, for 200 equally spaced points over the range $[450, 950]$nm and with values in $[0, 1]$. Thus, the output space is $\mathbb{R}^{2\times2\times3\times200}$.

For each $\ell \in \{1, .., 5\}$, we use 106.820 samples for training, 45.780 samples for validation, and 2.000 samples for computing the metric. The seeds needed by GIDNET are, as

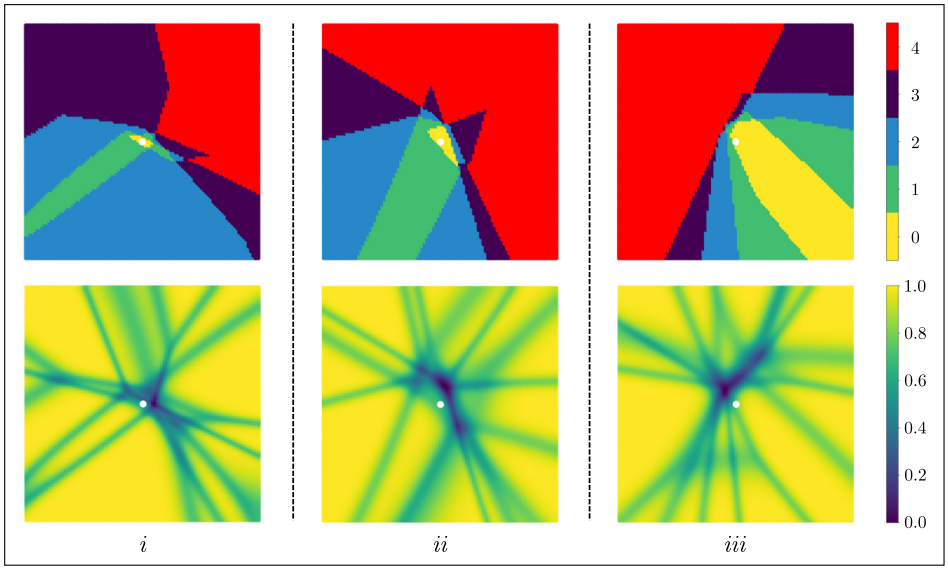


Figure 1.6: A snapshot of the latent space for a sample with $\ell = 4$ layers, centered in the initial point of the exploration. The plots report the "distance" between the corresponding point in the latent space and the given initial point: plots on the top refer to the distance in terms of the number of layers with different materials, while plots on the bottom refer to the average *squared Euclidean norm*. The latent space has 12 dimensions (see Figure 1.5), and each setting (*(i)*, *(ii)*, and *(iii)*) explores two dimensions among them, while keeping fixed all the remaining ones.

usual, computed over the traning and validation samples.

For a closer look at the dataset of Lininger et al. ([3]), note that about 5% of the samples have very similar spectra, with a significant portion of them being associated to structures different from each other.The absence of clearly defined inverse values on these samples is congenial to stress the capabilities of inverse design approaches.

Compared Methods and Metric. The GIDNET configurations resulting from the model-selection phase are reported in Figure 1.5. During the inverse computation step, GIDNET explores a space with $k \in \{3, 6, 9, 12, 18, 30, 50\}$ and learning rates in $\{0.01, 0.05, 0.1, 0.5\}$, by taking the best results achieved over them. A similar exploration over the learning rates has been implemented for the other output-dependent methods being tested. Performances of the methods have been compared via the spectral root mean squared error [3], srmse for short, between the spectra (in $\mathbb{R}^{2\times2\times3\times200}$) associated to the metamaterial designed by the methods and the actual ones.

In particular, to take care of the feasibility of the solutions, if $\bar{\boldsymbol{x}}$ are the values returned by the tested methods, then we first pre-process them by suitably rounding the components associated to the one-hot encodings of the materials over the various layers. Subsequently, for the resulting values (in fact, correctly encoding some device structures), we use the physical simulator at hand [53] to compute their associated spectra, and the quality of the results (srmse) is eventually assessed over them.

A Look at the Latent Space. As the crucial ingredient of GIDNET is its ability to work at the level of a suitably defined latent space, we took a closer look at it in Figure 1.6 by considering a sample taken from the dataset with $\ell = 4$ layers. In particular, the plots in

ℓ	**3**	**4**	**5**
GIDNET	0.009 ($\pm$0.024)	0.009 ($\pm$0.022)	0.011 ($\pm$0.022)
GIDNET no $\texttt{N}_{\texttt{ls}}$	0.060 ($\pm$0.099)	0.046 ($\pm$0.072)	0.043 ($\pm$0.064)

Table 1.3: Impact of the exploration of the latent space.

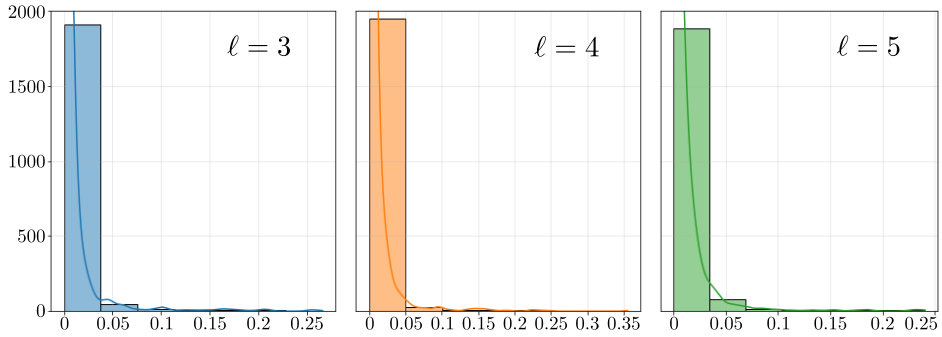


Figure 1.7: Histograms of srmse for GIDNET, in the photonic scenario, for different number ℓ of material layers.

ℓ		(a)	(c)	(d)	(e*)	(f)	[3]	GIDNET	$\lambda_0 = 0$
1	srmse	0.007 ($\pm$0.008)	0.062 ($\pm$0.09)	0.132 ($\pm$0.21)	0.033 ($\pm$0.054)	0.132 ($\pm$0.21)	0.009	**0.006** ($\pm$0.031)	0.019 ($\pm$0.051)
	one-hot	0.924 ($\pm$0.527)	0.236 ($\pm$0.023)	0.798 ($\pm$0.335)	0.302 ($\pm$0.114)	0.798 ($\pm$0.335)	-	**0.999** ($\pm$0.017)	0.275 ($\pm$0.012)
2	srmse	0.010 ($\pm$0.012)	0.098 ($\pm$0.117)	0.117 ($\pm$0.133)	0.049 ($\pm$0.050)	0.117 ($\pm$0.133)	**0.007**	0.008 ($\pm$0.032)	0.077 ($\pm$ 0.057)
	one-hot	0.921 ($\pm$0.567)	0.233 ($\pm$0.020)	0.353 ($\pm$0.139)	0.261 ($\pm$0.035)	0.353 ($\pm$0.139)	-	**0.992** ($\pm$0.037)	0.396 ($\pm$0.075)
3	srmse	0.015 ($\pm$0.032)	0.084 ($\pm$0.104)	0.031 ($\pm$0.034)	0.059 ($\pm$0.053)	0.083 ($\pm$0.09)	0.016	**0.009** ($\pm$0.024)	0.052 ($\pm$ 0.047)
	one-hot	0.902 ($\pm$0.563)	0.225 ($\pm$0.022)	0.235 ($\pm$0.013)	0.249 ($\pm$0.017)	0.230 ($\pm$0.065)	-	**0.974** ($\pm$0.053)	0.321 ($\pm$0.048)
4	srmse	0.038 ($\pm$0.162)	0.078 ($\pm$0.100)	0.042 ($\pm$0.043)	0.061 ($\pm$0.051)	0.078 ($\pm$0.081)	0.127	**0.009** ($\pm$0.022)	0.057 ($\pm$ 0.040)
	one-hot	0.825 ($\pm$0.538)	0.226 ($\pm$0.021)	0.220 ($\pm$0.009)	0.250 ($\pm$0.016)	0.232 ($\pm$0.068)	-	**0.940** ($\pm$0.068)	0.433 ($\pm$0.077)
5	srmse	0.050 ($\pm$0.087)	0.053 ($\pm$0.065)	0.050 ($\pm$0.054)	0.059 ($\pm$0.053)	0.069 ($\pm$0.071)	0.095	**0.011** ($\pm$0.022)	0.047 ($\pm$0.056)
	one-hot	0.663 ($\pm$0.267)	0.229 ($\pm$0.019)	0.222 ($\pm$0.010)	0.251 ($\pm$0.013)	0.232 ($\pm$0.067)	-	**0.911** ($\pm$0.082)	0.647 ($\pm$0.145)

Table 1.4: Performances of GIDNET, contrasted with earlier approaches in the literature—recall that spectra values are in $[0, 1]$. Lininger et al. referred to as [3].

Figure 1.6 are centered in the initial point of the exploration defined by GIDNET (with $k = 3$); then, the three settings (*i*, *ii*, and *iii*) in the figure correspond to an exploration of the latent space over an interval of that point defined by moving along two different dimensions and by fixing all remaining ones. For each setting, the top part reports the number of materials that change compared to the configuration in the initial point, while those at the bottom report the average *squared Euclidean norm* of the encoding associated with the materials over the various layers. Note that the latent space changes all (4) materials in the region, and that the encoding is always very close to a true one-hot encoding (with a squared Euclidean norm rather close to 1), except in the tiny regions associated with the frontiers of different configurations of the materials (where the squared Euclidean norm approaches to 0).

A Look at the Selection Layers. The other crucial ingredient of GIDNET is the use of k seeds to define an appropriate initial configuration for the exploration (cf. $\boldsymbol{s}$ in Equation (1.3)). Figure 1.8 illustrates the behaviour of the selection layer in terms of the percentage of samples for which it converges into a linear combination of the seeds—formally, in the inverse computation phase, no weight associated with some seed gets a value greater than the reported threshold. In some cases, GIDNET converges on some specific seed; but, in many cases, it retains the ability to explore a large portion of the search space by converging into a proper linear combination of the seeds; hence, the selection layer cannot be replaced by an approach

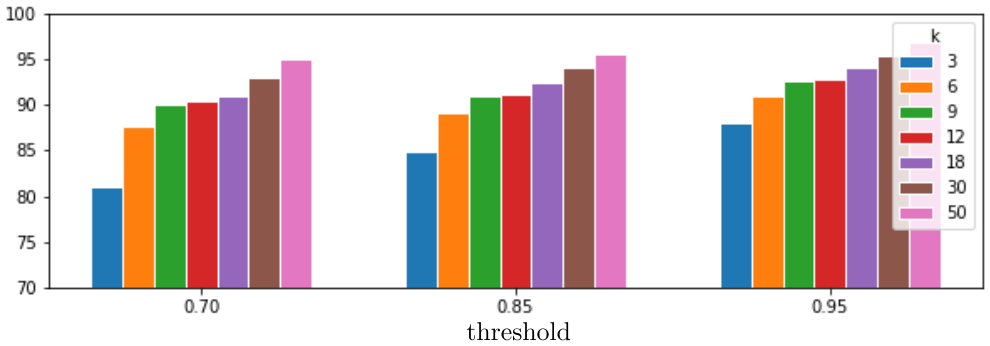


Figure 1.8: Percentage of samples for which the selection layer does not converge to a specific seed, but to a proper linear combination of them (at the varying of their number k).

ℓ		★	GIDNET	GIDNET no kNN
3	srmse	0.044 ($\pm$0.051)	**0.009** ($\pm$0.024)	0.024 ($\pm$0.035)
	one-hot	0.248 ($\pm$0.005)	**0.974** ($\pm$0.053)	0.926 ($\pm$0.083)
4	srmse	0.109 ($\pm$0.122)	**0.009** ($\pm$0.022)	0.052 ($\pm$0.059)
	one-hot	0.229 ($\pm$0.01)	**0.940** ($\pm$0.068)	0.953 ($\pm$0.0643)
5	srmse	0.058 ($\pm$0.059)	**0.011** ($\pm$0.022)	0.028 ($\pm$0.042)
	one-hot	0.205 ($\pm$0.004)	**0.911** ($\pm$0.082)	0.964 ($\pm$0.047)

Table 1.5: Comparison with (★) [Gómez-Bombarelli *et al.*, 2018].

that iterates over the seeds, by considering each of them as a fixed initial configuration for the exploration. Moreover, note that the higher is the number k of seeds, the more probable is converging to a proper linear combination rather than just to a specific seed.

A Look at the Conditional Generator. Finally, the last architectural component of interest is the conditional generator $\texttt{N}_{\texttt{ls}}$. The significance of its role can be appreciated by looking at Table 1.3, where the performances of GIDNET are shown depending on whether the module is enabled. In particular, without $\texttt{N}_{\texttt{ls}}$, GIDNET lacks the ability of searching over the latent space and it conceptually reduces to optimizing a random point (as in [36]).

Performances. Let us first consider Figure 1.7 reporting the histograms of the srmse for $\ell \in \{3, 4, 5\}$. It emerges that GIDNET is capable of designing metamaterials whose spectral responses are quite close to the desired ones; errors are rather small in general, and negligible in most of the cases. Moreover, there is a trend of (very mild) deterioration in the performances at the growing of the complexity (in terms of number of layers) of the space to be explored by GIDNET.

A summary of the results comparing GIDNET with earlier approaches is then reported in Table 1.4—for the domain specific method, we report the results that are published by the authors. Note that GIDNET produces solutions with very low $\mathbf{srmse}$, and improvements are significant given that spectra values are in $[0, 1]$. In particular, note that the cases $\ell \in \{1, 2\}$ are solved efficiently by all methods, while significant differences emerge on $\ell \in \{3, 4, 5\}$, that is, at the growing of the complexity of the underlying search space. Eventually, the summary also reports the feasibly of the solutions in terms of the quality of the one-hot encodings (prior that the results are rounded and adjusted for computing the associated

ℓ	(c)	(e)	(f)	GIDNET
1	8.976 ($\pm$0.297)	2.510 ($\pm$0.097)	7.621 ($\pm$0.113)	7.191 ($\pm$0.332)
2	8.876 ($\pm$0.151)	2.466 ($\pm$0.079)	7.639 ($\pm$0.111)	7.605 ($\pm$0.263)
3	8.809 ($\pm$0.199)	2.504 ($\pm$0.103)	7.728 ($\pm$0.099)	7.841 ($\pm$0.334)
4	13.01 ($\pm$0.185)	2.549 ($\pm$0.122)	7.588 ($\pm$0.103)	7.916 ($\pm$0.255)
5	16.16 ($\pm$0.198)	2.490 ($\pm$0.093)	7.707 ($\pm$0.089)	8.074 ($\pm$0.175)

Table 1.6: Average timings (s) per sample required by the output-dependent methods for inverse computation.

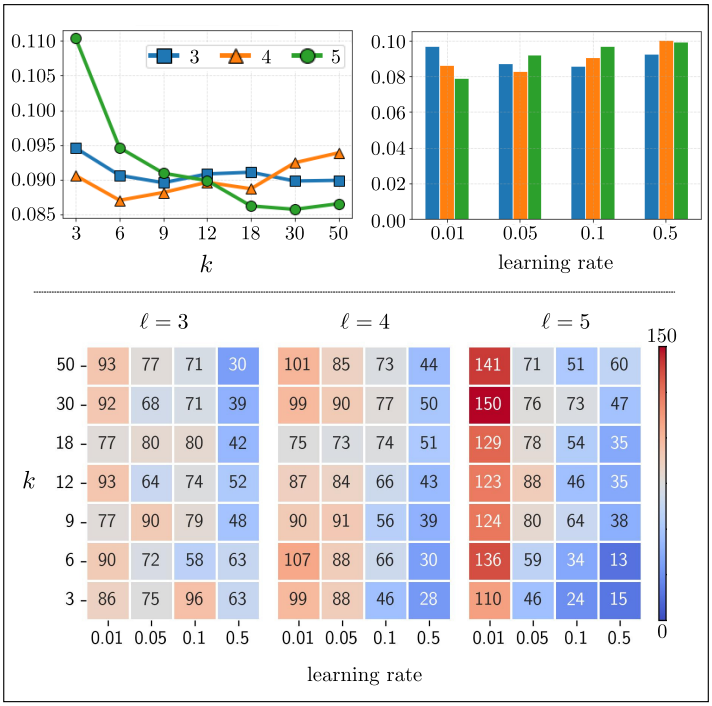


Figure 1.9: GIDNET performances by averaging on learning rates (top-left) and number of seeds (top-right). At the bottom, each heatmap reports the number of samples for which the best solution has been found for a given combination of k and learning rate.

spectra), which confirms the intuition we have derived from Figure 1.6.

And, finally, it also evidences the significant impact of the strategy we implemented in the autoencoder to deal with the categorical features (cf. $\lambda_0 = 0$ in Equation (1.1)). Indeed, by loosing the ability to map the latent space into feasible designs, performances of GIDNET rapidly deteriorates.

In addition to the methods reported in Table 1.4, we also assessed the performances of the method proposed by [36] (and implemented by fine-tuning the building blocks trained for GIDNET), which has been indeed specifically conceived to deal with categorical attributes. That methods works at the level of the latent space (see Section 1.2), but lacks of a mechanism to explore that space starting from a meaningful initialization. Results in Table 1.5 confirm that the exploration of GIDNET is crucial to achieve higher levels of performance and, as in Figure 1.4, they further confirm the benefits of picking k nearest neighbors for the initialization compared to a random initialization (column "no kNN").

The impact of the number k of seeds and of the learning rates is then analyzed in Fig-

ure 1.9. Observe that increasing the number of seeds as well as reducing the learning rate lead to better performances; in particular, this emerges more clearly over the most complex scenario with $\ell = 5$.

Finally, we shed lights on the time requirements, by conducting comparative experiments, on the same phyton environment, over an Intel(R) Xeon(R) Processor E5-2670 v3 (2.30GHz), 24 cores single thread and 96GB of RAM—for training the architectures (before inverse computation), we also use an NVIDIA 2080 GPU. In fact, GIDNET is an output-dependent method, which requires some computational efforts on every structure that has to be designed. While this is hardly relevant in practical design applications (unless timings make the approach unviable), for the sake of completeness, we report in Table 1.6 such information for GIDNET and all other output-dependent methods.

1.5 Conclusions and Discussion

We have proposed a deep-learning architecture improving on the performance of existing methods by addressing inverse design by means of a guided exploration of the latent space. In fact, it is known that search-based approaches are effective in domains where the space of the parameters is rather narrow, but they are outperformed by deep learning methods over high dimensional spaces. An interesting avenue of research is, therefore, to define an hybrid method where GIDNET is coupled with a genetic algorithm to explore the search space (in the spirit, e.g.,of [54]).

Similarly, it would be interesting to couple GIDNET with logic-based reasoning engines, in settings where inverse design amounts at looking for the (truth) value of some variables that will eventually lead to satisfy some desired logical goal; in particular, we are interested in considering cost-based settings [55] and precedence constraints to express temporal properties [56].

We conclude by noticing that a different—though related—problem is sometimes addressed in the context of inverse design, consisting of learning distributions of existing designs in order to synthesize new designs via sampling (e.g. [57, 58]). In this case, we do not a-priori enforce complex properties such as the spectral responses; however, conditioning methods can be used to enforce that samples meet some scalar requirement, such as a performance metric. A natural avenue of further research is, therefore, to assess whether GIDNET can be adapted to effectively address this related problem too.

1.6 Additional Material: A Deeper Dive into Choices and Metrics

We advocate this final section to the choices made during the hyperparameter tuning phases of models. Additional details about the datasets and the metrics are provided for experimental activity conducted on real-valued settings as well as on a setting characterized by categorical attributes.

1.6.1 GIDNET on Real-Valued Functions

Performances of GIDNET and the other methods have been assessed on the datasets in Table 1.1.

Dataset Description $\mathcal{D}_1, ..., \mathcal{D}_4$

Let us first consider a setting consisting of four synthetic datasets, $\mathcal{D}_1, ..., \mathcal{D}_4$, each one comprising 10.000 pairs $(\boldsymbol{x}_i \in \mathbb{R}^3, \boldsymbol{y}_i \in \mathbb{R}^2)$ defined to stress the absence of a univocally-defined inverse value, as it clearly emerges from Figure 1.10. Indeed, we proceeded as follows:

- $\boldsymbol{x}_{i,1}$ and $\boldsymbol{x}_{i,2}$ are random variables uniformly distributed in the interval $[-1, 1]$;
- $\boldsymbol{x}_{i,3}$ is constrained to take values from a function $f_d : \mathbb{R}^2 \rightarrow \mathbb{R}$ (defined over the above components), hence simulating physical/technological constraints on the input space; in particular,
  - $f_1(\boldsymbol{x}_{i,1}, \boldsymbol{x}_{i,2}) = \boldsymbol{x}_{i,1}^2 + \boldsymbol{x}_{i,2}^2$;
  - $f_2(\boldsymbol{x}_{i,1}, \boldsymbol{x}_{i,2}) = \boldsymbol{x}_{i,1}^3 + \boldsymbol{x}_{i,2}^3$;
  - $f_3(\boldsymbol{x}_{i,1}, \boldsymbol{x}_{i,2}) = \sin(\boldsymbol{x}_{i,1}) \times e^{\boldsymbol{x}_{i,1}^2 + \boldsymbol{x}_{i,2}^2}$;

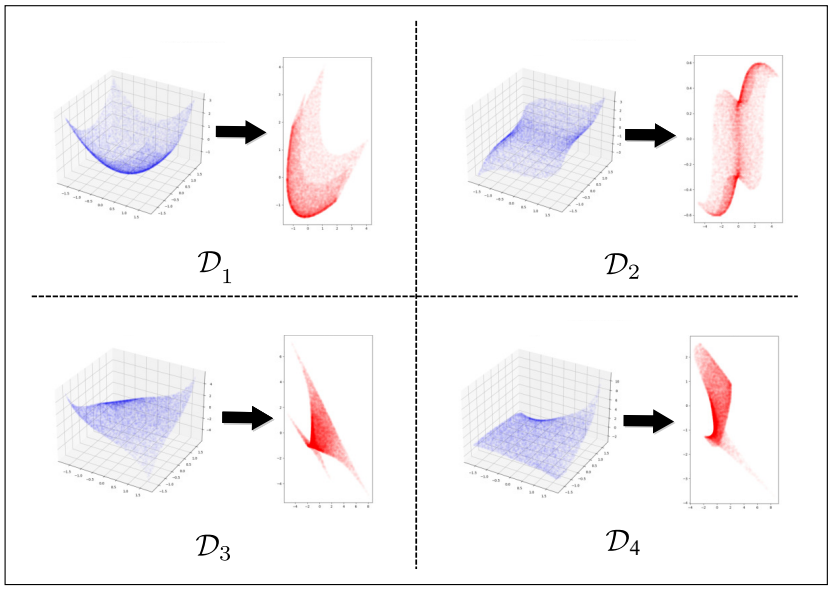


Figure 1.10: Datasets $\mathcal{D}_1, ..., \mathcal{D}_4$.

- $f_4(\boldsymbol{x}_{i,1}, \boldsymbol{x}_{i,2}) = \sin(\boldsymbol{x}_{i,1}) \times e^{\boldsymbol{x}_{i,1}^3 + \boldsymbol{x}_{i,2}^3}$;

- $\boldsymbol{y}_i$ is obtained from $\boldsymbol{x}_i$ by applying a linear transformation (with random coefficients uniformly distributed in the interval $[-1, 1]$), and by subsequently truncating its components at the second decimal digit; the rationale here is to enforce that a number of pairs exists in $\mathcal{D}_d$, for each $d \in \{1, ..., 4\}$, with the same output value and different input values—as to stress the capacity of the networks to deal with the non-uniqueness problem.

Dataset Description $\mathcal{D}_5, ..., \mathcal{D}_8$

In addition to the above datasets, which are novel in the literature, we tested GIDNET performances on four well established state of the art benchmark datasets for inverse design. These datasets are described below.

- **Ballistics** $\mathcal{D}_5$ [50]: this is a physically motivated problem in the 2D plane which arises when an object is thrown from a starting position $(x_1; x_2)$ with angle x_3 and initial velocity x_4. The object's trajectory $\boldsymbol{T}(t)$ can be computed as

  - $T_1(t) = x_1 - \frac{v_1 m}{k} \cdot \left(e^{\frac{kt}{m}} - 1\right)$
  - $T_2(t) = x_2 - \frac{m}{k^2} \cdot \left((gm + v_2 k) \cdot \left(e^{\frac{kt}{m}} - 1\right) + gtk\right)$, and
  - $y = T_1(t^*)$ where the t^* is the solution of $T_2(t^*) = 0$

  with $v_1 = x_4 \cdot cos(x_3)$, $v_2 = x_4 \cdot sin(x_3)$, and for a given gravity g, mass m and air resistance k. The dataset was generated by the python simulator provided by [48] choosing $x_1 \sim \mathcal{N}(0, \frac{1}{4})$, $x_2 \sim \mathcal{N}(\frac{3}{2}, \frac{1}{4})$, $x_3 \sim \mathcal{U}(9, 72)$ and $x_4 \sim Poisson(15)$.

- **Robotic Arm** $\mathcal{D}_6$ [50]: this benchmark was originally introduced by [59]. It is inspired by a geometrical problem which aim is to compute the starting height x_1 and three joint angles x_2, x_3, x_4 given a robotic arm's final position (y_1, y_2). The forward relationship is defined as:

  - $y_1 = l_1 \sin(x_2) + l_2 \sin(x_3 - x_2) + l_3 \sin(x_4 - x_3 - x_2) + x_1$
  - $y_2 = l_1 \cos(x_2) + l_2 \cos(x_3 - x_2) + l_3 \cos(x_4 - x_3 - x_2)$

  with $l_{1,2} = 0.5, l_3 = 1$, $\boldsymbol{x} \sim \mathcal{N}(\iota, \sigma^{\in})$ and $\sigma^2 = \left[\frac{1}{16}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right]$. The dataset was generated by the python simulator provided by [48].

- **Sine Waves** $\mathcal{D}_7$ [48]: this benchmark problem consists of a simple 2-dimensional sinusoidal function, defined as:

  - $y = \sin(3\pi x_1) + \cos(3\pi x_2)$

  The dataset was generated by the python simulator provided by [48].

- **Mulrilayer Stacks** (graphene–Si3N4 2D multi-layer stack) $\mathcal{D}_8$ this inverse design problem was originally introduced by [51], which provided an analytical transfer matrix simulator. The geometry of a multi-layer stack of alternating graphene and Si3N4 dielectric layers is optimized to produce a target absorption spectra under an incident beam s-polarized light.

The geometry of each stack is parameterized by paired graphene–Si3N4 subunits of infinite width and adjustable Si3N4 thickness. As in [48], We consider the 5-parameter version of this problem, in which spectra are discretized by 256 uniformly spaced outputs between the wavelengths of 240–2000 nm. By design, it is possible that different permutations of the stack layers, and so different design specifications, leads to very similar, or rather identical scattering (i.e., non-uniqueness). The dataset was generated by the python simulator provided by [48].

Instantiations for $\mathcal{D}_1, ..., \mathcal{D}_4$

All architectures defined in Figure 1.2 as well as the architecture of GIDNET consist of a number of building blocks (namely E, D, $\texttt{N}_\texttt{F}$, and $\texttt{N}_\texttt{ff}$) that have to be pre-trained before they are assembled together. These blocks have been instantiated by means of a model-selection phase, based on a gridsearch approach over the hyperparameter space of feedforward networks, defined by the number of layers and neuron per layer, the activation functions of the layers, the learning rate in the stochastic gradient descent algorithm, the initialization strategies: we first select the configuration in the hyperparameter space with the best average performances (Euclidean distance between the predictions and the actual values) computed via cross validation; and, on that configuration, we then select the weights of the cross validation iteration with the best overall performances. For each cross validation iteration, the blocks were trained for 100 epochs with *Adam* as optimizer. The latent space dimension is $h = 2$ and is the output layer of the encoder (E).

		Layers [n. of neurons]
$\mathcal{D}_1$	E	$\{[16, 8, 4, 2], [24, 16, 8, 2]\}$
	D	$\{[4, 8, 16], [8, 16, 24]\}$
	$\texttt{N}_\texttt{F}$	$\{[4, 4], [8, 8]\}$
	$\texttt{N}_\texttt{ff}$	$\{[4, 4], [8, 8], [8, 8, 8], [16, 16, 16], [24, 24, 24]\}$
$\mathcal{D}_2$	E	$\{[16, 8, 4, 2], [24, 16, 8, 2], [30, 20, 10, 2]\}$
	D	$\{[4, 8, 16], [8, 16, 24], [10, 20, 30]\}$
	$\texttt{N}_\texttt{F}$	$\{[4, 4], [8, 8]\}$
	$\texttt{N}_\texttt{ff}$	$\{[4, 4], [8, 8], [8, 8, 8], [16, 16, 16], [24, 24, 24]\}$
$\mathcal{D}_3$	E	$\{[16, 8, 4, 2], [24, 16, 8, 2], [30, 20, 10, 2]\}$
	D	$\{[4, 8, 16], [8, 16, 24], [10, 20, 30]\}$
	$\texttt{N}_\texttt{F}$	$\{[4, 4], [8, 8]\}$
	$\texttt{N}_\texttt{ff}$	$\{[8, 8], [8, 8, 8], [16, 16, 16], [24, 24, 24]\}$
$\mathcal{D}_4$	E	$\{[24, 16, 8, 2], [30, 20, 10, 2]\}$
	D	$\{[8, 16, 24], [10, 20, 30]\}$
	$\texttt{N}_\texttt{F}$	$\{[4, 4], [8, 8]\}$
	$\texttt{N}_\texttt{ff}$	$\{[4, 4], [8, 8], [8, 8, 8], [16, 16, 16], [24, 24, 24]\}$

Table 1.7: Configurations for layers and neurons in the model-selection phase for the blocks E, D, $\texttt{N}_\texttt{F}$, and $\texttt{N}_\texttt{ff}$: each sequence of numbers in square brackets represents a different configuration.

		Layers [n. of neurons]	lr	activation	init
$\mathcal{D}_1$	E	$[24, 16, 8, 2]$	0.0005	*tanh*	*Glorot Normal*
	D	$[8, 16, 24]$	0.0005	*tanh*	*Glorot Normal*
	$\mathtt{N_F}$	$[4, 4]$	0.0005	*relu*	*Glorot Normal*
	$\mathtt{N_{ff}}$	$[16, 16, 16]$	0.0005	*tanh*	*Glorot Normal*
$\mathcal{D}_2$	E	$[24, 16, 8, 2]$	0.0005	*tanh*	*Glorot Normal*
	D	$[8, 16, 24]$	0.0005	*tanh*	*Glorot Normal*
	$\mathtt{N_F}$	$[8, 8]$	0.0005	*relu*	*Random Normal*
	$\mathtt{N_{ff}}$	$[24, 24, 24]$	0.001	*tanh*	*Glorot Normal*
$\mathcal{D}_3$	E	$[30, 20, 10, 2]$	0.0005	*tanh*	*Glorot Normal*
	D	$[10, 20, 30]$	0.0005	*tanh*	*Glorot Normal*
	$\mathtt{N_F}$	$[8, 8]$	0.0005	*relu*	*Random Normal*
	$\mathtt{N_{ff}}$	$[24, 24, 24]$	0.0005	*tanh*	*Glorot Normal*
$\mathcal{D}_4$	E	$[30, 20, 10, 2]$	0.001	*relu*	*Glorot Normal*
	D	$[10, 20, 30]$	0.001	*relu*	*Glorot Normal*
	$\mathtt{N_F}$	$[8, 8]$	0.0005	*relu*	*Random Normal*
	$\mathtt{N_{ff}}$	$[24, 24, 24]$	0.001	*relu*	*Glorot Normal*

Table 1.8: Best configurations for the blocks E, D, $\mathtt{N_F}$, and $\mathtt{N_{ff}}$, over $\mathcal{D}_1$,...,$\mathcal{D}_4$.

The hyperparameter space for training the blocks E, D, $\mathtt{N_F}$, and $\mathtt{N_{ff}}$ was defined as follows[3]:

- **layers and neurons**: defined on the basis of the different datasets, as reported in Table 1.7;
- **learning rate**: $\{0.01, 0.005, 0.001, 0.0005, 0.0001\}$;
- **activation function**: $\{$*relu*, *tanh*$\}$;
- **weights initialization**: $\{$*Glorot Normal*, *Random Normal*$\}$.

The best configurations for that blocks are reported in Table 1.8. For the other blocks (that are in fact not pre-trained), configurations were naturally derived as follows: $\mathtt{N'_{ff}}$ and $\mathtt{N}_*$ have the same architecture as $\mathtt{N_{ff}}$, $\mathtt{N_c}$ use the same architecture of the autoencoder obtained by combining E and D, while $\mathtt{N_{ls}}$ has a fixed configuration (as $\mathtt{N_{N_{ls}}}$ occurs in GIDNET only, its results are amenable to improvements by performing model selection over that block too). A summary of these configurations is depicted in Table 1.9.

After that all basic building blocks have been defined, we assemble them in order to build the architectures in Figure 1.2 as well as the architecture of GIDNET. In particular, architecture (d) needs to undergo some further training to properly set the weights of $\mathtt{N'_{ff}}$ and $\mathtt{N_c}$. To this end, we performed model selection by looking for different learning rates in $\{0.01, 0.005, 0.001, 0.0005, 0.0001\}$ over 100 epochs, with *Adam* as optimizer. At this point, for (d) and for the other output-independent method, namely (a), we just picked the best architectures resulting from model selection and evaluate them over the test set.

For the output-dependent methods, instead, we fine-tuned the best architectures (resulting from the model selection discussed above) on each given value $\bar{\boldsymbol{y}}$ for which we have to compute $\boldsymbol{x}$ such that $\boldsymbol{y} = \mathtt{F}(\boldsymbol{x})$; in particular, fine-tuning has been conducted by considering the different learning rates depicted in Table 1.10 and by taking the best result over them. In the particular case of method (e*)—which requires re-initializations—we use $T = 50$.

[3]Keras default parameters can be taken into account when the values are not mentioned.

		Layers [n. of neurons]	activation	init
$\mathcal{D}_1$	$\mathtt{N}'_{\mathtt{ff}}$	$[4, 4]$	*relu*	*Glorot Normal*
	$\mathtt{N}_{\mathtt{c}}$	$[24, 16, 8, 2, 8, 16, 24]$	*tanh*	*Glorot Normal*
	$\mathtt{N}_{*}$	$[4, 4]$	*relu*	*Glorot Normal*
	$\mathtt{N}_{\mathtt{ls}}$	$[16, 8]$	*relu*	*Glorot Uniform*
$\mathcal{D}_2$	$\mathtt{N}'_{\mathtt{ff}}$	$[8, 8]$	*relu*	*Random Normal*
	$\mathtt{N}_{\mathtt{c}}$	$[24, 16, 8, 2, 8, 16, 24]$	*tanh*	*Glorot Normal*
	$\mathtt{N}_{*}$	$[8, 8]$	*relu*	*Random Normal*
	$\mathtt{N}_{\mathtt{ls}}$	$[16, 8]$	*relu*	*Glorot Uniform*
$\mathcal{D}_3$	$\mathtt{N}'_{\mathtt{ff}}$	$[8, 8]$	*relu*	*Random Normal*
	$\mathtt{N}_{\mathtt{c}}$	$[30, 20, 10, 2, 10, 20, 30]$	*tanh*	*Glorot Normal*
	$\mathtt{N}_{*}$	$[8, 8]$	*relu*	*Random Normal*
	$\mathtt{N}_{\mathtt{ls}}$	$[16, 8]$	*relu*	*Glorot Uniform*
$\mathcal{D}_4$	$\mathtt{N}'_{\mathtt{ff}}$	$[8, 8]$	*relu*	*Random Normal*
	$\mathtt{N}_{\mathtt{c}}$	$[30, 20, 10, 2, 10, 20, 30]$	*relu*	*Glorot Normal*
	$\mathtt{N}_{*}$	$[8, 8]$	*relu*	*Random Normal*
	$\mathtt{N}_{\mathtt{ls}}$	$[16, 8]$	*relu*	*Glorot Uniform*

Table 1.9: Configurations for blocks $\mathtt{N}'_{\mathtt{ff}}$, $\mathtt{N}_{\mathtt{c}}$, $\mathtt{N}_{*}$, and $\mathtt{N}_{\mathtt{ls}}$, over $\mathcal{D}_1$,...,$\mathcal{D}_4$.

method	Optimizer	lr	epochs
(c)	Adam	$\{0.1, 0.01, 0.001, 0.0001\}$	1000
(e*)	Adam	$\{0.001, 0.0001, 0.00001\}$	1000
(f)	Adam	$\{0.1, 0.01, 0.001, 0.0001\}$	1000
GIDnet	Adam	$\{0.1, 0.01, 0.001, 0.0001\}$	1000

Table 1.10: Configurations for inverse computation in the output-dependent methods, over $\mathcal{D}_1$,...,$\mathcal{D}_4$.

Finally, note that the constants used in the loss functions of GIDNET were defined as follows: $\lambda_0 = 0$, $\lambda_1 = 1$, and $\lambda_2 = 1$.

Instantiations for $\mathcal{D}_5, ..., \mathcal{D}_8$

A wide set of methods have been previously benchmarked on these dataset. We reproduced the identical experimental settings in recent literature to compare GIDNET performances against the two state of the art best performing methods, such as Neural Adjoint by [48] and conditional Variational Autoencoder (cVAE) [37]. In particular:

- For $\mathcal{D}_5, ..., \mathcal{D}_7$ we use the same experimental setting and the results reported by [48]
- For $\mathcal{D}_8$ we use the same experimental setting and the results reported by [52]

In both cases, the forward simulators $\mathtt{N}_{\mathtt{F}}$ for GIDNET were instantiated with the same hyper-parameters found in the repositories linked by the corresponding authors. The inverse computation of GIDNET is performed for 500 epochs with learning rates in $\{0.1, 0.01, 0.001, 0.0001\}$ and Adam as optimizer.

Metrics for $\mathcal{D}_1, ..., \mathcal{D}_4$

For each dataset, we used 7.000 samples for training and 2.900 for validation during training; seed computation for GIDNET was carried out over these 9.900 samples. The remaining 100 samples were, instead, used for assessing the value of the metrics after the training phase has been completed. In particular, by denoting as $\mathcal{T} = \{(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{y}}_i)\}_{i \in \{1,...,t\}}$ the test set and by $\mathtt{M}$ the generic method being evaluated (and with $\mathtt{F}$ being the given forward function), we consider:

$$\begin{aligned} \textbf{L2-norm}: &\quad \tfrac{1}{|\mathcal{T}|} \textstyle\sum_{(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{y}}_i) \in \mathcal{T}} ||\mathtt{F}(\mathtt{M}(\bar{\boldsymbol{y}}_i)) - \bar{\boldsymbol{y}}_i||_2 \\ \textbf{mae}: &\quad \tfrac{1}{|\mathcal{T}|} \textstyle\sum_{(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{y}}_i) \in \mathcal{T}} |\mathtt{F}(\mathtt{M}(\bar{\boldsymbol{y}}_i)) - \boldsymbol{y}_i| \\ \textbf{rmse}: &\quad \sqrt{\tfrac{1}{|\mathcal{T}|} \textstyle\sum_{(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{y}}_i) \in \mathcal{T}} (||\mathtt{F}(\mathtt{M}(\bar{\boldsymbol{y}}_i)) - \bar{\boldsymbol{y}}_i||_2)^2} \end{aligned}$$

Note that all the above metrics do not depend on $\bar{\boldsymbol{x}}_i$, as our goal is not to reconstruct $\bar{\boldsymbol{x}}_i$, but actually *any* possible input whose associated output coincides with $\bar{\boldsymbol{y}}_i$.

Metrics for $\mathcal{D}_5, ..., \mathcal{D}_8$

As previously pointed out, for datasets $\mathcal{D}_5, ..., \mathcal{D}_8$, we used the same experimental settings introduced by [48, 52]. In these works, the performances of Neural Adjoint and cVAE are benchmarked by using a variable number T of random initialization, with $T \in \{1, 10, 20, ..., 50\}$. They hence define the following metric:

$$\hat{r}_T = \frac{1}{|\mathcal{T}|} \sum_{(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{y}}_i) \in \mathcal{T}} [\min_{z \in Z_T} \mathcal{L}(\mathtt{F}(\mathtt{M}(z)), \bar{\boldsymbol{y}}_i)]$$

where Z_T is a randomly drawn sequence of z initializations of length T, and $\mathcal{L}$ is the mean square error function.
We recall that GIDNET choose the starting point for the exploration by selecting k nearest neighbor from the training set, hence relying on a deterministic initialization. For a fair comparison with the previous mentioned methods, we tested the performance of GIDNET for $k \in \{1, ..., 50\}$ and we define an analogous metric considering the number of different initialization:

$$\hat{r}'_T = \frac{1}{|\mathcal{T}|} \sum_{(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{y}}_i) \in \mathcal{T}} [\min_{k \in K_T} \mathcal{L}(\mathtt{F}(\mathtt{GIDnet}(\bar{\boldsymbol{y}}_i, k)), \bar{\boldsymbol{y}}_i)]$$

where K_T is a randomly drawn sequence of k values of length T, such that $K_T \subset K'_T$ whenever $T < T'$.

1.6.2 GIDNET on the Photonic Application Scenario

In order to assess the behaviour of GIDNET on a real application domain, we considered the photonic scenario and, in particular, the experimental setting discussed in the recent work by Lininger et al. [3], where the goal is to build a thin-film structure that gives rise to some desired reflectance/transmittance spectra. In this setting, the input space is associated with the thickness of the various layers as well as with the one-hot encodings of the corresponding materials being used for that layer, so that we can evaluate all ingredients of the GIDNET architecture. Moreover, the high-dimensionality of the input and output spaces required the use convolutional approaches and posed specific challenges in the design of the various blocks.

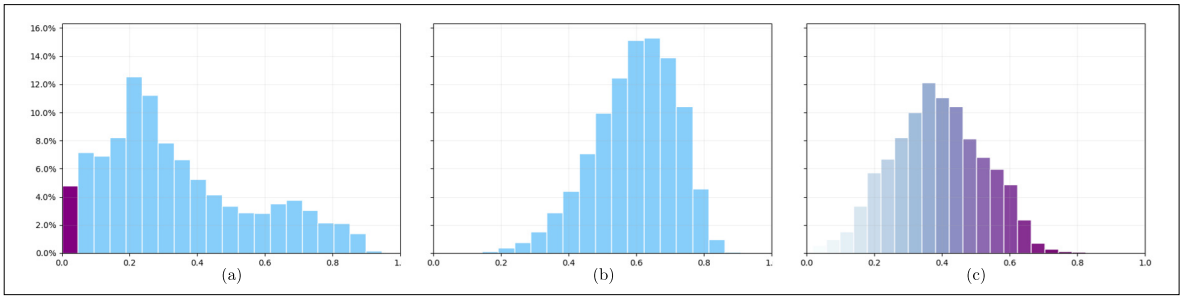


Figure 1.11: Metamaterials Dataset Analysis: (a) distribution of the pairwise distance (normalized in [0,1]) between 10000 randomly sampled spectra; (b) distribution of the pairwise distance between the corresponding structures; (c) distribution of the pairwise distance (normalized in [0,1]) between structures such that the distance of their spectra belongs to the left-most bar of (a).

Dataset Description

In the dataset considered by Lininger et al. [3], each structure is made of some layers (up to 5 distinct layers), each with thickness within the range $[1, 60]$nm and whose material can be Ag, Al2O3, ITO, Ni, or TIO2. Concerning the representation, it is natural to use a one-hot encoding for the material, so that the input space is $\mathbb{R}^{\ell\times(1+5)}$, with ℓ being the number of layers—indeed, for each layer, we have to represent its thickness and the material as a one-hot encoded vector over 5 alternatives. Each device structure is eventually associated with a reflectance spectrum and a transmittance spectrum, obtained via the transfer matrix method [53], for two polarizations, at the incident angles of 25, 45, and 65 degrees for 200 equally spaced points over the range $[450, 950]$nm and with values in $[0, 1]$. Thus, the output space is $\mathbb{R}^{2\times2\times3\times200}$.

In order to have a closer look at the dataset, Figure 1.11 reports some useful statistics on the distributions of the waveforms. In particular,

- Figure 1.11.(a) reports the distribution of the pairwise distance (normalized in [0,1]) between the spectra, by evidencing the significant percentage (the left-most bar) of the spectra that are almost similar, while being associated to different device structures; note that to compare spectra we use the standard Euclidean distance.

- Figure 1.11.(b) reports the distribution of the pairwise distance (again normalized in [0,1]) between the structures associated to the spectra; note here that to compare the structures, we sum two terms: the Euclidean distance (normalized in [0, 0.5]) between the thickness values and the Euclidean distance (normalized in [0, 0.5]) between the one-hot-encodings of the materials.

- Figure 1.11.(c) reports the distribution of the pairwise distance between the structures associated to the spectra that fall into the left-most bar of Figure 1.11.(a); note that distances are normalized with the same scale used for Figure 1.11.(b), so that the values are comparable and it emerges that a significant number of rather different structures are associated with very similar spectra.[4]

[4]Hence, the setting is very congenial to assess the efficacy of methods for inverse design.

ℓ	n	h
1	6	3
2	12	6
3	18	9
4	24	12
5	30	15

Table 1.11: Dimensions of the spaces in the photonic scenario.

	Layers [n. of neurons]	**lr**
E	$\{$**[72, 60, 36, 12]**, $[60, 48, 24, 12]\}$	$\{0.01, 0.005,$ **0.001**$\}$
D	$\{$**[36, 60, 72]**, $[24, 48, 60]\}$	$\{0.01, 0.005,$ **0.001**$\}$
$\mathtt{N_F}$	$\{[24, 48, 60],$ **[33,50,60]**$\}$	$\{$**0.01**$, 0.001\}$

Table 1.12: Configurations for layers and neurons in the model-selection phase for the blocks E, D, $\mathtt{N_F}$, and $\mathtt{N_{ff}}$, by fixing $\ell = 4$: each sequence of numbers in square brackets represents a different configuration. We report in bold the configurations associated with the best performances.

	Layers [n. of neurons]	**lr**	**activation**	**init**	**epochs**	**batch**
E	$[18 \cdot \ell, 15 \cdot \ell, 9 \cdot \ell, 3 \cdot \ell]$	0.001	*tanh*	*Glorot Normal*	150	256
D	$[9 \cdot \ell \cdot, 15 \cdot \ell, 18 \cdot \ell]$	0.001	*tanh*	*Glorot Normal*	150	256
$\mathtt{N_F}$	$[\lfloor n \cdot 1.4 \rfloor, 50, 60], 4 \times conv1d[64, 128]]$	0.01	*relu*	*Glorot Normal*	100	256

Table 1.13: Configurations for the blocks E, D, and $\mathtt{N_F}$, over the photonic application scenario with $\ell \in \{1, 2, 3, 4, 5\}$ ("4×" stands for 4 parallel layers). Each *conv1d* layer is followed by a max pooling 1d layer with pool size 2.

Instantiation of the Methods

On the photonic scenario, we instantiated GIDNET by considering fully connected blocks as well as convolutional blocks. For each value of ℓ, Table 1.11 summarizes the input dimension $n = \ell + (\ell \times 5)$ and the correspondent latent space dimension, which we defined as $h = n/2$.

Recall that, in order to build the architecture in Figure 1.2 as well as the architecture of GIDNET, we need to pre-train the blocks E, D, $\mathtt{N_F}$, and $\mathtt{N_{ff}}$. For the blocks E, D, and $\mathtt{N_F}$, we proceeded with a model-selection phase focused on the case of 4 layers, by using *Adam* as optimizer (with early stopping with patience of 50 epochs for E and D, and 20 for $\mathtt{N_F}$). The hyperparameter space explored to optimize the blocks is as follows:

- **layers and neurons**: as reported in Table 1.12;
- **learning rates**: as reported in Table 1.12;
- **activation function**: $\{$*relu*, *tanh*$\}$;
- **weights initialization**: $\{$*Glorot Normal*, *Random Normal*$\}$.

The best configurations of E, D, and $\mathtt{N_F}$ which emerged for $\ell = 4$ have been then generalized for each $\ell \in \{1, ..., 5\}$ according to the rules shown in Table 1.13.

Concerning $\mathtt{N_{ff}}$, instead, we used the configurations suggested by [3], which are indeed specific and already optimized for the photonic application scenario; actually, we modified the learning rate scheduler by starting the decaying approach proposed by the authors after 10 epochs rather than starting immediately, we introduced early stopping with a patience

ℓ	Layers [n. of neurons]	lr	lr decay	activation	dropout	batch
1	$[4 \times conv1d[64, 64, 64], [379, 315, 315], 2 \times [601, 200]]$	0.001	0.005	*relu*	0.039	512
2	$[4 \times conv1d[64, 64, 64], [379, 315, 315], 2 \times [601, 200]]$	0.001	0.005	*relu*	0.039	512
3	$[4 \times conv1d[64, 64], [1238, 397], 2 \times [286, 200]]$	0.001	0.009	*relu*	0.029	512
4	$[4 \times conv1d[64, 64, 64, 64, 64, 64], [421, 865, 865, 865], 2 \times [193, 200]]$	0.001	0.005	*relu*	0.013	128
5	$[4 \times conv1d[64, 64, 64, 64, 64, 64, 64, 64, 64], [276, 838, 838], 2 \times [837, 200]]$	0.001	0.009	*relu*	0.004	512

Table 1.14: Configurations for the block $\mathtt{N_{ff}}$ over the photonic scenario ("4×" and "2×" stand for 4 and 2 parallel layers, respectively). Blocks have been trained for 300 epochs, with Adam as optimizer, a learning rate of 0.001 and *relu* as activation function (as in [3]). Each network has two parallel output layers of 100 and 50 neurons, fully connected to the last reported layer. Each *conv1d* layer is followed by a max pooling 1d layer with pool size 2.

of 30 epochs, and we use an increased batch size with respect to the proposed one of 128 (experiments will evidence a great benefit of these minor improvements). In the particular case of $\mathtt{N_{ff}}$, for consistency with [3], we normalized the real valued part of $\bar{\boldsymbol{x}}_i$—corresponding to the thickness of the metamaterials layers—in the range $[0, 0.6]$.[5] Configurations for $\mathtt{N_{ff}}$ are reported in Table 1.14, at the varying of ℓ.

	Layers [n. of neurons]	activation	init
$\mathtt{N'_{ff}}$	see $\mathtt{N_{ff}}$ in Table 1.14		
$\mathtt{N_c}$	$[18 \cdot \ell, 15 \cdot \ell, 9 \cdot \ell, 3 \cdot \ell]$	*tanh*	*Glorot Normal*
$\mathtt{N_*}$	$\{[[256, 512], conv1dT[16, 16, 8, 4]], [[64, 128], conv1dT[8, 8, 4, 4]]\}$	*leaky relu (0.2)*	*Random Uniform*
$\mathtt{N_{ls}}$	$[3 \cdot \ell, 3 \cdot \ell]$	*relu*	*Random Normal*

Table 1.15: Configurations for blocks $\mathtt{N'_{ff}}$, $\mathtt{N_c}$, $\mathtt{N_*}$, and $\mathtt{N_{ls}}$, over the photonic scenario with $\ell \in \{1, 2, 3, 4, 5\}$ (for $\mathtt{N_*}$, two variants are reported—and we take the best results over them). Each *conv1dT* layer (convolutional 1d trasposed) is followed by a batch normalization layer with 0.1 as momentum.

As with the former application scenario, the remaining blocks of the architectures do not need to be pre-trained, and we proceed as follows: $\mathtt{N'_{ff}}$ has the same architecture as $\mathtt{N_{ff}}$, $\mathtt{N_c}$ uses the same architecture of the encoder $\mathtt{E}$, while $\mathtt{N_{ls}}$ has a fixed configuration. Concerning $\mathtt{N_*}$, we considered two different block configurations, the one proposed by the authors of the corresponding methods [41, 49] and a variant (and we shall take the best results over them, when evaluating the performances of the method (f)). The resulting configurations are shown in Table 1.15, so that—in particular—the final configuration for GIDNET is the one in Figure 1.5.

As usual, after that all basic building blocks have been defined, we assemble them in order to build the architectures in Figure 1.2 as well as the architecture of GIDNET. Moreover, we have to train the architecture (d) in order to properly set the weights of $\mathtt{N'_{ff}}$ and $\mathtt{N_c}$. In our experiments, performances of (d) were not satisfying for any fixed learning rate value; therefore, we decided to train 6 different models, one for each possible learning rate in $\{0.01, 0.001, 0.0005, 0.0001, 0.00005, 0.00001\}$ (with 150 epochs and *Adam* as optimizer), then picking (for each given value $\bar{\boldsymbol{y}}$ for which we have to compute the inverse) the best result that is achieved over them. That is, we essentially implemented a kind of output-dependent method, though just confined to the dynamic selection of the most appropriate pre-computed model.

[5] For all the other building blocks and for the inverse computation phase, thicknesses are normalized in $[-1, 1]$.

method	Optimizer	lr	epochs	batch
(c)	Adam	$\{0.01, 0.001, 0.0005, 0.0001, 0.00005, 0.00001\}$	100	-
(e*)	Adam	$\{0.01, 0.001, 0.0005, 0.0001, 0.00005, 0.00001\}$	100	-
(f)	Adam	$\{0.01, 0.001, 0.0005\}$	100	32
★	Adam	$\{0.1, 0.5, 0.05, 0.01\}$	100	1
GIDnet	Adam	$\{0.1, 0.5, 0.05, 0.01\}$	100	32

Table 1.16: Configurations for inverse computation in the output-dependent methods, over the photonic application.

For the output-dependent methods, instead, we fine-tuned the weights of best architectures (resulting from model selection) on each given value $\bar{\boldsymbol{y}}$ for which we have to compute $\boldsymbol{x}$ such that $\boldsymbol{y} = \mathtt{F}(\boldsymbol{x})$; in particular, fine-tuning has been conducted by considering the learning rates depicted in Table 1.16 to properly set the weights, and by taking the best result over them.

In particular, in the fine-tuning phase for all the output-dependent methods—and to train method (d)—, we used a learning rate scheduler, which reduces the value of **lr** by a factor of 0.6 when the loss function reaches a plateau. In the particular case of method (e*)—which requires re-initializations—we use $T = 7$ (which is the same number of different seeds initializations used for GIDNET) The constants used in the loss functions of GIDNET were defined as follows: $\lambda_0 = 1$, $\lambda_1 = 1$, and $\lambda_2 = 0$.

Finally, we compare GIDNET with the VAE-based method provided by [36], referred as ★. Such a method relies on two main building blocks which are the VAE and the property predictor used for conditioning the the latent space. We build the VAE by assembling the pre-trained $\mathtt{E}$, a sampling layer and the pretrained $\mathtt{D}$. For the property predictor we use instead the same achitecture of $\mathtt{N_F}$, changing the input layer to work with the latent space. We fine-tune the VAE while jointly training the property predictor, and by introducing the KL-loss [5]. Such a training phase is repeated for three different values of the learning rate $\{0.01, 0.001, 0.0001\}$ and with the same hyperparameters used for the training of $\mathtt{E}$ and $\mathtt{D}$. The final optimization is done with a variant of method in Figure 1.2(f). Here we use the original pretrained $\mathtt{N_F}$ and the decoder of the VAE as $\mathtt{N}_*$. While the latter two blocks are freeze, we optimize directly into the latent space by starting from a randomly sampled point. Hyperparameter for the optimization phase are reported in Table 1.16.

Evaluation Metric

For each $\ell \in \{1, .., 5\}$, we use 106.820 samples for training, 45.780 samples for validation, and 2.000 samples for computing the metric. The seeds needed by GIDNET are, as usual, computed over the traning and validation samples. In particular, experiments have been focused on measuring the **srmse** (spectral root mean squared error), as defined in [3], between the spectra associated to the metamaterial designed by GIDNET and the spectra provided as input.

More formally, by denoting as $\mathcal{T} = \{(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{y}}_i)\}_{i \in \{1,...,t\}}$ the test set and by $\mathtt{M}$ the generic method being evaluated (and with $\mathtt{F}$ being the given forward function), and by recalling that $\boldsymbol{y}_i \in \mathbb{R}^{2\times2\times3\times200}$, the **srmse** is defined as:

$$\textbf{srmse}: \frac{1}{|\mathcal{T}|}\sum_{(\bar{\boldsymbol{x}}_i,\bar{\boldsymbol{y}}_i)\in\mathcal{T}}\frac{1}{4}\sum_{\alpha=1}^{2}\sum_{\beta=1}^{2}\sqrt{\frac{\sum_{k=1}^{3}\sum_{h=1}^{200}\left(\mathtt{F}(\mathtt{M}(\bar{\boldsymbol{y}}))_{i,\alpha,\beta,,k,h}-\bar{\boldsymbol{y}}_{i,\alpha,\beta,k,h}\right)^2}{3\times 200}}$$

Actually, a subtle issue here comes into play. Given that we have a physical simulator [53], we can do much better than just using the spectra $\bar{\boldsymbol{y}}$ computed by the methods for measuring the error. Indeed, there is no a-priori guarantee that the corresponding values $\bar{\boldsymbol{x}}$ are associated with some correct one-hot encodings in $\mathbb{R}^{\ell\times(1+5)}$ (i.e., with some feasible device structures). Accordingly, for all tested methods, such values $\bar{\boldsymbol{x}}$ are pre-processed by suitably rounding the components associated to the one-hot encodings of the materials over the various layers. For the resulting values (correctly encoding some device structures), we then use the transfer matrix method to compute their associated spectra, and the quality of the results is eventually assessed over them.
In Figure 1.12 we report a real example of generated versus actual material.

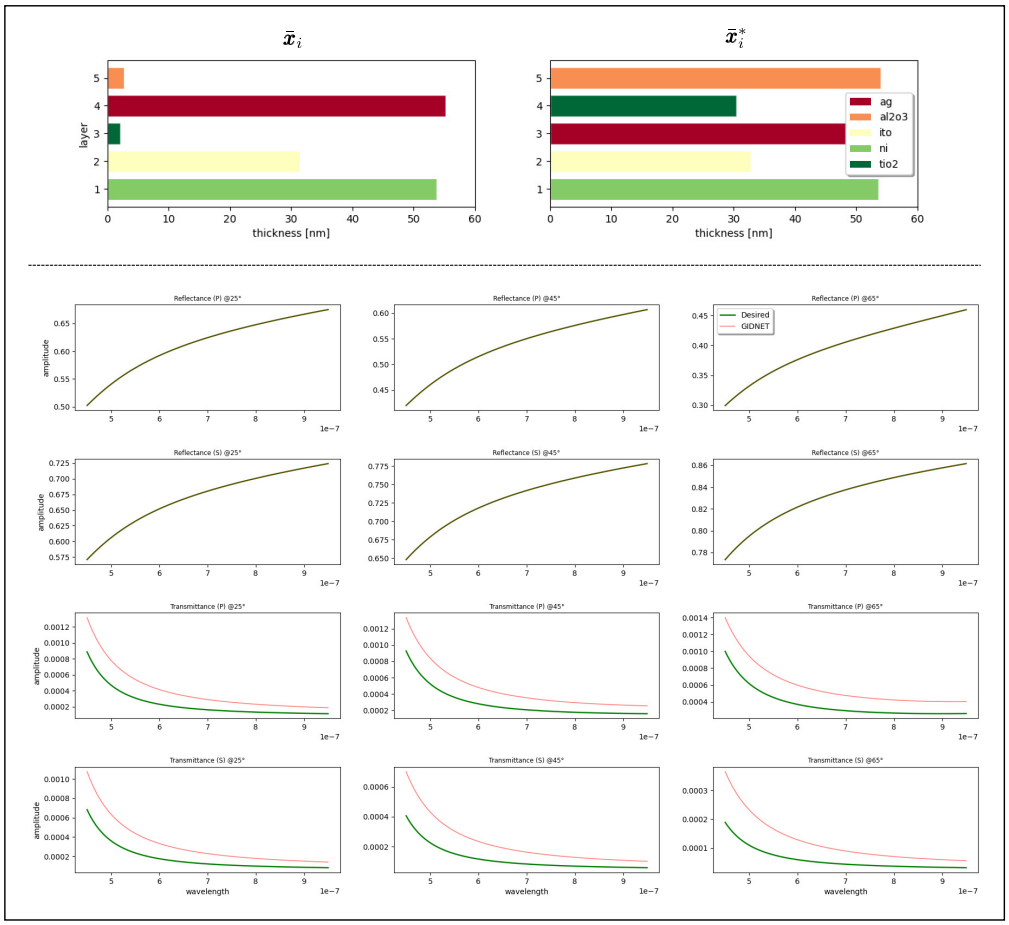


Figure 1.12: Exemplification of GIDNET behaviour on a pair $(\bar{x}_i, \bar{y}_i) \in \mathcal{T}$: The device structure x_i^* being computed by GIDNET is different from $\bar{x}_i$, but still enjoys the desired spectral responses $\bar{y}_i$—six spectra are identical, while the others are quite similar (differences are amplified by the scale).

Chapter 2

Automatic Medical Report Generation via Latent Space Conditioning and Transformers

This paper presents a comprehensive exploration of integrating artificial intelligence (AI) in the healthcare sector, focusing on the development and implementation of a novel framework called VAE-GPT. Our architecture combines Variational Autoencoder (VAE) and Generative Pre-trained Transformer (GPT), to generate high-quality medical reports. The VAE component enables the model to learn a latent space representation of the images, capturing the underlying patterns and structures. The GPT component leverages the power of transformer-based language models to generate coherent and contextually relevant text. Additionally, a novel metric, Medical Embeddings Attention Distance (MEAD), is proposed in order to capture the semantic similarity between the generated and training medical reports, taking into account the importance of specific words determined by the attention module. Experiments on real dataset demonstrate that our framework achieves state-of-the-art comparable performances in generating accurate and informative medical reports.

2.1 Introduction

Artificial intelligence (AI) profoundly impacts us by revolutionizing how we process information, make decisions, and interact with technology. This leads to transformative advancements across various industries and shapes the future of human-machine interactions.

The applications of AI, particularly those involving deep learning, are extensive and offer significant benefits to several sectors such as healthcare, transportation, finance, and more. In the healthcare domain, deep learning and AI have been successfully employed, providing numerous advantages such as improved diagnostics, personalized treatment plans, and the prevention of adverse events. A recent study[60] assesses that the market is projected to reach $194.4 billion by 2030.

The main drivers contributing to the growth of the AI healthcare market include, but not is limited to, the rising requirement for remote patient monitoring systems, the expanding set of patient health-related digital information, increasing demand for customized medicine, and the rising demand for reduced care costs.

In particular, AI has demonstrated its effectiveness in medical imaging tasks by enhanc-

ing precision, faster diagnosis, improving image interpretation, and the potential to identify subtle patterns and anomalies that may aid in early disease detection and personalized treatment planning[61, 62]. Recently, the combination of imaging analysis and NLP techniques has led to the exploration of a new research field namely "Automatic Medical Report Generation". Automatic Medical Report Generation is a complex task that involves the use of natural language processing (NLP) and machine learning techniques to create detailed and accurate medical reports without human intervention. Automated report generation can benefit both inexperienced and experienced radiologists. Inexperienced radiologists can benefit from automated assistance in acquiring the knowledge of normal anatomy, pattern discovery, and evaluation of temporal evolution. Experienced radiologists can save time and effort by automating the process of report analysis and generation. As show in figure2.1, the medical reports involves various types of information, including impressions, findings, and tags for grouping reports with common characteristics. The format and content of medical reports can vary widely between different healthcare institutions and specialties and standardization remains a challenge.

Generating accurate and meaninful medical reports is a not trivial task, because the difficulty of finding a "general" model that is accurate and robust w.r.t. an input (images, text etc.) is joined with a more ambitious goal, that is the automated systems should be able to understand the context, relationships between medical concepts, and the implications of certain findings. We addressed this challenging problem by proposing an approach that is able to properly navigate the latent space of the images and guides the building of a solution from the knowledge extracted from the medical reports associated to the such latent space. In other words, in addition to managing the multimodality of the input, it is necessary to guide the navigation of the space of solutions by exploiting knowledge in the text. It should be noted that having an approach that achieves the aforementioned goal assumes great importance also from the point of view of building interpretable models.

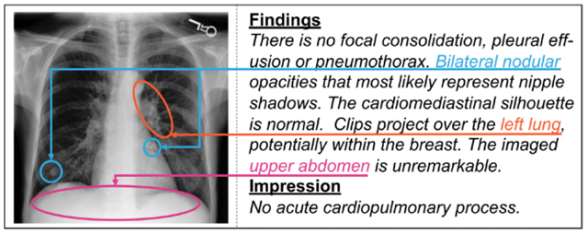


Figure 2.1: Medical report generation example

Interpretable models is another challenge to be addressed in healthcare, as transparency is essential to comprehend the underlying reasoning behind predictions, ensuring that decisions are based on reliable and understandable information. In detail, our proposal apply Transformer model, a well known deep learning model recently used in natural language processing and explored it in medical imaging as Visual Transformers (ViTs)[63] in order to achieve interpretability.

2.1.1 Contributions

Specifically, the paper introduces several contributions in the field of medical report generation. These contributions can be summarized as follows:

- **VAE-GPT:** The paper proposes a novel architecture called VAE-GPT for medical report generation. Our architecture combines Variational Autoencoder (VAE) and Generative Pre-trained Transformer (GPT), to generate high-quality medical reports. The VAE component enables the model to learn a latent space representation of the images, capturing the underlying patterns and structures. The GPT component leverages the power of transformer-based language models to generate coherent and contextually relevant text. By combining these two components, VAE-GPT is designed to distribute images into a conditioned latent space. Furthermore, by means of a tags predictor, it enforces the continuity into such a space with respect to the context associated with each image. Our framework achieves state-of-the-art comparable performances in generating accurate and informative medical reports.

- **MEAD Metric:** The paper introduces a new metric called MEAD (Medical Embeddings Attention Distance). MEAD leverages a weighted combination of BLEU scores [64], focusing on words that receive high attention scores from an attention module, build from the tags in the generated and training texts. The metric measures the semantic similarity between the generated and training medical reports, taking into account the importance of specific words determined by the attention module. By using this weighted combination approach, MEAD provides a comprehensive evaluation of the generated reports, considering both linguistic fluency and semantic relevance to the medical context. The utilization of attention scores enhances the metric's ability to capture important clinical information in the generated reports, facilitating improvements in the generation process and promoting the production of clinically meaningful reports.

- **Proposed Attention Mechanism:** To improve the generation process, the paper proposes a novel attention module, both for conditioning the latent space of the VAE with tags and for comparing texts. In fact, we designed a mechanism such that: The VAE encodes the images in a latent space which is conditioned on the tags and on the generated text. In such a way, "we enhance the continuity of the latent space w.r.t. the text."

- **Enhancing Model Interpretability:** The approach intrinsecally enhances model interpretability. It explores techniques that provide insights into the internal workings of the VAE-GPT model (e.g. attention mechanism), allowing researchers and medical professionals to understand how the model builds prediction and generates the reports. These interpretable models contribute to the trustworthiness and transparency of the generated reports, facilitating their adoption in real-world clinical settings.

These contributions collectively advance the field of medical report generation, offering a powerful framework, a comprehensive evaluation metric, an enhanced attention mechanism, and improved model interpretability. The proposed VAE-GPT framework, along with the MEAD metric and the combined attention module, provides a solid foundation for generating accurate, coherent, and clinically meaningful medical reports, ultimately benefiting healthcare professionals and improving patient care.

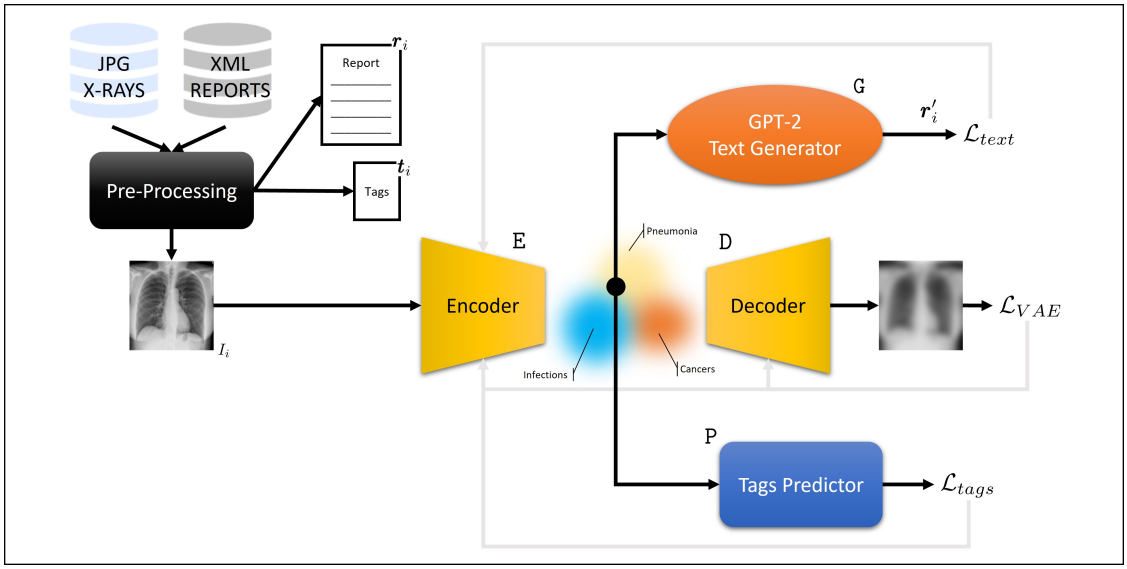


Figure 2.2: VAE-GPT architecture: the latent space, conditioned by the tags and the reports, identifies different regions for different contexts (e.g. diseases).

2.2 Related Works

To properly understand the problem addressed and the advance respect to the existent literature, this paper presents the state-of-the-art in medical report generation, considering related research works in the field.

Automatic generation of medical imaging reports has traditionally relied on computer-aided detection (CAD) systems and Electronic Health Records (EHR) to interpret medical images and provide diagnoses[65, 66] . Recently, exploiting deep learning techniques offers the ability to learn from features and generalize the context in a powerful way[67, 68]. In fact, deep learning models, such as multilayer perceptron (MLP), recurrent neural network (RNN), and convolutional neural network (CNN), have shown promising results in this domain[69, 70].

Different architectures have been proposed for medical report generation [65],[71]. A widely approach in the automatic report generation is the usage of a CNN for multi-label classification of tags, which are then mapped to embedded vectors via embedding matrix lookup. Thus, the report generation module has access to these tag vectors in order to build the whole text[72]. In most of case, these approaches have the tendency to describe more normal findings rather than abnormalities[73, 74]. To address this, researchers have proposed techniques such as the co-attention mechanism, which helps locate areas in images where abnormal elements are present. Thus, CNNs are used for learning visual features from medical images, while multi-label classification (MLC) techniques are employed for predicting tags. The combination of visual and semantic features is achieved using co-attention modules[75]. The architechure proposed by Li et al.[76] first successful exploits co-attention mechanism with LSTM to achieve better accuracy in the text generation. Specifically, [76] and [77] classify and look up tag embedding vectors, but unlike the previous works, the language component uses co-attention to access both tags vectors and visual features simultaneously. Their ablation analysis showed that the semantic information provided by these tags complements the visual information and improves the model's performance in report

generation. In [78, 79] an approach with Recurrent BiLSTM-attention-LSTM is used. The basic idea is to have a LSTM generate one sentence at a time, each time conditioned on a BiLSTM based encoding of the previous sentence and the output of an attention mechanism. The process is repeated recurrently sentence by sentence until the full report is generated. Other works [80, 81] used graph neural networks immediately after the CNN to encode the visual information in terms of medical concepts and their relations. Thus, the language component receives the intermediate graph representation instead of the raw visual features. In Vispi [82] a two-stage procedure with two distinct CNNs is used. In the first stage a DenseNet121 classifies abnormalities in the image, and then Grad-CAM is used to localize and crop a region of the image for each detected class. Then, in the second stage the multiple image crops are treated as independent images and processed by a typical CNN+LSTM architecture, with ResNet101. Moreover, NLP Researchers are shifting from recurrent deep learning model to transformer based one, with the attention concept[66]. This is very important in terms of calculation times, as we move from sequential models such as the classic MLPs or recurrent networks to more flexible architectures that allow GPU parallelization in a much more efficient way. Pre-trained models [83] are also being used effectively, eliminating the need for specific vocabularies and enabling faster learning from large-scale datasets.

In addition to the discussed approaches, pre-trained language models to image captioning have been also explored in medical report generation. Visual GPT architecture [84] could be used to segment and recognize region of interest in medical imaging, and thus enhancing models with explainability through the use of attention maps and heatmaps, and integrating medical knowledge graphs for improved accuracy and interpretability [85].

In our approach we propose an innovative approach that uses a novel attention mechanism that improve accuracy and enhance explainability. To the best of our knowledge this is the first time that the solution is find by navigate latent space conditioned by tags through a Variational Autoencoder, rather than using static feature extraction.

Overall, the research works presented in this paper demonstrate advancements in medical report generation and their potential to improve healthcare workflows, enhance diagnostic accuracy, and facilitate efficient reporting processes.

2.3 Proposed approach

The VAE-GPT model is an innovative text generation architecture designed for generating reports of X-ray scans. It combines the strengths of VAEs and pretrained GPT architectures. Figure 2.2 shows the proposed architecture, its components, and their relationship. The VAE component serves as an image encoder, extracting meaningful latent representations from X-ray scans. The VAE is jointly trained with a pretrained text generator (GPT) and a tags predictor such that images belonging to the same context (e.g. diseases) are placed in the same region of the latent space. Once VAE-GPT is trained, the encoder and GPT are used to generate coherent and contextually appropriate textual descriptions (reports).

2.3.1 Formal Framework

Let be $\mathcal{D}$ a dataset of triple $(I_i, \boldsymbol{t}_i, \boldsymbol{r}_i)$ with $i \in \{1, .., N\}$, where I_i is an x-ray image, $\boldsymbol{t}_i \in \mathbb{R}^m$ is its associated fixed-length vector of tags and $\boldsymbol{r}_i$ is the actual report. Our objective is to generate a report $\boldsymbol{r}'_i$ such that a text-related distance function $d(\boldsymbol{r}_i, \boldsymbol{r}'_i) \in \mathbb{R}$ is minimized.

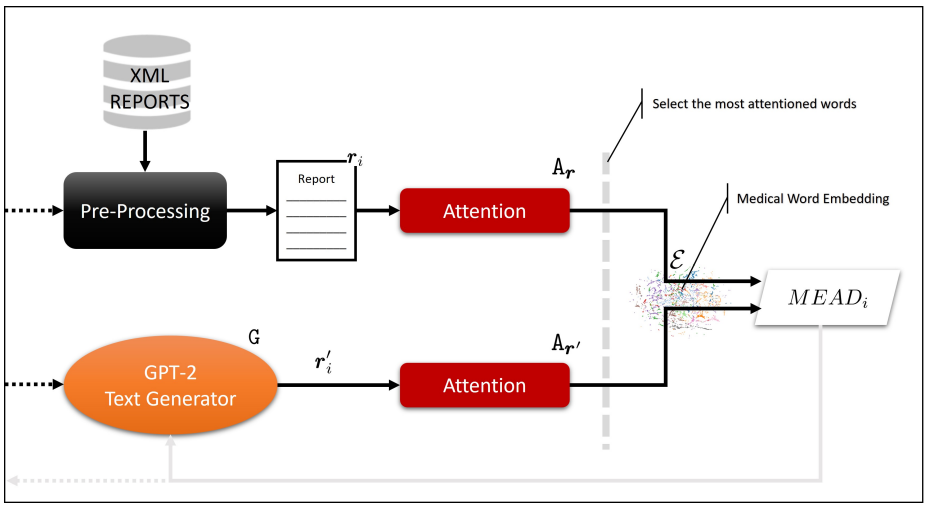


Figure 2.3: Architecture of the $MEAD$ block.

For this purpose, we define E and D as the encoder and the decoder of a Convolutional VAE to be trained on images; G as a text generator; P as a predictor of tags.
The generated report is obtained by

$$\boldsymbol{r}'_i = \mathtt{G}(\mathtt{E}(I_i))$$

after training the architecture with the following loss terms:

$$\mathcal{L}_{VAE} = \sum_{i=1}^{N} ||(\mathtt{D}(\mathtt{E}(I_i)) - I_i||_2 + \boldsymbol{KL}(q(\boldsymbol{z}|I_i)||p(\boldsymbol{z}))$$

$$\mathcal{L}_{tags} = \sum_{i=1}^{N} cce(\mathtt{P}(\mathtt{E}(I_i)) - \boldsymbol{t}_i)$$

$$\mathcal{L}_{text} = \sum_{i=1}^{N} d(\boldsymbol{r}'_i, \boldsymbol{r}_i)$$

where $\boldsymbol{KL}$ is the well-known Kullback-Leibler distance used for training VAE architectures [86], p and q are the respectively the prior and posterior distributions, $\boldsymbol{z}$ is a normally distributed random variable, and cce is the categorical cross-entropy function. Here the $\mathcal{L}_{VAE}$ is used to define an encoding of the x-ray images into a continuous latent space. The $\mathcal{L}_{tags}$ and $\mathcal{L}_{text}$ are used to reduce the distance between the latent image representations associated with similar tags and reports. Moreover the term $\mathcal{L}_{text}$ is introduced to optimize the text-related distance between the actual and the generated report. All the components of our architecture are jointly trained by using such loss terms.

2.3.2 Novel Proposed Metric

This study proposes a new metric called Medical Embeddings Attention Distance (MEAD) for comparing generated and actual reports. The MEAD exploits two attention blocks $\mathtt{A}_{\boldsymbol{r}}$ and $\mathtt{A}_{\boldsymbol{r}'}$ to select the most attention words into both the actual and the generated text. Such selected words are then projected into a Medical Embedding space by a function $\mathcal{E}$. Finally, a text-related distance d is computed between such medical embeddings of the actual and the generated reports. Our new metric is defined as:

$$MEAD_i = \sum_{n=1}^{N} d(\mathcal{E}(\mathtt{A}_{\boldsymbol{r}'}(\mathtt{r}'_\mathtt{i})), \mathcal{E}(\mathtt{A}_{\boldsymbol{r}}(\mathtt{r}_\mathtt{i}))) \tag{2.1}$$

The attention is used to select the most meaningful words in the two texts and the medical embedding is introduced to give more context in measuring the distance between words.
by adding the $MEAD$ block, reported in Figure 2.3, to the VAE-GPT framework, the following loss function can be considered for training the whole architecture:

$$\boldsymbol{\mathcal{L}} = \mathcal{L}_{VAE} + \mathcal{L}_{tags} + \mathcal{L}_{text} + \alpha * MEAD \tag{2.2}$$

where α is a real-valued factor in the range $[0, 1]$.

2.4 Experimental Evaluation

2.4.1 Dataset

Performances have been evaluated over a public dataset made by Indiana University[87]. The benchmark includes two main sources of data: jpg images (7470 elements) and associated diagnostic medical XML reports (3955 files). The XML reports contain structured sections such as Comparison, Indication, Findings, and Impression labels.

2.4.2 Pre-Processing

The pre-processing of the text data includes cleaning the dataset and performing the following actions:

- **Removing stopwords** using the "nltk" library. Stopwords are common words with little significance in conveying meaningful information and are often removed for better analysis.
- **Lemmatization** which transforms words into their base or dictionary form, reducing inflected or variant forms.
- **Removing punctuation** and special characters.
- Converting text to **lowercase**.
- **Tokenization** which involves splitting text into individual units called tokens.

Images undergo several functions to ensure consistency:

- Conversion to a **single channel** (black and white images).
- **Resizing** to a size of 512x512 pixels.
- **Normalization** of pixel values in the range $[0, 1]$.

For the tags, the initial 53 classes in the XML reports are reduced to 23 classes to address class imbalance issues.
These pre-processed datasets are then used for training the whole VAE-GPT architecture.

2.4.3 Metrics

The text loss is evaluated using the Rouge-L and BLEU metrics. These metrics are used to compare the model's performance with those already present in the literature. Briefly:

- *ROUGE-L* evaluates the similarity between predicted texts and true texts based on the length of the longest common subsequence between them. It considers word order in texts and assesses how well the predicted text captures the information present in the true text.
- *BLEU* evaluates the quality of predictions by comparing the predicted text with one or more true reference texts. It uses n-grams to calculate the accuracy of matches between the predicted text and reference texts, providing a score indicating the closeness of the generated text to the true reference texts.

2.4.4 Experiments

The experiments were performed by using all the VAE-GPT components with the following configurations:

- the VAE consists of 3 convolutional layers for the encoder `E` and 3 transposed convolutional layers for the decoder `D` with the following number of filters: 64,128 and 256 for the encoder and 256,128 and 64 for the decoder; each convolution operation is performed with *stride* of 1 pixel and *padding* 2 pixels; the *relu* activation function is used for each layer; the latent space dimension is set to 128.
- the Tags Predictor `P` is a three-layer convolutional network with a softmax output to provide multi-class classification over the tags. An attention mechanism is used to identify the most influential tokens in the classification process.
- the GPT-2 architecture (`G`) remains unchanged with respect to [88].

The whole VAE-GPT architecture was trained on the dataset introduced in Section 2.4.1 by using 80% of the data for training and validation while the remaining 20% was used for test. The *learning rate* was 0.001 and *reduce learning rate on plateau* and The architecture was trained over 10 epochs with a batch size of 64.
The loss function defined in equation 2.3.2, Section 2.3.2, was optimized by jointly training `E`, `D`, `P`, and by fine-tuning the pretrained `G`. The text-related distance metric d for both the $\mathcal{L}_{text}$ and the $MEAD$ terms, was defined as the sum of *BLEU-1*, *BLEU-2*, *BLEU-3*, *BLEU-4*, and *ROUGE-L* calculated over the actual and the generated report.

Other text-related hyperparameters were chosen by empirical evaluation. In particular we used:

- max_length = 1024: The maximum length of the generated text sequence.
- temperature = 1: A parameter controlling the level of randomness in the generated text. Higher values increase randomness, while lower values make the text more deterministic.

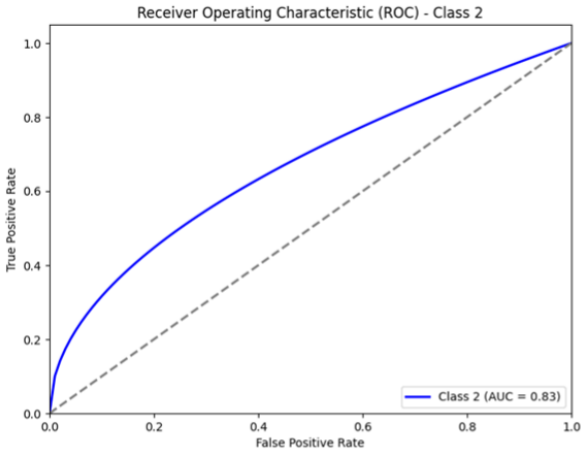


Figure 2.4: Example AUC-ROC plot for class "calcinosis"

- num_beams = 5: The number of beams used in beam search decoding. Increasing this value can improve the quality of the generated text but also increases computation time.
- no_repeat_ngram_size = 2: Controls the prevention of repeating n-grams (contiguous sequences of n words) in the generated text. A value of 2 means that generated sequences should not contain repeating two-grams.

2.4.5 Results

In this section we investigate performances of our proposed architecture. We first compute performance on classification tags and then we compare our approach against state-of-the-art deep learning model. Performance on text classification and tag prediction is show in the following table:

Metrics	**Value**
Test Accuracy	0.884
Precision	0.857
Recall	0.75
F1-score	0.799

In figure 2.4 and 2.5 are reported the AUC-ROC curve[89] and a confusion matrix for class "calcinosis", by choosing the worst case among the all predicted classes. This results for the classifying network of the tags impact and condition the latent space, and it is most important in implementing valuable strategy for report generation.

Confusion matrix in a 23x23 matrix that visually reports the results for all 23 classes, with a heatmap that goes from 0 to 1, with normalized correct prediction percentages, is presented in figure 2.6.

Table 2.1 reports the values obtained on the test set over ROUGE-L and BLEU metrics between the actual and generated medical report. Moreover, we compare our results with those available in the literature [66]. In such a table, the best results overall are reported in bold while the underlined values refer to the models which are outperformed by our VAE-GPT. It can be seen that our approach reaches comparable performances to the state-of-the-art approaches, and in several cases, it is able to outperform them. In fact, considering the

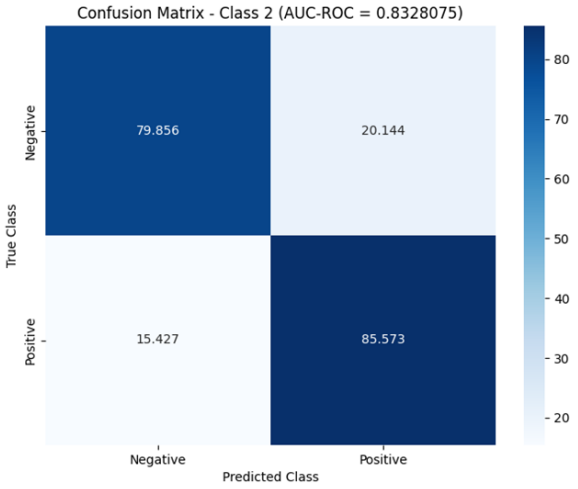


Figure 2.5: Confusion matrix for class 2 [0;100]

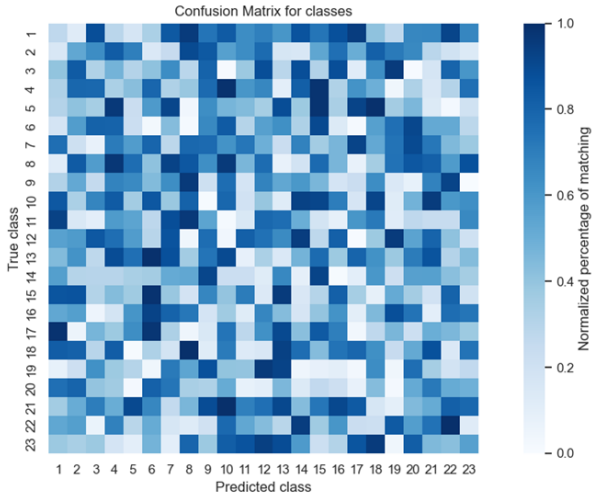


Figure 2.6: Confusion matrix for all classes

BLEU-4 and the ROUGE-L metric, our approach outperforms, respectively, 6 and 8 of the 12 existing approaches.

In figure 2.7 we provide a qualitative evaluation of a test reconstracted by our VAE.

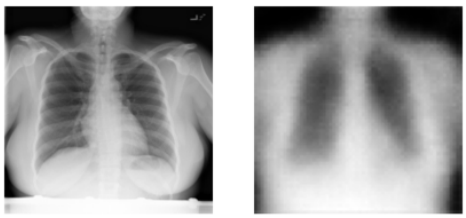


Figure 2.7: VOriginal vs reconstructed test image

The VAE outputs are often blurry (unfortunately it is a structural limitation of the model: these outputs should then pass through another network for output improvement), however, we are not interested in the reconstruction of the image as a task, but rather in the correct

Reference	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L
CoAtt [83]	0.517	0.386	0.306	0.247	0.447
Huang et al. [90]	0.476	0.340	0.238	0.169	0.347
Yuan et al. [71]	**0.529**	**0.529**	0.315	0.255	0.453
Xue et al. 2018 [78]	0.464	0.358	0.270	0.195	0.366
Vispi [82]	0.419	0.280	0.201	0.150	0.371
Singh et al. [72]	0.374	0.224	0.153	0.110	0.308
Yin et al. [77]	0.445	0.292	0.201	0.154	0.344
MLMA [75]	0.500	0.380	**0.317**	**0.278**	0.440
Harzig et al. 2019a [73]	0.373	0.246	0.175	0.126	0.315
A3FN [74]	0.443	0.337	0.236	0.181	0.347
Xue et al. 2019 [79]	0.489	0.340	0.252	0.195	**0.478**
Zhang et al. [81]	0.441	0.291	0.203	0.147	0.367
VAE-GPT (ours)	0.420	0.289	0.199	0.177	0.372

Table 2.1: Comparison with state-of-the-art approaches

construction of a continue latent space and extraction of the image features for the automatic generation of the reports.

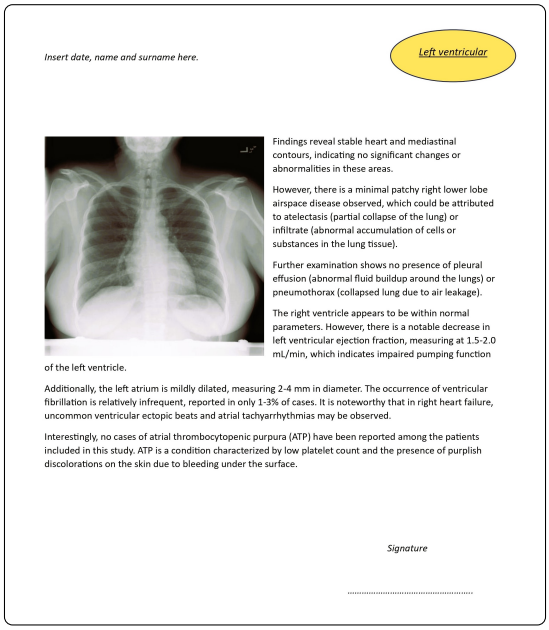
Figure 2.8 shows an example of a generated final report over a test image, using our report visualization algorithm.

To work with embeddings, the model uses FastText[91], a word embedding specifically designed and trained on biomedical data, and Ontology Sequence Generation to enrich the word vectors with medical domain-specific knowledge.

In figure 2.9 we report random variable selection combined with pca visualization of our embeddings. It can be see that our PCA preprocessing tend to cluster entities of the same category together, that indicates a semantic association between related entities within each category. The relative proximity and arrangement of displayed vectors provide insights into the relationship between specific organs and the medical specialties that primarily deal with them.

2.4.6 A Deeper Look at *MEAD*

We propose an ablation study executing different trainings of VAE-GPT by varying α. We compare the validation loss obtained by such trainings against VAE-GPT trained without $MEAD$ ($\alpha = 0$). The results are reported in Figure 2.11, which evidences how the introduction of $MEAD$ improves the VAE-GPT performances when $\alpha = 0.4$.



Insert date, name and surname here.

Left ventricular

Findings reveal stable heart and mediastinal contours, indicating no significant changes or abnormalities in these areas.

However, there is a minimal patchy right lower lobe airspace disease observed, which could be attributed to atelectasis (partial collapse of the lung) or infiltrate (abnormal accumulation of cells or substances in the lung tissue).

Further examination shows no presence of pleural effusion (abnormal fluid buildup around the lungs) or pneumothorax (collapsed lung due to air leakage).

The right ventricle appears to be within normal parameters. However, there is a notable decrease in left ventricular ejection fraction, measuring at 1.5-2.0 mL/min, which indicates impaired pumping function of the left ventricle.

Additionally, the left atrium is mildly dilated, measuring 2-4 mm in diameter. The occurrence of ventricular fibrillation is relatively infrequent, reported in only 1-3% of cases. It is noteworthy that in right heart failure, uncommon ventricular ectopic beats and atrial tachyarrhythmias may be observed.

Interestingly, no cases of atrial thrombocytopenic purpura (ATP) have been reported among the patients included in this study. ATP is a condition characterized by low platelet count and the presence of purplish discolorations on the skin due to bleeding under the surface.

Signature

.....

Figure 2.8: Example of Automatic Report Generation on test result

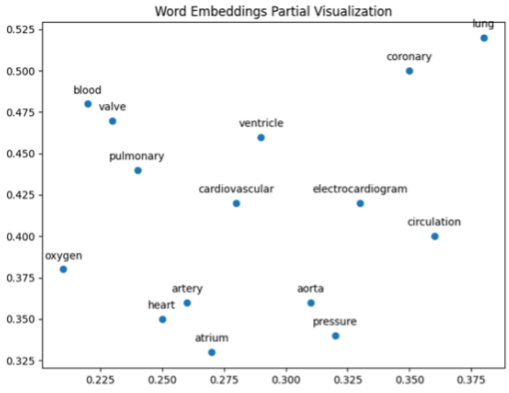


Figure 2.9: Random variable selection combined with pca visualization of our embeddings

2.5 Conclusion and future work

This paper started with an introduction to artificial intelligence in healthcare, highlighting the significance of research in this field from social, technological, and economic perspectives. The application of Transformers, Variational Autoencoder, and text generation in medical imaging is discussed. The literature review presents existing models and their results, with a focus on architectures, algorithms, and explainability. The proposed solution, VAE-GPT, is designed and implemented in Python. The main modules cover data pre-processing, Variational Autoencoder for images, and text generation with GPT-2. A new evaluation metric, Medical Embeddings Attention Distance (MEAD), is introduced to compare the distance between tokens with high attention.

The limitations of the framework are acknowledged, including data scarcity and quality, generalization issues, explainability challenges, the need for extensive testing, and the underutilization of VAE output. It would be interesting to extend and apply the model with

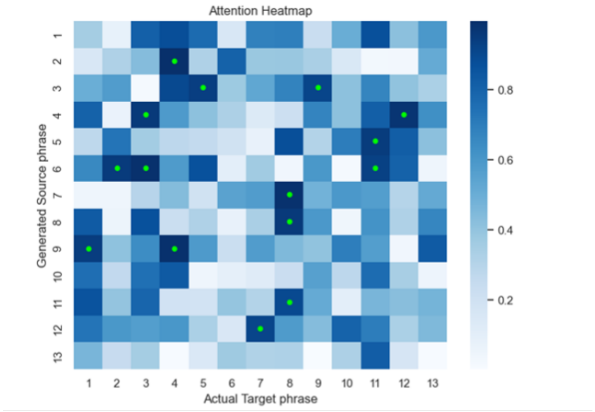


Figure 2.10: Test phrase attention result with a 0.9 threshold selection (green pointed on the attention heatmap)

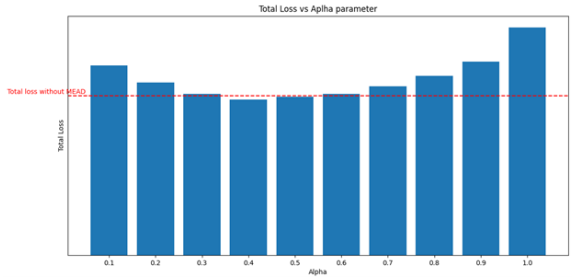


Figure 2.11: Total loss variation over alpha parameter tuning (each 0.1 pass)

Federated approaches [92, 93], in order to have each health facility serve as a client and train the model by sharing their weights without doing it with data. The conclusion emphasizes the importance of collaboration between AI researchers, clinicians, and industry stakeholders to overcome challenges responsibly and ethically. The integration of AI in medicine should complement healthcare professionals' expertise and improve patient care, without replacing them. This paper serves as a starting point for future developments in the field of AI and medicine in an innovative way, encouraging researchers to explore areas of improvement and pave the way for transformative advancements in medical practices. It is a call for responsible and ethical integration of AI technologies in healthcare, aiming to enhance efficacy, interpretability, and reliability for the benefit of patients and medical professionals alike.

Chapter 3

The Advent of Generative Agent in Agent-Based Modeling: Overview, Validation and Emerging Challenges

The advent of Generative Agents (GAs) based on Large Language Models (LLMs) has significantly influenced the evolution of Agent-Based Modeling (ABM), offering new perspectives across various domains, including engineering and social sciences. This paper provides an extensive overview of the integration of GAs into ABMs, emphasizing the advancements and emerging challenges in their validation. Traditional ABMs, characterized by their simplistic yet powerful approach to modeling complex systems, have been redefined with the introduction of GAs. This new generation of agents is often equipped with conversational capabilities. These agents, capable of simulating believable human behaviors and interactions, present unique opportunities and hurdles, especially in the context of urban simulations and social dynamics. We explore the nuanced differences between conventional ABMs and Generative ABMs (GABMs), delve into the state-of-the-art implementations of GAs, and discuss various validation methods. A critical focus is placed on the qualitative dimensions of these models, particularly the simulation and validation of emotions, to enhance the realism and applicability of GABMs in real-world scenarios. Through this comprehensive examination, we aim to shed light on the potential and limitations of GAs, advocating for a systematic approach to their validation.

3.1 Introduction

Various definitions of what is an artificial agent were given during the nineteens [94], highlighting features of agents such as autonomy, social ability, reactivity, and pro-activeness [95].
Such characteristics were formalized by [96] according to the following general definition:

"An agent is a computer system, situated in some environment, capable of flexible autonomous action to meet its design objectives."

that emphasizes two fundamental properties agreed upon by experts in the field, namely autonomy and social ability. Autonomy refers to the agent's ability to operate, decide, and con-

trol its actions independently, without external intervention [97, 98]. Social ability pertains to an agent's integration within a community, facilitating interaction with others to accomplish tasks and assist fellow agents [99, 98, 96].
Despite it being a widely accepted definition, the advent of Large Language Models (LLMs) suggests we rethink the entire Agent-Based Model (ABM) processes, from the design of new *humanized* agents able to perceive, think, memorize, and act as likely as possible to humans [100, 101], to the methodologies and approaches by which we validate their behaviors. Indeed, an emergent trend in research is the design of Generative Agents (GA) for ABM, such that agents' interactions and decision-making processes rely on LLMs. In this context, two classes of GA can be identified according to whether or not the agents are provided with conversational abilities, which allow agents to interact through natural language.
The term *"generative"* in ABMs was introduced by [102] to describe ABMs' ability to generate emergent social behaviors in the domain of social sciences. Although GAs and *generative* ABM share the same generative nature in this field, they are independent concepts and they should not be confused as (1) the first is agent-level while the second is system-level, and (2) one does not imply the use of the other. A more detailed description of both will be provided later in this work.
Focusing once again on GAs, we can now refer to the definition, provided by [103], by which generative agents are:

> *"Computational software agents that simulate believable human behavior."*

The idea of agents *believable behaviors* has been associated since from its origin to the art of animation [104, 105] (e.g. Disney characters), where animators aim to design apparent living creatures able to show emotions and act according to them, to create the illusion of life. This idea, while straightforward, lacks a formal interpretation of the agent's believability, making it difficult to validate the interactions within the system——especially in systems that involve human-like conversational interactions. A commonly used approach to address this issue is to engage human validators who manually review and assess the simulation history of each agent, ranking the agent's behavior based on perceived believability [103, 106].

This work aims to provide an overview of the evolution of ABMs and their validation in the GAs' era, shedding light on:

- The nature of ABMs and generative ABMs, their differences and definitions.
- The advent of GAs and their evolving status mechanisms.
- The validation methods for ABMs and their characteristics in the transition to generative ABMs involving GAs.
- A discussion on the need for formal and automated approaches for validation, which can be quantitative or qualitative.

This work is organized as follows: Section 3.2 will highlight the differences between traditional ABM and generative ABM. In Section 3.3, we will review state-of-the-art realizations of GAs underlying distinctive aspects of their different architectures. In Section 3.4, we review some of the most prominent validation methods for both ABMs and generative ABMs. The last section of this paper will explore the qualitative dimension of generative ABM.

3.2 From ABMs to Generative ABMs

The word *"generative"* in the field of ABMs was introduced by [102] and later extended in [107] referring to the ability of ABMs, framed into the field of social sciences, to generate emergent social behaviors. Epstein's work emphasized how ABM can be considered a generative approach to social science. He advocates for building models where social phenomena are "grown" from the interactions of individual agents. Indeed, his work is highly regarded for demonstrating how simple local interaction rules can generate complex social behaviors, making it a foundational text for those interested in the principles and practices of ABM. His interpretation of the field has inspired numerous subsequent works, where authors refer to generative ABMs as systems capable of producing behaviors from simple rules [108, 109, 110]. These systems, although generative, do not necessarily involve the use of generative algorithms or generative Artificial Intelligence (AI) in their implementation.

The contemporary use of the term *"generative"* can lead to ambiguity in this research field, particularly with the advent of generative AI, which can be used to develop enhanced agents or scenarios in complex systems. In recent works, authors developed ABMs by relying on GAs, referred to as GABM [111].
In the following, we aim to clarify the distinctions and establish a systematic terminology for different uses. For this purpose, we start from the basic definition and purposes of ABMs.

3.2.1 Definitions and Purposes

Since the beginning of the field, ABMs have been used to simulate systems with a broad and high-level approach, observing emergent properties and social phenomena by using extremely simplistic models to replicate real-world phenomena. This high level of abstraction grants the flexibility to use ABMs for a wide range of purposes, which [112] categorizes into seven distinct modeling purposes:

- **Explanation:** Involves creating a causal chain from an event to its consequences using the simulation's structure. This is crucial for understanding complex social phenomena, such as the causal architectures of bullying analyzed by [113]. At the same time, Macal (2016, p. 146) argues that ABMs provide a "framework for explicitly specifying causal mechanisms.

- **Prediction:** Defined as the ability to foresee aspects of unknown data with a valid level of precision using a computational model. A model's predictive capacity often indicates its accuracy. For instance, [114] predict bitcoin price trends using an ABMs by simulating rational agents' behaviors. While [115] presents a dynamic optimization approach for agent-based models for real-time bus locations and arrival times predictions.

- **Description:** Simulation models represent key aspects of a system without aiming to replicate it fully, focusing instead on documenting significant elements. [116] uses ABMs to describe important factors in tenant relocations.

- **Theoretical exposition:** Entails a systematic mapping and evaluation of mechanisms' consequences, aiding in formulating and testing hypotheses. [117] utilize ABMs to develop theories and test hypotheses about historical patterns.
- **Illustration:** Focuses on simplifying and exemplifying ideas to clarify complexities without making assertive conclusions. [118] use an ABM to demonstrate the effects of resource scarcity on group cooperation.
- **Analogy:** Uses simulation processes to informally reflect on concepts, often borrowing ideas from different domains to offer new perspectives. [119] computational game is an example of using analogy in ABMs to rethink cooperation.
- **Social learning:** Emphasizes the participatory aspect, helping diverse groups achieve a shared understanding. [120] employs an ABM for conflict management between herders and foresters in Thailand, fostering mutual understanding of land-use dynamics.

Using these purposes of modeling as a basis for our discussion, we assert that while all of these ABMs can be implemented by using GAs, the generative nature of the models depends on how they are used and how their results are explained. The presence of GA is not a sufficient condition to make a system generative. According to the definitions provided by [121], models are considered generative of an observed social phenomenon when they describe it in terms of the external (environmental and social) and internal (behavioral) mechanisms that generate it, rather than inferring causes from observed covariations—running the systems under different conditions and studying the variations. For example, when describing agent behavior through logical or numerical formalisms, the behavior is described from the outside, as perceived by an observer, but the internal generation process is not captured. The ability to explain behavior in terms of emerging internal processes makes an ABM system generative. Under this definition, and counterintuitively, even a model designed for **Explanation** or **Theoretical exposition** purposes is not necessarily generative. Therefore, we argue that the focus, when referring to generative ABM, should be on the generative properties of a system rather than the modeling purposes or the mere presence of GA.

As shown in Figure 3.1, we synthesize the different approaches by considering Generative ABM as the subset of ABM designed to generate emergent behavior and internal processes explanations. Within the context of Generative ABM, Generative Agents (GA) can be utilized and implemented in various forms and developed through different approaches (e.g. based on reinforcement learning, evolutionary algorithm, machine learning algorithm, etc.). With the advent of Large Language Models (LLMs), a promising new trend is to develop LLM-based GA. In some cases, these models can be equipped with conversational abilities, allowing them to interact with each other through natural language. Such agents can be referred to as social, conversational, and interactive.
In line with the current literature, we will refer to ABM designed with generative agents as GABMs and ABM when discussing the broader class.

AI represents a promising new frontier for ABMs, offering numerous opportunities to enhance their development. In [122], the authors provide an overview of Machine Learning (ML) techniques useful for addressing various challenges based on the purposes of ABM

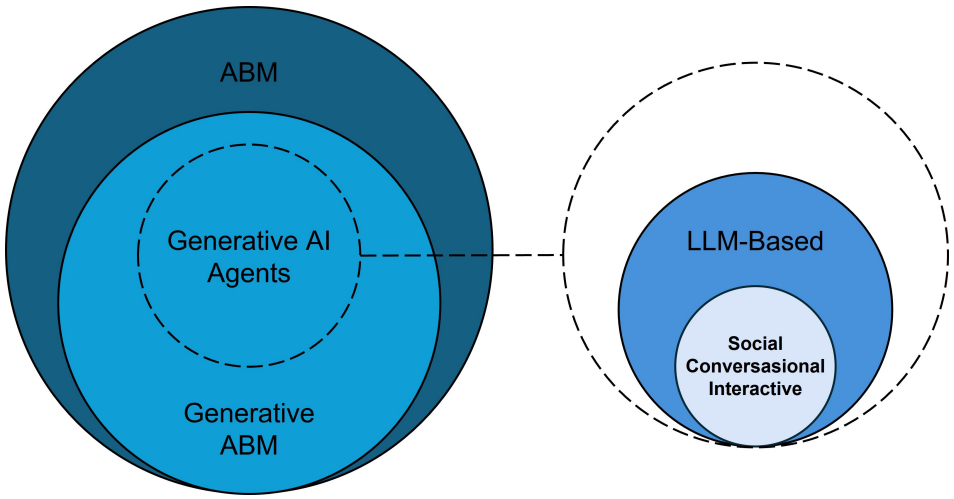


Figure 3.1: The diagram highlights Generative ABM as a specialized subset of the broader ABM set, aimed at producing emergent behaviors and internal process explanations. Generative AI Agents can be employed to develop Generative ABM. In the case of LLM-Based Agents, such agents can be equipped with social, conversational, and interactive capabilities.

modeling. Specifically, two primary areas of ML application can be identified: *structural specification* and *output analysis*. While this interpretation focuses on ML, it serves as a valuable starting point for our discussion, which extends beyond ML to focus on generative AI. Moreover, in this context, we will not consider *output analysis*, which is more related to data science. Instead, we will discuss the integration of Generative AI into structural specifications, focusing on agents' decision-making processes and interaction modeling. Therefore, the following brief discussion about the role of generative AI in designing ABMs will provide a foundation for understanding what we mean when referring to GA.

3.2.2 The role of Generative AI in ABM

Figure 3.2 visualizes how generative AI can be utilized in three key areas:

- *Population Synthesis*: generative AI can create synthetic datasets of agent characteristics and plans, providing ABMs with rich agent representations (e.g., [123]). In this case, even though agents and their behaviors are generated by a generative AI model, it is not accurate to refer to them as GA, as the evolution of agents within the system does not leverage generative processes, and their behavior is defined ex-ante.

- *Decision Making Through Interactions*: generative AI can model interactions between agents, and between agents and their environment. These interactions can be modeled through simple inference or systems that allow agents to interact using natural language (e.g., [103]). In this case, a generative algorithm is responsible for generating decisions dynamically during the simulation, considering its evolving history, making it appropriate to refer to these as GA.

- *Scenario Generation*: generative AI can aid in creating and characterizing environments (e.g. [111]).

The following discussion will focus on GAs, delving deeper into decision-making and interaction processes modeled using LLMs.

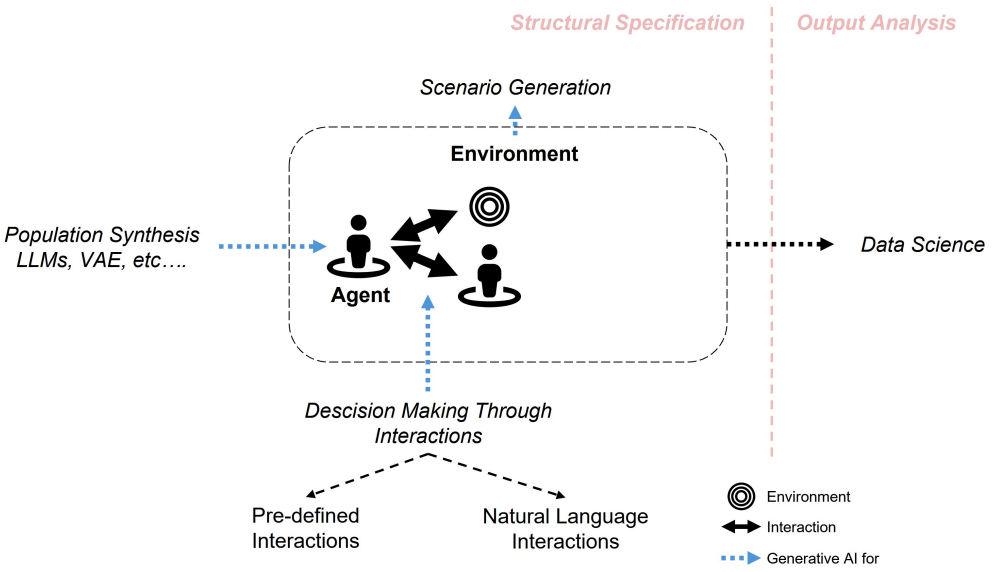


Figure 3.2: AI can be used in two distinct areas of ABM development: Structural Specification and Output Analysis. Over them, generative AI can contribute to Population Synthesis, Decision Making Through Interactions, and Scenario Generation.

3.3 Generative Agents

Since their appearance, LLMs have demonstrated remarkable capabilities in text comprehension and generation, which nearly match human-level performance. Indeed, they have emerged in the last two years as a promising research avenue for developing artificial agents that can exhibit human-like behavior [101]. We refer to such agents as GAs. Their potential into the field of ABM can be attributed to three main characteristics of these models: firstly, LLMs can perceive and apprehend the world, although this is limited to environments that can be adequately described through text; secondly, LLMs can plan and organize tasks by considering the task requirements and rewards. They keep and update a memory, whose mechanisms are guided by prompts based on human reasoning; lastly, LLMs can generate text that resembles human-produced language, enabling agents to interact with each other by entertaining natural language conversations.

Therefore, adopting an agent-based simulation paradigm that utilizes LLMs to simulate individual agents holds significant promise for capturing not only their behavior but also the complex interactions among them.

The connection between this technology and social dynamics is straightforward, and researchers can design GAs based on a feedback loop that improves individual decision-making using environmental data and historical information from evolving simulations and interactions. As proposed by [124], this feedback loop involves the environment and agents exchanging feedback about actions and current states with LLMs. In more detail, a mechanistic model—a rule-based environment defined by developers—captures the system's state, provides information to the LLM, and uses LLM-informed decisions to shape the system's state. A notable feature of this feedback loop is that decision rules emerge from the extensive knowledge embedded in the LLMs rather than being imposed by modelers.

Based on Ghaffarzadegan contribution, we propose an extension of the feedback loop by focusing on GA's decision-making processes. In particular, as shown in Figure 3.3, we dis-

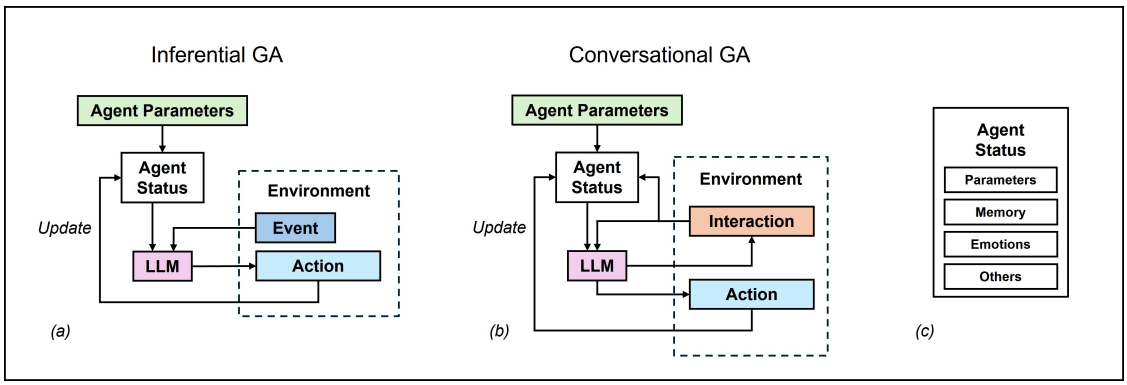


Figure 3.3: LLM-Based Feedback loop and Status of GAs: *(a)* Inferential GA feedback loop where the agent receives in input the status and an event e produces an action; such action updates the agent's current status. *(b)* Conversational GA feedback loop where the LLM decides based on agent status and natural language interactions. Such interaction can directly affect the status. *(c)* Example of an Agent Status content.

tinguish between the *inferential* case, in which LLMs are only used as decision-makers (Figure 3.3 *(a)*), and the case in which GAs are equipped with *conversational* capabilities, hence when LLMs are used to enable natural language interactions between GAs (Figure 3.3 *(a)*). When GAs are developed with an inferential approach, agent decision-making is confined to a discrete GA characterized by Agent Parameters and holds Agent Status in both case regimes, where LLMs generate actions based on ongoing conversation dynamics and environment perception.

3.3.1 Parameters and Status

A GA is characterized by Agent Parameters and holds Agent Status in both cases. Agent Parameters typically include demographic information that identifies personas, such as high-level data like age, sex, income level, race, employment, and location, as well as social information detailing the relationship between agents [125]. This information, when combined with the extensive knowledge embedded in LLMs, can guide the decision-making process of an agent to accurately emulate a human-like personality. Agent Status, on the other hand, represents the current state of the agent within the evolving simulation (see Figure 3.3 *(c)*), and in particular, it can be seen as the GA representation itself, excluding its decision-making processes (which are implemented by the LLMs). The status can be characterized differently depending on the application, potentially involving parameters and an evolving memory updated during the simulation. It may even include representations of a discrete emotional state. These sentiment states can be generated by the LLMs themselves, as demonstrated in the work by [126], which employs natural language processing techniques (BERT [127]) to perform frequent global measurements of emotional states through sentiment, assessing the impacts of the pandemic and related policies.
LLMs can exploit the information contained in the Agent Status to enhance and magnify the GA's decision-making processes, which can vary according to different human-like personalities.

3.3.2 GAs Examples

We advocate this section to explore the application of GAs for GABMs.
Concerning the inferential GAs, an interesting implementation has been proposed by [128]. The authors developed an epidemic model that integrated human behavioral dynamics in response to evolving outbreaks by defining an inferential LLM-based architecture. These agents effectively mirrored human actions, exhibiting adaptive behaviors such as self quarantining when unwell and isolating as cases escalated. A crucial contribution of this work is the systematic method used for defining personas, where agents' personalities are defined on an Agent Status based on the Big Five personality traits by [129] from the field of psychology. This study is good example on how personality traits rooted in psychology literature can influence the LLM's responses.

More recent studies concern the development of conversational GAs to be used in GABMs. An initial breakthrough in this field was recently achieved by [103]. In this work, the authors proposed an original architecture for GAs to design entities that emulate believable human behavior in interactive settings. These GAs can carry out everyday activities like eating breakfast, working, and conversing with each other. Such innovative agents' architecture strongly relies on natural language processing and generation in its key components: *memory streams*, *reflection*, and *planning mechanisms*.
The *memory stream* is a module that records experiences in natural language. To query such a memory, the authors designed a retrieval function based on the triple $\{$*Recency, Importance, Relevance*$\}$. This module enables the GA to reflect on past events and plan future actions based on their accumulated experiences. With this, the authors claim to address the need for AI systems to manage complex and evolving memories, aligning with the focus in the current state-of-the-art on developing AI systems with personalized and customized experiences.
The *reflection* is instead a system for incorporating reflective knowledge synthesis within the architecture. This feature empowers the GA to generalize and draw higher-level inferences from their memories. Agents are equipped with a reflection module to generate trees (hierarchy) of reflections where the leaf nodes are base observation, and the non-leaf nodes represent thoughts that become more abstract and higher-level as they ascend the tree, summaries of the previous experiences. This way, the agents can derive meaningful insights from their experiences.
Lastly, the authors designed a *planning and reaction* mechanism based on the *reflection* and the current situation of surrounding gas and the environment. These mechanisms enable the GA to create plans and responses that are consistent and appropriate over time and w.r.t. the autonomously evolving scenario.
Through the described components, the agents can record experiences, reflect upon them, and plan future actions based on past events. A key characteristic of Park's GA architecture is that it couples LLM capabilities and a continuously updated memory stream that stores agents' experiences. Hence, in a continuous regime, items in the memory stream are retrieved to guide the LLM generating actions, plans, and reflections.
Furthermore, the authors developed a virtual sandbox environment inspired by "The Sims" in which they simulated agents' behavior. In particular, 25 agents were observed and interacted with by testing for their individual and social behaviors (e.g., planning a Valentine's Day party autonomously). The simulation showed remarkable agent autonomy, coherence, and believability.

The work from [103] has been a pioneering and disruptive contribution to the GABM and Computational Social Sciences field and is highly inspiring for the research community.

By relying on Park's GA architecture, [111] recently introduced Concordia, a library that facilitates the development of language-mediated GABMs based on LLMs, to provide agents with more sophisticated cognitive capacities. These capacities enable the agents to perform actions grounded in physical, social, or digital spaces with complexity and realism that traditional ABMs cannot achieve. Concordia GAs consist of associative memories, inspired by [103], and flexible components that mediate between LLM calls and memory retrieval. The Agent Status is equipped with what the authors call *components*, which are statements in natural language expressing the current state of an agent in terms of identity, plans, observation, and call for action. Each of these statements serves as a base knowledge to generate context for decision-making. All interactions between GAs are facilitated by a unique Game Master (GM) whose behavior can be grounded by physical, chemical, financial, etc. mathematical models of the real world. The GM simulates the environment, handling the world's state, agent actions, and interactions between GAs. It generates event statements representing agent actions' outcomes, mediates interactions, and updates environmental variables. Special attention in this work is given to such environmental variables, referred to as *Grounded Variables*. They represent the environment that can influence and be influenced by agents' actions within the model. They are important in enabling state tracking, mapping interaction effects, simulation integrity checking, and control and manipulation.

Another interesting work inspired by Park's agents recently proposed by [130]. They designed a new agent architecture, called Lyfe, which relies on a sophisticated memory system and Hierarchical Action Selection component. The latter allows GAs to be cost-effective by delegating only necessary decisions and interactions to LLM, avoiding the wide number of text generation calls usually needed to allow system continuity. As an even more interesting original contribution, and conversely to existing prominent GAs (e.g. [103, 131]), the Lyfe agents were provided the ability to entertain conversational interaction with multiple agents.

A comprehensive review of existing architectures for autonomous agents (not necessarily framed in the field of ABMs) has been recently provided by [132].

3.3.3 Simulating Emotions - The Qualitative Dimension of GABMs

Another interesting implementation of GAs has been proposed by [17]. In this work, the authors designed an architecture for LLM-powered agents to simulate social network dynamics. These agents could observe content posted by other agents, adjust their attitudes and emotions, create their own content, or remain inactive. They successfully replicated the diffusion of information and the shifts in emotions and attitudes of users in response to two events: the release of nuclear wastewater in Japan and the incident involving the mother of eight chained children.
Although their approach shows promise for flexible behavioral modeling, in contrast with the GAs from [103], the agent's reasoning here relies solely on the LLM without incorporating explicit computational mechanisms that represent human cognitive processes, such as memory retrieval based on relevance, recency, and frequency.
Interestingly, the authors introduce Emotions information into the GA's status. It goes with-

out saying that to effectively simulate real-world scenarios it is essential to equip agents with their own cognition, attitudes, and personalities. However, to make the behavior of GAs more human-like, we must consider how humans are often emotionally triggered and how they translate these emotions into their choices. Emulating emotions is crucial for simulating human behavior, as it significantly influences how agents convey their intended actions. Although this is challenging due to the myriad factors and complex relationships involved in human emotions, the rich knowledge of human behavior embedded in LLMs provides an opportunity to introduce an individual emotional dimension in GAs, enhancing the realism of simulations [133, 134, 135].
We want to stress how the inclusion of emotions in the GABMs framework underscores the importance of qualitative aspects to understand agent behavior in social scenarios. In the following we will discuss the challenges that this brings in the simulation results analysis. Moreover, we will provide a deeper discussion on the opportunity to realize a qualitative representation of the real world to enhance the potential impact of simulation practices into the field of complex systems and social dynamics.

3.4 Validation

3.4.1 Informal Theory of ABMs Validation

In ABMs, validation is essential for gaining trust and credibility in simulation models, particularly ensuring that the model meets its intended purposes and generalizes to real-world social dynamics. In this field, validation is an intricate problem because of the complex mechanisms and scenarios these models attempt to represent. Moreover, there is a lack of universally accepted standards for ABMs validation, which makes it even more difficult for model designers to demonstrate the validity of their system. This introduces the need for formal methods and rigor in the validation processes for ABMs [136, 137].

Furthermore, an explicit hierarchy of evidence should guide ABMs validation. This can be inspired by the one used in evidence-based medicine and policy-making, where the strongest evidence comes from ecological validity and rigorous experiments, while observational data and consistency with previous theories are considered weaker [138]. It is worth noting that the rigor required for validation depends, again, on the purpose of the ABM. In a model intended only to guide further research, the threshold for evidence can be reasonably lower. It's crucial to note that the model's hypothesis should not only be evaluated based on direct evidence but also on their "productivity" or their ability to stimulate further empirical research and theoretical developments [139]. On the other hand, when the model is used to guide real-world decisions with significant consequences, validation should require higher rigor and evidence.

A recent attempt to define guidelines for ABM validation has been proposed in [140], where the authors, after reviewing different validation approaches, discuss how the choice of validation methods should be tailored to the specific needs of the simulation, the characteristics of the model, and the available data. In particular, they classify nine key validation-supporting methods into foundational and advanced, offering insights into their application and effectiveness. Such a classification follows, along with a brief description of each method.

Foundational Methods

- **Data Analytics**: essential for handling input and output data, data analytics supports the validation by ensuring data integrity and relevance. This method involves extensive data cleaning, organization, and analysis, including statistical tools.
- **Docking**: a method for comparing the outputs of the simulation with the ones of other state-of-the-art ABMs to verify if they align with each other, thus validating the assumptions and results of the designed new model.
- **Empirical Validation**: involves using real-world data to fit and validate model outcomes. This method is crucial for ensuring the model's predictions are realistic and grounded in actual data.
- **Sampling**: covers the various sampling techniques used to explore the simulation space and understand the impact of different input variables on the model's output.
- **Visualization**: focuses on using graphical representations to understand and analyze the behavior and outcomes of simulations. Visualization helps stakeholders easily interpret complex model behaviors.

Advanced Methods

- **Bootstrapping**: this technique uses resampling to create empirical distributions of model outputs, which helps assess the simulation results' accuracy and reliability.
- **Causal Analysis**: Look at the model's causal relationships to identify and correct any erroneous or unexpected behaviors. This method is vital for understanding the causal dynamics within the model and ensuring its validity.
- **Inverse Generative Social Science**: A novel approach that uses techniques like evolutionary algorithms to explore various potential agent behaviors and their consequences within the model, helping to uncover all feasible configurations that lead to particular phenomena, backward tracking behaviors, and decision from an expected output to the input.

The idea of tailoring the validation methods to the specific needs and purposes of the simulations is further supported by previous theory on ABMs validation [137]. In this work, the authors provided guidelines for rigorous validation—meaning the model corresponds to reality—of ABM in marketing research, which can be extended to a wider set of applications. According to them, there are four steps for considering a model to be rigorously validated.
Two of these steps focus on *empirical validation* of ABMs, specifically concerning the validation of the model's input and output. Using statistical measures, they ensure that these elements align with real-world data or are comparable to other models' outcomes. These validation steps can be easily performed by using the **Foundational Methods** previously described. However, they require the availability of data or the existence of a similar model for comparison.

The remaining steps from [137] involve the concept of *"face"* validation, and they do not necessarily rely on data availability. Rather, they rely on human intelligence through expert assessments and structured walk-throughs. Face validation ensures that model processes and outcomes appear reasonable and plausible based on expert and stakeholder knowledge. Such validation steps are namely:

- **Micro-face Validation**
  This process ensures that the individual-level mechanisms and properties of the model correspond to the real-world ones. It checks if the agents and their actions realistically reflect human-like dynamics.

- **Macro-face Validation**
  This step involves validating that the aggregate patterns and dynamics of the model correspond to real-world phenomena. The focus is on the emergent properties and overall behavior of the system.

The model is not directly compared to real-world data in microscopic and macroscopic validation. Instead, the emphasis is on showing that the general model's attributes and processes reasonably correspond to the ones of the real world.
Micro-face validation is particularly relevant because it allows us to examine the system's most detailed and specific aspects (such as the individual behavior of GAs).
Conversely, macro-face validation is essential because it offers a broader perspective of the system. This is especially important in situations where direct individual validation is challenging or impractical, and validation granularity is not required (e.g., prediction of land uses in city simulations). Moreover, this proves useful when empirical validation is not possible due to the absence of data. In such a case, **Advanced Methods** such as the one based on causality, can be applied to infer the system's dynamics and validate its functionalities.
The next sections will provide a further and more detailed discussion on the uses of *face* validation in GABMs.

According to [141], *face* validation involves three key methodological elements:

1. **Animation Assessment**: Experts review simulation animations to ensure that the system's or its components' behavior appears realistic. Visualization methods can help observe the simulation from a general perspective to assess dynamics like passenger flows or traffic jams and potentially track individual agents to scrutinize their specific behaviors.

2. **Output Assessment**: This involves human experts examining the simulation outputs for plausibility. This includes checking whether their relationships and the dynamics and trends observed in the simulation outputs are coherent. This assessment can be automated if the relationships among the inputs, the evolution, and the outputs can be formally defined (e.g. through constraints and rules definition).

3. **Immersive Assessment**: Experts immerse themselves in the simulation by adopting the perspective of an individual agent within the system. This allows them to directly evaluate the believability of an agent's behavior and reactions, as well as the interactions and reactions of other agents within the simulation environment. The effectiveness of this method depends significantly on the design of the user interface, which

may need to be customized for different types of agents.

Animation and Immersive Assessments share the fundamental principle wherein human validators replay simulations by impersonating agents. During this process, they make informed decisions that are subsequently compared to those made by the agents. While conceptually similar, Immersive Assessment provides a more advanced framework that allows validators to fully experience the simulation and have a complete sensorial perception, for example, through virtual reality [142]. This approach is particularly valuable when validating human emotions is necessary, but relevant data is unavailable.

Overall, these validation techniques help ensure that the model is not only statistically and structurally sound but also intuitively plausible to human experts, crucial for its acceptance and usefulness in real-world applications.

3.4.2 GABMs Validation

Many aspects of model validation concern both GABMs and ABMs, although GABMs introduce additional unique challenges (e.g. believable behaviors).
When discussing GAs performing actions or visiting places, we use simpler language for easier understanding. However, this does not mean these agents have human-like abilities or intentions. Their actions are designed to seem realistic and engaging, similar to characters in animated Disney movies, but this does not imply they possess true independence or consciousness.
Current state-of-the-art GAs rely on LLMs to enhance the illusion of human-like behavior, as this new generation of language models encapsulates information on human behavioral patterns. However, this is only believable to a certain extent due to issues such as hallucinations and biases in LLMs. Additionally, there is no automatic method for validating LLMs, and the current trend involves employing a large number of human validators to assess these models, followed by a voting system to compute performance metrics based on human evaluations [143, 144].
Recent studies [145, 146] discuss how current LLMs lack cognitive competence in ways that sometimes produce fragile and even bizarre behavior. In particular, the study demonstrated that GPT-3 could solve many tasks reasonably and showed some human-like behaviors. However, it highlighted significant limitations in the model's cognitive abilities, particularly in context sensitivity and causal reasoning.
These problems are inherited by GAs in GABMs, and they become even more severe in the case of conversational agents, where the validation can require the analysis of natural language interactions. Again, human validators can be employed in this case, but it is easy to understand that this approach does not scale to simulation involving thousands of interacting agents (e.g. in cities).
Currently, there is no agreed-upon method for interpreting the outcomes of simulations that use LLMs to model human populations. The critical question remains: what criteria should we use to determine if, how, and under what conditions the findings from simulations might apply in real-world settings? These are not questions that can be addressed by a single group of researchers alone; instead, they must be collectively discussed and resolved by the entire research community.

While this remains a topic of debate, the research community should focus on the discovery of formally defined methods that will allow us to confidently predict that a model's outcomes can be generalized to real-world dynamics. Identifying these methods should be the foremost priority for this emerging field [147].
As of today, while there exists a wealth of literature on ABM validation methods, it is challenging to find new specific validation methods proposed for GABMs. Therefore, we leverage ABM literature as a basis to detail the challenges introduced by GAs. We do this by re-analyzing *face* validation in the context of GABMs. In particular, we discuss how the previous theory can be applied to the transition to GABMs and the new challenges it presents:

- **Micro-face Validation** When transitioning from traditional rule-based agents to GAs, two dimensions of this type of validation emerge: validation of individual decision-making and natural language interaction. For decision-making, the validation should focus on the decisions made by GAs and assess the believability of social dynamics. For interaction, it should examine the conversations between agents at a deeper level of granularity to ensure that decisions are coherent with the GAs' natural language interaction history. Additionally, validating emotions (or other qualitative aspects) can be seen as a third dimension, and it is crucial to determine if agents align with human emotional dynamics.

- **Macro-face Validation** While this approach does not significantly differ from traditional ABM to GABM, it is essential because it enables the validation of systems where individual agent validation is impractical (e.g., simulations with thousands of agents). Advanced validation methods such as Bootstrapping, Causal Analysis, and Inverse Generative Social Science prove invaluable.

Face validation generally provides a framework for validating systems where empirical validation data may not be available. With the advent of LLMs, we can leverage the embedded knowledge within these models to simulate scenarios where data is lacking accurately. Populations can generalize to different places and scenarios without relying on real-world data. Explaining *face* validation is essential to take full advantage of this great opportunity.

GAs Validation - Examples

This Section details how the state-of-the-art GAs discussed in Section 3.3 validate their systems.
As a first example, [128] uses visualization, data analysis, and sampling methods to validate their simulations. Interestingly, this demonstrates how, in the case of inferential GAs, GABMs can be validated even only employing one or a set of foundational methods for traditional ABMs that are widely accepted by the community (see Section 3.4.1).

Delving into Conversational Agents and in line with what we mentioned about leveraging human validators, [103] recruited 100 evaluators (paid 15.00$ per hour) for the validation process to demonstrate the effectiveness of the proposed architecture in generating believable behaviors. The validation process encompasses controlled experiments and end-to-end assessments.

- *Controlled Evaluation* to answer the question: *"Do individual agents properly retrieve past experiences and generate believable plans, reactions, and thoughts that shape their behavior?"*. For this purpose, they interviewed the agent in five question categories (self-knowledge, memory, plans, reactions, and reflections) using an LLM. They use *believability* as an evaluation metric, measuring it according to human evaluators watching the replay of a 2-day simulation of a randomly sampled subset of agents. The system ran for five different conditions (ablation), and the evaluators ranked the conditions as most or least believable.

- *End-To-End Evaluation* to answer the question: *"Does a community of agents demonstrate information diffusion relationship formation and agents coordination across different pockets of community?"*. For this purpose, they run a 2-day simulation by tracking 2 pieces of information from two single agents: the organization of a party and the candidacy for mayor. They test for (1) information diffusion: they interview the 25 agents with "do you know about the party?" and "do you know about the candidacy?"; (2) relationship information: they constructed an undirected graph of agents where edges represent mutual knowledge between two of them and they measure the density of the network (increases as the simulation keep going); (3) agent coordination: they count the agent which actually went to the party.

While they designed a state-of-the-art architecture for GAs, overcoming the limits of existing GAs whose believability falls short in extended simulation, they validated their system using a qualitative measure of coherence between the agents' behavior and the expected behavior of humans in similar scenarios, often influenced by historical context, rather than a formal representation of believability. Their method recalls the micro face validation approach at a conversational level and the concept of Immersive Assessment discussed in Section 3.4.1, where human validators assess the agents' behavior by experiencing the evolving simulation. Indeed, in this particular case, the immersive experience engages validators in the conversation between agents and their resulting decisions.
As previously mentioned, [103] proposed an effective validation method involving 100 human validators to inspect the behavior of 25 agents. This indicates that for GABMs, the number of human validators needs to grow with the number of artificial agents involved in the simulation. This makes it difficult to validate the agent's behavior for large systems, and even for smaller ones without such disposal of human contributors.
Additionally, specialists are involved in designing the interviews, making the process subjective and qualitative rather than formal.
However, a promising automatic approach in this work involves using LLMs to interview GAs. Despite the known issue of hallucinations in LLMs and the current lack of automated tools to detect them, this method offers potential for scalable validation.

In [111], particular emphasis is placed on model validation, providing guidance on evaluating simulation outcomes. The work discusses best practices such as measuring generalization and conducting sensitivity analysis.
The validation framework that the authors propose is based on Empirical Validation, referred to as the *Similarity Principle*. This principle emphasizes the similarity between tested and untested samples. A model makes a family of related predictions, and testing one prediction can provide confidence in similar, untested predictions. In the context of human-behavior validation, the authors advocate that the gold standard is the direct measurement of model

predictions on new test data that could not have influenced the model's parameters. For example, if a model predicts how humans will behave in a particular situation, the best evidence is measuring actual human behavior in that situation. This approach is emphasized by the authors as the most robust form of validation.
We underscore how this would require the availability of data or the development of a framework, tailored to the specific GABM, for immersive validation where real human behavior can be compared.
They also provide validation recommendations that extend beyond empirical testing by discussing the concept of *Algorithmic Fidelity*. Algorithmic fidelity describes how well a model can be conditioned using socio-demographic backstories to simulate specific human groups. This fidelity must be measured anew for each research question, as it may vary across different topics and experiences. One direction for further research in this context would be based on the work by [148], which involves checking for the fidelity of human personality traits in LLMs. As a concrete validation practice, it is essential that models be checked for consistency with prior theory, whether they pertain to psychological aspects (e.g., testing human behavior) or structural aspects of the real world (e.g., validating a GABM modeling consumer behavior by showing that prices in the model move according to classic microeconomic theories).
Additionally, the authors in [111] discuss open challenges and unresolved issues, focusing on three main problems: (1) *Train-Test Contamination*: it is crucial to ensure that LLMs have not been contaminated or biased by the training data. (2) *Representation of Stereotypes*: LLMs likely represent stereotypes of human groups, which may inadvertently lead to studying stereotypes rather than real lived experiences. (3) *Detail Limitation*: understanding what happens when models are pushed to their limits in detail remains an unresolved issue.
Finally, they emphasize the importance of addressing *Hard-to-Reach Populations*. For populations that are difficult to recruit for experiments, more flexible approaches to validating GABMs representing these populations may be necessary.

In the work by [17] the authors used Empirical Validation for different aspects of their system, such as: *information propagation*: the system simulates the temporal dissemination of events and compares these simulations with empirical data to validate the accuracy of its predictions. For instance, simulations of the Eight-child Mother Event and the Japan Nuclear Wastewater Release Event showed good accuracy in forecasting propagation patterns, capturing distinct emotional peaks and dissemination trends; *emotion and emotion Propagation*: emotions are classified into three levels: calm, moderate, and intense. The model predicts the user's emotion level in subsequent time steps by considering the current emotion level, user profiles, user history, and messages received. This prediction mechanism is validated using real-world data. The emotional propagation process, on the other hand, is simulated by extracting emotional density from textual interactions among agents. Again The effectiveness of the model is demonstrated by its ability to reproduce the dynamic process of emotion propagation observed in real-world data; *interactive behavior*: The model simulates users' decision-making processes regarding whether to forward content, create new content, or remain inactive. Validation is performed through metrics like accuracy, AUC, and F1-Score, demonstrating robust performance in scenarios like Gender Discrimination and Nuclear Energy.
Although, in this case, the authors collected real-world labels, obtaining data on emotions and other qualitative aspects of GAs and models in different scenarios is always challeng-

ing. This underscores the need for new face validation methods for qualitative aspects at the macro and micro levels.

A promising approach to GAs validation has been proposed by [106], where the authors introduce a comprehensive framework for evaluating agent interactions across various scenarios. Interestingly, they designed an *holistic* social agent evaluation framework, SOTOPIA-EVAL, using seven dimensions grounded in established literature from sociology, psychology, and economics. Both human annotators and GPT-4 are used to evaluate agents based on this set of dimensions:

- **Goal Completion**: evaluators consider whether the agent met its explicit goals by the end of the interaction.
- **Believability**: it includes assessments of naturalness (how realistic the interactions are) and consistency (alignment with the agent's personality, values, and background).
- **Knowledge**: evaluators look at what new information the agent has gained, its relevance, and its importance to the agent's goals.
- **Secret**: evaluators assess whether the agent successfully kept its secrets and the impact of any leaks on the interaction.
- **Relationship**: evaluates the impact of the interaction on the agent's personal relationships. Social interactions often aim to maintain or improve relationships, and this dimension measures whether the interaction has a positive or negative effect. Includes changes in the perceived quality of the relationship, social status, and reputation resulting from the interaction.
- **Social Rules**: evaluators consider whether the agent violated any social norms or legal rules and the severity of such violations.
- **Financial and Material Benefits**: includes short-term monetary gains, long-term economic benefits, and overall financial outcomes of the interaction.

This comprehensive framework for evaluating artificial agents is holistic, yet it misses important qualitative dimensions such as emotions, mood, attitude, etc. [149, 17, 150]. Despite this, among these various dimensions, GPT-4 demonstrated a strong correlation with human judgments, particularly when dealing with the impact on relationships. This includes evaluating the agent's perceived quality of relationships, social status, and reputation resulting from interactions. This finding further proves how LLMs can understand and handle qualitative aspects of social interactions.

3.5 Discussion

The integration of generative agents enhanced by Large Language Models (LLMs) offers a promising approach to creating more nuanced, context-aware, and emotionally intelligent urban simulations. This advancement has significant implications for urban social science, enabling the examination of complex social phenomena in a controlled and replicable manner [132, 151].

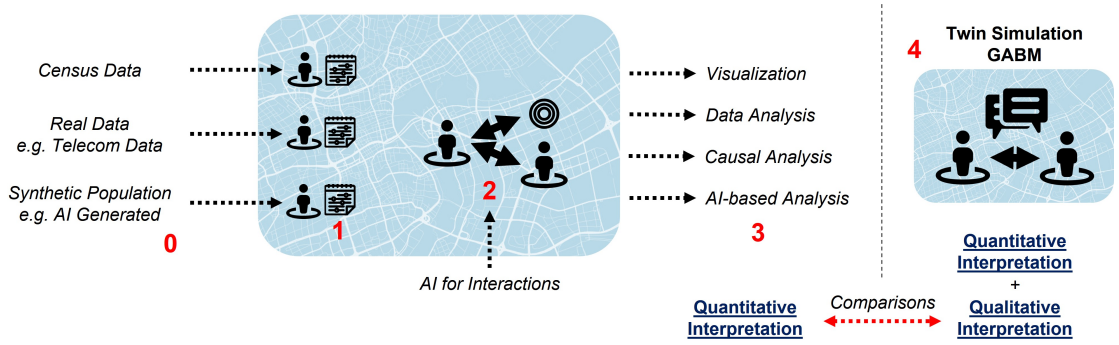


Figure 3.4: A comprehensive city simulation example.T his diagram illustrates the traditional city simulation framework, starting with data collection and integration (census, real-world, and AI-generated data) in Step 0. Next, a schedule of decisions for agents is created (Step 1), followed by modeling agent-environment interactions (Step 2). Results are analyzed through visualization, data, causal, and AI-based analysis (Step 3). Finally, a twin GABM simulation (Step 4) combines quantitative validation with qualitative interpretation of human-like behaviors, enhancing overall insights.

3.5.1 From Quantitative to Qualitative Interpretation

Generative agents unlock the potential for qualitative interpretations in urban contexts, addressing a notable limitation of previous approaches [152]. By incorporating models of human cognition, LLMs can develop cognitively plausible agents that exhibit behavior consistent with human decision-making processes, including emotional responses and the influence of incomplete information [153, 150].
The concept of agent cognition, introduced by Russell and Norvig [154], provides a foundation for understanding how agents examine their environment and make decisions. In real-life scenarios, this process is influenced by qualitative features such as emotions [150]. While current approaches often rely on categorical representations of emotions [17], future challenges lie in representing these qualitative variables as continuous embeddings [155]. LLMs' natural language processing capabilities make them well-suited for this task.

Potential Impact on Urban Planning and Policy Making

GAs present a valuable opportunity for urban social science. These simulations can shed light on social phenomena, such as community formation, social networks, and the impacts of urban policies and intervention in a controlled and replicable manner [151]. These applications can enhance human decision-making processes (e.g., in urban planning) and enable detailed modeling of social dynamics.
In Figure 3.4 we show a complete example of how urban simulation are traditionally designed, highlighting how the advent of GAs enables for comparison between qualitative and quantitative interpretation of the results.
In urban planning and policy-making contexts, the impact of GAs could be transformative. By creating representations of residents' emotions and perceptions towards urban spaces, we move beyond simplistic models to a more nuanced understanding of urban environments. Indeed, the qualitative interpretation facilitated by these advanced models aims to comprehend subjective experiences, feelings, and the underlying meanings behind actions in urban contexts.
This approach enables urban designers to create detailed experience maps, visualizing the

emotional highs and lows that users experience when engaging with various urban spaces [156]. Such qualitative interpretation of emotions helps identify opportunities for improving the overall urban experience, leading to more responsive and human-centered urban planning. This holistic approach promises to transform urban planning from a primarily functional exercise to one that truly enhances quality of life for city dwellers, allowing for the development of more targeted and effective urban improvement strategies, policies that consider emotional well-being, and public spaces that cater to diverse needs [157, 158].

3.5.2 GAs: Emerging Challenges

While the advent of generative agents in ABM presents significant opportunities, several challenges remain. One of the primary challenges is the computational capacity required to run and validate simulations at scale, where thousands of GAs are freely interacting with each other (e.g. in urban simulations). But besides the need for computational resources, actual challenges concern the qualitative aspects of GAs. In the context of emotion representation, existing agents are often characterized by discrete emotional states (e.g., happy, sad, etc.), relying on categories that are typically shaped and constrained by the available data. From this, two key challenges emerge: First, categorical modeling should adhere to scientifically accepted definitions of emotions [149] rather than on the only availability of data. Second, future work should aim to represent emotions and other qualitative variables as continuous embeddings rather than discrete categories [155]. This shift would allow for capturing nuances and intensity levels on a continuous spectrum, moving beyond simple labels like ”happy” or ”sad”. LLMs' capabilities make them particularly well-suited for the task of generating and interpreting continuous emotional representations. However, there is a need for more advanced approaches to achieve more reliable emotional representations and, notably, their validation.

Validation

The need for GAs equipped with cognitive abilities is growing, but it's important to note that while agent perception is a significant source of qualitative dimensions in our models, it's not the only one. Qualitative aspects can arise from various sources: agent interpretations of their environment (e.g., perceived comfort levels in a crowded street), inherent characteristics of the environment (such as cultural significance), and our interpretation of agent behaviors [**?**]. Agents also process quantitative data, like distances or population counts. The richness of our urban simulations stems from this interplay between qualitative and quantitative elements, combining agent perceptions, predefined parameters, expert knowledge, and interpretive analysis.
This complexity makes the validation of GABMs an even more intricate problem when the GAs are equipped with qualitative features.
When it comes to cognition and emotion, it is challenging to find appropriate data for the empirical validation of GABMs. On the other hand, while face validation methods can be useful to ensure the model's validity. However, it is also essential to compare the agents' emotional and cognitive outputs with human experiences and perceptions, in addition to the traditional face validation methods.
Following the principles of face validation, researchers should focus their attention on the design of new advanced approaches for validation of qualitative outcomes of GABMs. Ex-

isting approaches can be leveraged as the basis for the development of this field. For instance survey-based validation collects subjective reports from participants regarding their emotional responses to various urban environments. It allows for a direct comparison between the agents' simulated emotions and human-reported emotions. Another used validation approaches involves observational studies of human interactions in different settings and comparing these observations with the agents' behavior in the simulation to ensure that agents' actions are consistent with human behaviors. Also expert reviews can be used to engage urban planners and social scientists to review the agents' behavior and emotional responses, providing feedback on their plausibility and accuracy.
To validate GABMs, immersive simulations offer a promising strategy. Virtual or augmented reality environments incorporating LLM-generated emotion representations can be used to capture participants' physiological responses and subjective reports [156, 159]. Combining crowdsourcing approaches [160] with continuous emotion tracking techniques [161] could help assess the accuracy of LLM-generated emotional representations within simulated urban contexts.

To conclude, we advocate that, when it comes to GABMs, it is crucial to consider validation methods from various disciplinary perspectives, which may be independent from the specific application field of the simulation, but which can depend on the human-like behavior we aim to reproduce. Qualitative approaches for interpretation and validation, either in place of or alongside quantitative ones, are essential since not everything can be quantified.

3.6 Conclusion

The integration of GAs into ABM has ushered in a transformative era for simulating complex systems, particularly within the social sciences. This paper has explored the significant advancements, methodologies, and emerging challenges associated with GABMs. Traditional ABMs, known for their ability to simulate emergent behaviors through simplistic models, are being revolutionized by GAs that leverage the sophisticated capabilities of LLMs. These GAs not only enhance the realism of individual agent behaviors but also introduce a new dimension of qualitative interpretation, such as simulating human-like emotions and social conversational interactions.
The transition from traditional ABMs to GABMs presents unique challenges, particularly in the realm of validation. While foundational validation methods like empirical validation, data analytics, and sampling remain relevant, the introduction of GAs necessitates the development of additional validation frameworks. These frameworks must account for the believability and coherence of agent behaviors, as well as the qualitative aspects of simulations, such as emotional responses.
Face validation, both at the micro and macro levels, proves essential for ensuring the plausibility of GABMs, especially when empirical data is scarce or unavailable. The use of LLMs to interview GAs offers a promising automated approach to validation, though it must be complemented by human evaluators to ensure accuracy and believability. The study of conversational agents, in particular, highlights the need for scalable and formalized validation methods to handle large-scale simulations involving numerous agents.
This paper advocates for a systematic approach to the development and validation of GABMs, emphasizing the importance of interdisciplinary collaboration. By combining insights from

psychology, sociology, and computational sciences, researchers can create more robust and realistic simulations. Future research should focus on formalizing validation methods following along with the practice of addressing the limitations of LLMs.

In conclusion, the advent of GAs represents a significant leap forward in the field of ABM, offering opportunities for modeling complex social systems. As the research community continues to refine these technologies and their validation methods, GABMs are poised to become indispensable tools for understanding and addressing real-world challenges in urban planning, social dynamics, and beyond.

Chapter 4

On the Effectiveness of Compact Strategies for Opinion Diffusion in Social Environments

An opinion diffusion scenario is considered where two marketers compete to diffuse their own opinions over a social network. In particular, they implement *social proof* marketing approaches that naturally give rise to a strategic setting, where it is crucial to find the appropriate order for targeting the individuals to which provide the incentives to adopt their opinions. The setting is extensively studied from the theoretical and empirical viewpoint, by considering strategies defined in a *compact way*, such as those that can be defined by selecting the individuals according to their degree of centrality in the underlying network. In addition to depicting a clear picture of the complexity issues arising in the setting, several compact strategies are empirically compared on real-world social networks. Results suggest that the effectiveness of compact strategies is moderately influenced by the characteristic of the network, with some centrality measures naturally emerging as good candidates to define heuristic approaches for marketing campaigns.

4.1 Introduction

The opinions that individuals populating a social environment form and express are significantly impacted by the social pressure of the opinions manifested by their friend/neighbors. Such pressure leads them to exhibit a kind of conformist behaviour, resulting in an *opinion diffusion* process over the underlying network [162, 163]. In fact, the dynamics of opinion diffusion is a central topic of research in areas such as social psychology and political sciences, but it has been recently attracting much attention in the artificial community too (see, e.g., [164, 165, 166, 167, 168, 169, 170] and the references therein).

By abstracting from their specific technical differences, diffusion models can be classified in two main groups [171, 172], namely *progressive* and *non-progressive* ones. In a non-progressive model, an individual that has adopted and manifested an opinion can well change her mind and adopt a different opinion later [173]. Instead, progressive models assume that once an individual adopts an opinion, she remains with that opinion forever. This perspective is appropriate in contexts such as viral marketing [174, 175] or to predict the adoption of new technologies or trends, where the crucial problem is influence maximization [176, 177, 178,

179] via *target set selection*, that is, to identify a small number of individuals that can be profitably used as seeds for a marketing campaign.

In the paper, we precisely consider a progressive scenario in a context where two opinions, say b (black) and w (white), compete for diffusing over the social environment [180, 181, 176, 182, 183]. However, we depart from classical studies related to target set selection, by assuming that the seeds are given and they are not under the control of the marketers, who can instead provide incentives to the individuals to change their opinions in some desired order. In particular, this can practically be done by exhibiting a *social proof* of the opinion, that is, a list of "friends" or influential individuals that have already adopted it. In fact, this setting has been considered in some earlier works in the literature too [184, 185], where it is shown that the specific order used to pick individuals for changing their mind can dramatically affect the number of individuals that eventually hold some desired opinion. However, the questions of how to define an optimal *strategy* for a marketer and of how the strategies of the marketers interplay have been not explored so far, neither from the theoretical viewpoint nor empirically by analyzing the dynamics of some real-world networks.

Our work embarks in a systematic study of the above questions within a setting where marketing strategies are defined in a *compact way*. Indeed, in a general setting, a diffusion strategy might well depend on the history of the evolution of the network as well as by the specific configuration given to hand. However, modeling and reasoning with such arbitrary strategies would require extremely demanding computational resources and an assumption of complete knowledge, which is unrealistic in real-world scenarios. In fact, in order to identify the individuals to target for the propagation of the opinion, marketers are often guided by some heuristic parameters aimed at estimating the social "power" of the individuals in the network. A noticeable example is when such power is modeled in terms of some well-known *centrality measures* [186], and where a strategy might be specified by just picking the individual with the highest rank (according to the desired measure) over all possible individuals that can potentially change their mind with a social proof marketing strategy.

In more details, we provide the following contribution:

- We define a strategic setting for reasoning about progressive dynamics determined by compact strategies. Our modeling takes care of the *speed* of the propagation, which reflects the efforts spent by the marketer to spread her opinion, and of the interplay between the strategies of the competing opinions. Moreover, to define the social pressure of the individuals, we assume a deterministic *linear threshold* [171] setting, that is, an individual can adopt an opinion only if (at least) a given fraction of her neighbors did.

- We formalize some relevant problems arising in our strategic setting and we study their computational complexity. The study is conducted for strategies defined in a compact way, but not necessarily restricted to those induced by centrality measures.

- And, finally, we focus on some well-known centrality measures and we performed an extensive experimental evaluation aimed at assessing the quality of the strategies they naturally induce. Our campaign considers real social networks, and diffusion processes with different characteristics determined by the initial seeds and the speed of propagation.

The rest of the paper is organized as follows. The model for opinion diffusion is presented in Section 4.2. Our formal and empirical studies are reported in Section 4.3 and Section 4.4,

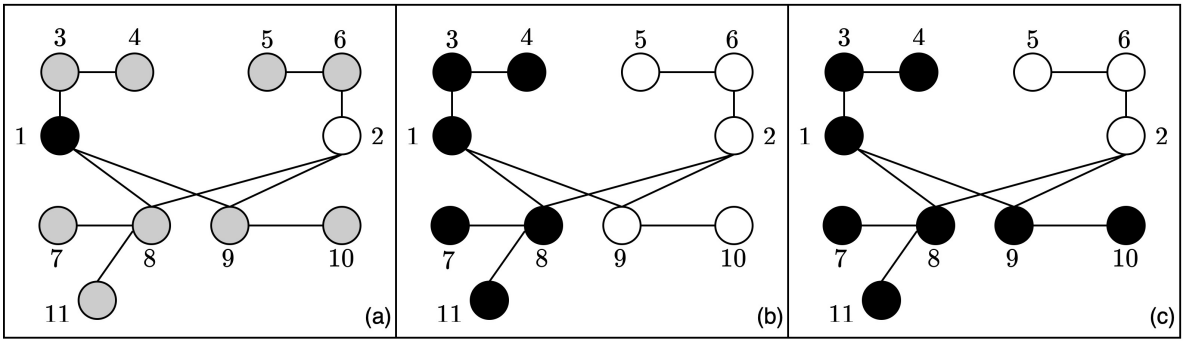


Figure 4.1: Illustrations for examples in Section 4.2.

respectively. Some final remarks on our findings as well as some directions for further works are eventually discussed in Section 4.5.

4.2 Compact Strategies for Opinion Diffusion

Social networks and dynamics. A social network is modeled as an undirected graph $G = (N, E)$ over a set N of individuals/nodes. Two competing opinions, denoted as b (*black*) and w (*white*), are spreading over the network, by starting from some initial seeds. A special opinion g (*gray*) is associated with any individual $v \in N$ that has not already adopted an opinion in $\{\mathsf{w}, \mathsf{b}\}$; in this case, v is influenced by her *neighbors* in G, i.e., by the individuals in the set $\delta(v) = \{x \mid \{v, x\} \in E\}$, under a *linear threshold model* [171] for some fixed threshold $0 \leq t \leq 1$.

Formally, a *configuration* for G is defined as a pair $S = (S_{\mathsf{b}}, S_{\mathsf{w}})$ such that $S_{\mathsf{b}}, S_{\mathsf{w}} \subseteq N$ is the set of all individuals that hold opinion b and w, respectively. For each individual $v \in N \setminus (S_{\mathsf{b}} \cup S_{\mathsf{w}})$ that has not already adopted an opinion, let $\sigma(v) = \lceil t \cdot |\delta(v)| \rceil$. Then, we say that v is *stable* with respect to the configuration $(S_{\mathsf{b}}, S_{\mathsf{w}})$ if $|\delta(v) \cap S_{\mathsf{b}}| < \sigma(v)$ and $|\delta(v) \cap S_{\mathsf{w}}| < \sigma(v)$. The configuration S is stable if all individuals in $N \setminus (S_{\mathsf{b}} \cup S_{\mathsf{w}})$ are stable. A *dynamic* for G is a sequence of configurations $\pi = (S^0, ..., S^k)$ such that S^k is stable and, for each $i \in \{1, ..., k\}$, S^i is obtained from S^{i-1} by picking an individual in $N \setminus (S^{i-1}_{\mathsf{b}} \cup S^{i-1}_{\mathsf{w}})$ that is not stable in S^{i-1} and by setting her opinion to b or w. Note that we are considering *progressive* dynamics; therefore, for each *initial configuration* S^0, $k \leq |N \setminus (S^0_{\mathsf{b}} \cup S^0_{\mathsf{w}})|$ always holds.

Example 1. Consider the network in Figure 4.1(a), the configuration $(\{1\}, \{2\})$ and the threshold $t = 0$ (meaning that one neighbor that is not g is enough to make the node not stable). Then, the configuration is not stable, as individuals in $\{3, 8, 9\}$ can change their opinion to b, and individuals in $\{6, 8, 9\}$ can change to w. $\lhd$

Compact Startegies. We consider a strategic setting for opinion diffusion, where dynamics originate from the interactions of two players, say P_{b} and P_{w}, competing to maximize the spread of b and w, respectively. Formally, a strategy for P_{b} (resp., P_{w}) is a function τ_{b} (resp., τ_{w}) associating with any configuration S over the network G a node $\tau_{\mathsf{b}}(S) \in N$ (resp., $\tau_{\mathsf{w}}(S) \in N$) that is not stable and that can change her opinion to b (resp., w). In particular, note that we are considering strategies that do not depend on the *history* of the evolution of the network, which is a rather natural assumption in all those settings where strategies

are a-priori defined in terms of structural/topological properties of the social network. For instance, a strategy of interest to our analysis can be the one of selecting the individual that can change the opinion and that have the maximum possible degree. Such strategies will be hereinafter called *compact*. In formal terms, a strategy τ_{b} (resp., τ_{w}) is compact if it is given as a polynomial-time computable function defined over some internal encoding, say $\epsilon(\tau_{\mathsf{b}})$ (resp. $\epsilon(\tau_{\mathsf{w}})$), whose size is polynomially bounded in the size of G. In fact, if a strategy is not compact, than its encoding would naturally require to list all possible network configurations with their associated outcomes, hence requiring exponential space (rather than polynomial). We refer the reader to Section 4.4 for further relevant compact strategies that we consider in our experimentation.

Speed of Diffusion. We assume that players act in turns. Moreover, as a way to formalize the efforts spent in spreading their opinions, we define the *speed* of diffusion as a pair $\rho = (\rho_{\mathsf{b}}, \rho_{\mathsf{w}})$ of natural numbers characterizing, at each turn, the number of individuals selected by P_{b} or by P_{w}, respectively, to change their mind.

In fact, given an initial configuration S^0 and the speed ρ, the strategies τ_{b} and τ_{w} univocally determine a dynamic $S^0, ..., S^k$ for G, which we hereinafter denote as $\pi[S^0, \rho, \tau_{\mathsf{b}}, \tau_{\mathsf{w}}]$ and which is defined as follows. W.l.o.g., the first turn of player P_{b} starts in S^0. When the turn of P_{b} (resp., P_{w}) starts in some configuration S^i, then the dynamic evolves by iteratively changing the mind to m individuals according to τ_{b} (resp., τ_{w}), such that either $m = \rho_{\mathsf{b}}$ (resp., $m = \rho_{\mathsf{w}}$) or S^{i+m} contains no individual that can change her opinion to b (resp., w); eventually, the turn of the other player starts in S^{i+m+1}.

Example 2. Consider a strategy D_{b} (resp., D_{w}) for P_{b} (resp., P_{w}) that selects, for each configuration S, the node that is not stable in S and can change her opinion to b (resp., w) having the maximum degree. By starting from the configuration in Figure 4.1(a), and by considering the speed $(1, 1)$, the network evolves as follows: $8 \mapsto \mathsf{b}; 9 \mapsto \mathsf{w}; 3 \mapsto \mathsf{b}; 6 \mapsto \mathsf{w}; 4 \mapsto \mathsf{b}; 5 \mapsto \mathsf{w}; 7 \mapsto \mathsf{b}; 10 \mapsto \mathsf{w}; 11 \mapsto \mathsf{b}$. Eventually, the dynamic induced by D_{b} and D_{w}, say $\dot{\pi}$, will lead to the stable configuration reported in Figure 4.1(b). $\lhd$

Coverage. In the following, we shall study opinion diffusion from the perspective of maximizing the spread of the opinions b and w. Hence, in order to finalize the formalization of the framework, it is natural to define the *coverage* of b (resp, w) over G of the given dynamic $\pi = S^0, ..., S^k$ as the number $\gamma_{\mathsf{b}}(\pi)$ (resp., $\gamma_{\mathsf{w}}(\pi)$) of individuals holding opinion b (resp., opinion w) at the end of π, that is $\gamma_{\mathsf{b}}(\pi) = |S^k_{\mathsf{b}}|$ (resp., $\gamma_{\mathsf{w}}(\pi) = |S^k_{\mathsf{w}}|$).

Example 3. The coverage of b (resp., w) in the dynamic $\dot{\pi}$ of Example 2 is $\gamma_{\mathsf{b}}(\dot{\pi}) = 6$ (resp., $\gamma_{\mathsf{w}}(\dot{\pi}) = 5$). $\lhd$

4.3 Reasoning about Opinion Diffusion

Now that we have defined a formal framework for reasoning about opinion diffusion under compact strategies, we can turn to study some relevant computational problems arising therein. In particular, we next embark on the definition and study of the opinion maximization problem by considering two kinds of setting determined by the strategic interplay emerging between players P_{b} and P_{w}.

4.3.1 Brave and Cautious Reasoning

Let S^0 be an initial configuration and ρ be the speed. Recall that we are considering strategies τ_{b} and τ_{w} that are functions of the configuration S at hand only. We take the perspective of player P_{b} and we assume that τ_{w} is private to w. In particular, P_{b} is in charge of determining her best possible strategy and, given the uncertainty about τ_w, two approaches can be considered.

- On the one hand, player P_{b} might take an optimistic perspective, according to which her coverage is defined as the maximum possible coverage over all the possible strategies of τ_{w}. Accordingly, we say that the strategy $\tau_{\mathsf{b}}^{\top}$ is *brave*-optimal for P_{b} if there exists a strategy $\tau_{\mathsf{w}}^{\top}$ for P_{w} such that:

$$(\tau_{\mathsf{b}}^{\top}, \tau_{\mathsf{w}}^{\top}) = \arg\max_{\tau_{\mathsf{b}}, \tau_w} \gamma_{\mathsf{b}}(\pi[S^0, \rho, \tau_{\mathsf{b}}, \tau_{\mathsf{w}}]).$$

- On the other hand, player P_{b} might take a pessimistic viewpoint, in that she assumes that P_{w} always plays the strategy that maximally reduce her coverage. Accordingly, we say that the strategy $\tau_{\mathsf{b}}^{\perp}$ is *cautious*-optimal for P_{b} if:

$$\tau_{\mathsf{b}}^{\perp} = \arg\max_{\tau_b} \left(\min_{\tau_w} \gamma_{\mathsf{b}}(\pi[S^0, \rho, \tau_{\mathsf{b}}, \tau_{\mathsf{w}}]) \right).$$

Example 4. Consider the network and the initial configuration reported in Figure 4.1(a). Assume that P_{b} adopts the strategy D_{b} of selecting, for each configuration S, the individual having the maximum degree that is not stable in S and can change her opinion to b. Then, according to a cautious perspective, the maximum coverage that can be obtained is the one in Figure 4.1(b) and discussed in Example 2.

Consider now the brave perspective. In this case, the maximum coverage for P_{b} is associated with the strategy M_{w} for P_{w} that selects, for each configuration S, the node that is not stable in S and can change her opinion to w, having the minimum degree. By starting from the configuration in Figure 4.1(a), and by considering the speed $(1, 1)$, the network evolves as follows: $8 \mapsto \mathsf{b}; 6 \mapsto \mathsf{w}; 9 \mapsto \mathsf{b}; 5 \mapsto \mathsf{w}; 3 \mapsto \mathsf{b}; 4 \mapsto \mathsf{b}; 7 \mapsto \mathsf{b}; 10 \mapsto \mathsf{b}; 11 \mapsto \mathsf{b}$. Eventually, the dynamic $\overline{\pi}$ induced by D_{b} and M_{w}, will lead to the stable configuration reported in Figure 4.1(c) where the coverage of b is $\gamma_{\mathsf{b}}(\overline{\pi}) = 8$. $\lhd$

Armed with the above notions, we can naturally define the following two (Opinion Maximization) problems, receiving as input G, the initial configuration S^0, the speed ρ, and a real number $\alpha \in [0, 1]$:

BRAVE-OM: Is $\gamma^b(\pi[S^0, \rho, \tau_{\mathsf{b}}^{\perp}, \tau_{\mathsf{w}}^{\perp}]) \geq \alpha \times |N|$?

CAUTIOUS-OM: Is $\gamma^b(\pi[S^0, \rho, \tau_{\mathsf{b}}^{\perp}, \tau_{\mathsf{w}}]) \geq \alpha \times |N|$, for each possible strategy τ_{w} for P_{w}?

The complexity of these two problems will be next analyzed.

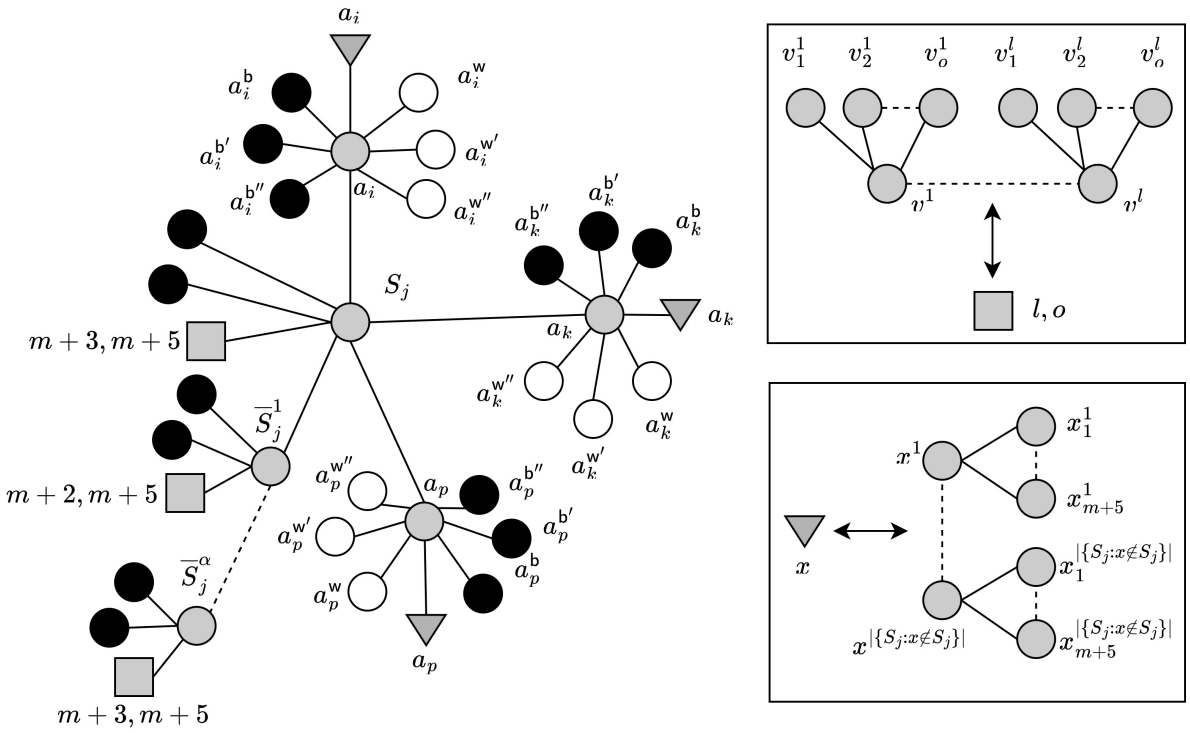


Figure 4.2: Illustration of the reduction in the proof of Theorem 5.

4.3.2 Complexity Analysis

Given that we are considering compact strategies, it is immediate to check that problem BRAVE-OM belongs to the class NP of all problems that can be solved in polynomial time by a non-deterministic Turing machine. Indeed, we can just guess the strategy $\tau_{\mathsf{b}}^{\perp}$ (whose encoding $\epsilon(\tau_{\mathsf{b}}^{\perp})$ requires polynomially-many bits) and then check in polynomial time whether $\gamma^b(\pi[S^0, \rho, \tau_{\mathsf{b}}^{\perp}, \tau_{\mathsf{w}}^{\perp}]) \geq \alpha \times |N|$ actually holds. Things are more complex with CAUTIOUS-OM. Indeed, in this case, we can still guess in polynomial time $\tau_{\mathsf{b}}^{\top}$ over a non-deterministic Turing machine; but, now the problem of checking whether $\gamma^b(\pi[S^0, \rho, \tau_{\mathsf{b}}^{\perp}, \tau_{\mathsf{w}}]) \geq \alpha \times |N|$ holds for each τ_w requires solving another problem in NP (which amounts at checking whether there exists some τ_{w}^* such that $\gamma^b(\pi[S^0, \rho, \tau_{\mathsf{b}}^{\perp}, \tau_{\mathsf{w}}^*]) < \alpha \times |N|$). Hence, CAUTIOUS-OM belongs to the complexity class Σ_2^{p} [187].

We next complete the picture by showing the above results are tight. In fact, we start by showing that BRAVE-OM is NP-hard, by exibithing a reduction to the 3Hitting Set problem [188] (shortly 3HS), that, given a collection $C = \{S_1, \ldots, S_m\}$ of subsets of size three of a finite set $S = \{a_1, ..., a_n\}$ and an integer k, is the problem of checking whether there exists a subset $S' \subseteq S$ such that $|S'| \leq k$ and S' contains at least one element from each subset in C.

Theorem 5. BRAVE-OM *is* NP*-complete.*

Proof (Sketch). Let $C = \{S_1, ..., S_m\}$ be a collection of subsets of size three of a finite set $S = \{a_1, ..., a_n\}$ and let k be a positive integer. Consider the network $G = (N, E)$, depicted in Figure 4.2, where we have one node a_i for each element $a_i \in S$ and a node S_j for each subset $S_j \in C$ – in the figure nodes a_i, a_k, a_p represent the elements belonging to the set S_j.

Note that the network is built such that each node has exactly 1 or $m + 6$ neighbors, and from each S_j node there is a chain of g nodes $\overline{S}_j^1, ..., \overline{S}_j^\alpha$ of length α, where α is chosen to be far greater than $\max\{m, n\}$. Consider a threshold $t = 3/(m + 6)$, then, according to such a threshold, only a_i, S_j and $\overline{S}_j^t$ nodes can change their opinions, since the gadgets reported in the right part of Figure 4.2 prevent other nodes of being able to change their opinion. We now claim that the (C, S, k) is a *yes* instance of 3HS if, and only if, BRAVE-OM returns *yes* on G, with the initial configuration reported in Figure 4.2, where all nodes are $3/(m + 6)$-individuals, and by considering a speed of diffusion $(k, n - k)$ and a final coverage threshold of $3 * m * \alpha/|N|$.

(if part) Let S' be a set witnessing that (C, S, k) is a *yes* instance to 3HS. A strategy for b is to diffuse to the nodes $a_i \in S'$ in the first k steps (if $|S'| < k$, the remaining $k - |S'|$ nodes can be chosen randomly among the remaining a_i). Then, the only possibility for w in the subsequent $n - k$ steps is to diffuse in the a_i that are still g. Then, w cannot diffuse anymore. In fact, note that all three a_i, a_k, a_p must be w to enable S_j to switch to w, while just one of them is required to be b for S_j being able to switch to b. Since S' is a solution to 3HS, it means that at least one element, say a_i, for each $S_j \in C$ is in S' and, thus, the corresponding node a_i has switched to b in the first k steps of the dynamic. Thus, in the subsequent steps all nodes S_j will switch to b thus enabling the m $\overline{S}_j$ chains of length α to also switch to b. This conclude the proof since at the end of the dynamic the number of nodes holding opinion b is greater than $3 * m * \alpha$.

(only-if part) Let $(S_k^{\mathsf{b}}, S_k^{\mathsf{w}})$ be the final configuration of a dynamic witnessing that BRAVE-OM returns *yes*. Note that, to be able to reach the coverage threshold, all S_j nodes had to become b at some point so enabling the m $\overline{S}_j$ chains of length α to switch to b too. Thus, it means that for each node S_j there is at least one a_i neighbor holding opinion b. The set $S' = \{a_i \mid a_i \in S_k^{\mathsf{b}}\}$ covers all the subsets $S_j \in C$. To conclude, note that according to the speed of diffusion the nodes a_i that have opinion b are at most k (and these are the nodes that changed their opinion in the first k steps), since opinion w can only cover the other a_i nodes, that are (at least) $n - k$, since they are the only nodes that are enabled to switch to w in every dynamic witnessing a *yes* instance of BRAVE-OM. □

We next complete the picture by showing that CAUTIOUS-OM is Σ_2^{p}-hard, by exhibiting a (rather elaborated) reduction to the problem of deciding the validity of a Quantified Boolean Formula (QBF) having the form $\exists x \forall y.\varphi$ and where φ is given in disjunctive normal form [187]. Practically, this means that—unlike BRAVE-OM, that is a "classical" NP-complete problem–we cannot design a *flat-backtracking* algorithm for CAUTIOUS-OM, i.e., where the search-space is a tree having a polynomial number of levels (and such that moving along the tree edges does not take exponential time).

Theorem 6. CAUTIOUS-OM *is* Σ_2^{p}*-complete.*

Proof (Sketch). Let $\exists x_1, ..., x_n \forall y_1, ..., y_n \varphi$ be a 2QBF formula in disjunctive normal form with m clauses[1]. Consider the network $G = (N, E)$, depicted in Figure 4.3, where we have two nodes v_i^T and v_i^F for each variable v_i, that are meant to encode the truth assignment for variable v_i. Moreover, there is a node d_j for each disjunct (in the figure nodes l_j^1, l_j^2, l_j^3 represent the nodes associated to the variables directed or negated that appear in the disjunct

[1]Note that, w.l.o.g., we consider a formula with $2n$ variables, n quantified existentially and n quantified universally.

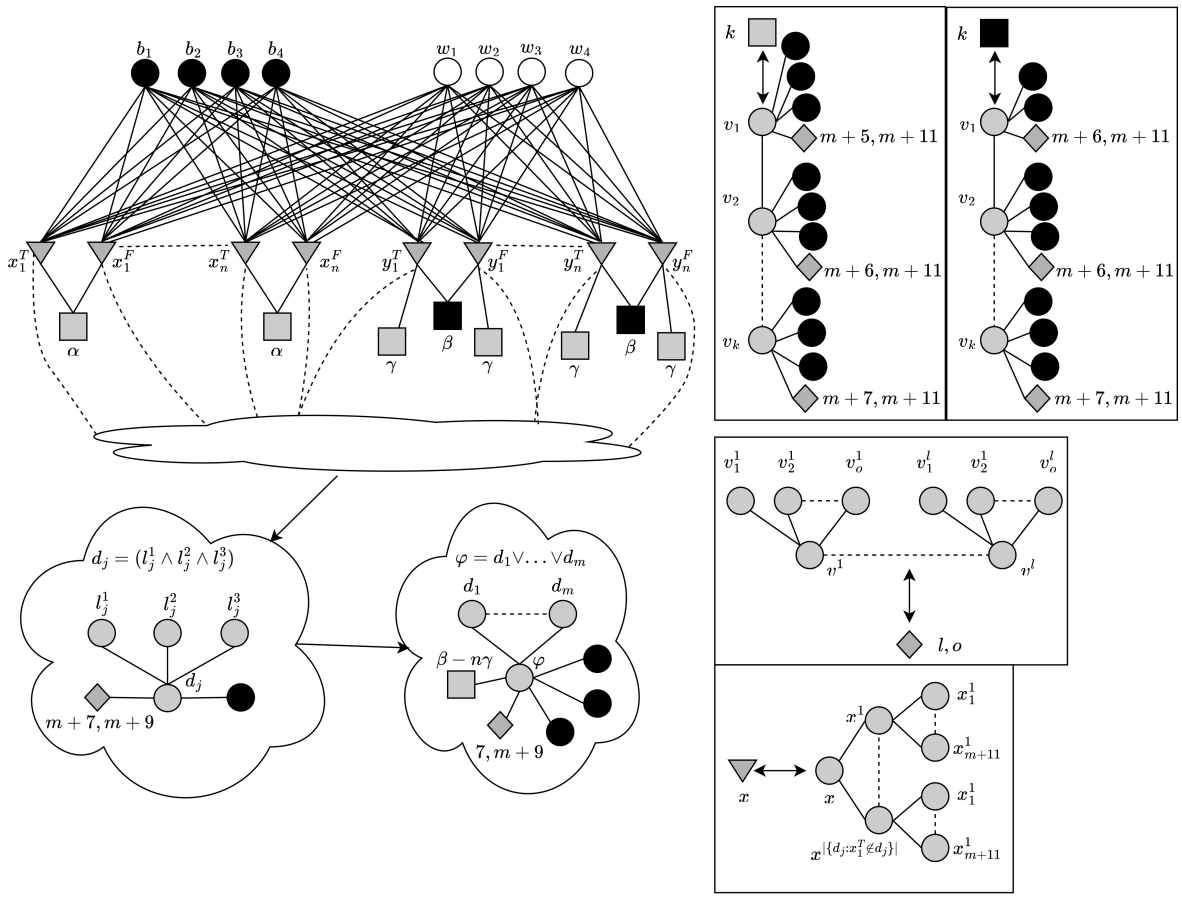


Figure 4.3: Illustration of the reduction in the proof of Theorem 6.

d_j) and a node φ representing the formula. Note that the network is built such that each node has exactly 1 or $m + 11$ neighbors (see the gadgets reported in the right part of the figure), and from each x_i^T and x_i^F node there is a chain of g nodes of length α, while from each y_i^T and y_i^F node there are a chain of g nodes of length γ and a chain of g nodes of length β with $\alpha \gg \beta \gg \gamma \gg \max(n, m)$. Furthermore, there is also a chain of g nodes of length $\beta - n\gamma$ starting from the node φ. If we consider a threshold $t = 4/(m + 11)$, only the nodes x_i^T, x_i^F, y_i^T, y_i^F, d_j and φ (as well as the α, β and γ chains) can change their opinions, since the gadgets reported in the right part of Figure 4.3 prevent other nodes of being able to change their opinion.

We now claim that $\exists x_1, ..., x_n \forall y_1, ..., y_n \varphi$ is valid if, and only if, the answer to CAUTIOUS-OM is *yes* on G, with the initial configuration reported in Figure 4.3, where all nodes are $4/(m + 11)$-individuals, and by considering a speed of diffusion (n, n) and a final coverage threshold of $(n * \alpha + \beta)/|N|$.

(if part) Let $\mathbb{X}$ be a satisfying assignment for the existentially quantified variables witnessing the validity of $\exists x_1, ..., x_n \forall y_1, ..., y_n \varphi$. A strategy for b is to diffuse to nodes x_i^T (resp., x_i^F) for each x_i that evaluates true (resp., false) in $\mathbb{X}$ in the first n steps. Then, w can diffuse to the x nodes that are still g in the subsequent n steps. From this point, we can consider whatever truth assignment for variables y, thus we can assume that b will diffuse to $y_1^T, ..., y_n^T$ and w to $y_1^F, ..., y_n^F$. From this configuration only b is enabled to diffuse to the n chains of length α connected to the x nodes and to the n chains of length γ connected to the y_i^T nodes.

Moreover, since $\mathbb{X}$ is witnessing the validity of the formula, it means that there is at least one disjunct, say d_j, that evaluates true in $\mathbb{X}$. This means that the three nodes l_j^1, l_j^2, l_j^3 associated to the literals that appear in d_j hold opinion b and enable node d_j to adopt opinion b too. To conclude, note that after d_j adopts opinion b also node φ becomes not stable and can adopt opinion b, thus enabling the last chain of length $\beta - n\gamma$ to change its opinion to b.

(only-if part) Let $(S_{\mathsf{b}}^k, S_{\mathsf{w}}^k)$ be the final configuration of a dynamic π witnessing that the answer to CAUTIOUS-OM is *yes*, and consider the truth assignment $\mathbb{X}$ such that x_i evaluates true (resp., false) in $\mathbb{X}$ if x_i^T (resp., x_i^F) becomes b in the first n steps of π. By definition of π, we have that $|S_{\mathsf{b}}^k| \geq n * \alpha + \beta$. Note that, to meet such a requirement it is mandatory that all α chains connected to the x nodes must be b in S_{b}^k and thus, for each x_i at least one among x_i^T and x_i^F must be b. Moreover, according to the CAUTIOUS-OM setting, the strategy selected from b must allow to obtain a valid solution for whatever strategy adopted by w, and thus b must diffuse in the first n steps to exactly one node between x_i^T and x_i^F for each x_i. In fact, suppose that b diffuse to both x_i^T and x_i^F for some x_i, it means that there exists an x_j for which both x_j^T and x_j^F are still g after the first n steps and to which w can diffuse by preventing b to subsequent diffuse in the corresponding α chain. Then, we will show that whatever strategy played by w is always a winning strategy for b.

If w in the subsequent n steps will leave free two nodes y_i^T and y_i^F for some y_i, then b can diffuse to both of them thus enabling the corresponding β chain to become b and meeting the coverage requirement. If, on the contrary, w diffuses to either y_i^T or y_i^F for each y_i (thus, for each valid truth assignment), then b can diffus e to the y nodes that are still g in the subsequent n steps, enabling n chains of length γ to adopt opinion b too[2]. Then, to meet coverage requirement there are still $(\beta - n\gamma)$ nodes missing that can obtained only via the $(\beta - n\gamma)$ chain connected to the φ node. To enable such a chain to switch to b, node φ must switch to b too, meaning that at least one d_j is enabled to switch to b because its three literals nodes l_j^1, l_j^2, l_j^3 became b in the previous steps of the dynamics. □

4.4 Compact Strategies via Centrality Measures

In this section, we turn to study opinion diffusion from an empirical viewpoint by focusing on an important class of compact strategies, namely those that are naturally identified by the ranking induced by a centrality measure [186]. In fact, each measure naturally induces a strategy where the next individual to be picked is the one with the highest rank over the individuals that can adopt the given opinion.

In particular, in our analysis, we shall consider the following centrality measures, which—without ambiguity—will be hereinafter transparently referred to as strategies:

- *degree centrality* (`deg`), which counts the number of connections of an individual;
- *betweenness centrality* (`bet`), which measures the number of shortest paths that pass through a particular individual;
- *closeness centrality* (`cls`), which looks at the average distance from a particular individual to all others in the network;

[2]Note that, if b decides to diffuse in some x_i^T/x_i^F still g then w can occupy an y_p^T/y_p^F left g thus preventing b to obtain a γ chain and, thus, it is not a valid strategy for b.

Network	$\|N\|$	$\|E\|$	r_{kk}	$\langle k \rangle$	k^*	λ
dblp	317080	1049866	0.27	6.6	343	1.5
fb	134873	1380293	0.07	20.5	1469	1.3
deezer	143884	846915	0.33	11.8	420	1.3
fb-Art	50521	819306	-0.02	32.4	1469	1.2
fb-Ath	13868	86858	-0.03	12.5	468	1.3
fb-Com	14120	52310	0.01	7.4	215	1.4
fb-Gov	7058	89455	0.03	25.3	697	1.2
fb-NS	27930	206259	0.02	14.8	678	1.3
fb-Pol	5908	41729	0.02	14.1	323	1.3
fb-PF	11573	67114	0.20	11.6	326	1.4
fb-TvS	3895	17262	0.56	8.9	126	1.3
dz-HR	54573	498202	0.20	18.3	420	1.2
dz-HU	47538	222887	0.21	9.4	112	1.2
dz-RO	41773	125826	0.11	6.0	112	1.4

Figure 4.4: Dataset characteristics for the experiments in Section 4.4.

- *eigenvector centrality* (`eig`), which considers both the number of connections that an individual has, as well as the centrality of the individuals that it is connected to;
- *vote rank* (`vr`), which is based on the concept of voting, where each individual in the network has the ability to cast a vote for other individuals; the more votes a node receives, the higher its vote rank centrality measure.

The quality of these measures/startegies (from the perspective of opinion diffusion) is next assessed by considering a thorough experimentation conducted on several real-world social environments.

4.4.1 Experimental Setting

Dataset. To test the effectiveness of different compact strategies on opinion diffusion we used several real networks. We considered a benchmark consisting of 13 graph datasets, whose main features are summarized in Figure 4.4. In particular, for each dataset, we report the number of nodes $|N|$, the number of edges $|E|$, the assortativity r_{kk}, the average node degree $\langle k \rangle$, the maximum node degree k^* and the coefficient of an (approximate) underlying power law distribution λ. In more detauils, assortativity (or degree correlation) r_{kk} is the Pearson correlation between the degrees of connected nodes. In assortative networks ($r_{kk} > 0$) nodes are connected to nodes having similar degree, while in disassortative networks ($r_{kk} < 0$) they link to nodes having dissimilar degree. Note that two networks having the same degree distribution can differ for their assortativity. Moreover, the coefficient of the power law distribution (i.e., λ) that better approximate the degree distribution of the network has been computed according to the Bhattacharyya distance [189].

The datasets fb-Art, fb-Ath, fb-Com, fb-Gov, fb-NS, fb-Pol, fb-PF, fb-TvS have been extracted from the Facebook (fb) [190] dataset by considering artists, athletes, companies, governments, new sites, politicians, public figures and TV shows pages only, respectively. The datasets dz-HR, dz-HU and dz-RO have been extracted from the Deezer dataset [190] by

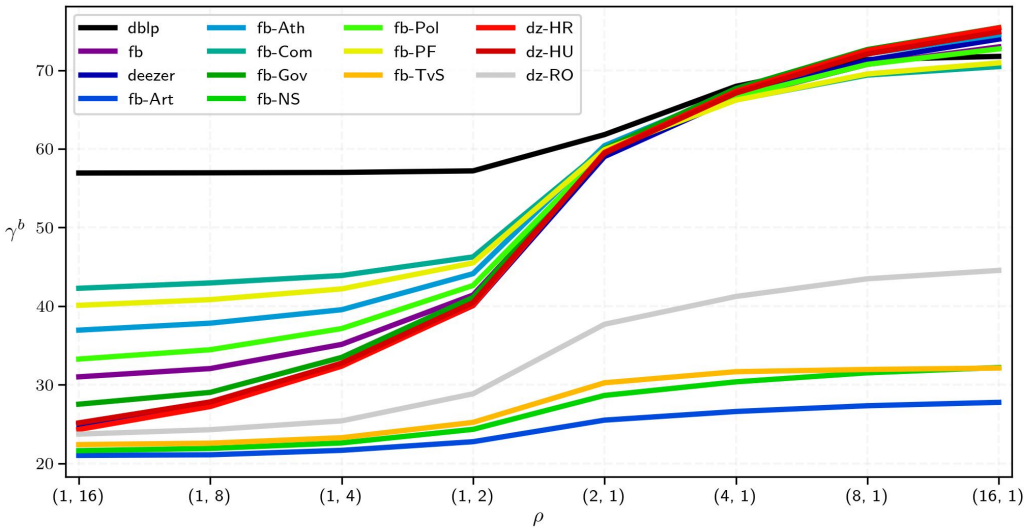


Figure 4.5: Final percentage of individuals holding opinion b according to different speed of diffusion $(\rho_{\mathsf{b}},\rho_{\mathsf{w}})$, when the strategy for both opinions is `deg`, the threshold is 0.1 and initial seed ratios for both opinions is 0.2.

considering the friendships networks of users in Croatia, Hungary and Romania, respectively. The largest dataset is dblp [191] having more than 300K nodes and 1M edges.

Experimental Setup. Experiments have been conducted as follows. For each pair of strategies $\tau_{\mathsf{b}}, \tau_{\mathsf{w}} \in \{\texttt{deg}, \texttt{bet}, \texttt{cls}, \texttt{eig}, \texttt{vr}\}$, we varied the number of individuals having opinions b and w in the initial configuration. In particular, for each pair of real numbers $\eta_{\mathsf{b}}, \eta_{\mathsf{w}} \in \{0.1, 0.15, 0.20, 0.25\}$, the initial configuration $(S^0_{\mathsf{b}}, S^0_{\mathsf{w}})$ was determined as follows: S^0_{b} consists of the $\eta_{\mathsf{b}} * |N|$ nodes having the highest value according to τ_{b}, while S^0_{w} consists of the $\eta_{\mathsf{w}} * |N|$ nodes having the highest value according to τ_{w} in $N \setminus S^0_{\mathsf{b}}$ (individuals still holding opinion g after b initialization). Then, for each pair of strategies and each initial configuration, we considered different values for the threshold t determining when a node is not stable, by considering $t \in \{0.1, 0.3, 0.35, 0.4, 0.45, 0.5\}$. Finally, we also varied the relative speed of diffusion of b and w by considering:
$(\rho_{\mathsf{b}}, \rho_{\mathsf{w}}) \in \{(1, 16), (1, 8), (1, 4), (1, 2), (2, 1), (4, 1), (8, 1), (16, 1)\}$.

Overall, for each network in Figure 4.4, we considered 19.200 different experimental settings from which we simulated the diffusion dynamics according to the strategies and the speed given at hand.

Execution environment. All experiments were executed in an Anaconda3 virtual environment, where an opinion diffusion framework has been implemented in Python (v3.8.13) by taking advantage of the NetworkX (v2.8.6) library. All experiments ran on high-performance computing node with Intel(R) Xeon(R) Gold 5118 CPU (2.30GHz), 4x12 cores double thread (for a total of 96 threads) and 512GB of RAM. In order to take full advantage of the available resources, the experiments were fully parallelized such that 96 different settings for a graph ran simultaneously on separate CPU threads.

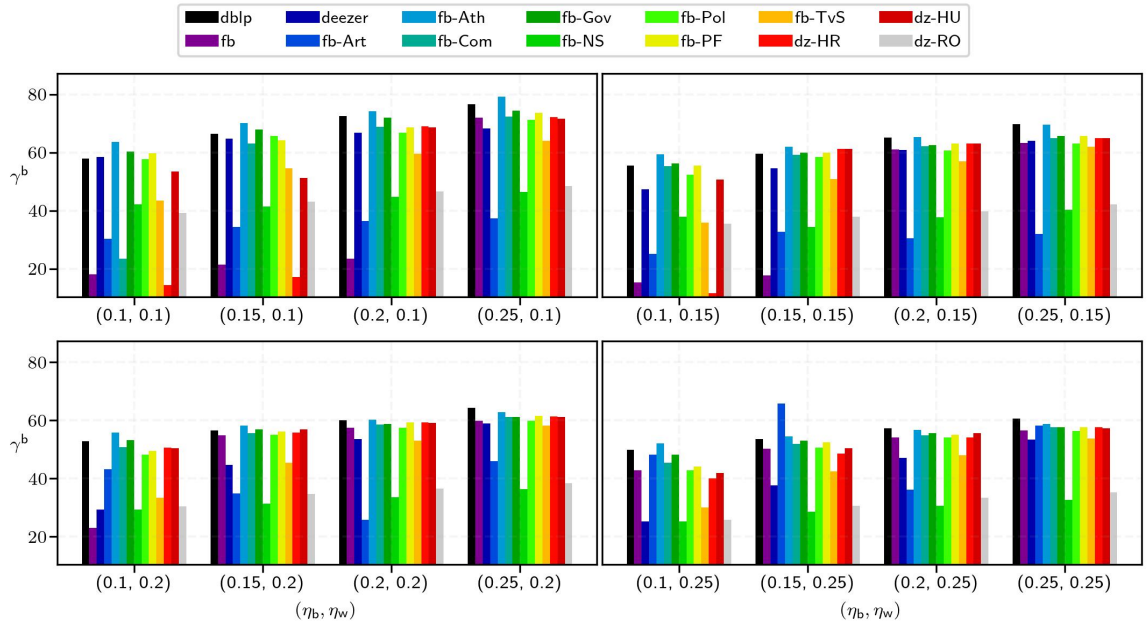


Figure 4.6: Final percentage of individuals holding opinion b according to different ratios (η_b,η_w) of nodes in the initial configuration, when the strategy for both opinions is `deg`, the threshold is 0.3, and the speed of diffusion is $(2, 1)$. Each subplot is obtained by fixing η_w, and varying η_b.

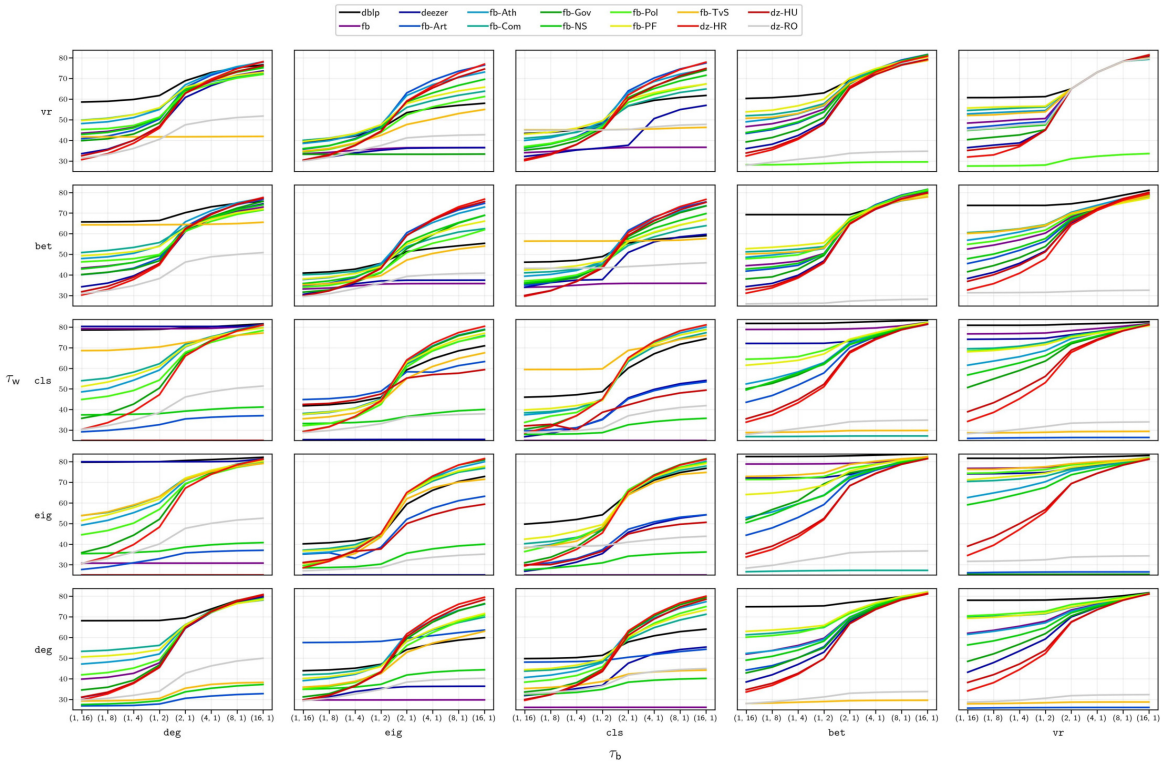


Figure 4.7: Final percentage of individuals holding opinion b for each pair of strategies for b and w (on outer x-axis and y-axis respectively), according to different speed (on the subplots x-axis), for $t = 0.1$, $\eta_b = 0.25$ and $\eta_w = 0.15$.

Figure 4.8: Total number of settings in which `vr` and `bet` resulted to be brave- or cautious-optimal.

4.4.2 Results

In order to shed lights on the behaviour of the various compact strategies, we start by discussing the results of our experimental campaign by looking at how the final coverage of opinion b is related to the different parameters that determine the experimental settings. Figure 4.5 reports the final coverage of b (in terms of percentage of nodes holding opinion b at the end of the diffusion process) for increasing values of the relative speed of diffusion for b; in particular, we depict the results for $\tau_b = \tau_w = \texttt{deg}$, $\eta_b = \eta_w = 0.2$, and $t = 0.1$. Our findings—which are representative of the behaviour manifested over all the other settings—evidence that higher speeds lead to an increase of the coverage. Moreover, note that a steeper increase in the coverage emerges as soon as ρ_b becomes greater than ρ_w. Figure 4.6 shows, instead, the coverage of b, when both opinions adopt the `deg` strategy and diffuse with a relative speed of $(2, 1)$, by considering the threshold 0.3, w.r.t. different initial configurations. In particular, each sub-chart considers η_b as a variable for a fixed η_w. As can it be noted, an increase in the number of initial seeds does not necessarily correspond to an increase of the final coverage. In fact, when η_w is 0.1 (upper-left chart) and 0.15 (upper-right chart) the final coverage of b in the fb-NS network decreases when η_b increases from 0.1 to 0.15. Similarly, also for the graph fb-Art for all η_w except 0.1, the final coverage of b decreases when η_b increases from 0.15 to 0.2.

A clear pictures of the effectiveness of the various strategies considered in the experimentation is then reported in Figure 4.7. In particular, we report the final coverage of b (again, as percentage of the whole network) for different strategies for b (outer x-axis) and w (outer y-axis), and speed of diffusion (x-axis of subplots)—results are referred to a threshold $t = 0.1$ and to the ratios of b and w in the initial configuration equals to 0.25 and 0.15, respectively. As it can be noted, the impact of the strategy τ_b on the final coverage heavily depends on the strategy τ_w. For example, on the Facebook network (fb), when considering $\tau_b = \texttt{deg}$ and the speed of diffusion $(16, 1)$, the final coverage of b approaches the 80% of nodes if $\tau_w \in \{\texttt{deg}, \texttt{cls}, \texttt{bet}, \texttt{vr}\}$ but is around the 30% of nodes when $\tau_w = \texttt{eig}$. A similar

behaviour can be noted on the deezer network when considering $\tau_{\mathsf{b}} = \texttt{eig}$. In fact, for the speed $(16, 1)$ the final coverage of b is around 0% of nodes if $\tau_{\mathsf{w}} \in \{\texttt{eig}, \texttt{cls}\}$ but reaches the $35\% - 40\%$ of nodes when $\tau_{\mathsf{w}} \in \{\texttt{deg}, \texttt{bet}, \texttt{vr}\}$. Finally, by looking at Figure 4.7, we note that the efficacy of some strategies depends on the characteristics of the network. For example, deg and bet are much less effective on dz-RO than on dblp, while cls on dz-RO is more effective than on dblp in several settings.

Finally, to sum up all the findings of our experimental campaign, we depict a synthetic picture in Figure 4.8 which can be used to identify the best strategies according to both brave and cautious reasoning. Indeed, Figure 4.8 reports the overall number of settings (over all input networks and by considering all initial configurations) in which each strategy has been identified as brave-optimal and cautious-optimal (for P_{b}) over the total of 1792 settings for each threshold. In Figure 4.8, subplots shows the results for different thresholds. In each subplot, the first five rows consider settings where the speed of diffusion of b is lower than the speed of diffusion of w (i.e., $(1, 2)$, $(1, 4)$, $(1, 8)$ and $(1, 16)$) while rows $6 - 10$ consider settings in which the speed of b is greater than the speed of w (i.e., $(2, 1)$, $(4, 1)$, $(8, 1)$ and $(16, 1)$). For example, by considering $t = 0.1$, bet was cautious-optimal in 938 settings, while for $t = 0.45$ it resulted to be cautious-optimal in 503 settings only. From the analysis we performed, bet seems to be the best *cautious* choice for low thresholds, while in all the other cases the best cautious choice is vr. As for brave reasoning, each cell reports the number of settings in which a strategy has been identified as brave-optimal together with the corresponding strategy for w. For example, by considering $t = 0.4$ and $\rho_{\mathsf{b}} > \rho_{\mathsf{w}}$, vr for b was brave-optimal in 569 settings together with eig for w (that resulted in this settings the worst strategy for w), while bet for b was brave optimal in 28 settings together with cls for w. From the analysis of the results on the one hand it emerged that, in general, according to a brave-reasoning, the worst strategy for w id eig, followed by deg, that are those allowing b to reach the maximum coverage for some strategy τ_{b}. On the other hand, again vr is the brave-optimal strategy for b in the majority of the settings, followed by bet.

4.5 Discussion and Conclusion

Opinion diffusion has been largely studied in earlier literature. Several studies considered a setting in which there are two opinions that compete [180, 181, 176], and some recent works also considered the scenario in which more than two opinions are available [192, 185, 193, 194]. In this paper, we have analyzed a progressive model of opinion diffusion, in which individuals can hold one of two competing opinions or can have no opinion at all. Given this model, we have investigated the effectiveness of *compact* strategies that, if adopted by marketers, suggests the sequence of individuals to target for maximizing the final diffusion of their opinion. We studied this problem from a theoretical point of view and complemented our analysis with an experimental evaluation that demonstrate how compact strategies can be effectively adopted in opinion maximization. In particular, our findings suggest that *vote rank* and *betweenness centrality* are very effective measure to characterize the power of the nodes in terms of their capacity to affect the final coverage of the diffusion process.

Our results open a number of avenues for further research. First, it would be relevant to investigate the impact of further network characteristics, such as network density, on the effectiveness of compact strategies. Furthermore, while we conducted experiments on large real social networks, it might be nonetheless interesting to consider small synthetic networks

on which it would be possible to check how the various strategies are far from the optimal coverage. Finally, another interesting avenue for future research is to develop more sophisticated models for compact strategies, such as hybrid models that combine multiple strategies to achieve even better results.

Part II: Explainable AI

Chapter 5

Nutrition Education Program and Physical Activity Improve the Adherence to the Mediterranean Diet: Impact on Inflammatory Biomarker Levels in Healthy Adolescents From the DIMENU Longitudinal Study

Adherence to Mediterranean diet (MD) and physical activity (PA) in adolescence represent powerful indicators of healthy lifestyles in adulthood. The aim of this longitudinal study was to investigate the impact of a nutrition education program (NEP) on the adherence to the MD and on the inflammatory status in healthy adolescents, categorized into three groups according to their level of PA (inactivity, moderate intensity, and vigorous intensity). As part of the DIMENU (Dieta Mediterranea & Nuoto) study, 85 adolescents (aged 14–17 years) participated in the nutrition education sessions provided by a team of nutritionists and endocrinologists at T0. All participants underwent anthropometric measurements, bio-impedentiometric analysis (BIA), and measurements of inflammatory biomarkers such as ferritin, erythrocyte sedimentation rate (ESR), and C-reactive protein (CRP) levels. Data were collected at baseline (T0) and 6 months after NEP (T1). To assess the adherence to the MD, we used KIDMED score. In our adolescents, we found an average MD adherence, which was increased at T1 compared with T0 (T0: 6.03 ± 2.33 vs. T1: 6.96 ± 2.03, $p = 0.002$), with an enhanced percentage of adolescents with optimal ($\geq$8 score) MD adherence over the study period (T0: 24.71% vs. T1: 43.52%, $p = 0.001$). Interestingly, in linear mixed-effects models, we found that NEP and vigorous-intensity PA levels independently influenced KIDMED score (β = 0.868, $p < 0.0001$ and β = 1.567, $p = 0.009$, respectively). Using ANOVA, NEP had significant effects on serum ferritin levels ($p < 0.001$), while either NEP or PA influenced ESR ($p = 0.035$ and 0.002, respectively). We also observed in linear mixed-effects models that NEP had a negative effect on ferritin and CRP (β = -14.763, $p < 0.001$ and β = -0.714, $p = 0.02$, respectively). Our results suggest the usefulness of promoting healthy lifestyle, including either nutrition education interventions, or PA to improve MD adherence and to impact the inflammatory status in adolescence as a strategy for the prevention of chronic non-communicable diseases over the entire lifespan.

5.1 Introduction

Adherence to the Mediterranean diet (MD) and physical activity (PA) in adolescence represent powerful indicators of healthy lifestyles in adulthood [195, 196, 197]. The MD, characterized by a high intake of vegetables, fruits, legumes, dairy products, and nuts, a moderate intake of fish and poultry, along with a low intake of red meat, processed foods, and saturated lipids [198], has been accepted as one of the healthiest dietary patterns in the world [199]. The inverse association between the MD adherence and a wide range of chronic and metabolic diseases is well-known [200], and it may be, at least in part, attributed to the anti-inflammatory properties of MD components [201]. For instance, it has been reported that low adherence to the MD is directly associated with a worse profile of circulating inflammation-related biomarkers [202]. Although the relationship between MD and inflammatory markers within a population of European adolescents has been recently investigated [203], the impact of MD and PA on inflammatory status in healthy adolescents remains to be clarified.

5.2 Materials and Methods

5.2.1 Study Population

The DIMENU (Dieta Mediterranea & Nuoto) project was funded by the EU Regional Operational Programme Calabria, Italy (prot. #52243/2017), for investigating the impact of the adherence of MD and PA on health status in a sample of adolescents from Southern Italy. Based on an in-depth collaboration with the public high school "Istituto Istruzione Superiore"—Castrolibero, and three swim and sport centers (sports club in Cosenza, Paola, and Crotone of Calabria Region, Italy), we were able to recruit and to select sedentary adolescents and subjects performing recreational sport activities or competitive sports, between December 2018 and January 2019 [204]. The exclusion criteria were cognitive or physical/motor limitation, health-related problems, use of medications, restrictive diet (i.e., hypocaloric, low carbohydrate, and low fat). All participants and their parents received a detailed explanation of study purposes. Prior to the enrolment of adolescents in the DIMENU trial, their parents provided written informed consent. All adolescents were subjected to study visits and data collection at baseline (T0) in which the NEP on MD-related issues and sports nutrition was also included. This study was conducted according to the guidelines laid down in the Declaration of Helsinki and approved by the Ethic Committee of the University of Calabria, Italy (#5727/2018).

5.2.2 Nutritional History Assessment and Nutrition Education Sessions

To collect the nutritional and medical history, participants were orally interviewed at baseline and after 6 months by a team of nutritionists through a nutritional history record, as previously reported [204]. Using the KIDMED test [198, 205], we assessed the adherence to the MD in the study population. The score of MD adherence, ranging from 0 to 12, was based on a 16-point paper questionnaire in which a value of +1 was assigned for the consumption of fruits, vegetables, fish, legumes, whole cereals or grain, nuts, oil, dairy products, and yogurt and a negative value -1 for skipping breakfast, consumption of baked goods, sweets, and going to fast food. At baseline, two 30/40-min education sessions, consisting of seminars and

interactive lectures structured to cover knowledge of food sources of macro- and micronutrients included in the healthy eating pattern and benefits of MD (basic nutrition concepts, eat at regular intervals, maintain adequate hydration, healthy food choices), were provided for all participants by study nutritionists (EA, AG, GC, EM, and SF) and by three endocrinologists (DB, SA, and SC). In addition, we have created an official website of DIMENU project [206] and a Facebook page [207] as innovative ways for assuring additional support and information on MD-related issues to all participants.

5.2.3 Physical Activity Intensity Levels

The intensity of PA levels was estimated following WHO recommendations [208] as physical inactivity [<3 metabolic equivalents (METs)], moderate-intensity (3–6 METs), and vigorous-intensity PA (>6 METs). Specifically, using a questionnaire to assess PA habits that we have described elsewhere [204], the enrolled adolescents were classified into three groups: physical inactivity (PAi = 23), moderate-intensity PA (Pam = 34) [subjects performing at least 60min daily of bicycling (n = 2), dancing (n = 2), brisk walking (n = 2), gymnastics (n = 4), aquatic aerobics (n = 3), recreational swimming (n = 21)] and vigorousintensity PA (PAv = 28) [subjects engaged in at least 60 min daily of jogging or running (n = 2), boxing (n = 1), tennis (n = 1), soccer (n = 2), basketball (n = 2), squash (n = 1), swimming (n = 16), aerobic dancing (n = 2), and volleyball (n = 1)]. The same PA intensity levels of adolescents were confirmed through the interview at T1.

5.2.4 Anthropometric Parameters and Bioelectrical Impedance Analysis

A detailed description of the anthropometric measurements and bio-impedentiometric analysis (BIA) performed has been reported elsewhere [204]. BIA estimated phase angle (PhA), total body water (TBW), body cell mass (BCM), fat-free mass (FFM), and fat mass (FM). Data obtained by BIA test were analyzed using version 1.2.2.8. of the software Bodygram Plus (Akern Srl; Florence, Italy).

5.2.5 Biochemical Measurements, Erythrocyte Sedimentation Rate, and Interleukin Assays

In the detailed explanation of study purposes, participants were informed to have not eaten at least for 8 h before blood collection. Additionally, participants were reminded 1 week before the study visit. Venous blood samples were collected at T0 and T1, and in order to obtain serum, samples were centrifuged as previously reported [204].

samples were centrifuged as previously reported [204]. ESR was measured by Wintrobe method. Serum C-reactive protein (CRP) levels were detected by immunonephelometry (GOLDSITE Diagnostics, Inc., Shenzhen, China). Serum ferritin levels were measured by enzyme-linked immunosorbent assay (ELISA) (Monobind Inc., Lake Forest, CA, United States), with a detection sensitivity limit of 0.17 ng/ml. Serum iron was determined on a Konelab 20i Chemistry Analyzer (Thermo Electron Corporation, Vantaa, Finland) according to the standardized procedures (Method Iron "Ferene S," Sclavo Diagnostics, Siena, Italy).

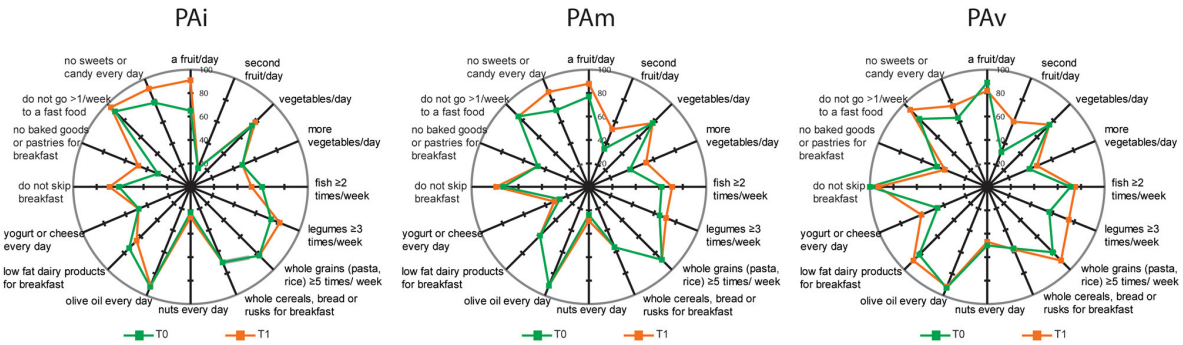
		PAi (n = 23 subjects)	Pam (n = 34 subjects)	PAv (n = 28 subjects)	p-value		
					NEP	PA	NEP + PA
BMI (Kg/m2)	T0	24.87 ± 5.53	21.91 ± 2.20	21.80 ± 2.30	0.007	0.001	0.501
	T1	25.25 ± 5.33	22.31 ± 2.25	21.92 ± 1.88			
PhA (◦)	T0	5.85 ± 0.52	6.13 ± 0.83	6.36 ± 0.54	0.051	0.004	0.518
	T1	5.87 ± 0.67	6.34 ± 0.72	6.55 ± 0.8			
BCM (%)	T0	52.88 ± 2.71	54.11 ± 4.17	55.39 ± 2.54	0.074	0.004	0.466
	T1	52.90 ± 3.59	55.19 ± 3.41	56.19 ± 3.61			
BCM (Kg)	T0	36.34 ± 10.4	32.96 ± 5.43	33.97 ± 5.41	<.001	0.151	0.394
	T1	37.26 ± 10.37	34.51 ± 5.79	35.24 ± 5.18			
FFM (%)	T0	73.62 ± 9.27	78.63 ± 9.01	82.37 ± 7.43	<.001	<.001	0.365
	T1	71.69 ± 7.17	75.98 ± 7.38	81.29 ± 6.38			
FFM (Kg)	T0	49.96 ± 11.3	46.49 ± 7.79	50.35 ± 7.91	0.322	0.159	0.663
	T1	49.91 ± 11.7	46.49 ± 8.40	50.89 ± 7.08			
FM (%)	T0	26.38 ± 9.3	21.37 ± 9.02	17.63 ± 7.43	<.001	<.001	0.380
	T1	28.30 ± 7.17	23.99 ± 7.41	18.70 ± 6.38			
FM (Kg)	T0	18.92 ± 9.84	12.75 ± 5.74	10.87 ± 4.54	<.001	<.001	0.478
	T1	20.51 ± 9.38	14.64 ± 4.74	11.80 ± 4.41			
TBW (%)	T0	54.9 ± 7.54	57.68 ± 6.45	62.62 ± 5.12	<.001	<.001	0.341
	T1	53.2 ± 6.2	54.92 ± 5.67	60.94 ± 5.04			

Table 5.1: Anthropometric characteristics and body composition parameters of participants according to the three physical activity (PA) groups at baseline (T0) and after 6 months (T1). Data are presented as mean ± SD. T0, Baseline; T1, 6 months of follow-up; PAi, Physical inactivity; PAm, moderate Physical Activity; PAv, vigorous Physical Activity; NEP, Nutrition Education Program; BMI, body mass index; PhA, phase angle; BCM, body cell mass; FFM, fat-free mass; FM, fat mass; TBW, total body water. The statistical differences were evaluated by two-way repeated-measures ANOVA. In boldface are reported statistically significant values. The effects of Nutrition Education Program (NEP) and PA on them are reported.

Quantification of interleukins was performed using ELISA kits for human IL-6, human TNF-α, human IL-1β, and human IL-10 (Merck Life Sciences, Darmstadt, Germany). The respective sensitivities of these assays were 1.6 pg/ml for IL-6, 0.2 pg/ml for TNF-α, 0.2 pg/ml for IL-1β, and 2 pg/ml for IL-10.

5.2.6 Mediterranean Diet Meal Plan

All participants received a personalized MD meal plan according to their different PA intensity levels. During the entire period of the NEP, the nutritionists gave verbal and written dietary instructions on the choice of typical Mediterranean foods. The dietary approach was based on the MD pattern according to the last guidelines [209, 210]. Each diet plan provides 15–20% of calories through protein, 45–60% of calories through carbohydrates, and 25–30% of calories through fat, with the respective distribution of macro- and micronutrients according to the different energy expenditure of each subject. We calculated the total daily energy expenditure (TDEE) using the formula: TDEE = Basal Metabolic Rate $\times$ Physical Activity Level, as recommended by the Italian Society of Human Nutrition [211]. Meals included an abundance of plant food (fruits, vegetables, whole grains, nuts, and legumes); fish, poultry, and eggs in moderate amounts; olive oil as the primary source of fat; low consumption of red meats, saturated fats, and sweets. Meals and food plans were designed using MetaDieta software version 4.2.1. (Meteda S.r.l, Roma, Italy).



	KIDMED score (Mean ± SD)		Source of variation	p-value
	T0	T1		
PAi	5.96±2.10	6.61±2.35	NEP	<0.001
PAm	5.57±2.30	6.94±1.66	PA	0.201
PAv	6.30±2.16	7.61±1.77	NEP + PA	0.692

Figure 5.1: Compliance with items from KIDMED test according to the three physical activity (PA) groups (PAi: inactive; PAm: moderate; PAv: vigorous) at baseline (T0) and after 6 months (T1). The radar chart plots the values of each item of Mediterranean diet score along a separate axis that starts in the center of the chart (0% compliance) and ends at the outer ring (100% compliance). KIDMED score is presented as Mean ± SD; statistical differences were evaluated by two-way repeated-measures ANOVA. NEP, Nutrition Education Program.

5.2.7 Statistical Analysis

Sample size was calculated by considering the whole adolescent population (aged 14–17 years) living in the Mediterranean area of Calabria region (76,000 adolescents) by fixing the confidence level to 95% and the confidence interval to 10% [212]. *Post-hoc* power analysis was performed, by G*Power software version 3.1.9.4 (University of Heinrich-Heine, Germany), to evaluate the adequacy of the analyzed sample; the effect size was calculated on a website [213] on the mean values and standard deviation obtained at baseline (T0) and after 6-month follow-up (T1) for KIDMED score (d = 0.4210), ESR (d = 0.3621), and ferritin levels (d = 0.6584) using Cohen's d formula. Data were analyzed by SigmaPlot for Windows version 12.0 (Systat, San Jose, CA, United States) and reported as the mean and SD. Data normality was verified by Kolmogorov–Smirnov test (with Lilliefors' correction). The statistical differences between variables T0 and T1 were evaluated by using paired t-test. Frequencies (%) were used to describe qualitative variables that were graphically represented in radar plots. McNemar's chi-squared test was applied to evaluate the statistical differences. Two-way repeated-measures ANOVA was used to test for significant differences between KIDMED score with respect to NEP and PA and for their interaction. Spearman's correlation test was used to assess the association between variables. Linear mixed-effects models were used to test the association between dependent variables (KIDMED, ferritin, ESR, CRP, and cytokines) and independent variables such as NEP and PA along with a set of anthropometric parameters to improve the robustness of our analyses. In the modeling analysis, the p-value was adjusted with the Holm–Šidák method (extension of Holm–Bonferroni method). The latter analyses were carried out in a Python 3 environment taking advantage of the *statsmodels* module. Results are considered statistically significant when $p < 0.05$.

	Model 1					Model 2				
	β	se	p	CI (95%)		β	se	p	CI (95%)	
				Lower	Upper				Lower	Upper
Intercept	-11.924	13.278	0.369	-37.948	14.100	-12.191	13.355	0.361	-38.365	13.984
NEP	0.868	0.218	**0.000**	0.441	1.295	0.463	0.422	0.273	-0.365	1.291
PAm	0.507	0.552	0.359	-0.576	1.590	0.201	0.670	0.764	-1.112	1.514
PAv	1.567	0.599	**0.009**	0.392	2.741	1.099	0.707	0.120	-0.286	2.485
Gender M	-0.443	0.582	0.446	-1.584	0.697	-0.476	0.585	0.416	-1.623	0.671
Age	0.094	0.211	0.655	-0.319	0.508	0.434	0.536	0.418	-0.321	0.506
Weight	-0.076	0.097	0.436	-0.267	0.115	0.677	0.558	0.225	-0.271	0.112
Height	0.099	0.077	0.200	-0.052	0.250	0.093	0.211	0.661	-0.050	0.255
BMI	0.249	0.287	0.384	-0.312	0.811	-0.080	0.098	0.416	-0.302	0.827
PhA	-0.233	0.239	0.330	-0.702	0.236	0.103	0.078	0.186	-0.321	0.506
NEP:PAm						0.262	0.288	0.362	-0.616	1.485
NEP:PAv						-0.247	0.241	0.304	-0.416	1.770

Table 5.2: Mixed-effect linear regression model for the association between KIDMED score and NEP, PA, and a set of anthropometric parameters, considering T0 and T1 as a unique longitudinal dataset. Model 1: KIDMED vs. NEP, PA, Gender, Age, Weight, Height, BMI, PhA. Model 2: KIDMED vs. NEP, PA, Gender, Age, Weight, Height, BMI, PhA, NEP:PA (Interaction). PAm, moderate physical activity; PAv, vigorous physical activity; CI, confidence interval; The regression coefficient (β), the standard error (se), and the statistical significance (p) are reported. The p-value was adjusted with the Holm–Šidák method (extension of Holm–Bonferroni method). In boldface are reported statistically significant values.

5.3 Results

5.3.1 Characteristics of Participants

The longitudinal DIMENU project was conducted in 85 adolescents (44 girls and 41 boys) who completed the scheduled monitoring visit at T0 and T1. The enrolment at T0 included 92 adolescents, but seven participants dropped out over the course of the study. Table 5.1 summarizes the anthropometric characteristics and body composition parameters of 85 subjects categorized according to the intensity level of PA into the following three groups: inactivity (PAi), moderate-intensity (PAm), and vigorous-intensity (PAv) PA at T0 and T1. We observed that either NEP or PA had statistically significant effects on the majority of anthropometric measurements and body composition parameters, while no combined effects were found (Table 5.1).

5.3.2 Impact of NEP and PA on the Adherence to the Mediterranean Diet

Over the study period, we found that adherence to the MD evaluated by KIDMED score increased at T1 compared with T0 (T0: 6.03 ± 2.33 vs. T1: 6.96 ± 2.03, p = 0.002) in all adolescents. Based on the KIDMED values, we divided the population into optimal (score $\geq$8), medium (score 4–7), and poor (score $\leq$3) adherence to the MD [205], and we observed that the proportion of adolescents having optimal adherence to the MD was significantly higher after nutritional intervention with respect to baseline (T0: 24.71% vs. T1: 43.52%, p = 0.001).

In Figure 5.1, we reported the compliance with items from KIDMED test in the three separate PA groups. No significant changes were observed in the comparison between T0 and T1 for most items with the exception of an increase in the consumption of "a fruit/day"

		PAi (n = 23 subjects)	Pam (n = 34 subjects)	PAv (n = 28 subjects)	p-value		
					NEP	PA	NEP+PA
Ferritin (ng/ml)	T0	34.9 ± 19.6	27.7 ± 19.7	31.1 ± 21.3	<0.001	0.436	0.701
	T1	21.1 ± 20.5	16.2 ± 15.4	17.0 ± 15.0			
ESR (mm/h)	T0	20.1 ± 10.9	21.1 ± 11.1	12.2 ± 7.1	0.035	0.002	0.259
	T1	25.9 ± 20.5	21.8 ± 12.7	12.7 ± 8.2			
CRP (mg/L)	T0	2.6 ± 4.5	1.8 ± 2.5	1.62 ± 2.2	0.483	0.133	0.924
	T1	2.4 ± 5.1	1.1 ± 0.5	1.0 ± 0.1			
IL-1β (pg/ml)	T0	9.8 ± 8.9	25.9 ± 40.7	26.9 ± 53.2	0.083	0.152	0.605
	T1	7.9 ± 7.3	26.4 ± 54.6	30.2 ± 63.4			
IL-6 (pg/ml)	T0	124.3 ± 138.4	328.1 ± 930.8	503.7 ± 1170.3	<0.001	0.179	0.173
	T1	420.5 ± 456.5	647.0 ± 1321.1	918.1 ± 1592.7			
TNF$-\alpha$ (pg/ml)	T0	552.8 ± 499.4	744.5 ± 993.8	974.7 ± 1567.7	0.002	0.239	0.034
	T1	665.4 ± 517.8	839.9 ± 917.5	1275.7 ± 1909.6			
IL-10 (pg/ml)	T0	164.65 ± 189.9	180.2 ± 292.1	240.7 ± 360.2	0.449	0.514	0.748
	T1	195.5 ± 184.7	199.3 ± 207.9	268.6 ± 260.5			

Table 5.3: Serum inflammatory markers in adolescents according to the three physical activity (PA) groups at baseline (T0) and after 6 months (T1). Data are presented as mean ± SD. T0, Baseline; T1, 6 months of follow-up; PAi, Physical inactivity; PAm, Moderate Physical activity; PAv, Vigorous Physical activity; NEP, Nutrition Education Program; ESR, erythrocyte sedimentation rate; CRP, C-reactive protein; IL-1β, interleukin-1beta; IL-6, interleukin-6; TNF-α, tumor necrosis factor alpha; IL-10, interleukin-10. Statistical analysis was performed on ln-normalized samples. The statistical differences were evaluated by two-way repeated-measures ANOVA. In boldface are reported statistically significant values. The effects of Nutrition Education Program (NEP) and PA on them are reported.

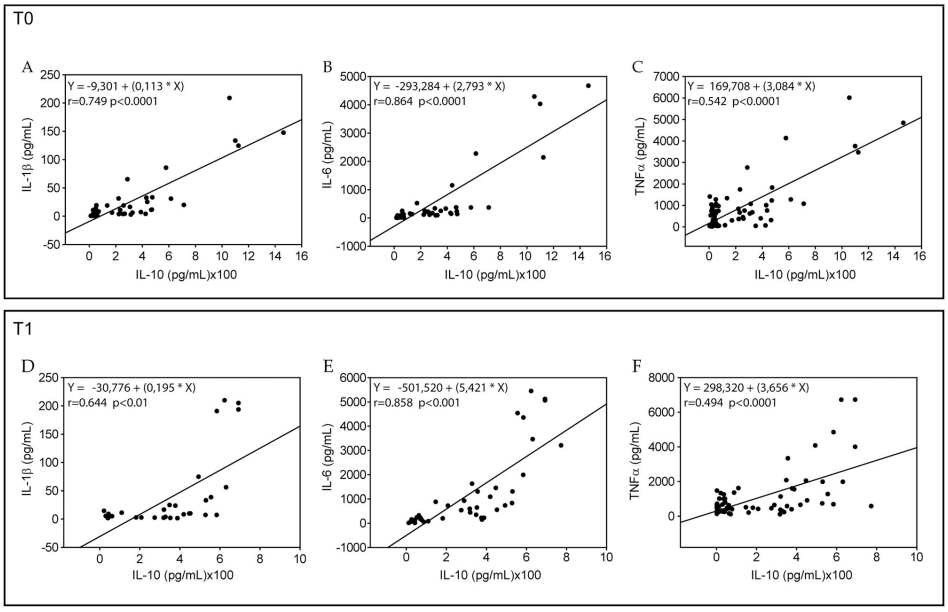


Figure 5.2: Correlations between serum cytokines in the adolescent population at T0 and T1. The correlation coefficients (r) between IL-1β, IL-6, TNF-α, and IL-10 are presented as a heatmap.

in PAi group and "second fruit/day" in PAv group (65 vs. 91%, p = 0.03 and 32 vs. 60%, p = 0.03, respectively). Statistical analyses evidenced that KIDMED score is influenced by NEP ($p < 0.001$) and not by either PA or by NEP + PA (Figure 5.1).

To evaluate the impact of NEP and PA on the adherence to the MD, we performed linear mixed-effects models including a set of anthropometric parameters (age, gender, weight, height, BMI, and PhA) as independent variables. Results show that NEP had a significant

		T0			T1		
		Ferritin (ng/ml)	ESR (mm/h)	CRP (mg/L)	Ferritin (ng/ml)	ESR (mm/h)	CRP (mg/L)
BMI (kg/m2)	r	0.19	0.189	0.218	0.08	0.073	0.282
	p	0.081	0.091	0.046	0.460	0.504	0.009
Pha (∘)	r	0.113	-0.201	-0.218	0.201	-0.289	-0.036
	p	0.304	0.071	0.046	0.065	0.007	0.740
BCM (%)	r	0.110	-0.210	-0.217	0.196	-0.273	-0.026
	p	0.316	0.060	0.047	0.072	0.011	0.810
FFM (%)	r	-0.002	-0.370	-0.341	0.166	-0.524	-0.066
	p	0.983	0.0007	0.0015	0.129	<0.00001	0.547
FM (%)	r	0.002	0.370	0.341	-0.167	0.524	0.066
	p	0.983	0.0007	0.0015	0.127	<0.00001	0.545
BCM (kg)	r	0.311	0.012	0.213	0.315	-0.229	0.127
	p	0.004	0.915	0.052	0.0035	0.035	0.246
FFM (kg)	r	0.277	-0.195	0.04	0.358	-0.484	0.151
	p	0.01	0.0812	0.717	0.0008	<0.00001	0.166
FM (kg)	r	0.076	0.384	0.369	-0.047	0.465	-0.0095
	p	0.484	0.0004	0.0006	0.668	<0.00001	0.931
TBW (%)	r	0.033	-0.392	-0.329	0.119	-0.511	-0.073
	p	0.765	0.0003	0.0022	0.276	<0.00001	0.505

Table 5.4: Correlations between serum ferritin, ESR, and CRP levels with body composition parameters in all the sample at baseline (T0) and after 6 months (T1). CRP, C-reactive protein; ESR, erythrocyte sedimentation rate; BMI, body mass index; PhA, phase angle; BCM, body cell mass; FFM, fat-free mass; FM, fat mass; TBW, total body water. Data were analyzed by Spearman's correlation test. The correlation coefficient (r) and the statistical significance (p) are reported. In boldface are reported statistically significant values.

positive effect on the KIDMED score ($p < 0.001$), considering constant all the other variables. Furthermore, PAv had a greater impact on the MD adherence with respect to PAi ($p = 0.009$), while there were no significant differences in KIDMED scores between PAm and PAi (Table 5.2). To further study these associations, we investigated the interaction between NEP and PA on the KIDMED score, concluding that there were no significant effects between the two variables (Table 5.2).

	Model 1				Model 2				Model 3			
	β	se	p	CI (95%)	β	se	p	CI (95%)	β	se	p	CI (95%)
Intercept	-248.402	100.597	0.014	[-445.5,-51.23]	48.384	76.752	0.528	[-102.04,198.81]	15.745	10.597	0.137	[-5.02,36.51]
NEP	-14.763	1.516	0.000	[-17.73,-11.79]	2.161	1.421	0.128	[-0.62,4.94]	-0.714	0.308	0.020	[-1.31,-0.11]
PAm	-4.019	4.658	0.388	[-13.14,5.11]	0.447	2.783	0.873	[-5.00,5.90]	0.222	0.426	0.601	[-0.61,1.05]
PAv	-5.708	5.000	0.254	[-15.50,4.09]	-4.286	2.983	0.151	[-10.13,1.56]	0.115	0.453	0.800	[-0.77,1.00]
Gender M	1.176	4.854	0.809	[-8.33,10.69]	-9.791	2.864	0.001	[-15.40,-4.17]	-0.184	0.474	0.698	[-1.11,0.74]
Age	-1.189	1.786	0.506	[-4.68,2.31]	-0.386	1.071	0.719	[-2.48,1.71]	0.003	0.161	0.986	[-0.31,0.31]
Weight	-1.356	0.718	0.059	[-2.76,0.05]	0.224	0.593	0.706	[-0.93,1.38]	0.148	0.079	0.060	[-0.00,0.30]
Height	1.441	0.585	0.014	[0.29,2.58]	-0.116	0.456	0.799	[-1.01,0.77]	-0.094	0.062	0.127	[-0.21,0.02]
BMI	5.216	2.120	0.014	[1.06,9.37]	-0.123	1.707	0.943	[-3.46,3.22]	-0.287	0.233	0.217	[-0.74,0.16]
PhA	4.680	1.692	0.006	[1.36,7.99]	-1.663	1.354	0.219	[-4.31,0.99]	-0.163	0.218	0.454	[-0.58,0.26]

Table 5.5: Mixed-effect linear regression model for the association between ferritin, ESR, CRP, and NEP, PA, and a set of anthropometric parameters, considering T0 and T1 as a unique longitudinal dataset. Model 1: Ferritin vs. NEP, PA, Gender, Age, Weight, Height, BMI, PhA. Model 2: ESR vs. NEP, PA, Gender, Age, Weight, Height, BMI, PhA. Model 3: CRP vs. NEP, PA, Gender, Age, Weight, Height, BMI, PhA. PAm, moderate physical activity; PAv, vigorous physical activity. CI, confidence interval; The regression coefficient (β), the standard error (se), and the statistical significance (p) are reported. The p-value was adjusted with the Holm–Šidák method (extension of Holm–Bonferroni method). In boldface are reported statistically significant values.

5.3.3 Correlations Between Inflammatory Biomarkers and Body Composition Parameters

Evaluating the inflammatory status by measuring ferritin, ESR, and CRP along with a panel of serum cytokines in our population longitudinally, we observed that ferritin and CRP levels were significantly reduced in all adolescents (50.66 ± 33.90 vs. 17.79 ± 16.75, $p < 0.001$ and 1.94 ± 3.07 vs. 1.53 ± 2.76, $p = 0.028$, respectively). Using ANOVA, we found that NEP significantly influenced ferritin, ESR, and serum IL-6 as well as TNF-α levels, and PA had effects on ESR, while NEP in combination with PA exerted the effects on serum TNF-α (Table 5.3). Using the correlation analysis, we found that the pro-inflammatory IL-1β, IL-6, and TNF-α cytokines were significantly associated with the anti-inflammatory IL-10 levels at both times of observation (Figure 5.2). As expected, serum IL-6, IL-1β, and TNF-α levels were also directly correlated.

5.4 Discussion

The results of this study highlight the importance of nutrition education programs (NEP) and physical activity (PA) in improving adherence to the Mediterranean diet (MD) and in reducing inflammatory markers in healthy adolescents. The NEP effectively increased the KIDMED score, indicating better adherence to the MD, and this effect was independent of PA levels. Vigorous-intensity PA also positively impacted MD adherence, although to a lesser extent than NEP.

Furthermore, the study demonstrated significant reductions in ferritin and CRP levels, which are markers of inflammation, following the NEP. These findings suggest that both NEP and PA can contribute to lowering the risk of chronic diseases by improving diet quality and reducing inflammation.

5.5 Conclusion

Promoting healthy lifestyle choices, such as adherence to the Mediterranean diet and engaging in regular physical activity, is crucial for the prevention of chronic non-communicable diseases. This study underscores the effectiveness of nutrition education programs in enhancing dietary habits and reducing inflammatory biomarkers among adolescents. Future research should continue to explore the long-term benefits of such interventions and their potential to improve public health outcomes.

Chapter 6

μ-Net: A Deep Learning-Based Architecture for μ-CT Segmentation

X-ray computed microtomography (μ-CT) is a non-destructive technique that can generate high-resolution 3D images of the internal anatomy of medical and biological samples. These images enable clinicians to examine internal anatomy and gain insights into the disease or anatomical morphology. However, extracting relevant information from 3D images requires semantic segmentation of the regions of interest, which is usually done manually and results time-consuming and tedious. In this work, we propose a novel framework that uses a convolutional neural network (CNN) to automatically segment the full morphology of the heart of *Carassius auratus*. The framework employs an optimized 2D CNN architecture that can infer a 3D segmentation of the sample, avoiding the high computational cost of a 3D CNN architecture. We tackle the challenges of handling large and high-resoluted image data (over a thousand pixels in each dimension) and a small training database (only three samples) by proposing a standard protocol for data normalization and processing. Moreover, we investigate how the noise, contrast, and spatial resolution of the sample and the training of the architecture are affected by the reconstruction technique, which depends on the number of input images. Experiments show that our framework significantly reduces the time required to segment new samples, allowing a faster microtomography analysis of the *Carassius auratus* heart shape. Furthermore, our framework can work with any bio-image (biological and medical) from μ-CT with high resolution and small dataset size.

6.1 Introduction

X-ray Computed tomography (CT) is a powerful and widely used imaging tool that provides 3D digital gray-scale images of an object's internal structure; such images can be quantitatively analyzed to identify specific components of the 3D morphology. Modern CT is a valuable diagnostic tool that provides meaningful information reducing X-ray doses. μ-CT is an even more powerful technique used in the study of human and animal anatomy in research and medicine [214], allowing to achieve higher resolutions. Computer-based approaches can ease and enhance the extraction of information and patterns from μ-CT, leveraging, for instance, accurate semantic segmentation of the anatomical parts. Moreover, in the latest years, the use of image segmentation algorithms proved to be promising in facilitating analysis and detection of abnormalities [215, 216]. Those methods can be applied voxel-wise in a 3D

context as well as pixel-wise, slice by slice [217]; however, instrumental noise, non-uniform intensity, and pixel discretization can limit the resolution of the image and obscure finer details. Traditional segmentation methods like thresholding and morphological filters are sensitive to parameter changes, leading to potential detail loss; conventional methods struggle with variations in phase/absorption contrast intensity [217]. Generally, 3D segmentation methods lack flexibility and adaptability, and determining the best method for a specific application is challenging, especially in medical imaging due to the heterogeneity of image characteristics and distributions [217]. Deep Learning (DL) approaches such as CNNs rapidly became the state-of-the-art (SOTA) for medical image segmentation, classification, recognition, and report generation [218, 219], and have been widely applied in the field of CT. However, few attempts have been made on μ-CT; indeed, the wealth of information presents a significant challenge in terms of analysis and interpretation. This is particularly evident in semantic segmentation tasks such as for kidney [220], cartilage [221], temporal bone [222], lung [223] and thorax mouse μ-CT [214]. Same applies to cardiac imaging, crucial for patient-specific intervention planning [224]; here, primary datasets are mainly magnetic resonance imaging (MRI), but tomography datasets have started to be acquired, which have a higher resolution. Notable contributions include DL segmentation structure for MRI cardiac datasets [225], U-net variant for short-axis MRI [226], DL approach for ECG-gated CT data [227], novel pipeline for whole heart CT segmentation [228].

This work aims at defining a general framework for DL-based processing of hi-resolution μ-CT images in presence of small datasets, a prevalent scenario in the medical domain. The underlying rationale is that we can improve performance and reliability of image segmentation by considering each class individually. Our approach not only contributes at enhancing μ-CT segmentation performances, but also fosters the usage of more lightweight models, in contrast to the current widespread usage of foundational models. The main contributions of this work can be summarized as follows.

− We build a new dataset consisting of μ-CT images from *C. auratus*'s heart, a teleost fish, also known as *goldfish*.

− We design a novel DL-based framework for extracting, enhancing, analyzing information from μ-CT. It extends SOTA semantic segmentation by defining multiple models and ensembling strategies. We present an implementation of the framework and assess it by designing and conducting an extensive experimental campaign over the newly introduced dataset. Results show that our proposal outperforms the SOTA methods, exhibiting improved performance and reduced misclassification errors.

− We show how the application of 2D CNNs followed by custom post-processing to achieve 3D continuity reduces computational costs if compared with 3D CNNs.

− We design an approach for feasible and robust segmentation, explicitly suited for use cases in which a limited number of labeled samples is available.

− We study how the quality of 3D tomographic images affects architectural performance, given that obtaining such images requires to collect multiple projection images over a wide range of projection angles.

To the best of our knowledge, this is the first approach that proposes a combination of multiple DL-based models and a comprehensive ablation study to assess the benefits of different architectures and parameter components in the context of μ-CT image segmentation, even with limited prior knowledge.

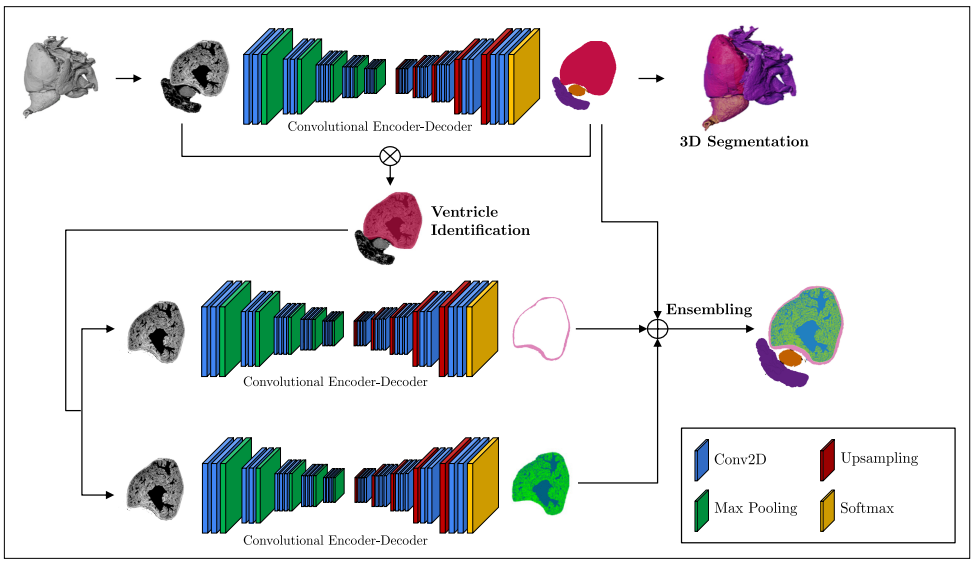


Figure 6.1: Initially, the μ-Net employs a CNN to identify the ventricles. Following this, two separate DL architectures carry out binary segmentation of various areas. The final result is obtained by applying an ensemble strategy to the different segmentations produced by each model.

6.2 Proposed approach

We present μ-Net, a novel DL-based framework for the analysis and semantic segmentation of μ-CT images. As already introduced, although the high resolution of μ-CTs offers many advantages, images can either be too rich or have little variability between different tissues; this can negatively affect CNN generalization capability, resulting in misclassifications; this is further exacerbated when only a few images are available. The herein proposed framework addresses such challenges by automatically extracting meaningful information from μ-CT images. μ-Net faces different tasks with different specialized models; each model is trained to automatically solve a small part of the whole task: each model segments a different area of the heart, and we defined an ad-hoc ensembling procedure to combine the results. One of the key advantages of our approach is versatility; indeed, our framework can be applied to any μ-CT images in the medical domain, regardless of the organs/tissues involved,thus resulting as a valuable tool for researchers across various disciplines. It is flexible and adaptable, as it can be configured with different architectures and customized according to the dataset; moreover, it is specifically tailored to preprocess and postprocess images of this kind with suitable filters, taking into account the 3D nature of the image. Fig. 6.1 shows the architecture of μ-Net whose aim is to automatically segment heart morphology in μ-CT images of goldfish. We propose the following five-step procedure for performing semantic segmentation of the goldfish μ-CT images: ***1. Data acquisition and preparation***: biological samples are collected and properly stained; subsequently, μ-CT projections are acquired and the volume is reconstructed and normalized (see Section 6.3). ***2. Data preprocessing***: filters are applied to images according to the different tasks. ***3. Segmentation model***: we defined and trained three different models by dividing the segmentation problems (see Section 6.4.2); ***4. Data post-processing***: different filters and volumetric techniques are used to obtain a 3D coherence starting from 2D CNN models. ***5. Ensembling models***: we defined an ensembling set of rules to merge the results and obtain the whole final semantic segmentation (see Section 6.4.2).

We experimentally validate the proposed methodology for the particular image segmentation task and compared the results with the SOTA methods (see Section 6.4.4).

6.3 Dataset building and description

In our experiments, we use μ-CT of the heart of *C. auratus* (Linnaeus, C. (1758)), a teleost fish, also known as *goldfish*. The scans were manually annotated under the supervision of 4 expert biologists for supervised learning.

The goldfish heart comprises four main components: sinus venosus, atrium, ventricle, and *bulbous arteriosus* [229]. Notably, the atrium features a spacious cavity with a muscular rim and a network of thin elastin and collagen fibers. Meanwhile, the ventricle consists of two distinct layers: the outer *compacta*, rich in blood vessels and muscle bundles oriented in various directions, and the inner *spongiosa*, which lacks blood vessels but contains numerous fibers. Sample preparation and dataset acquisition consist of different steps; it is worth noting that the entire procedure requires several hours. Given the anonymous nature of the submission, additional details on data sources will be disclosed in case of publication. The X-ray acquisition resulted in a challenging procedure, as the biological nature of the samples posed several technical issues. A staining procedure was applied to enhance the contrast of the samples; the absorption contrast technique was used to reconstruct 3D images. Each sample rotates with an angular step $\Delta\theta$, and is penetrated by an X-ray beam at each step; the attenuated X-ray beam's intensity is measured, creating a sinogram. The 3D structure is converted into a stack of 2D sinograms, that are fed to a reconstruction algorithm. We used the Filtered Back-Projection (FBP) algorithm for speed and simplicity. The resulting 2D image stack from FBP defines the 3D twin of the μ-CT sample. μ-CTs are manually segmented to define the ground truth labels, which hold clinical significance and include atrium, ventricle, *Bulbus arteriosus*, *compacta*, and *lacunary spaces*.

The analysis of such dataset is significantly interesting for studying cardiac pathologies. When subjected to oxygen deprivation [230, 231, 232], the goldfish accelerates its cardiac functions and, despite its small size, exhibits electrical activities similar to those of large mammals, which makes it relevant for translational research [233]. Moreover, the goldfish heart is influenced by many hormones and peptides acting as cardiac modulators in mammals under normal and stressful conditions [234]. These features make the goldfish an attractive model for exploring the mechanisms that give high flexibility to the heart, especially when facing internal and external challenges.

6.4 Experimental Design

6.4.1 Data acquisition, 3D reconstruction and data preparation

According to the previous section (see Sec. 6.3) and the difficulties that data acquisition and preparation hold, we acquired only 3 samples. For each sample, a total of $N_p = 3600$ projections with an angular resolution of 0.1 degrees were acquired. Samples were reconstructed in line with the section 6.3 using for each sample different projection dose. For each of the three samples, we used N_p, $\frac{N_p}{2}$, and $\frac{N_p}{3}$ projections to reconstruct three datasets, D_1, D_2, and D_3, respectively. This allowed us to test how well our architecture can handle tasks with varying levels of projection data. We also conducted an analysis on the impact of input dimensions

on the model's performance. This was achieved by generating new, cropped stacks from the original data, focusing on the region of interest. This approach enabled us to evaluate the adaptability of our model to tasks with varying input sizes. Each reconstructed sample consists of approximately 1500 slices of size 1300×1300 pixels, where each voxel corresponds to $5,55\mu m$. We implemented a normalization process for the dataset to ensure uniformity without introducing any additional bias. In the field of tomography, a specific range of absorption values is selected during the reconstruction phase. This selection process results in each sample appearing self-normalized. For our 16-bit images, any pixel in the reconstructed image that falls below this threshold is assigned a value of 0, while those above it are assigned a value of 2^{16}. Various strategies exist in the literature for this process, but we chose to reconstruct our samples using the same range of absorption values. This range serves as a normalization factor in our methodology. Furthermore, in our experiment, we also considered the 2D-trans-axial projection of μ-CTs: Axial View (corresponds to the XY plane, which is perpendicular to the rotation axis Z), Sagittal View (XZ plane), and Coronal View (YZ plane). Then, we split the entire dataset of 9 stacks of images (3 datasets $\times$ 3 stacks), composed of D_1, D_2 and D_3 into $D_{1,train}$, $D_{1,test}$, $D_{2,train}$, $D_{2,test}$, $D_{3,train}$ and $D_{3,test}$. In this setting, the train subsets consist of 2 stacks (2 heart samples). The third sample is used as a single heart sample that is common to all the datasets, except for the number of projections (N_p). To reduce redundancy, we select only one image out of every three from the input stacks for training, since the images are nearly identical when they are next to each other. For each training iteration, we randomly select one tile from each 2D slice to reduce its size. The training set is split into 70% for training and 30% for validation to monitor progress and prevent overfitting.

6.4.2 Training phase and evaluation metrics

The framework was developed using Pytorch (v1.13.0). A high-performance computing node with two Tesla V100-PCIE-16GB GPUs, Intel® Xeon® Gold 5118 CPU (2.30GHz), and 512GB of RAM was used for training. Jaccard index, also known as the Intersection Over Union (IoU) coefficient, is used as the evaluation metric during the training (i.e., 1 means perfect prediction, 0 worst prediction) [235, 236]. To assess the overall performance on the test set, after reconstructing the entire volume, we employed an IoU weighted by the frequency of each class in the 3D image.

6.4.3 Ablation study

As for the ablation study we conducted, we explored:

- **(A.1) Hyperparameter space:** learning rate, tile size, model architecture, preprocessing, and postprocessing. For each aspect, we compare several options and report the results on the validation set.
- **(A.2) DL-based models:** Segnet [237], DeepLabV3 [238] and U-net [239].
- **(A.3) Input parameters:** normalization type, tile size, number of slices for each stack (i.e., number of images are fed into the model at once), preprocessing and postprocessing methods. To reduce the computational cost of processing large images and to avoid

losing small-scale details relevant to our samples, we applied random cropping of sub-images, called tiles. We also experimented with different filters on input images (i.e., histogram equalization, median, unsharpmask filter).

- **(A.4) Number of projections:** variation of projection dose and, consequently, the generated 3D images (see Sec. 6.3); number of projections affects the quality of the image in terms of noise, spatial resolution, and artifacts. We found it useful to examine how performance changed with spatial resolution.

6.4.4 Experiments

As mentioned, we split the semantic segmentation task into separated sub-problems, focused on specific classes. Therefore, we have conducted several experiments: **(1) Semantic Segmentation of Atrium, Ventricle, and *Bulbus arteriosus***. The chosen model is the Segnet [237] (see subsection 6.4.3). The train ran for 150 epochs with a learning rate of 0.0001, Adam optimizer, and tiles dimension of 400×400 pixels. We trained our model on the XY view of the sample and inferred on all three views (XY, XZ, and YZ). To obtain the final prediction and the 3D continuity, first, we chose the pixel-based mode for the pixels of intersection between the views and then we applied a hole-filling algorithm. **(2) Binary Segmentation of *lacunary spaces***. We used the Segnet model with the same parameters as the previous step, except for the tile size of 224×224 pixels. The model received as input only the ventricle image part obtained from the previous step. To enhance the contrast between *lacunary spaces* and tissue, we applied an unsharp mask filter to the input image. **(3) Binary Segmentation of *compacta***. Similarly, we used a Segnet model with the same parameters and tile size. Also in this case we used as input only the ventricle part of the image. **(4) Overall Semantic Segmentation via Ensambling**. Once the results of the three experiments have been obtained, an ensembling strategy was performed to obtain a single segmentation result. Such strategy consists of a set of rules: *(I)* the Atrium class is always chosen over all other classes; *(II)* the *Bulbus arteriosus* class is preferred over the *lacunary spaces* and *compacta* ones; *(III)* the *compacta* class is selected in preference to the Ventricle and *lacunary spaces* classes; *(IV)* the *lacunary spaces* class is favored over the Ventricle class.

Comparison approaches To evaluate our approach, we compared it with two SOTA anatomical image segmentation tools: Biomedisa [240] and nnU-net [241]. Biomedisa is a platform designed for semi-automatic segmentation of large volumetric images, using smart interpolation of sparsely pre-segmented slices. nnU-net, on the other hand, is a DL-based method that self-configures for any new segmentation task, covering preprocessing, network architecture, training, and post-processing.

6.5 Results and Discussion

Among the tested architectures (see Sec. 6.4.3 (A.2)) we discarded DeepLabV3 due to its poor accuracy. Although U-net achieved good performance, Segnet obtained the best results on the test set. Therefore, we selected Segnet as the model for each experiment. Also, according to Sec. 6.4.3 (A.3), we tested two data normalization techniques: a normalize the data based on the mean and standard deviation of each stack and b normalize during the

Table 6.1: Performance achieved by μ-Net on the test set according to different training sets with a different number of projection doses.

Train set	IOU (%) on Test set		
	$D_{1,test}$	$D_{2,test}$	$D_{3,test}$
$D_{1,train}$	**87.9** ($\pm$3.7)	77.6 ($\pm$4.5)	44.6 ($\pm$7.8)
$D_{2,train}$	44.5 ($\pm$7.8)	73.7 ($\pm$3.4)	81.1 ($\pm$3.5)
$D_{3,train}$	30.3 ($\pm$8.5)	81.7 ($\pm$3.6)	86.8 ($\pm$3.6)
$D_{1,train} + D_{2,train}$	**87.9** ($\pm$3.7)	**86.4** ($\pm$3.4)	**88.6** ($\pm$3.7)

Table 6.2: Comparing IOU scores for our proposal, nnU-net, and Biomedisa methods. Best results for each class are reported in bold.

	IOU (%) on Test set					
	Ventricle	*Bulbus arteriosus*	Atrium	*Compacta*	*Lacunary spaces*	Total
μ-Net	**94.5** ($\pm$3.4)	77.5 ($\pm$3.7)	**87.9** ($\pm$2.4)	**77.2** ($\pm$2.4)	**84.8** ($\pm$3.6)	**87.6** ($\pm$3.7)
nnU-net	80.2 ($\pm$3.4)	**80.1** ($\pm$3.4)	61.1 ($\pm$5.2)	64.8 ($\pm$4.2)	72.9 ($\pm$4.4)	76.8 ($\pm$5.2)
biomedisa	63.5 ($\pm$5.4)	16.3 ($\pm$8.4)	20.5 ($\pm$7.4)	50.2 ($\pm$5.6)	56.9 ($\pm$5.4)	43.8 ($\pm$8.4)

reconstruction process. The technique described in b yielded better results in the ablation study so we adopted it. As stated by Sec. 6.4.3 (A.4), we trained different models varying on the number of projections used for the reconstruction stack.
Results of the first experiment (see Sec. 6.4.4) show that as the number of projection doses decreases, the spatial resolution deteriorates. The models were then evaluated on various test sets with different numbers of projection doses. Results are reported in Tab. 6.1. The models were tested by using the test sets for each of the 3 different datasets and performing 3-fold cross-validation over the 3 stacks in each dataset.

The table shows that models trained with high-resolution images obtain good performances on a high-resolution test set, while worse performances are reported for a low-resolution test set (see the first row of Tab. 6.1). Training on medium and low-resolution images results in better performance in similar-resolution test sets than the ones obtained on high-resolution (see the second and third row of Tab. 6.1). This evidences that as the number of projections (and hence the spatial resolution) decreases, the performance of the network deteriorates due to a lack of information. However, training a model using images with different resolutions results in a more stable performance, meaning that the network has more generalization capability across resolutions.
Tab. 6.2 shows performance results in terms of IOUs for each experiment performed (see Sec. 6.4.4). They largely hinge on the outcome of the initial experiment, the better the ventricle is detected in the first experiment the better the *compacta* and *lacunary spaces* will be detected in the same area of interest. Our workflow achieved an IOU value of 87.6% where a very good identification of ventricle (94.5%) allows a good detection of *compacta* and *lacunary spaces*, 77.2% and 84.8%, respectively. We compared our proposal with two SOTA semantic segmentation models, nnU-net and Biomedisa. We trained both models to segment atrium, ventricle, and *Bulbus arteriosus* regions only; our workflow achieved a higher IOU than both models, with 76.8% for nnU-net and 43.8% for Biomedisa.The lower performance of nnU-net may be attributed to its automatic selection of patch size, filters, and normalization. We used the 2D configuration of nnU-net, as the 3D one was not feasible due to the

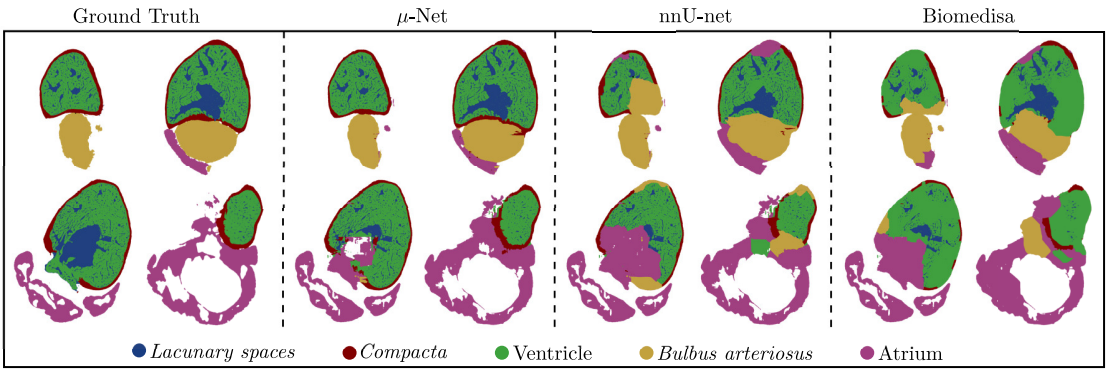


Figure 6.2: Visualization of ground truth, μ-Net and the comparison methods results. Each image represents a slice taken from the same quartile of slices within a single 3D stack

limited number of samples (less than 3) and the high computational demand. On the contrary, Biomedisa's performance was poor due to its use of a 3D U-net that standardized each sample size, leading to a reduction in resolution and distortion of shapes. A visual inspection of the results is shown in Fig. 6.2 where we compared from the left to the right: the manual segmentation (labels), the predicted segmentation of μ-Net, and the results of the comparison methods. We can observe that μ-Net demonstrates excellent segmentation of small and medium *lacunary spaces*, while it tends to confuse larger ones with the background. As for other anatomical regions, μ-Net notably outperforms the comparison methods. In accordance with our hypothesis, these results highlight how the strategy underlying our framework, which decomposes the problem into simpler sub-problems before utilizing ensemble techniques, is more effective than tackling the problem as a whole, as seen in the cases of nnU-Net and Biomedisa. Finally, our workflow can speed up the processing of new scans and is adaptable for additional segmentation tasks. The models are primed for training on more goldfish hearts or for transfer learning on other high-resolution μ-CT tasks. The data from our automatic segmentation can be quantitatively analyzed like manually segmented data.

6.6 Conclusions

We introduced μ-Net, an novel workflow built on Segnet for the semantic segmentation of biological μ-CT images. Training was performed on a new dataset, encompassing the collection of manually segmented 3D μ-CT scans of goldfish hearts. The experiments showed that μ-Net enhances efficiency and dependability of image segmentation techniques by treating each class separately. μ-Net significantly outperformed existing methods, setting the stage for a more precise and automated examination of goldfish heart morphology and potential diseases, thus facilitating translational studies.

Chapter 7

The Dilemma of Accuracy in Bankruptcy Prediction: A New Approach Using Explainable AI Techniques to Predict Corporate Crises

In today's highly dynamic and hyper-competitive markets, firms do not have any buffer of inefficiency, and wrong decisions can silently compromise corporate equilibrium, undermining survivorship. Since Altman's 1968 contribution, researchers have been striving for decades to implement new models and techniques for predicting corporate crises, but no superior model has emerged among the others. In this context, our work joins the stream of research that leverages Artificial Intelligence for corporate default prediction with a novel approach based on a mix of techniques, enabling it to achieve higher accuracy in anticipating crises compared to all previous contributions. We investigated models with sequence lengths that were both fixed and variable, and we chose a variable sequence length model that boasted an accuracy of 0.85 and a ROC-AUC score of 0.927, which is the most optimal predictor, superior to existing tools of bankruptcy prediction. Our results provide significant implications as they offer a noteworthy instrument for preventing business crises. Moreover, it shows key aspects that a manager should consider in order to guarantee the survival of their company.

7.1 Introduction

The topic of predicting corporate crises is very hot nowadays. The statistics revealing the number of businesses that cease to exist every year worldwide, especially startups, as well as the average lifespan of companies, are impressive [242]. Over the past decades, millions of companies have filed for bankruptcy with detrimental consequences to society and the economy as a whole [243]. Many papers have investigated the financial and non-financial causes of failure as it represents a multidimensional phenomenon [244], while others have focused attention on the process that brings firms to bankruptcy [245]. Bankruptcy hurts not only firms but also the personal lives of entrepreneurs [246] and employees [247, 248], the efficiency of the supply chain [249], society [250], and the global economy [251]. Thus, bankruptcy prediction represents a highly significant corporate activity with a growing awareness among businesses. No firm can reduce the risk of bankruptcy to zero, but efforts can be

made to minimize it as much as possible. Firms need to manage the risk of crises very well and choose the best model of bankruptcy prediction to discern the nature of the decisions and anticipate any negative consequences of the crises.

In this context, the research community contributed to predicting business crises, and several models and techniques have been developed to support executives of companies in managing potential collapses. Academics have paid attention to this topic since the 1960s, and some models for predicting corporate crises stand as milestones in this field of study [252, 253, 254, 255]. In particular, the noteworthy contribution of [252] laid the groundwork and has been an inspiration for numerous subsequent studies on the topic. Following Altman, over time, researchers and practitioners have developed crises prediction methods and techniques that go beyond the traditional mathematical and statistical models with the purpose of predicting the bankruptcy of firms quickly and more accurately [256]. In addition to traditional methods, since the 1990s, Machine Learning (ML) models such as decision trees, neural networks, and Support Vector Machine have been extensively applied in this field of research as tools to predict the bankruptcy of firms [257] and provide managerial support for businesses. Recently, Deep Learning (DL) has emerged and gradually developed into a powerful technique for a wide range of applications in economics studies [258] and in bankruptcy prediction. Indeed, a neural network can learn from data automatically, so it can be trained to recognize patterns, classify data, and forecast future events [259, 260, 261, 262, 263, 264, 265, 266]. A particular Neural Network architecture called Recurrent Neural Network (RNN) can process information with a temporal dimension in the form of time series. Such RNNs can identify data patterns that evolve by considering a set of features measured at subsequent time steps [264]. Some authors have used the machine learning algorithm to predict bankruptcy [256, 267, 268] and more in general there is extensive literature investigating bankruptcy prediction using Artificial Intelligence (AI) techniques [269, 270, 271, 272, 273, 274, 275, 264, 257, 265, 276, 277, 268, 266, 278].

All these contributions interestingly demonstrate that bankruptcy prediction models based on AI techniques outperform classic mathematical and statistical models, even considering different industries and both cross-country and single-country studies. Therefore, AI can be very important for the survival of businesses. The importance of these models based on computational solutions is additionally demonstrated by the fact that since their first use in this area of investigation, researchers have never stopped developing new and more accurate solutions. In a context where many models of bankruptcy forecasting have been developed [279], we questioned what drove researchers in this ongoing pursuit, reaching a point where to date it is difficult to find a universally accepted and standard model to prevent corporate crises. Certainly, one of the reasons is linked to the fact that previous models utilizing this research approach employ various AI techniques, each of which naturally yields different results. Furthermore, prior works have certain limitations, such as using short-term temporal sequences or a limited number of observations.

Within this strand of research, our innovative work leverages cutting-edge methodologies such as deep recurrent neural networks and explainable AI. This choice is driven by the rapid evolution of artificial intelligence techniques in practical applications. Thus, our aim is to develop a highly precise corporate crises prediction model that surpasses previous versions rooted in the forefront of technological advancements. Major challenges in bankruptcy prediction concern sample size, quality of input data, and the choice of time series length, which is critical in the context of historical data of companies. In contrast with previous state-of-the-art approaches, this work stands out by utilizing a large sample (4,172,046 observations)

over a broad time span (from 2012 to 2021) for building a prediction model able to process variable length time series of financial indexes. Decisions made by such a model are then interpreted by using an explainable AI methodology. This is very important also to understand the contribution of each variable to the bankruptcy prediction and provide useful managerial implications. Another unique aspect of our work involves the utilization of data extracted from the Orbis database by Bureau Van Dijk. Orbis, which is Moody's analytics company, represents the most extensive database of financial and business information across Europe as it contains detailed and well-harmonized accounting, financial, and business information for firms. Moreover, it is the world's most powerful comparable data resource on private companies.

Our sample also consists of small and medium-sized enterprises (SMEs). This is important because SMEs have a key role in economic growth as they represent 99% of businesses in Europe. Hence, their bankruptcy prediction could be very relevant both for local and global economies. Additionally, SMEs are businesses that are particularly affected by asymmetric information problems within financial markets [280, 281], hence suffering more during difficult periods such as the recent COVID-19 crisis [282]. The coronavirus emergency amplified the attention of the media and business community to the precarious situation for businesses, highlighting once more the negative consequences of financial crises.

Our findings demonstrate that the artificial techniques implemented lead to greater accuracy in predicting business crises compared to all previous research efforts, even those utilizing long-time sequences or a high volume of observations. Furthermore, our results highlight the key variables that need monitoring to prevent business crises. Finally, with this contribution, we aim to open a new avenue of research that starting from the use of these techniques can implement models incorporating non-accounting variables such as governance, ESG, and information from companies' social media to prevent business crises.

The paper is structured as follows: the second section presents the theoretical framework and research gap, while the third section outlines the materials and methods used. The fourth section provides results, with a focus on explainability in the fifth section. The sixth section offers conclusions, discussions, and the managerial contribution of the work.

7.2 Theoretical Framework and Research Gap

Bankruptcy prediction in the business community is a field of growing interest, particularly since the financial scandals at the beginning of 2000 and even more the global financial crises of 2008, after which the number of studies on the topic has greatly increased, becoming a significant area of study within the field of management and corporate finance [283]. However, investigations into predicting corporate crises date back a long time. Indeed, a significant amount of research has focused on the prediction of corporate financial distress since Altman's influential introduction of his bankruptcy prediction model in 1968. Altman found that predicting bankruptcy can be done by using discriminant analysis and liquidity, profitability, productivity, leverage, and asset turnover ratios to establish the so-called Z-score. The Z-score is calculated using five financial ratios assessing a company's liquidity, profitability, efficiency, solvency, and turnover, thereby providing a comprehensive view of its financial health. A higher Z-score indicates better financial strength and stability, while a lower score suggests higher bankruptcy risk. Altman pioneered a field of studies that aimed to describe the financial variables that lead to the default of companies during different years. After

Altman, other models have been particularly influential. In [254], for instance, the authors investigated the bankruptcy prediction for American firms from 1970 to 1976 using the logistic regression model and nine financial ratios . Begley et al. (1996) investigated bankruptcy in three of the major stock markets in the U.S. (NYSE, AMEX, and NASDAQ), observing that the Altman model performed better than the Ohlson model for data from 1980 to 1989 [284]. Some other papers went beyond the use of financial variables. Authors in [285], for instance, observed financial distress in the Hong Kong Growth Enterprise Market from 2000-2010 and found that a logistical model that used financial, non-financial, and macroeconomic variables over-performed the other models [285]. Liang et al. in [286] used the combination of financial ratios and corporate governance indicators for bankruptcy prediction, finding that combining such ratios and indicators is more efficient than using only financial data.

The main techniques employed following the seminal contribution of Altman and the subsequent influential contribution mainly aim to distinguish firms into bankrupt or non-bankrupt and include multivariate discriminant analysis, logit, probit, and neural networks [287]. However, analysis of the most recent literature on anticipating business failure reveals a wide array of models employed that extend far beyond the ones observed by the review of Bellovary. This diversity stems from advancements in statistical techniques and information technology over time that have tried to establish more accurate bankruptcy prediction models compared to earlier attempts aiming to contribute to firms in their crisis prevention decisions. In particular, information technology-based techniques have been developed since the 1990s, after which neural networks have been the most widely used methods to predict corporate crises. Obviously, in the initial stages of using this technology, the models adopted were less complex than the more recent ones. Indeed, in the 1990s, some artificial intelligence models were developed [288, 289], followed by subsequent models in the succeeding decade. For instance, in the 2000s, Atiya [259] shows that neural network (NN) models performed well in predicting bankruptcy for Credit Risk. Authors in [268] investigated bankruptcy prediction in Korea by using support vector machines (SVM) and a back-propagation neural network (BPN). They found that the SVM performed better than BPN when the training set size got smaller. Tsai and Wu [266] in another study show that single neural networks perform better than multiple neural networks. Nanni and Lumini in [267] reported that the ML models outperformed the traditional statistical analysis methods in predicting bankruptcy in three credit markets (Australian, German, and Japanese). In the following decade, there was an increase in studies on the prediction of corporate crises, particularly considering that the negative effects on businesses caused by the global financial crises exacerbated and underscored the gravity of this phenomenon. Barboza in [256] show that ML models have a higher capacity for predicting Bankruptcy compared to ANN, LR, and MDA models for data from 1985-2013. Also, Mai et al. used ML models with non-financial variables to predict bankruptcy [265]. Jabeur et al. in [263] studied the bankruptcy prediction of French companies from 2014-2016 using fuzzy convolutional neural networks (FCNN). They concluded that FCNN performs better than neural networks, logistic regression, partial least square discriminant analysis, support vector machines, or discriminant analysis.

As a matter of fact, the use of Artificial Intelligence provided a higher explanatory power in crises prediction in comparison to a deterministic model based on financial ratios. Considering this superiority, a fight for the best technology-based approach started. The vast amount of research has led in recent years to the use of highly advanced artificial intelligence techniques to predict corporate crises, leading up to the models characterizing this research trend in the last two years. In a recent study, Kim et al. [264] used textual sentiment analysis

(BERT) to predict bankruptcy. They found that BERT-based analysis performed better than dictionary-based analysis and Word2Vec-based analysis combined with a convolutional neural network for data from 1995-2020. Yang et al. in [290] show that the HDNN algorithm was a good solution for higher dimensional corporate credit risk during the entire sample period (from 1 January 2009 to 31 December 2019). Chen et al. in [291] investigated the corporate bankruptcy prediction of U.S. firms in the period from 1994 to 2018 by including the text-based communicative value of annual reports in four machine-learning models. They reported improvements in the performance of XGBoost and Random Forest models. They confirmed the importance of text-based annual reports for banks' corporate loan underwriting decisions. Charalambous et al. in [260] observed the U.S. public firms for data from 1990–2015 and found that structural models like the Black-Scholes-Merton and the Down-and-Out option models perform better than a standard neural network. They reported that the neuro-structural model performed better than a sample neural network. Dube et al. in [262] used Artificial Neural Networks (ANN) to investigate the financial distress models on the Johannesburg Stock Exchange (JSE) between 2000–2019. They found that ANN had good accuracy and predicted financial distress for up to five years for the financial services and manufacturing companies. Kim et al. in [264] found that from January 2007 to December 2019, the recurrent neural network (RNN) and long short-term memory (LSTM) increased the performance of bankruptcy prediction compared to the use of logistic regression, support vector machine, and random forest methods. They concluded that the RNN and LSTM methodologies cannot detect the importance of each explanatory variable for bankruptcy prediction. Elhoseny et al. in [292] in another study found that combining the DL and Whale optimization algorithm (AWOA-DL) overperformed the TLBO-DL, DNN, LR, and RBF Network models in predicting bankruptcies and assessing credit risk. du Jardin in his study [261] found that the convolutional neural network CNN performed better at corporate bankruptcy and financial failure compared to the traditional model.

Despite the attention on crises prediction from both academics and the community in general, no superior model has emerged among the others to forecast corporate crises. Every year, researchers develop models that aim to be more capable of predicting corporate failure better than others, but these efforts have not led to the development of a universal model used by all companies today. Veganzones and Severin in [293] highlight that the way researchers design experiments is a crucial factor since it can have a significant impact on the outcomes. They believe that differences among the key elements (definition of failure, sample size, prediction methods, variables, evaluation metrics, and performance) of such kinds of analysis are at the core of different outcomes.

Thus, here comes into play the need to develop a model that can encompass as much information as possible to predict corporate crises more accurately than prior contributions. With this in mind, we try to fill this research gap by investigating bankruptcy prediction using a mix of explainable artificial intelligence techniques among the most recent and advanced solutions. Indeed, we use ML and DL models at the same time, while other papers use just one technique. In doing so, we pay a lot of attention to variable selection and data characteristics, using the Orbis database provided by Bureau Van Dijk. Moreover, Italian studies on the topic are very few and use a shorter timespan or a lower sample size. The Italian context is well suited for this analysis as in Italy there are significant differences among provinces in terms of institutional development. Moreover, our work differs from previous studies by using data about firms that include the Covid-19 period.

7.2.1 Artificial intelligence

Neural networks

Neural Network (NN) is one of the most popular DL methods. NN is inspired by the human brain function and structure, by using interconnect nodes called neurons in a layered structure that resembles a graph. In each layer, several neurons use the outputs of all neurons in the previous layer as inputs, such that all neurons interconnect with each other through the different layers. Each neuron is typically assigned a weight that is adjusted during the learning process by a decrease or an increase. A neural network can learn from data, so it can be trained to recognise patterns, classify data, and forecast future events. It breaks down the input into layers of abstraction and defines a wide range of models characterised by an extremely high number of parameters, especially in the so-called deep models, which can be based on supervised or unsupervised learning paradigms. In the supervised learning approach, a neural network employs a sample of data consisting of corresponding inputs to outputs. By manipulating input parameters, the neural network finds the best non-linear predictive model that generates output consistent with the sample. This model has generalisation properties, which means that it can be used to predict an output when we add a new set of inputs that the model has never seen. This approach is typically used to solve regression and data classification problems. Instead, the unsupervised learning approach provides the computational model with a sample that does not include output information. In this case, it identifies statistical structures within the sample, such as correlations or associations, producing an output that describes such relationships. The unsupervised strategy is typically applied to solve clustering problems and it is a useful tool to assess industry similarities and data analysis in the financial field. There exist different types of Neural Networks.

Recurrent neural networks

A Recurrent Neural Network (RNN) is a neural network that adopts the following principle: it processes sequences by iterating through the sequence elements and maintaining a state containing information relative to what it has seen so far. In effect, an RNN is a type of neural network that has an internal loop. Unlike traditional neural network algorithms which are limited in their ability to handle ordered data, such as time-series data, music, or sentences, RNNs can manage such data by exploiting these loops in their structure. The state of the RNN is reset between processing two different, independent sequences, so you still consider one sequence a single data point: a single input to the network. What changes is that this data point is no longer processed in a single step; rather, the network internally loops over sequence elements. There exist different ways to implement a Recurrent Neural Network. The simplest one is Simple RNN, but it is never used in practice because it suffers from the vanishing gradient problem, as you keep adding layers to a network the network will struggle to remember information seen many timesteps before, so long-term dependencies are impossible to learn. The LSTM and GRU layers are designed to solve this problem. Indeed, they work similarly: they save information for later, thus preventing older signals from gradually vanishing during processing.

SHapley additive explanation

We employed SHAP to elucidate the interpretability of neural network models. SHAP values, derived from cooperative game theory, were used to quantify the contribution of individual features to the model's predictions. By calculating Shapley values for each input feature, we discerned their impact on the neural network's output, thereby enhancing the interpretability of complex model decisions. This approach facilitated a comprehensive understanding of the neural network's behaviour, offering insights into the relative importance of features in driving predictions, thereby contributing to the transparency and interpretability of our model.

7.3 Materials and Methods

7.3.1 Data and Variables

The data used in this study were collected from the Orbis European database by Bureau Van Dijk (BVD). We collected annual accounting data from 2012 to 2021. The dataset counts about 4,172,046 Italian firm-year observations, 66,226 of which experienced bankruptcy. In this dataset, each financial index is collected for 10 years from 2012 to 2021. From such a dataset, we extracted sequences of annual accounting data for different periods considering period length from 2 to 9 years. The dataset consists of 24 indexes of financial performance: intangible assets, non-current plant and equipment, inventories, current assets, trade receivables, cash and cash equivalents, total assets, equity, share capital, long-term indebtedness, non-current liabilities, current liabilities, debts, trade payables, total value of production, revenue sales and services, operating profit (EBIT), financial income, financial charges, total taxes, profit/loss for the year (net profit), inventory rotation, cash-out times (days), and payment times (days). In this dataset, we distinguish between two classes: Class 0 is the set of firms for which bankruptcy occurred (66,226 firms), while Class 1 is the set of firms in good health (4,105,820 firms). This dataset is characterized by unbalanced classes, meaning that the number of firms in Class 1 is significantly lower than the number of healthy businesses.

7.3.2 Preprocessing

Along with the need for large amounts of data, DL models, and in particular neural networks, need to work with data values distributed in a well-defined range and with data balanced among the classes. For this reason, we first normalized each variable in the range [0, 1] and then re-balanced the classes by generating synthetic data samples (firms). We augmented Class 0 by using the Synthetic Minority Over-sampling Technique (SMOTE) [294]. Thanks to this technique, new synthetic instances are created starting from existing ones that are in the minority class and small perturbations are added to the new data points.

7.3.3 Experiments and Models

In this section, we discuss the development of optimization of an RNN model for predicting bankruptcy. We conducted experiments on four RNN architectures by considering different numbers and types of recurrent layers. We selected the best-performing model over these in terms of accuracy and subsequently performed an automated model selection process with Cross-Validation to ensure the robustness and generalization capabilities of the selected

model across different data partitions. Finally, we used an Explainable AI method called SHAP (Shapley Additive explanations) [279] to explain the contribution of each financial index to the prediction.

7.3.4 Training and Hyperparameter Optimization

In the development of bankruptcy prediction models, we implemented and evaluated a set of four models, each characterized by a distinct set of hyperparameters. Our dataset comprised 84,000 actual data instances augmented by 26,000 synthetically generated samples using SMOTE. The dataset was partitioned into training (80%) and test (20%) sets, with the test set being exclusively constituted of real (non-synthetic) samples. All the models shared the following configuration. First, as a loss function, we used Binary Cross-Entropy, suited to the binary classification nature of the bankruptcy prediction task. Second, the Metric Accuracy is defined as the proportion of correct predictions out of the total predictions made. Finally, we used Adam as an optimizer. The architectural details about the models used follow:

- **Model 1:** 2 subsequent Long Short-Term Memory (LSTM) layers of 128 and 64 blocks respectively, followed by 3 fully connected layers of 64, 128, and 64 neurons with dropout.
- **Model 2:** 3 subsequent Conv1D layers of 64 blocks respectively, followed by 3 Batch-Normalization of 64 blocks, and finally GlobalAveragePooling1D of 64 blocks.
- **Model 3:** 2 subsequent LSTM layers of 256 and 128 blocks respectively, followed by 2 fully connected layers of 256 and 128 neurons with dropout.
- **Model 4:** 2 subsequent Gated Recurrent Units (GRU) of 128 followed by 2 fully connected layers of 128 and 128 neurons with dropout.

Model 1 performed best with the best accuracy in each test. We trained it several times by varying the input data, exploring models with fixed sequence length (from length 2 to 9) as well as models with variable sequence length. To properly evaluate its performance, we used the K-fold Cross Validation technique with K=10.

7.3.5 Metrics

In this work, different models were created by varying the input data. As evaluation metrics, we used Accuracy, which is the ratio between the number of correct predictions and the total number of observations, and ROC-AUC.

7.4 Results

This section reports the performance of Model 1 in two cases: 1. When the model is trained and tested on fixed sequence lengths; 2. When the model is trained and tested on sequences of variable lengths.

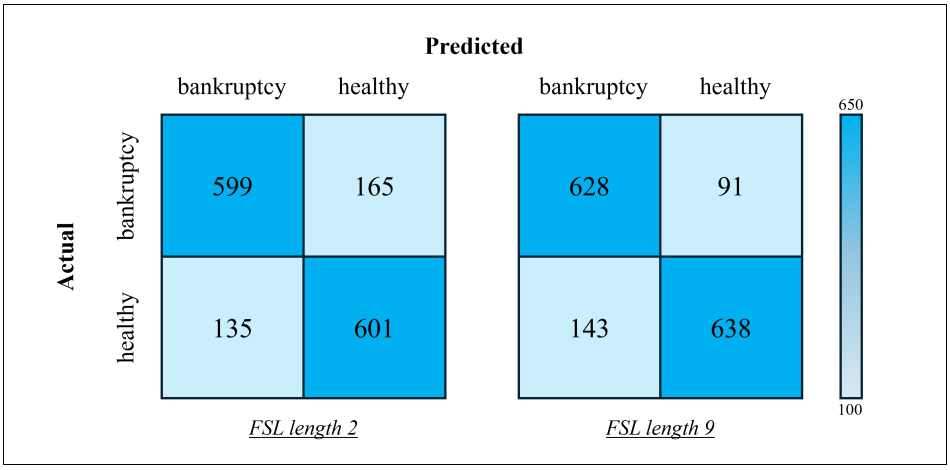


Figure 7.1: Confusion matrix of the fixed length best models for length 2 and length 9 of the sequence.

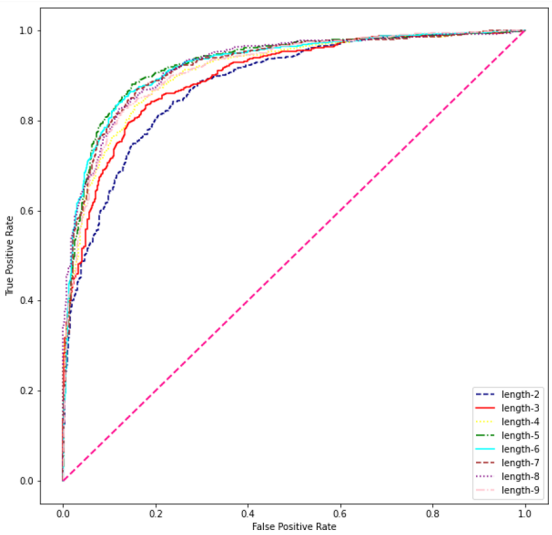


Figure 7.2: ROC curves for the fixed sequence length models for all the sequence lengths.

7.4.1 Fixed Sequence length models

As the first report, we want to compare results between models in which input data are only data with fixed lengths. We will refer to these as fixed sequence length (FSL) models. Each of these models is specialized for input sequences of fixed length, so a model whose input data length is 2 learns from data with complete information only for 2012 and 2013, and accordingly a model whose input data length is 9 learns from data with complete information from 2012 to 2020. Each FSL model have been trained and tested on 15000 firms. Specifically, 90% of data were used for train and validation and 10% of samples were used for testing its performances. Training and testing were performed by using a cross validation approach with 10 iterations.

We can notice that as the length of the sequence increases (i.e. the information becomes more complete), the performances of the neural networks increase. This difference is more evident between the "low-length model" and the "high-length model". Indeed, as shown in the confusion matrices of Figure 7.1, and the ROC curve in Figure 7.2, the best FSL model

with length 9 is much more accurate than the length 2 model. Detailed results are shown in Table 7.1.

Input Sequence Length	Accuracy	95% C.I.	ROC-AUC
2 (2012-2013)	0.802	[0.778, 0.794]	0.880
3 (2012-2014)	0.826	[0.792, 0.816]	0.897
4 (2012-2015)	0.836	[0.799, 0.821]	0.910
5 (2012-2016)	0.862	[0.831, 0.846]	0.926
6 (2012-2017)	0.859	[0.844, 0.853]	0.924
7 (2012-2018)	0.851	[0.823, 0.838]	0.924
8 (2012-2019)	0.850	[0.832, 0.845]	0.924
9 (2012-2020)	0.844	[0.826, 0.841]	0.914

Table 7.1: Results of the FSL models on test sets. Each model has been tested by using 1500 samples of the same sequence length. The table reports the accuracy of the best model (over cross-validation) on the bankruptcy binary classification followed by the confidence interval of accuracy over the results of cross-validation and the ROC-AUC score of the best performing model.

7.4.2 Variable Sequence length models

Now we want to compare results between models whose input data are mixed, so as to have more flexible models that can perform with sequences of different lengths, that we refer to as VSL models. Each VSL model have been trained and tested on a different dataset where 90% of data were used for train and validation and 10% of samples were used for testing. Training and testing were performed by using a cross validation approach with 10 iterations. In Table 7.2, we highlight the performance of the best VSL model, which is the one trained on sequences with complete information and time series which comprises data in the ranges: 2012-2017, 2012- 2018, 2012-2019, and 2012-2020. In Figure 7.3 we show the ROC curve to compare the results of the different VSL models and the confusion matrix of the best performing model on the test set. It is possible to notice how the model trained with sequences of length 3, 4, and 5 is the worst model in this case, and this can be explained by the fact that it is the model with less complete information than all the other models, so it is more likely to be wrong in its predictions. In Figures 7.4 we show the results of the explainability method over the best performing VSL model. It is possible to notice that the

Number of Firms	Input Sequence Length	Accuracy	95% C.I.	ROC-AUC
110k	all lengths	0.835	[0.829, 0.832]	0.912
55k	6-7-8-9	0.850	[0.828, 0.842]	0.927
43k	3-4-5	0.834	[0.816, 0.828]	0.908
54k	3-5-7-9	0.836	[0.812, 0.827]	0.910

Table 7.2: Results of the VSL models on test sets. Each model has been trained and tested by using the number of firms in the first column. The table reports the accuracy of the best model (over cross-validation) on the bankruptcy binary classification followed by the confidence interval of accuracy over the results of cross-validation and the ROC-AUC score of the best performing model.

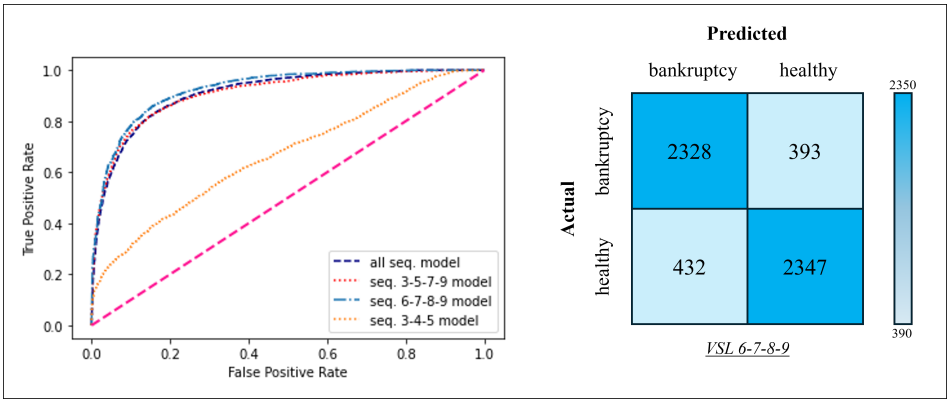


Figure 7.3: ROC curve comparison over the VSL models (on the left) and confusion matrix of the best VSL model on the test set.

most important feature in bankruptcy prediction is the Inventory, and the firms with higher inventory have less chance of financial distress. This result can be explained because firms that have an important purchase-transformation-sale cycle with high stock levels are able to work well and survive within the market

Moreover, the right inventory management system can save a company time and money. We also observe that a firm with higher Net Profit has a lower probability of financial distress. This result indicates that those companies that succeed in having revenues significantly higher than costs presumably are less subject to face situations of difficulty. When the value of Inventory Turnover is high the chance of financial distress increases, and this can be explained by the fact that if this ratio is high in difficult periods the company might not have the stock necessary to supply the production department or the points of sale in a timely manner. Moreover, a higher turnover of inventory could make the company riskier. Another important variable is the Share Capital. Higher values are related to higher default probability, indicating that the economic contribution of shareholders is not sufficient to guarantee proper resource management. Despite this, we interestingly notice that the higher the Equity the lower the financial distress. Typically, firms with a capital structure having prevailing eq-

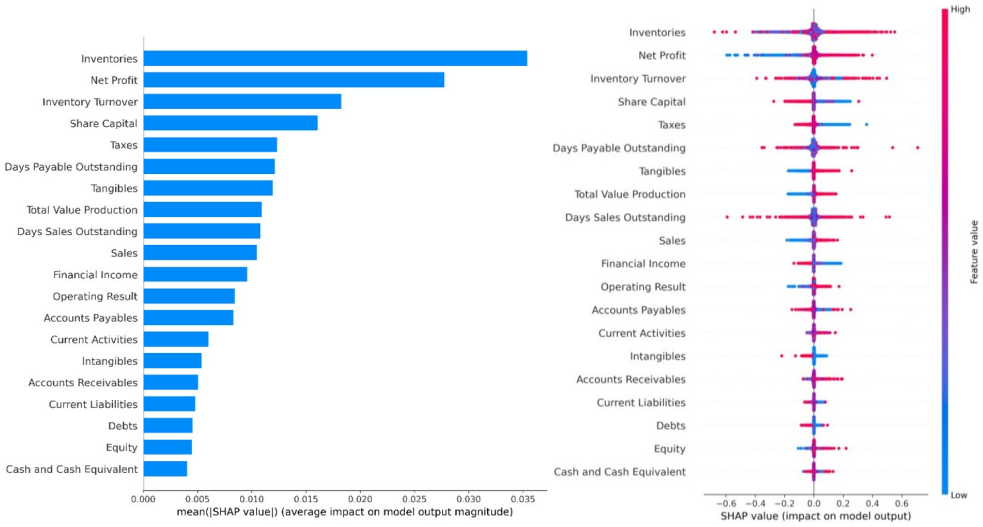


Figure 7.4: Global feature importance plot (right) and Global summary plot (left).

uity are stronger because throughout the years the company has achieved high profits which became reserves. It seems that reserves play a key role in survival more than the liquidity invested by shareholders in the form of share capital. Additionally, when Taxes is low this means that the company was good at leveraging the tax shield and saving cash to invest. The higher the value of Days Payable Outstanding the higher the probability of financial distress, and this can be explained by the fact that taking a lot of time to pay bills and invoices of trade suppliers could make them dissatisfied with receiving the money with so much delay, leading them to undertake adverse conduct towards the company. The higher the Tangibles the lower the financial distress, as firms having higher values of physical and measurable assets have more resources to be competitive in today's dynamic markets. More assets are also linked to a higher availability of collateral, which is useful to get loans and receive the cash necessary to pay debts to prevent bankruptcy. The higher the Total Value Production the lower the financial distress. This means that businesses that are able to sell as much and add value to production costs are less likely to fail. Similarly, the higher the Sales the lower the financial distress, because firms with no difficulty selling products or services undoubtedly have a competitive advantage in the market. Additionally, the higher the Days Sales Outstanding the higher the risk of bankruptcy, because if the average number of days taken by a firm to collect payment from their customers after the completion of a sale is excessive the company might not have cash buffers both for current activities and new investments that are crucial to survive. The lower the Financial Income the higher the financial distress, indicating that the revenue generated by the cash invested in financial investments could play an important role. This interesting finding suggests that non-operating activities also matter. The higher the Operating Result the lower the financial distress. Operating results are an essential metric for investors to understand how much money a company is potentially capable of generating from its core business; moreover, it shows the ability to generate profit from operating activities that can remunerate both debtholders and shareholders. The higher the Accounts Payables the higher the probability of default, if the amount of money that the company owes to suppliers is high, suppliers could be dissatisfied and might decide to interrupt the supply or not grant deferred payment when the firm needs it. We also observe that the higher the Current Assets the lower the financial distress, because a firm that operates in the market using many resources could potentially have a competitive advantage. The lower the Intangible Assets the higher the financial distress. The intangibles allow growth opportunities in the market and increase corporate value. In the current context, patents, managerial skills, and know-how are essential to survive. The higher the Accounts Receivables the lower the financial distress, meaning that the company that converts accounts receivables to cash faster can use such cash for growth purposes. The higher the Current Liabilities the higher the financial distress; we can explain this result by the fact that if a firm needs to pay debts within the next 12 months it could need more money in the short run and if the cash is not available it could suffer from financial constraints. Moreover, as expected, the higher the Debts the higher the financial distress, as the most indebted companies may have difficulty coping with their debts and subsequently go into crises. Finally, Cash is negatively related to default, as firms with more liquidity have greater possibility to make investments in order to grow and to pay their debts of any nature, avoiding corporate crises.

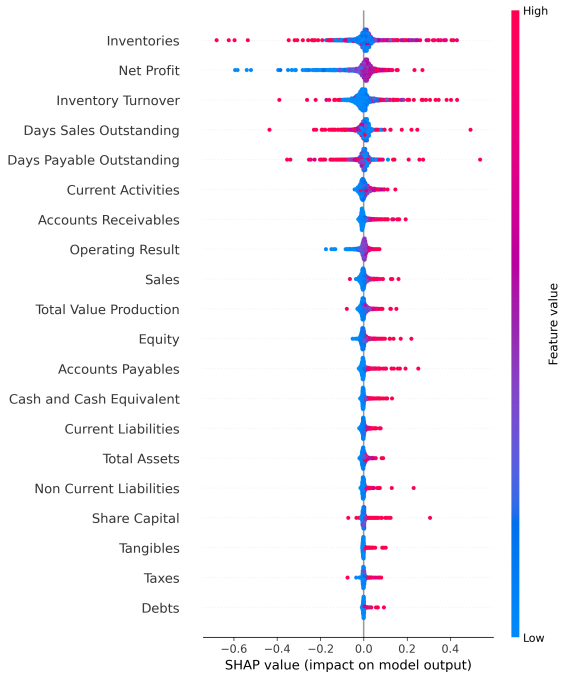


Figure 7.5: Global summary plot for time step 5 (2017).

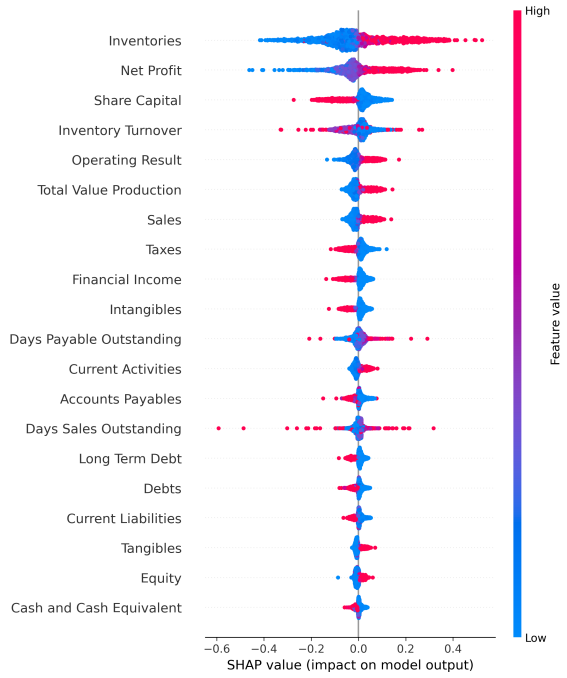


Figure 7.6: Global summary plot for time step 6 (2018).

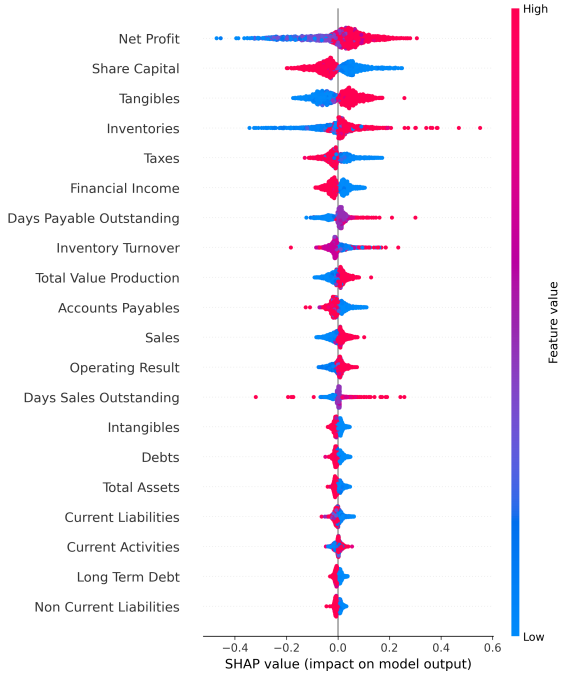


Figure 7.7: Global summary plot for time step 7 (2019).

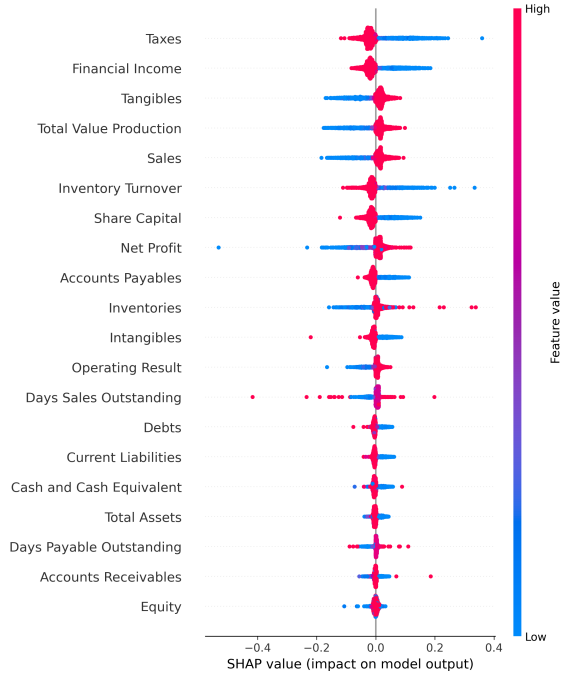


Figure 7.8: Global summary plot for time step 8 (2020).

7.5 Explainability

In this section, SHAP results will be reported and interpreted. SHAP was used to interpret the output of the best model with a variable sequence length. Interpretation is very important in order to get practical implications from the research conducted. For each SHAP plot in this section, we show the top 20 variables that most affected the model's prediction.

7.5.1 Global Feature Importance

Figure 7.4 (right) shows a waterfall plot of absolute mean SHAP values, reporting the average importance of each financial index in the model, and hence the contribution of each to the predictions, evaluated by using SHAP. Indexes are reported in order of importance. For example, the model was strongly influenced by "Inventories" and "Net Profit", moderately influenced by "Tangibles" and poorly influenced by the others below. The feature importance plot is useful but contains no information beyond global importance. As shown in Figure 7.4 (right), higher values of Profit/Loss Inventories are associated with positive SHAP values, meaning that they will increase the prediction towards 1 (healthy firms). Moreover, lower values of the variable are associated with negative SHAP values, meaning that they will decrease the prediction towards 0 (bankruptcy). Conversely, for the Taxes lower values are associated with positive SHAP values, increasing the prediction towards 1. Higher values of the latter index, instead, are associated with negative SHAP values, meaning they will decrease the forecast towards 0, which means bankruptcy.

7.5.2 Summary plot for each time step

We analysed the impact of variables at each different time step. As can be seen, each time step ranks features. Indeed, as we said before, "Inventories" is the most impacting feature at a global level, and this is confirmed by data of time steps 2017-2018 (Figure 7.5, 7.5). As shown by time step 7 (Figure 7.7), the feature "Net Profit" is the most impacting one. This feature indicates the net profit of the company, and its contribution is directly related to the output of the model: it means that as the value of the feature increases, the Firm is pushed towards a status of health. In other words, a company whose net profit assumes a high value tends to be in financial health and vice versa. In time step 8 (Figure 7.8), instead, the feature "Taxes" occupies the first position. As noted previously, the contribution of this item is inversely related to the value of the prediction: as the value of the feature increases, the output tends to be negative. In other words, a company with a lot of taxes to pay is more likely to fail. An interesting thing to discuss is that the feature "Taxes" has the most impact in the last time step (2020) see Figure7.7. Indeed in 2020, the first year of the COVID-19 pandemic, the amount of debt (for most companies) drastically increased due to a lack of income and higher taxes, Each time step ranks the features. Indeed, as noted, "Inventories" is the most impacting feature at a global level, and this is confirmed by data of time steps 5-6 (2017-2018). The story is different for time step 7-8 (2019-2020), in which other features are more influential. As is evident from the time step 7 plot (Figure 7.7), the feature "Net Profit" is the most crucial. This feature indicates the net profit of the company, and its contribution is directly related to the output of the model: it means that as the value of the feature increases, the contribution becomes positive. In other words, a company whose net profit assumes a high value tends to be in financial health and vice versa. In time step 8, instead, the feature

"Taxes" occupies the first position. Here again, its contribution is inversely related to the value of the prediction: as the value of the feature increases, the output tends to be negative. In other words, a company with a lot of taxes to pay is more likely to fail. An interesting thing to discuss is that the feature "Taxes" has the most impact in the last time step(2020). Indeed in 2020, the first year of the COVID-19 pandemic, the amount of debts (for most companies) drastically increased due to a lack of income and heavier taxes, resulting in a large number of bankruptcies. These results open new research prospects as they suggest investigating business crises situations before, during, and after the COVID-19 crises using the techniques we have employed. Surprisingly, in the last two time steps the feature "Inventories" is not even in the top five impacting features. This proves the fact that SHAP is also useful at the local level to understand better what happens time step after time step.

7.6 Discussion, Managerial Contribution, and Conclusions

In this work, we presented a DL approach to address the Bankruptcy Prediction problem utilizing Recurrent Neural Networks. Extensive experimentation and a review of the literature on ML and DL models underscored the significance of our chosen methodology. Notably, our best model outperformed other architectures, and we explored fixed and variable sequence length models, ultimately selecting a variable sequence length model with an accuracy of 0.85 and a ROC-AUC score of 0.927 as the optimal predictor. Leveraging the SHapley Additive Explanations method, we elucidated the impact of each feature on model predictions, enhancing the interpretability of the results and that which is important to obtain managerial implications.

Our findings demonstrate the viability of RNNs in bankruptcy prediction, offering a valuable tool for decision-making in financial contexts. The use of a long time-series also allows understanding how many years before the crises there are signals of possible financial distress. We have identified potential variables with a higher predictive power so as to prevent corporate financial crises. In particular, we observed that firms with higher inventories, net profit, sales, value of production, and operating results have a lower probability of distress, being successful in the market. We also find that management of net working capital is also important as having too many trade debts (or paying such debts with delay) could make suppliers dissatisfied, and having too many receivables (or collecting such receivables with a delay) could bring cash difficulties. We also find that having reserves (and so positive net profits throughout the years) is more important for survival than the amount of money provided by shareholders in terms of share capital. It is also interesting to notice the crucial role of intangibles in survival. Indeed, firms with non-tangible assets such as patents, skills, know-how, etc. have key resources to pursue growth opportunities and take on market competition.

Concerning potential practical implications, the first important implication of the research is to provide a model/tool that assesses a possible business crisis in advance through a monitoring and alert system. In this way, it is possible to provide indications on the critical business areas and exploit an easy-to-use management support tool. Another important application is for policymakers who can use our research's output as a tool to combine with current credit-scoring systems and to assess the effectiveness of the new corporate crisis reforms that are upcoming in many European countries. This would favour public and private collaboration. In addition, it would support the current control system of the business crisis,

reducing and making the assessments more efficient. In particular, in Italy, such a model would make it possible to improve the "assisted settlement of the crisis," a system introduced in Italy through Legislative Decree 14/2019. More generally, the model can more effectively support the entire Italian control system with regard to managers and the Corporate Crisis Settlement Organisation (the Italian OCRI). The timeliness of the alerts could even reduce the crisis reports sent to the OCRI, which occur in the presence of ex-post financial data or events that reveal a status of insolvency. Moreover, our model can be used by Italian financial institutions to elaborate the existing Indices of Reliability for companies, going beyond the techniques currently employed. A further social impact of the model is its greater simplicity and effectiveness in forecasting crises. In fact, with this model, it is possible to avoid/mitigate the negative consequences of socio-economic effects from business failure, such as less time spent by the courts on insolvency proceedings, lower unemployment, fewer social consequences for entrepreneurs whose project fails, fewer psycho-emotional reactions, and personal traumas that often characterize "failed" entrepreneurs. Therefore, the potential practical/technological applications are considerable, and the tool can be available to companies in order to reduce corporate crises with important socio-economic benefits. The tool can also be made available to banks, public entities, and consulting firms that interact with companies and are interested in the evaluation of a firm's financial health. Our results also provide implications with a strong impact on the scientific/technological community at the international level. The output of the research has an important impact on the scientific community as it represents the starting point of new challenges in technological innovation aimed at applying AI techniques to predictive models of business strategies, such as investments, financial decisions, marketing choices, and so on. It is possible to start a new line of scientific and technological research that would lead to the use of AI in various areas of management, exploiting deep learning techniques for applications that guide the entrepreneur not only toward the correct quantitative choices but also provide support for strategic/qualitative decisions with a consequent strong impact on business and economic growth. Future research prospects involve the integration of governance, ESG, and social media information into our RNN model, which could provide real-time insights overcoming the limitations of delayed financial reporting and improving predictive accuracy. Utilizing LSTM for processing unstructured text from governance indicators, ESG, social media, and news articles holds promise for capturing economic trends, thus enhancing the overall predictive capabilities of neural networks in identifying risky business situations and signalling potential financial distress. This research opens avenues for more accurate and timely bankruptcy prediction methodologies crucial in today's dynamic economic landscape. A research limitation consists of the fact that this paper is based only on accounting data and that for this reason we suggest using other non-financial variables to further improve accuracy.

Chapter 8

A New Graph Neural Network (GNN) Based Model for the Evaluation of Lateral Spreading Displacement in New Zealand

The increased availability of high-quality data from post-disaster field reconnaissance enabled the use of deep learning algorithms in the field of geotechnical earthquake engineering. The 2010-2011 Canterbury earthquake sequence in New Zealand caused significant damage due to abundant manifestation of liquefaction-induced lateral spreading. The data available from this sequence is an ideal case study for deep learning analyses due to the amount and quality of information available through the New Zealand Geotechnical Database (NZGD). A dataset of about 7500 datapoints was collected and organized by the authors to develop a new Graph Neural Network (GNN) algorithm for lateral spreading in the Canterbury area. The comparison between predicted and observed data is performed using feed-forward Neural Network. Several GNN models with different features are used and presented in this paper and Explainable Artificial Intelligence is applied to the model that provides the best performance. These computationally expensive analyses were carried out utilizing cloud-based computing capabilities offered by the Texas Advanced Computing Center (TACC) available to the natural hazard community through the cyberinfrastructure DesignSafe.

8.1 Introduction

Soil liquefaction is a phenomenon that occurs in saturated loose sand deposits during strong and rapid cyclic excitations such as earthquakes. The sudden motion generates excess pore water pressure that causes a rapid decrease of the stiffness of the soil with a consequent failure. In the presence of a gentle slope or near a free face, liquefaction triggering facilitates lateral spread displacements. Such a mechanism can generate significant damages to the built environment, increasing the loss associated with the earthquake [295, 296]. The 2010-2011 Canterbury earthquake sequence in New Zealand is known in the geotechnical earthquake engineering community as one of the best documented case studies for earthquake-induced lateral spreading phenomena (Figure 8.1(a)). Remote sensing techniques were used to obtain ground displacement from image correlation of satellite images before and after each event

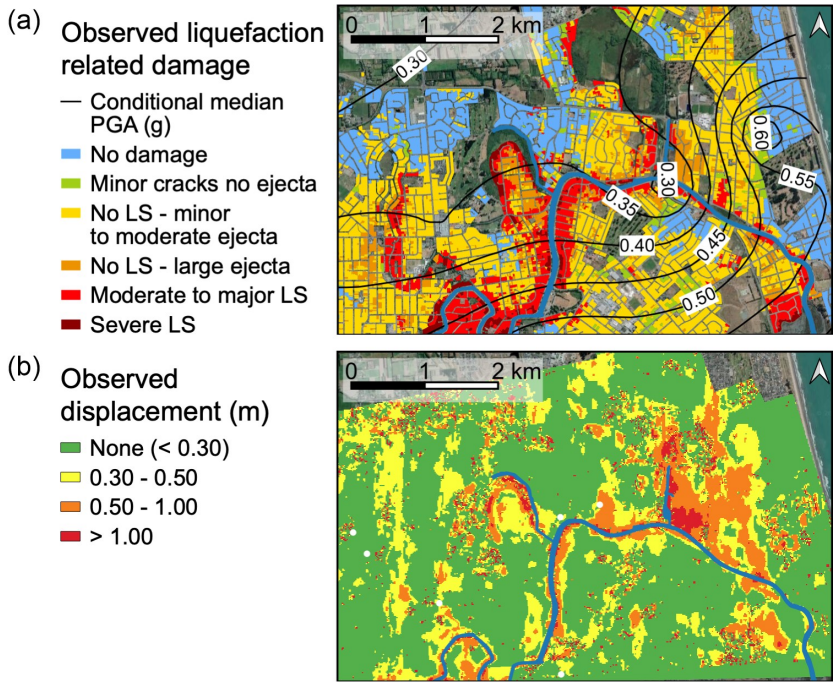


Figure 8.1: (a) Observed liquefaction-related damage (data from NZGD 2013) and (b) lateral spreading horizontal displacement observed from optical image correlation (data from Rathje et al. 2017 [1]) in the Avon River area for the 2011 Christchurch earthquake [2].

of the sequence (Figure 8.1(b)) [1].

The New Zealand Geotechnical Database (NZGD) is a publicly-available collection of geotechnical site characterization data. The availability of high-quality data collected in the post-disaster field reconnaissance combined with the numerous field investigation data performed in the whole area struck by the earthquake represents a precious dataset that enables the use of innovative artificial intelligence (AI)-based techniques to better guide lateral spreading hazard analysis. Previous studies were carried out in the area of interest using both standard methods [297, 298, 299] and more innovative approaches such as using AI-based techniques [300, 2].

The research presented in this paper aims to develop a new Graph Neural Network (GNN) algorithm to predict the occurrence of lateral spreading based on geometrical and event-specific conditions. More specifically, a dataset of about 7500 datapoints was collected and organized with the following information: (i) distance from the river (ii) ground slope (iii) ground elevation (iv) ground water table - GWT (v) Peak Ground Acceleration - PGA. The event-specific information (i.e., GWT and PGA) are referring to the 22 February 2011 Christchurch earthquake (Mw 6.2) and were extracted from the NZGD. Lateral spreading occurrence is evaluated at each datapoint based on the remote sensing analyses presented by Rathje et al. 2017 using a 0.3m threshold to discern between occurrence and non-occurrence of lateral spreading. As in different subject areas, prior research conducted the application of GNNs exploiting geometrical and event-specific variables [301, 302, 303].

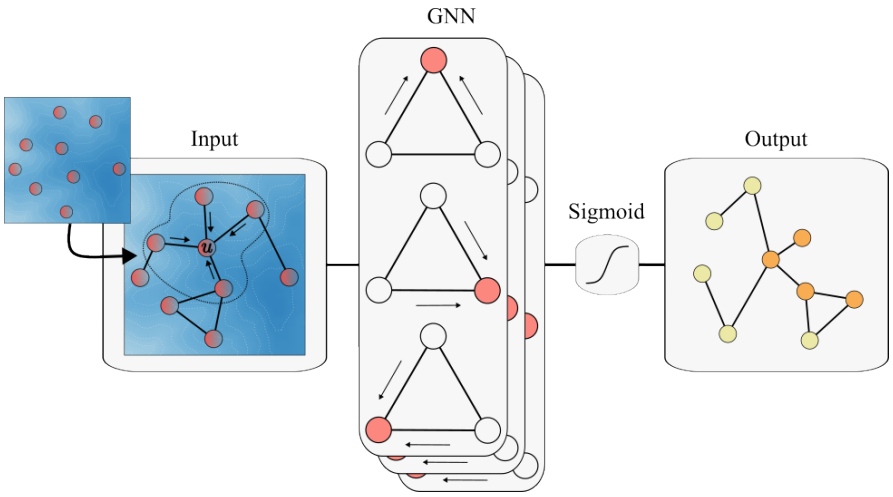


Figure 8.2: Schematic representation of Graph Neural Network algorithm.

8.2 AI Methods

Random Forests (RFs) are Machine Learning (ML) models belonging to the ensemble learning family which combines individual models aiming at more accurate and robust predictions [304]. RFs consist of ensembles of Decision Trees which recursively splits the data based on features resulting in a tree-like structure.

Deep Learning (DL) is a subset of ML methods which are able to consider numerous variables and accounting for non-linear interaction between them. Based on Artificial Neural Networks (NNs), DL models consist of consecutive layers of interconnected artificial neurons which process and propagate information [305]. Such models are not immediately interpretable, meaning that it is difficult to understand the causal relationship between input and their output. Explainable Artificial Intelligence (XAI) methods are designed to overcome this limitation. Graph Neural Networks are DL models able to process graph representations. These models are particularly suitable for the analysis of geospatial datasets (e.g., geotechnical damage maps) making it possible to consider the relationship between neighboring nodes. As shown in Figure 8.2, through GNNs the information can be propagated, aggregated, and processed among neighbors such that the prediction on a specific point takes into consideration the influence of its immediate surrounding nodes. As the number of GNN layers increases, the information spread is extended beyond the immediate neighborhood of nodes as an iterative process. The local correlation assumption [306, 307] which characterizes spatial data enables the use of distances to define neighbor relationships between points in geospatial datasets. In this study, Graph Convolutional Networks [308] are adopted to model liquefaction-related damage observed in New Zealand following the February 2011 Christchurch earthquake (Mw 6.2) and predict for each node the occurrence of lateral spreading.

8.3 Experiments

This study provides a performance comparison between RF, NN, and GNN in the prediction of lateral spreading occurrence by binary classification (yes/no). The baseline for the comparison refers to the RF Model 3 developed by Durante and Rathje (2021) which makes use

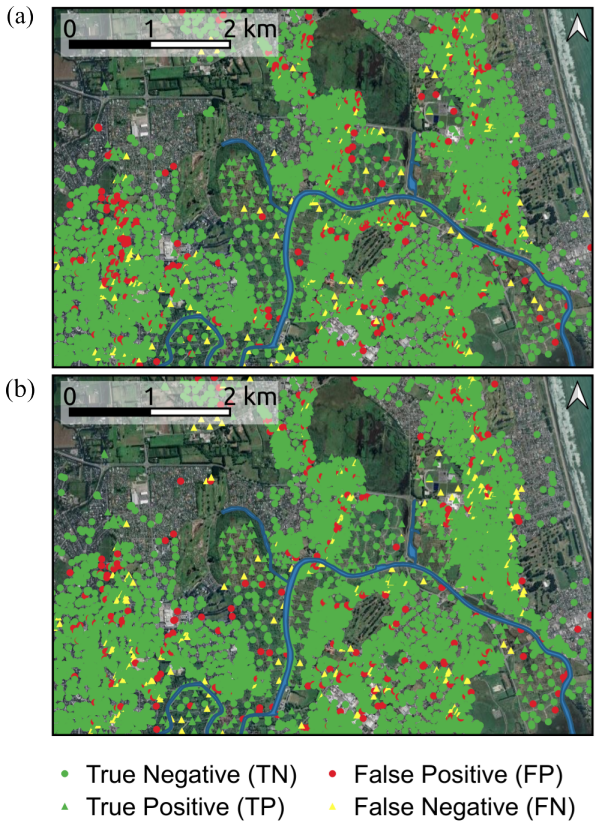


Figure 8.3: Prediction over the whole dataset by the best (a) NN and (b) GNN models.

of available geometrical and event-specific features. For fair comparison, the same feature set is used to train all the models being evaluated. In this experimental setting, NNs input consists of a set of feature vectors while the GNNs input is a graph. The latter is obtained from the geospatial map by considering the latitude and longitude of each datapoint such that: each datapoint is a graph node (characterized by the set of features) and edges are added so that each node is linked to its $k = 5$ nearest neighbors according to their Euclidean distance.

In order to find the best hyperparameter (i.e., an input parameter of the model) configuration for the networks, a grid-search approach was used. The hyperparameters space for both NN and GNN was defined by the number of trainable parameters (50k, 67k, 100k), the learning rate, the activation function, and the coefficient of the $l2$ regularization term. Models were trained by using the binary cross-entropy as loss function and with a maximum number of epochs (650) under an early stopping mechanism to avoid overfitting.

For each different hyperparameter configuration, we performed 10-fold Cross-Validation (CV) which assesses the model's ability to generalize by repeatedly training and testing the model on different subsets of the dataset. Models' performance was assessed by considering Cohen's kappa (κ) coefficient [309] coherently with the reference work. The best hyperparameters configuration is chosen as the one with the highest average performance (according to coefficient κ); the best model is ultimately selected as the best performing model among the CV iterations of the chosen configuration.

		NN		GNN	
Metric	Ref	Best	CI	Best	CI
Accuracy	0.88	0.85	0.82-0.83	**0.91**	0.89-0.90
Recall (Yes)	0.82	0.79	0.75-0.76	**0.87**	0.85-0.88
Recall (No)	0.93	0.89	0.86-0.88	**0.94**	0.90-0.92
Precision (Yes)	0.89	0.84	0.80-0.81	**0.91**	0.87-0.89
Precision (No)	0.87	0.85	0.83-0.84	**0.91**	0.90-0.91
ROC AUC	0.90	0.92	0.89-0.90	**0.97**	0.94-0.96

Table 8.1: Test results in predicting liquefaction induced lateral spreading occurrence (CI=95% Confidence Interval)

.

8.4 Results

This section discusses results obtained applying the best NN and GNN models presented previously to the extensive dataset available to predict the occurrence of liquefaction induced lateral spreading. Fig. 8.3 shows the distribution of model predictions for NN (Fig. 8.3(a)) and GNN (Fig. 8.3(b)) in terms of True Negative (TN – green dots), True Positive (TP – green triangles), False Positive (FP – red dots) and False Negative (FN – yellow triangles) among data points. The distribution of the errors (FP and FN) is not concentrated in any specific area meaning that the models are both able to predict the general pattern on a regional scale. The comparison between NN and GNN shows that the latter is able to perform better especially in the western part of the selected area thanks to the fact that due to its formulation it is able to include in the analysis the relationship between neighboring nodes.

Table 8.1 reports the test results for the models considered in terms of accuracy, recall, precision, and ROC AUC for the best NN and GNN models together with the 95% confidence interval (CI) that represents the range of estimates of each metric sampled by CV iterations for the chosen hyperparameter configuration. In this table it is also reported the performance of the reference model (Ref) that is the RF presented in Durante and Rathje 2021. The numbers show that while the best NN model is on average less performing compared to the reference model, the best GNN model always provides higher metrics compared to the reference RF model.

8.4.1 Explainability

DL models as mentioned are not immediately interpretable. For this reason, XAI methods are often used to better understand the contribution of each feature in the model response. In this study, Shapely Additive exPlanation (SHAP) is used to interpret the models. SHAP is a game theory-based approach for interpreting black-box models that assigns an importance value to each feature called a SHAP value according to its contribution to the predictions [13]. The implemented SHAP adaptation to GNN models provided by Duval and Malliaros (2021) enables the use of graph data and was used to compute explanations for the best GNN.

Nevertheless, a powerful tool to visualize feature importance and their effects over the prediction is the beeswarm plot. Such a plot can be interpreted as follows: the features are listed from the most (top row) to the less relevant (bottom row). Each point on the plot is

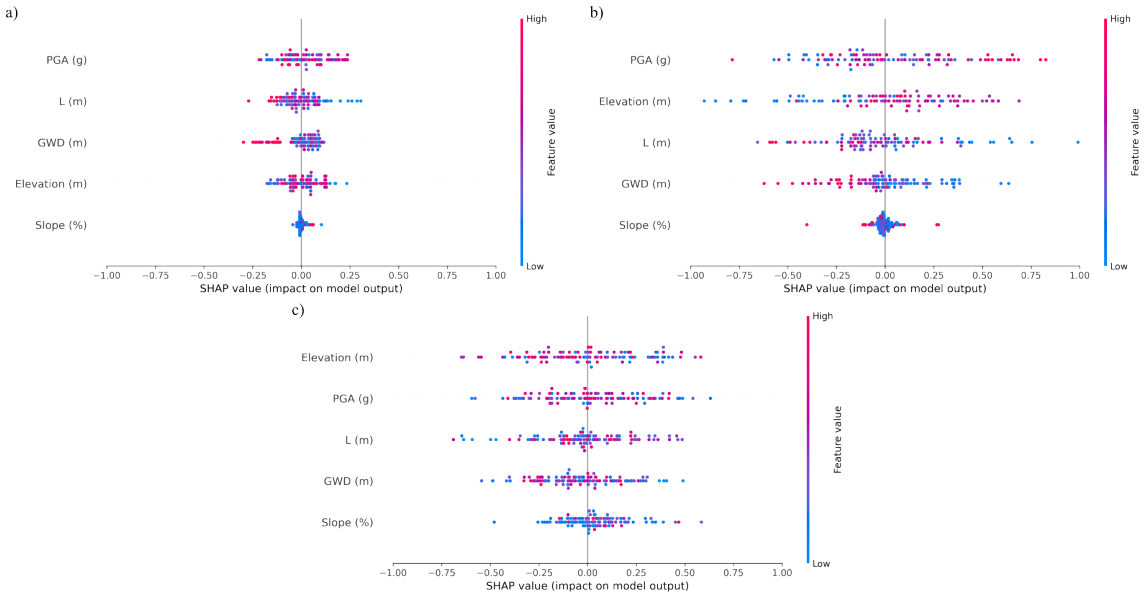


Figure 8.4: SHAP-based beeswarm plots showing the relation between dataset samples and their influence over the model predictions for (a) RF, (b) NN and (c) GNN.

a SHAP value for a feature and a sample distributed along the horizontal axis with color coding the feature value (as shown in the legend of the plot, high values in red and low values in blue). In binary classification, points associated with positive SHAP values increase the prediction towards the occurrence of the event while ones associated with negative values decrease the prediction towards the absence of the event.

Fig. 8.4 shows the beeswarm plots for the reference RF model (Fig. 8.4(a)), the best NN model (Fig. 8.4(b)) and the best GNN model (Fig. 8.4(c)). For the RF and NN models, the most important parameter is PGA, while for the GNN is the elevation with PGA as the second most important. For all the models, the less influential feature is the slope but it is worth noticing that while both RF and NN model predictions are not clearly influenced by this feature (SHAP values are concentrated around zero), the GNN model is able to leverage its contribution to the prediction. More specifically, on average, higher values of slope increase the GNN model predictions towards lateral spreading occurrence. Similarly, for RF and NN models, high values of PGA, smaller distance from the river (L) and small values of GWD all increase the prediction in the same direction. This model behavior is ensuring that the models shown in this study are able to respect the physics beyond the problem analyzed. Nevertheless, the interaction between SHAP values and feature values is less evident for the GNN explanation which is attributable to SHAP definition not considering the influence of the neighboring nodes to the prediction.

8.5 Conclusions

Deep learning algorithms are used in this paper to predict the occurrence of liquefaction induced lateral spreading for the February event of the 2010-2011 Canterbury earthquake sequence in New Zealand. Data from the New Zealand Geotechnical Database are used to create a dataset of about 7500 datapoints with some basic geometrical and event specific information. The DL algorithms selected for this study are NN and GNN. The selection of

the best models is based on a parametric investigation that included several hyperparameters for each algorithm. These computationally expensive analyses have been carried out utilizing cloud-based computing capabilities offered by the Texas Advanced Computing Center (TACC) available to the natural hazard community through the cyberinfrastructure DesignSafe [310]. The comparison of the best models used SHAP analyses as a tool to provide explanation to the different models. It was shown that GNN is able to correctly consider the contribution of each feature and that the added information from modeling the relationship between neighboring nodes is increasing the overall performance of the model in the regional scale.

Chapter 9

A New Deep Learning and XAI-Based Algorithm for Features Selection in Genomics

In the field of functional genomics, the analysis of gene expression profiles through Machine and Deep Learning is increasingly providing meaningful insight into a number of diseases. The paper proposes a novel algorithm to perform Feature Selection on genomic-scale data, which exploits the reconstruction capabilities of autoencoders and an ad-hoc defined Explainable Artificial Intelligence-based score in order to select the most informative genes for diagnosis, prognosis, and precision medicine. Results of the application on a Chronic Lymphocytic Leukemia dataset evidence the effectiveness of the algorithm, by identifying and suggesting a set of meaningful genes for further medical investigation.

9.1 Introduction

In the field *functional genomics*, starting from the results of the Human Genome Project, the evolution of sequencing techniques provides big volumes of data for each single patient by taking advantage of the *high-throughput* and *next-generation sequencing* i.e., a set of time and cost-effective techniques for sequencing DNA and RNA. By means of them, it is possible to measure the expression of thousands of genes for each individual and hence to collect quantitative gene expression profiles (GEP) to be used for research and clinical purposes. But despite GEP datasets represent a valuable source of information in healthcare—they are indeed used for diagnosis, prevention, and precision medicine—their analysis results challenging for three main reasons. The first one is the *course of dimensionality*: a genomics dataset typically consists of a very large number of features (genes) and a small number of samples (patients); the second problem concerns *imbalanced classes*: in the analysis of different groups of patients, genomics data are often stratified in classes according to different pathologies. In most cases, there is a significant difference between the number of instances in each class; finally, sequencing data are typically collected from multiple sources, different laboratories, and sequencing tools. This results in *noisy datasets* which are difficult to analyze [311].
In recent years, Machine Learning (ML) and Deep Learning (DL) have been widely adopted in this field, providing breakthrough results and meaningful insights into the relationship be-

tween genomics and cancer [312, 313]. Although still very promising, DL models are in general not immediately interpretable, meaning that it is difficult to understand the causal relationship between the inputs and their outcomes. This is an even more severe problem in the bioinformatics domain, where it is crucial to understand, for example, in the case of genomics, how the expression of a gene can affect the progression of oncological patients.
We propose a new algorithm, based on DL and Explainable Artificial Intelligence (XAI), for genomics whose aim is threefold: first, select the most meaningful genes for a regression/classification problem; second, provide a more accurate prediction model; third, quantify and evaluate the effect of features on the predictions, through XAI. We used our algorithm for the GEP analysis of Chronic Lymphocytic Leukemia (CLL) patients, identifying a meaningful subset of genes for the disease prognosis. The following sections are organized as follows. First, we review the most relevant related works in Section 9.2, and we then give a formal definition of the algorithm in Section 9.3. The application and the results obtained by the algorithm for the CLL study are discussed in Section 9.4. Finally, directions for further research are proposed in Section 9.5.

9.2 Related Works

A number of recent studies propose and evaluate new approaches for feature selection (FS) on GEP datasets for cancer diagnosis and prognosis[312]. Such methodologies mainly aim at selecting the most informative genes, which are able to characterize classes and identify groups of patients. In this context, the adoption of XAI methods has started to gain momentum for interpretability purposes as well as to enhance FS[314, 315, 316]. A widely used approach to overcome the *course of dimensionality* problem is to perform dimensionality reduction using AEs [317]. While this has been proven to be effective, the encoding is typically a non-linear projection of the variables into a lower-dimensional space, which makes it difficult to provide the proper interpretations of the results. In this work we propose a novel approach, which uses AEs for selecting the most informative genes without any change into the original features space, hence enhancing the explainability of the results, and still exploiting the representation abilities of AEs.
We moreover use an ad-hoc defined XAI-based score in order to iteratively select the features by taking advantage of the Shapely Additive ex-Planation method (SHAP)[279], a cooperative game theory-based approach for computing the *shapely values*. Such values measure, locally (at the sample level), the contribution of each feature to the predictions of an ML model. In particular, for a given sample x, the set of features F, the contribution of the feature $j \in F$ is defined as:

$$\phi_j = \sum_{S \subseteq F \setminus \{j\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} \left[f_{S \cup \{j\}}(x_{S \cup \{j\}}) - f_S(x_S) \right] \tag{9.1}$$

with $\phi_j \in \mathbb{R}$ and where $f_{S \cup \{j\}}$ and $x_{S \cup \{j\}}$ denote the prediction model and the sample considering the only subset of features S without the j-th one. In words, SHAP computes the contribution of a feature by comparing the model predictions obtained with and without a feature, for all the possible combinations S. Since the computation of the Equation 9.1 is inefficient in the case of NN as a prediction model—a NN should be re-trained for each combination of features ($2^{|F|}$)—the authors demonstrate in [279] that shapely values can be

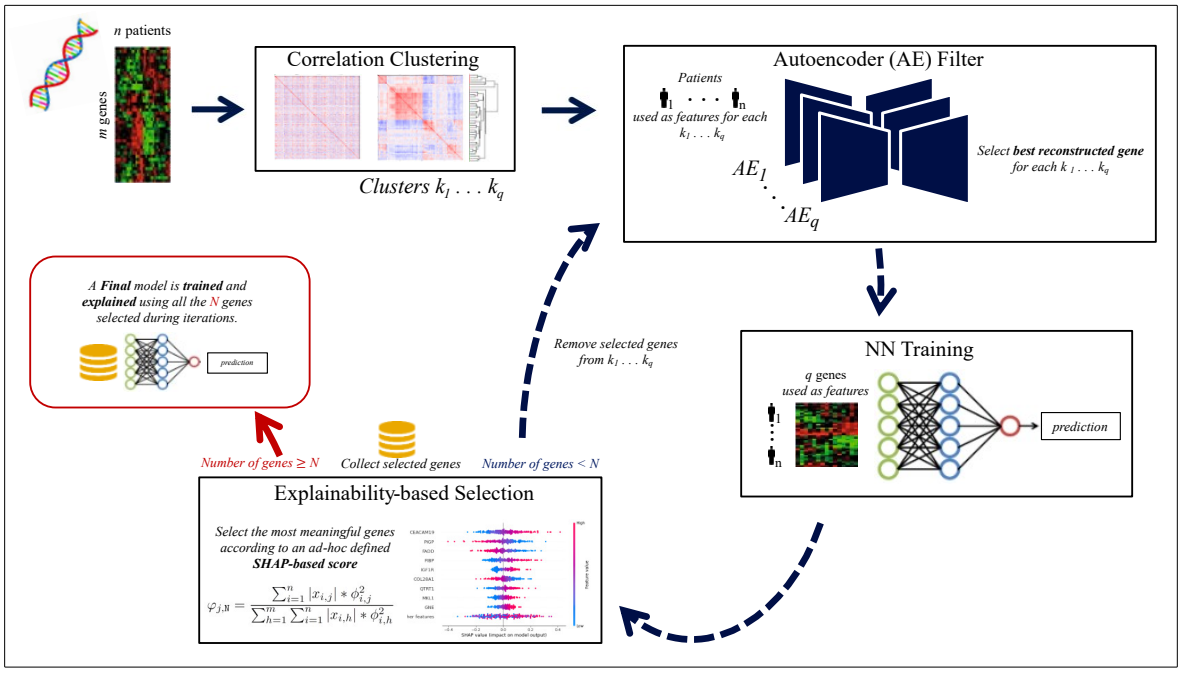


Figure 9.1: The Proposed Algorithm.

computed by solving a weighted linear least square regression with the proper shapely kernel. Although we used such an alternative method, we omitted the details and focus on the only definition of shapely values.

9.3 The Algorithm

The proposed algorithm is based on two main ideas: (1) we use a clustered correlation matrix in order to group features that enclose similar patterns and we then filter the redundant information for each group by using AEs. In contrast with previous works, in which AEs are used for dimensionality reduction, we still work at the level of the original features. In particular, we take advantage of the encoding and reconstruction abilities of the AEs assuming that the more accurate is the reconstruction of a feature, the more that feature is representative of the cluster it belongs to. We hence provide a more treatable dataset in terms of dimensionality, without loss of representativeness, by filtering redundant features; (2) we train NNs and we iteratively select the most meaningful features using a new ad-hoc defined SHAP score. We repeat the analysis by removing at each iteration, the previously selected features. We eventually use the set of selected features (from all the iterations) to train and explain a final model. Figure 9.1 shows the main algorithm phases.

9.3.1 Formal Setting

Let be $\mathcal{D} = \{X, Y\}$ a dataset such that $X \in \mathbb{R}^{n \times m}$ is the matrix of inputs, and $Y \in \mathbb{R}^{n \times l}$ is the matrix of the corresponding labels. Let us further assume $m \gg n$ meaning that the dataset is characterized by a way larger set of features with respect to the number of samples.

As a novelty contribution, we introduce a new impact score, which, by means of the

SHAP local explanation, measures the global impact of each feature on model predictions. We hence associate to each feature (column) j of X, used to train a model $\mathtt{N}$, a couple $(\rho_{j,\mathtt{N}}, \varphi_{j,\mathtt{N}})$ were $\rho_{j,\mathtt{N}}$ is the correlation between the j-th columns of X and their shapely values $\{\phi_{1,j}, ..., \phi_{n,j}\}$, and $\varphi_{j,\mathtt{N}}$ is defined as follows:

$$\varphi_{j,\mathtt{N}} = \frac{\sum_{i=1}^{n} |x_{i,j}| * \phi_{i,j}^2}{\sum_{h=1}^{m} \sum_{i=1}^{n} |x_{i,h}| * \phi_{i,h}^2} \tag{9.2}$$

With $\rho_{j,\mathtt{N}}$ and $\varphi_{j,\mathtt{N}}$ we want to emphasize *how* and *how much*, respectively, a feature globally affect the predictions of $\mathtt{N}$.

9.3.2 Algorithm

For sake of clarity, we introduce our algorithm by first defining a set of sub-procedures. The first one (Algorithm 1) computes the pairwise correlation matrix $C \in \mathbb{R}^{m \times m}$ between the features (columns) of a generic real-valued matrix M. Finally it clusters C in order to return a set $K = \{k_1, ..., k_q\}$ such that for each $i = 1, .., q$, k_i is a set of indexes—a partition (cluster) for the columns of M.
The second sub-procedure, defined in Algorithm 2, trains an AE for each cluster, by using the transpose of the input matrix M—meaning that, for the AE model, each feature represents a sample and vice versa. The rationale here is that we assume the best-reconstructed feature (over the samples) to be the most representative of the cluster it belongs to. We denote $M_{k_i} \in \mathbb{R}^{n \times |k_i|}$ as a matrix including the only columns of M which indexes are in k_i. The $evaluate$ function provides the column indexes of M_k^T associated with the best-reconstructed feature. Finally, the sub-procedure returns a set J of q indexes—one for each cluster.

Algorithm 1 Corr. & Clustering

function CORRCLUSTERING(M)
  $C \leftarrow corr(M)$
  $K \leftarrow clustering(C)$
  return K
end function

Algorithm 2 AE Filtering

function AEFILTERING(M,K)
  $J \leftarrow \emptyset$
  for $k \in K$ **do**
    $\mathtt{AE} \leftarrow train(M_k^T)$
    $J \leftarrow J \cup evaluate(\mathtt{AE}, M_k^T)$
  end for
return J
end function

The last sub-procedure, reported in Algorithm 3, takes as input: the data, a matrix of shapely values Φ and the threshold $\beta \in \mathbb{R}$, with $\beta \in [0, 1]$. It first computes the correlations between each column of M and the corresponding columns of Φ. Subsequently, it computes the intensity for each feature following the definition of equation 9.2. It then selects the column indexes according to β and the mean intensity, to finally provide a set $\tilde{J}$ of column indexes for M.

Algorithm 3 Selection

function SELECT(Φ, M, β)
  $\boldsymbol{c} \leftarrow computeCorrelation(\Phi, M)$
  $\boldsymbol{d} \leftarrow computeIntensity(\Phi, M)$
  $\mu \leftarrow \frac{1}{|\boldsymbol{d}|} \sum_{d \in \boldsymbol{d}} d$
  $\tilde{J} \leftarrow \{j \mid |\rho_j| > \beta \ \wedge \ \varphi_j > \mu, \ \forall \rho_j \in \boldsymbol{c}, \forall \varphi_j \in \boldsymbol{d}\}$
  return $\tilde{J}$
end function

The main procedure is described by Algorithm 4. After clustering the correlation matrix, it selects a set of meaningful features index to be added to S. It then removes the selected indexes from their corresponding clusters in K and proceeds by repeating the analysis. Here we denote $X_J \in \mathbb{R}^{n \times |J|}$ (and accordingly X_S) as a matrix including the columns of X which indexes are in J, and $\mathtt{N}_J$ (and accordingly $\mathtt{N}_S$) as a NN trained on $\{X_J, Y\}$. The iterative analysis stops when $\alpha \in \mathbb{N}$, $\alpha \leq m$ features are selected or on a maximum number of iterations. The algorithm eventually trains and explains a final NN using the set S.

Algorithm 4

Require: X, Y
  $K \leftarrow$ CORRCLUSTERING(X)
  $S \leftarrow \emptyset$
  while $|S| < \alpha \vee$ **not** $maxIter$ **do**
    $J \leftarrow$AEFILTERING(X, K)
    $X_J^b, Y^b \leftarrow$DATABALANCING(X_J, Y)
    $\mathtt{N}_J \leftarrow$ FINDMODEL(X_J^b, Y^b) ▷ Model Selection & Training
    $\Phi \leftarrow$ SHAP($\mathtt{N}_J, X_J$) ▷ matrix of shapely values $\Phi \in \mathbb{R}^{n \times |J|}$
    $\tilde{J} \leftarrow$ SELECT(Φ, X_J)
    $S \leftarrow S \cup \{j_i \in J \mid i \in \tilde{J}\}$
    $K \leftarrow K \setminus S$ ▷ remove from their corresponding cluster
  end while
  $\mathtt{N}_S \leftarrow$ FINDMODEL(X_S, Y)
  $\Phi \leftarrow$ SHAP($\mathtt{N}_S, X_S$)
  $J^* \leftarrow$ SELECT(Φ, X_S)

9.4 A Use Case: Chronic Lymphocytic Leukemia

9.4.1 Materials and Methods

We applied our algorithm for analyzing GEP of patients affected from CLL. The dataset used for this analysis is composed of 97 patients GEP for 19367 genes. The such dataset was extracted from the observational O-CLL1 study (clinicaltrials.gov identifier NCT00917540), where a set of newly diagnosed Binet A CLL cases were prospectively enrolled from several Italian institutions and studied for GEP [318, 319]. For each patient a real-valued number is provided, indicating, as a factor of prognosis, the time interval after which the condition of the patient deteriorates. In particular, we distinguished two classes: *class 0* (23 samples) for the patients whose condition deteriorates in a period shorter than 24 months, and *class 1* (74) for the patients whose condition deteriorates in a period equal or longer than 24 months. We used the proposed algorithm for training a NN to solve such a classification problem as well as to identify a set of meaningful genes over the whole set of 19367. We additionally provide insight into the prognostic power of such genes. The genes were initially clustered in 500 groups by using a hierarchical clustering technique—Figure 9.2. The AE filtering selects 500 genes and we further applied a statistical filter in order to select 50 genes. After re-balancing the classes with the Synthetic Minority Over-sampling Technique (SMOTE), we perform model selection with 10-fold cross-validation in order to find the best (in terms of binary accuracy on the test set) NN for solving the classification problem. We finally use our SHAP scores (defined in Section 9.3.1) to select the most meaningful genes, by setting $\beta = 0.85$. After selecting a set of $\alpha = 50$ genes through the iterations of the algorithm, we use them to train and explain a final NN. During cross-validation, the dataset was split into a train set of 138 patients (after re-balancing) and a test set of 10 only-real patients equally

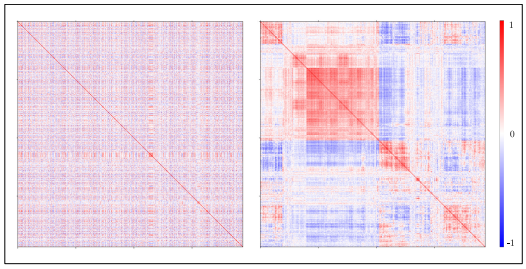


Figure 9.2: Genes clustered correlation matrix.

Iteration	Accuracy (CI 95%)
1	77.2%-92.7%
2	74.7%-91.2%
3	68.6%-89.3%
4	64.3%-83.6%
final	**79.1%-92.9%**

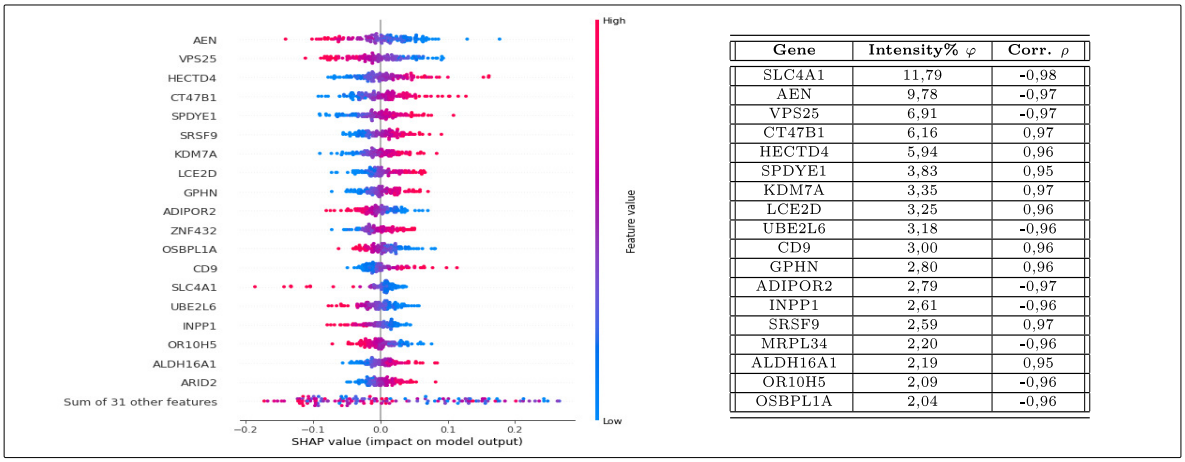
Table 9.1: Results over iterations.

divided among the two classes.

The algorithm has been implemented using the Python (v3.8.11) programming language. NNs have been implemented by taking advantage of the Tensorflow (v2.6.0) framework along with the Keras library. XAI analysis was performed by means of the SHAP library [279].

9.4.2 Results

The overall results are reported in Table 9.1. In particular, for each iteration of the algorithm, we measured the accuracy of all the models obtained during cross-validation, for which we report the confidence interval. As we expected, the classification accuracy decreases with the algorithm iterations: the reason is that the previously chosen features—expected to be the most representative of each cluster—are no more considered for the subsequent analysis. An improvement in accuracy is instead reported for the final step of the algorithm, by which a model is trained using the set of genes selected during each iteration. The accuracy of the best final model is 100%.



Gene	Intensity% φ	Corr. ρ
SLC4A1	11,79	-0,98
AEN	9,78	-0,97
VPS25	6,91	-0,97
CT47B1	6,16	0,97
HECTD4	5,94	0,96
SPDYE1	3,83	0,95
KDM7A	3,35	0,97
LCE2D	3,25	0,96
UBE2L6	3,18	-0,96
CD9	3,00	0,96
GPHN	2,80	0,96
ADIPOR2	2,79	-0,97
INPP1	2,61	-0,96
SRSF9	2,59	0,97
MRPL34	2,20	-0,96
ALDH16A1	2,19	0,95
OR10H5	2,09	-0,96
OSBPL1A	2,04	-0,96

Figure 9.3: Final selected genes.

Figure 9.3 reports, on the left side, a summarized representation of the shap values and, on the right side, the values for correlation and intensity for the most interesting genes found by our algorithm.

9.5 Conclusions

The algorithm proposed in this work can be used as a valuable tool in genomics to identify protective (or not) sets of genes for a disease, suggesting potential pathways for further medical investigation. A natural direction for future development is to perform a large-scale assessment of the algorithm performances, by using state-of-the-art benchmark GEP datasets.

Chapter 10

Genes Selection using Deep Learning and Explainable Artificial Intelligence for Chronic Lymphocytic Leukemia Predicting the Need and Time to Therapy

Analyzing Gene Expression Profiles (GEP) through Artificial Intelligence provides meaningful insight into cancer disease. This study introduces DeepSHAP Autoencoder Filter for Genes Selection (DSAF-GS), a novel Deep Learning and Explainable Artificial Intelligence-based approach for Feature Selection in genomics-scale data. DSAF-GS exploits autoencoders' reconstruction capabilities without changing the original feature space, enhancing results' interpretation. Explainable Artificial Intelligence is then used to select the informative genes for Chronic Lymphocytic Leukemia prognosis of 217 cases from a GEP database comprising roughly 20000 genes. The model for prognosis prediction achieved an accuracy of 86.4%, a sensitivity of 85.0%, and specificity of 87.5%. According to the proposed approach, predictions were strongly influenced by CEACAM19 and PIGP, moderately influenced by MKL1 and GNE, and poorly influenced by other genes. The 10 most influential genes are selected for further analysis. Among them, FADD, FIBP, GNE, IGF1R, MKL1, PIGP, and SLC39A6 were identified in the Reactome pathway database as involved in signal transduction, transcription, protein metabolism, immune system, cell cycle, and apoptosis. Moreover, according to the network model of the 3D protein-protein interaction (PPI) explored using the Network Analyst tool, FADD, FIBP, IGF1R, QTRT1, GNE, SLC39A6, and MKL1 appear coupled into a complex network. Finally, all the 10 selected genes showed a predictive power on Time To First Treatment (TTFT) in univariate analyses on a basic prognostic model including IGHV mutational status, del(11q) and del(17p), NOTCH1 mutations, β2-microglobulin, Rai stage, and B-lymphocytosis known to predict TTFT in CLL. However, only IGF1R [hazard ratio (HR) 1.41, 95% CI 1.08-1.84, P=0.013], COL28A1 (HR 0.32, 95% CI 0.10-0.97, P=0.045), and QTRT1 (HR 7.73, 95% CI 2.48-24.04, P$<$0.001) genes were significantly associated with TTFT in multivariable analyses when combined with the prognostic factors of the basic model, ultimately increasing the Harrell'c index and the explained variation to 78.6% (versus 76.5% of the basic prognostic model) and 52.6% (versus 42.2% of the basic prognostic model), respectively. Also, the goodness of model fit was enhanced (χ2=20.1, P=0.002), indicating its improved performance above the basic prognostic model. In conclusion, DSAF-GS identified a group of significant genes for CLL prognosis, suggest-

ing future directions for bio-molecular research.

10.1 Introduction

A precise prognostic methodology in chronic lymphocytic leukemia (CLL) patients is critical from the clinical standpoint since progression to a more advanced disease stage requires therapy and often implies an adverse prognosis. At first presentation/diagnosis, over three-quarters of CLL patients are classified as early/asymptomatic disease phase and not requiring immediate therapy [320]. Although most patients have a low-risk profile, as indicated by the high frequency of the immunoglobulin heavy chain variable (IGHV) gene mutated (IGHV-mut) status [321] and the low del(17p) occurrence involving the TP53 locus [322], the time to first treatment (TTFT) is rather heterogeneous and it can be partially predicted using combinations of risk-associated markers which include staging systems and β2-microglobulin (β2-M) [323].

Despite the proven prognostic power of this approach, the clinical course of a number of patients does not follow the pattern predicted, possibly indicating the requirement for additional prognosticators. In this respect, gene expression profiling (GEP) that is the measurement of the activity (the expression) of all genes of interest to depict a synthetic picture of cellular function is exploited to increase the ability to predict the prognosis of CLL patients [324, 325].

Although GEP datasets represent a valuable source of information in healthcare, being currently used for diagnosis, prognosis, and precision medicine of hematological malignancies [326], their analysis results challenging for three main reasons. The first one is the course of dimensionality: genomic-scale datasets typically consist of a very large number of features (genes) and a relatively small number of samples (patients); the second problem concerns imbalanced classes: genomics data are often collected from multiple sources and stratified based on pathologies. In most cases, there is a significant difference between the number of instances in each class; finally, sequencing data are typically collected from multiple sources, different laboratories and sequencing tools. This results in noisy datasets which are difficult to analyze [327].

A bioinformatic analysis is necessary to fully realize the potential of these large-scale sequencing data for prognosis in hematological malignancies [328] and solid tumors [329]. Machine Learning (ML) approaches have been widely used to enhance the performance of diagnostic and predictive models for different diseases and CLL as well [330]. Resources and guidelines for using ML in CLL have been made available [311].

However, most ML prognostic models for CLL fail to consider numerous variables and do not account for non-linear interactions between them [331]. This limits the accuracy of the models and the ability to make informed predictions about the disease progression. Therefore, promising tools, such as Deep Learning (DL) methods, a subset of ML methods based on Artificial Neural Networks (NNs), may be used to overcome the aforementioned ML limitations. DL approaches recognize hidden patterns in large-scale datasets that are typically difficult to detect with traditional statistical and ML models. Recent studies propose and evaluate new feature selection (FS) approaches on genomic-scale datasets for cancer diagnosis and prognosis [332]. Such FS methodologies mainly aim at selecting the most informative genes which can characterize classes and identify groups of patients.

Although very powerful, DL models are in general not immediately interpretable, mean-

ing that it is difficult to understand the causal relationship between the inputs and their outcomes. This is an even more severe problem in the bioinformatics domain where it is crucial to understand, for example in the case of genomics, how the expression of a gene can affect the progression of oncological patients. In this context, adopting Explainable Artificial Intelligence (XAI) methods have started to gain momentum for interpretability purposes as well as to enhance FS [333].

On the other hand, a widely used approach to overcome the course of dimensionality problem is to perform dimensionality reduction using Autoencoders (AE) [334]. While this has been proven effective, the encoding is typically a non-linear projection of the variables into a lower-dimensional space, making it difficult to provide the interpretations of the proper results.

This study introduces DeepSHAP Autoencoder Filter for Genes Selection (DSAF-GS), a novel DL and XAI-based FS method for genomics-scale data analysis. Such a method uses AEs for selecting the most informative genes without any change into the original features space, hence enhancing the explainability of the results and still exploiting the representation abilities of AEs. Such selection of genes is used to design and train a prediction model for diagnosis or prognosis. Eventually, the Shapely Additive ex-Planation (SHAP) [335] XAI method is applied to interpret the model results and select the most meaningful genes for the disease.

In the present paper, the proposed XAI method has been used to identify those genes whose expression levels are relevant for predicting the need of therapy in CLL patients from a prospective cohort of newly diagnosed Binet stage A CLL (O-CLL protocol) [336]. This innovative approach enabled meaningful insights into CLL prognosis from genomic data by locating a group of significant genes to boost the prognostic power of a basic prognostic model. We point out that while our contribution is fully positioned within the research in oncology, our XAI method has a broader applicability; in fact, from the bioinformatics and computational genomics point of view [337], an interesting avenue of further research is to assess its efficacy as a general feature-selection method, for instance by considering datasets for which we already have some a-priori semantic information on the most relevant features and by using classical comparison metrics for predictive models.

10.2 Materials and Methods

10.2.1 Patients

Among the 523 newly diagnosed Binet A CLL patients included in the observational O-CLL protocol (clinicaltrials.gov identifier NCT00917540) [338], 224 were ultimately enrolled and evaluated for GEP. All participants provided written informed consent, and the relevant institutional review boards approved the study. The inclusion and exclusion criteria were previously detailed [339]. The biologic review committee of the O-CLL trial confirmed the diagnosis using flow cytometry analysis centralized at the Ospedale Policlinico San Martino IRCCS Genova, Italy.

10.2.2 Assessment of biological markers

Cytogenetic abnormalities involving deletions at chromosomes 11q23 and 17p13 were evaluated by FISH in a purified CD19+ population as previously described [340]. IGHV gene mutational status was assessed on cDNA specimens [341]. Sequences were aligned to the IMGT directory and analyzed using IMGT/VQUEST software. NOTCH1 mutation hotspot was set by next-generation deep sequencing as previously described [342].

10.2.3 GEP analysis

GEP experiments were performed as previously described [343]. Briefly, total RNA fraction was obtained from CD19+-enriched B-cell samples (EasySep-Human B cell enrichment kit without CD43 depletion, Stem Cell Technologies, Voden Medical Instruments spa, Milan, Italy) using the fully automated protocol of immunomagnetic cell separation with RoboSepTM (Stem Cell Technologies). Purified B-cells (CD19+) exceeded 95% were employed as total RNA sources for GEP analysis.

10.2.4 O-CLL Dataset

For each patient, 19367 genes profiles were provided. Patients are labeled according to the occurrence of an event (or not). The considered outcome was the need of therapy starting or death (dichotomous, not (event=0) vs. yes (event=1)]. From the 217 patients of the final dataset, 120 were labeled as event=0 and 97 as event=1.

10.2.5 Feature Selection

GEP studies generate large, high-dimensional, and unbalanced datasets where each sample can have up to thousands of variables. This results in high computational costs and the possibility of overfitting. Such overfitting may mistake small changes in the data as significant differences, leading to misclassification errors. This study addresses these risks by applying FS techniques to reduce the dimensionality problem by selecting the most relevant features and removing noise and redundancy. FS techniques can be filter-based, wrapper, and embedded. The integration of multiple FS techniques is denoted as hybrid FS.

The proposed DSAF-GS approach is a hybrid FS method that combines filter-based and wrapper techniques to achieve a representative and meaningful subset of genes. DSAF-GS uses Autoencoders (AE) as wrappers, along with statistical filters to remove redundant genes. An NN is trained on the remaining genes as an event predictor. Finally, the SHAP XAI method is used to evaluate the contribution of genes to NN decisions. The genes with the strongest contribution are selected.

10.2.6 Neural Networks

NNs are ML computational models inspired by the structure and function of the human brain. They consist of consecutive layers of interconnected artificial neurons, which process and transmit information through weighted connections. Training a NN amounts at providing a dataset of input-output pairs and identifying via proper optimization methods the NN parameters that minimize some given loss function, usually meant to measure the distance between

the output at hand and the result of the NN computation on the given input. In a research context, NNs can be trained on large datasets of genetic data to identify patterns and predict the effects of genetic mutations on traits of interest. These predictions can then be used to understand further the genetic basis of diseases and other phenotypic traits, and inform the development of personalized medical treatments. In addition, by using NNs and other computational and experimental methods (e.g., clustering, statistical analysis such as F-test or XAI), researchers can gain deeper insights into the complex interactions between genetics and biology.

Autoencoders (AE) are particular architectures of NNs that uncover the underlying structure of the data and generate a latent code for further analysis [344]. An AE maps an input to a lower dimensional representation (latent code). Such code is expected to have uncorrelated features, being able to reconstruct the original input data. Therefore, AEs can be used for dimensionality reduction, denoising, and data generation.

10.2.7 SHapley Additive exPlanation (SHAP)

The black-box nature of NNs often limits the interpretability of their results. Advances in XAI provide various methods for interpreting black-box models, offering a clearer understanding of their predictions. For example, Shapely Additive ex-Planation (SHAP) is a game theory-based approach for interpreting black-box models. SHAP determines the importance of a feature by observing the variations in predictions when the feature is included or excluded from the model. It assigns an importance value, called a SHAP value, to each feature based on its contribution to the predictions [279]. SHAP values are quantitative estimates indicating 'how' and 'how much' every single gene contributes to the model decisions, providing insight into the gene's role in the event prediction. The SHAP method provides a way to understand the underlying workings of NNs predictions, leading to improved insights and better decision-making.

An alternative XAI approach named LIME (Local Interpretable Model-Agnostic Explanations) [12] approximates the behavior of complex models via local interpretable explanations. Such explanations are obtained by fitting simpler and interpretable surrogate models with perturbed input data and observing the resulting changes in the model's predictions.

While LIME approximates the behavior of complex models with simpler ones, SHAP provides a more direct and explicit connection between feature importance and predictions. This transparency, along with a solid theoretical foundation rooted in game theory, enhances a deeper understanding of the underlying mechanisms driving the model's decisions.

10.2.8 Proposed Algorithm

DSAF-GS consists of the following steps (Figure 10.1):

1. The pairwise correlation (Pearson) was computed over the whole set of genes.

2. The resulting correlation matrix was clustered using hierarchical clustering such that similarly correlated genes belong to the same cluster.

3. For each cluster, all patient data were retrieved from the original dataset. An AE is then trained for each cluster using patients as features and genes as samples. The AEs selected the most representative gene of the respective cluster. Such most representative

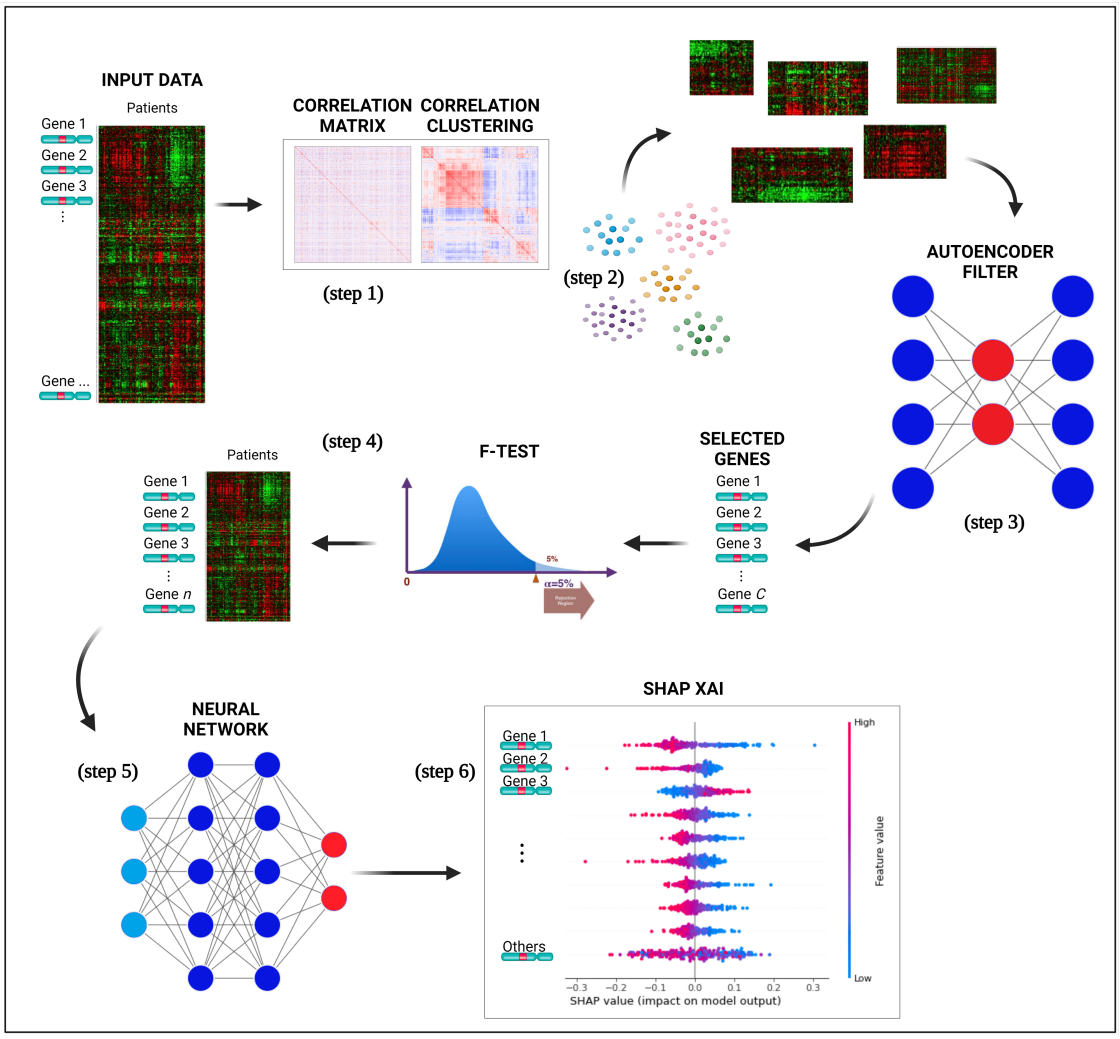


Figure 10.1: The pipeline proposed for selecting a subset of genes relevant to predict CLL events. The input data is used to compute the genes pairwise correlation matrix (step 1), and the correlation matrix is clustered (step 2) to group similarly correlated genes. The clusters are then mapped to the original input data and transposed. AEs are trained for each cluster to select the most representative gene, reducing dimensionality (step 3). The genes are ranked with F-test, selecting a subset with the highest F-value (step 4). A neural network is trained with a selected set of genes to perform binary classification of the CLL patients (event=0 and event=1) (step 5). The best NNs architecture is determined through model selection, and the SHAP XAI method explains each gene's importance in the predictions (step 6).

gene was the one associated with the lowest reconstruction error. This step eliminates redundant genes, hence reducing the dimensionality still working at the level of the original feature.

4. Genes selected in the previous step were then ranked with F-test. The F-value was computed by considering for each gene the ratio of the variance between and within the groups (event=0 and event=1). A subset of genes with the highest F-value was then selected. The final subset size was empirically selected by considering different subset sizes.

5. A NN was trained to perform binary classification on the event class using the previously selected set of genes as input variables and by considering the standard binary cross-entropy loss function. A model selection phase identified the most appropriate architecture of the NN. A grid-search approach is applied over a hyperparameters space defined by the number of layers and neurons per layer. In particular, we considered 2, 3, 4 layers and 8, 16, 32, 64, 128, 256 neurons for the first layer. In every successive layer, the number of neurons was determined as half of the number of neurons in the preceding layer. For each given configuration (layers/neurons), we built the corresponding NN whose performances were evaluated using a cross-validation (cv) algorithm, which assesses the model's ability to generalize by repeatedly training and testing on different subsets of the data for multiple iterations. The hyperparameters configuration is chosen as the one with the highest average performances (according to the average binary accuracy) over the cv iterations; the best model is finally selected as the one with the best performances among the cv iterations of the chosen configuration.

6. SHAP XAI method was used to explain the chosen NN classifier for the CLL event. SHAP evaluates the importance of each gene on the predictions by providing information on how such genes affect the prognosis.

Number of genes	Best accuracy (%)	95% Confidence Interval
5	79.54	70.88-77.74
10	84.09	72.60-78.30
50	86.36	65.60-75.30
100	79.54	65.82-72.36
300	75.00	66.36-70.90

Table 10.1: Models' accuracy in the binary classification of CLL event (i.e., therapy need or death).

10.2.9 Implementation

DSAF-GS algorithm has been implemented using the Python (v3.8.11) programming language. NNs have been implemented using the Tensorflow (v2.6.0) framework and the Keras library. XAI analysis was performed using the SHAP [279] library.

For GEP analysis, 500 clusters were identified. Each AE architecture consisted of five layers with the following number of neurons: 217, 43, 21, 43, and 217 respectively; relu was used as activation function and Adam as optimizer with a learning rate of 0.01; Mean Absolute Error was used as reconstruction loss and each AE was trained for 1000 epochs.

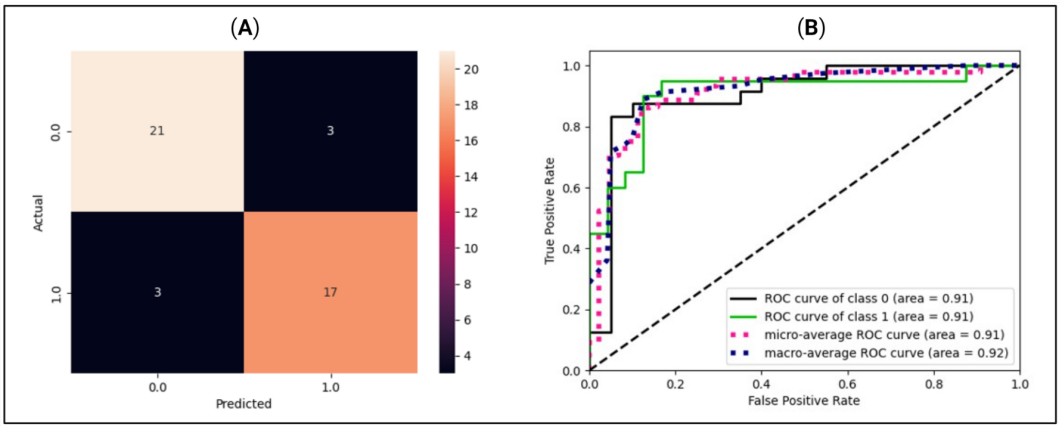


Figure 10.2: (A) Confusion matrix of model performance on the test set in predicting the event or non-event of new patients. Black squares refer to wrongly classified patients (false positives and false negatives), while colored squares refer to well-classified patients (true positives and true negatives). (B) ROC curves for the model. The graph plot sensitivity against specificity at various threshold settings. The classifier performs better as the curve approaches the upper left corner. An AUC value of 0.91 for GEP predictions indicates the solid overall performance of the model.

Out of the 500 genes selected by the AEs (one for each cluster), different subset sizes were used to train the NN event predictor. F-test was used to select a subset of genes of size 5, 10, 50, 100, and 300 (Table 10.1). Different NNs were trained for each gene subset. The best model for GEP takes in input 50 genes and consists of 4 layers of 46, 22, 12, and 1 neuron respectively.

During grid-search, Adam was used as an optimizer with a learning rate of 0.001 and relu was used as an activation function. A total of 10 cv iterations have been performed for each configuration by randomly splitting the data into a training set and test set (90% and 10% of the whole dataset respectively). While the train set has been enriched with synthetic samples (by using SMOTE) [294] to guarantee a balanced training of the NN, the test set only comprised real data samples. While the computation of SHAP values was found to be inefficient for NN models, the authors demonstrated that shapely values could be calculated through a weighted linear least square regression with a shapely kernel. Such a method was adopted for computing SHAP values using a subsample of 100 patients. Note, training and test sets are different parts of the same O-CLL dataset. Indeed, we are not aware of any further public dataset having the clinical and genomic information required by our method and that can be used as a validation set by fitting our needs and the prospective nature of our study.

10.3 Statistical analysis

TTFT was calculated during the watch and wait period from the date of the diagnosis to the date of therapy start or last follow-up. The prognostic impact of predictors was investigated by univariable and multiple Cox regression analysis. Data were expressed as hazard ratio (HR) and 95% confidence intervals (CIs). The predictive accuracy of the prognostic models was quantified by calculating the Harrell C-index (HC-index) ranging from 0.5 to 1.0 and the explained variation on the outcome (i.e., an index combining calibration and discrimination)

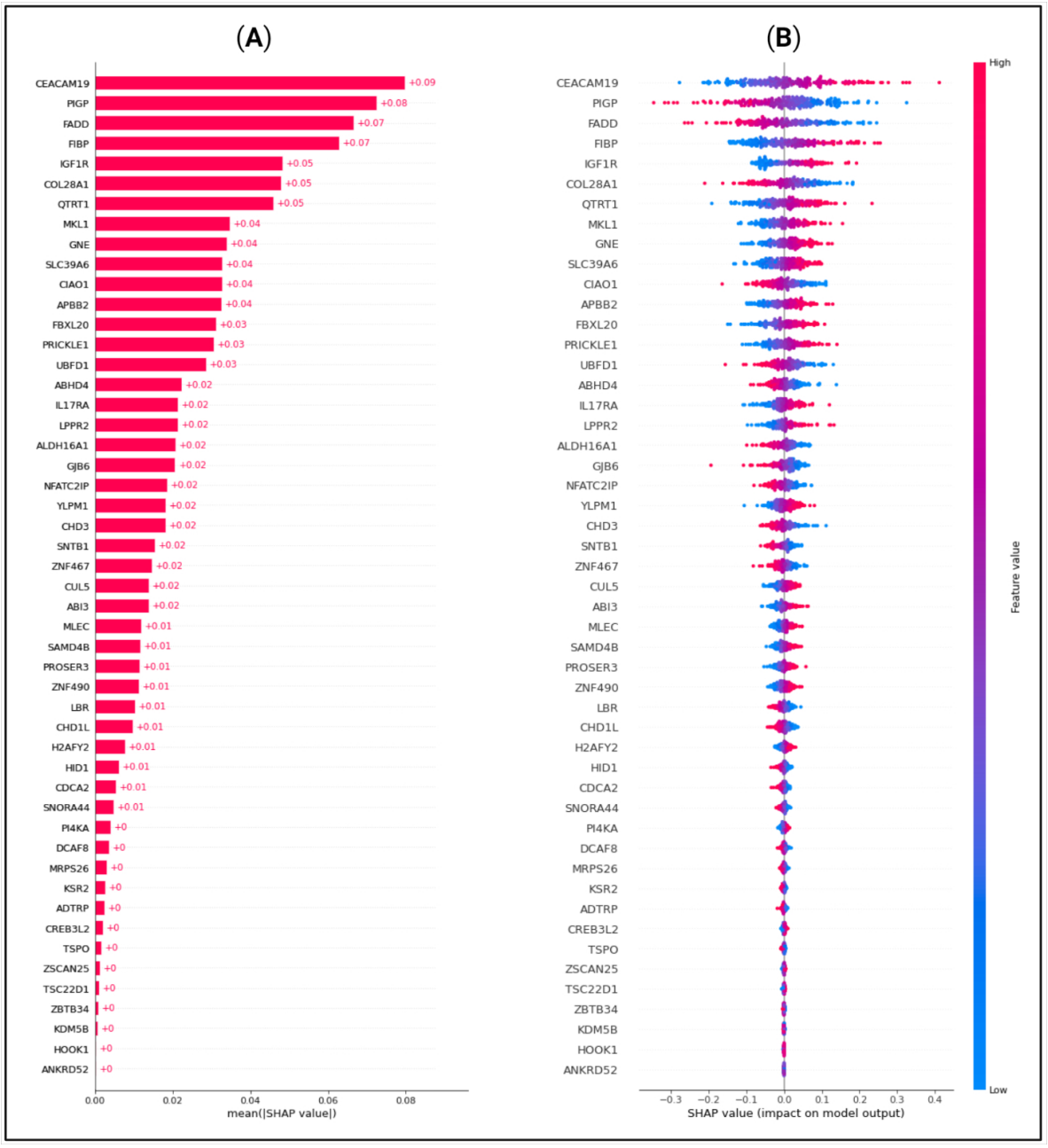


Figure 10.3: SHAP values were computed for the best model. (A) Waterfall plot of absolute mean SHAP values (average absolute importance of each gene in the model), (B) Beeswarm plot of SHAP values (shows how and how much each gene influences the predictions).

[345]. The improvement of model fitting due to the inclusion of specific genes was assessed by the log-likelihood ratio statistics. The receiver operator curves (ROC) in Figure 10.2B illustrate the model performance by plotting the actual positive rate (sensitivity) versus the false positive rate (1 - specificity).

A value of $P < .05$ was considered significant for all statistical calculations. Data analysis was performed by SPSS for Windows v.21, IBM, Chicago, Illinois, USA, and by Stata 16, StataCorp, Texas, USA.

10.4 Pathway and gene network analysis

Pathway analysis was performed using the Reactome Pathway Analysis tool (reactome.org) to group genes into specific pathways. Reactome, with statistical hypergeometric distribution test, determines whether certain pathways are over-represented in the submitted data. This test produces a probability score which is corrected for false discovery rate using the Benjamani-Hochberg method [336].

Gene network was constructed with the free NetworkAnalyst tool using IMEx Interactome (Literature-curated comprehensive data from InnateDB.

10.5 Results

10.5.1 Gene Selection by Explainable Artificial Intelligence

Of the 224 enrolled patients, 217 were used for the analysis. The remaining 7 were removed for defective gene profiles. The performance of the best model on the test set is reported in Table 1. The results are shown according to the number of genes used as independent variables (first column). The best model accuracy is 86.36% achieved using 50 genes as predictors with a sensitivity and specificity of 85% and 87.5%. The second column reports the 95% CIs computed using the model accuracy's mean and standard deviation over the cv iterations.

The model sensitivity and specificity are reported in Figure 10.2A terms of confusion matrix and ROC curve respectively. Despite the 3 false positives and 3 false negatives, the model is capable of detecting the underlying patterns in the data as shown by the overall performance. Figure 10.2B shows that the models have an area under the curve (AUC) of 0.91.

Figure 10.3A shows a waterfall plot of absolute mean SHAP values reporting the average importance of each gene in the model evaluated using SHAP. Genes are reported in order of importance. For example, the model was strongly influenced by CEACAM19 and PIGP (+0.09 and +0.08), moderately influenced by MKL1 (alias of MRTFA gene) and GNE (+0.04), and poorly influenced by others (< 0.03).

Moreover, as shown in Figure 10.3B higher values of the gene CEACAM19 are associated with positive SHAP values, meaning that they will increase the prediction towards the occurrence of the event. Moreover, lower values of the variable are associated with negative SHAP values, meaning that they will decrease the prediction towards the absence of an event. Conversely, for the PIGP gene, lower values are associated with positive SHAP values increasing the prediction towards the event's occurrence. PIGP higher values are asso-

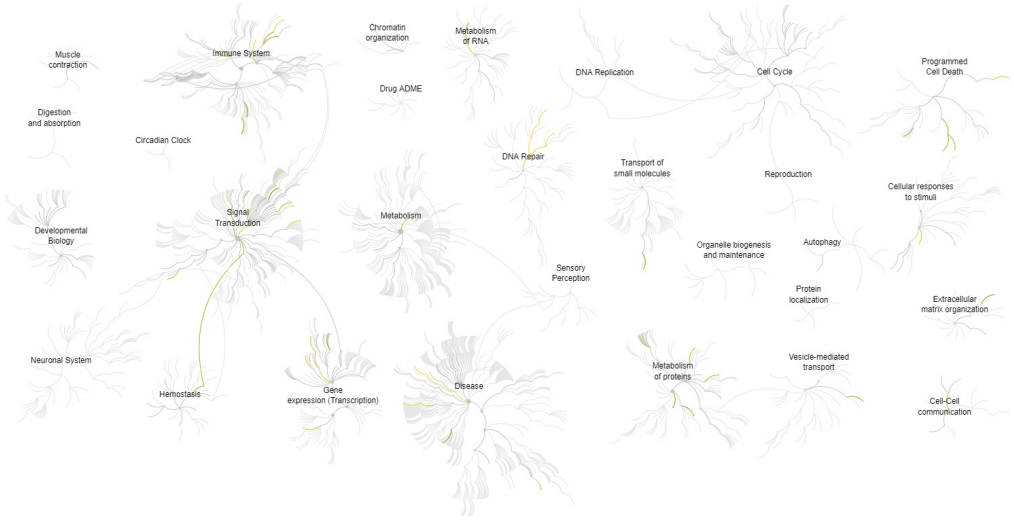


Figure 10.4: Pathways overview based on the Reactome database of the top 10 genes identified by the (SHAP) XAI method. A genome-wide overview of the results of pathway analysis is shown. Reactome pathways are arranged in a hierarchy. The center of each of the circular "bursts" is the root of one top-level pathway, for example, Cell Cycle. Each step away from the center represents the next lower level in the pathway hierarchy. The color code denotes the over-representation of that pathway in the input dataset. The closer the color is to yellow, the more significant the over-represented pathway is; light grey indicates pathways that are not significantly over-represented.

ciated with negative SHAP values, meaning they will decrease the forecast towards no event occurrence.

Gene name	Chromosome	Gene start (bp)	Gene end (bp)	Gene description
CEACAM19	19	44662278	44684359	CEA cell adhesion molecule 19
PIGP	21	37062846	37073170	Phosphatidylinositol glycan anchor biosynthesis class P
FADD	11	70203296	70207390	Fas associated via death domain
FIBP	11	65883740	65888531	FGF1 intracellular binding protein
IGF1R	15	98648539	98964530	Insulin like growth factor 1 receptor
COL28A1	7	7356203	7535873	Collagen type XXVIII alpha 1 chain
QTRT1	19	10701430	10713437	Queuine tRNA-ribosyltransferase catalytic subunit 1
MKL1 (MRTFA)	22	40410281	40636719	Myocardin related transcription factor A
GNE	9	36214441	36277042	Glucosamine (UDP-N-acetyl)-2-epimerase/N-acetylmannosamine kinase
SLC39A6	18	36108531	36129385	Solute carrier family 39 member 6

Table 10.2: Description and localization of the top ten genes derived from SHAP analysis.

10.5.2 Pathways and networks overview based on Reactome database of the top 10 genes

The description and chromosome localization of the top ten genes are described in Table 10.2. Eight of the top ten genes (i.e., FADD, FIBP, FIBP, GNE, IGF1R, MKL1, PIGP, SLC39A6) selected by the NN algorithm were found in the Reactome pathway database, showing an involvement in various pathways such as signal transduction, gene expression (transcription), protein metabolism, immune system, cell cycle, and apoptosis (Figure 10.4).

Interestingly, seven (i.e., FADD, FIBP, IGF1R, QTRT1, GNE, SLC39A6, MKL1) of the top 10 genes appear to be connected into a complex network as shown in Figure 10.5 by the

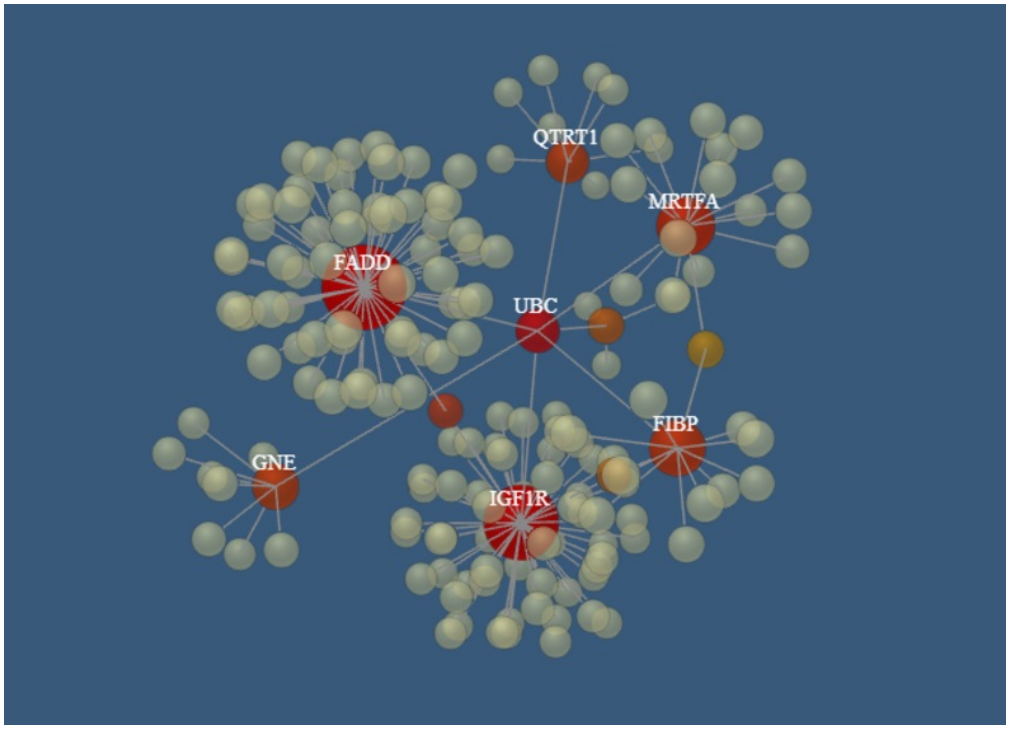


Figure 10.5: The PPI networks created by FADD, FIBP, IGF1R, QTRT1, GNE, SLC39A6, and MRTFA genes. Node size and color correspond to the number of connected edges; gene name is displayed only for nodes with ≥ 4 edges, and the closer the color is to red, the bigger the node size is.

network model of the 3D protein-protein interaction (PPI) explored using the NetworkAnalyst tool.

10.5.3 Multivariate analysis of the top 10 genes

The top 10 genes selected by the NN models were chosen to estimate their prognostic influence on noticeable clinical and biomolecular variables (named basic prognostic model) consisting of IGHV mutational status, del(11q) and del(17p), NOTCH1 mutation, β2-M, Rai stage, and B-lymphocytosis. Table 10.3 shows their prediction power on TTFT in univariable analysis.

As expected, all ten top genes showed a predictive power on TTFT in univariable analyses (Figure 10.5). Specifically, while COL28A1(HR 0.32, 95% CI 0.12-0.82, P=0.018), FADD (HR 0.21, 95% CI 0.07-0.62, P=0.005), and PIGP (HR 0.39, 95% CI 0.15-0.98, P=0.047) high expression was associated with a reduced risk to be treated, the remaining genes showed an inverse prognostic association with therapy need. However, CEACAM19 (HR 2.44, 95% CI 0.84-7.14, P=0.10), PIGP (HR 0.57, 95% CI 0.19-1.72, P=0.32), FADD (HR 0.45, 95% CI 0.12-1.64, P=0.22), FIBP (HR 1.95, 95% CI 0.96-3.96, P=0.06), MKL1 (HR 2.31, 95% CI 0.73-7.34, P=0.15), GNE (HR 1.86, 95% CI 0.82-4.26, P=0.14), and SLC39A6 (HR 1.44, 95% CI 0.64-3.22, P=0.37) lost their independent predictive power when analyzed with variables belonging to the basic prognostic model. Conversely, IGF1R (HR 1.41, 95% CI 1.08-1.84, P=0.013), COL28A1 (HR 0.32, 95% CI 0.10-0.97, P=0.045), and QTRT1 (HR 7.73, 95% CI 2.48-24.04, P<0.001) genes were significantly associated with TTFT in multivariable analyses (Table 10.4). When these three significant genes were evaluated in a final multivari-

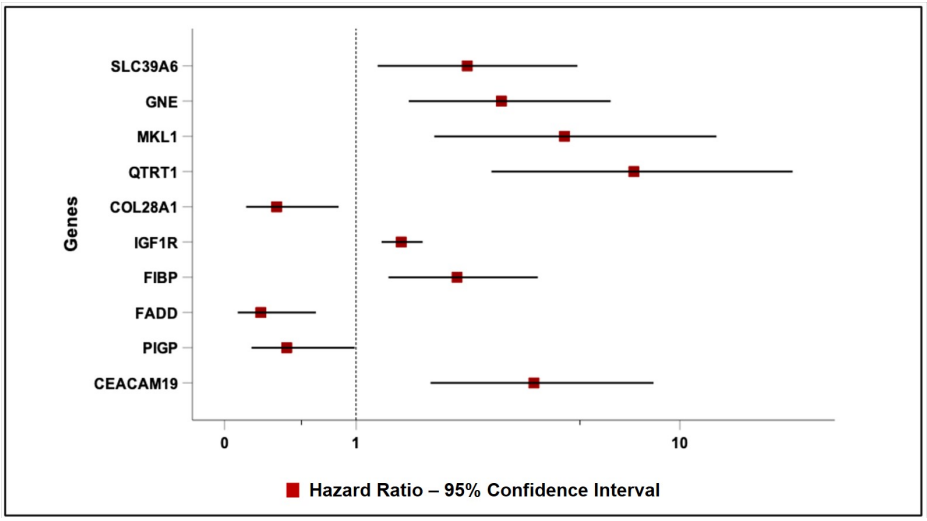


Figure 10.6: Forest plot of Cox univariable analysis for time to TTFT according to the top 10 genes selected by the NN algorithm.

able model including the basic prognostic variables, COL28A1 along with B-lymphocytosis lost its significance, while IGF1R and QTRT1 maintained their prognostic independence (Table 10.4).

The basic prognostic model provided an HC-index of 76.5% and an explained variation to predict the TTFT of 42.2%. When the three significant genes (i.e., IGF1R, COL28A1, and QTRT1) were jointly considered in the final multivariable model, the HC-index and the explained variation significantly increased to 78.6% and 52.6%, respectively, along with an improvement of the goodness of model fit (χ2=20.1, P=0.002). In a more parsimonious model only including IGF1R and QTRT1 (i.e., the two genes that remained significantly associated with the TTFT in the final model) and excluding CLO28A1, HC-index (78.2%) and the explained variation (52.4%) retained a better performance as compared with the basic prognostic model with a concomitant raising of the goodness of the model fit (χ2=18.8, P=0.001).

	HR	LL 95% CIs	UL 95% CIs	P-value
IGHV unmutated	5.35	3.95	7.26	<.001
del(11q)	5.31	3.51	8.04	<.001
del(17p)	5.20	2.42	11.17	<.001
NOTCH1 mutated	2.51	1.74	3.61	<.001
β2-M abnormal	2.21	1.56	3.14	<.001
Rai stage I-II	1.92	1.39	2.65	<.001
B-Lymphocytes (>5×10^9/L)	2.15	1.48	3.12	<.001

Table 10.3: Univariable Cox analyses for time to first treatment of several well-known clinical and biomolecular variables belonging to the basic prognostic model.

Variable	HR	LL 95% CI	UL 95% CI	P-value
Model 1				
IGHV unmutated	2.03	1.12	3.70	0.02
del(11q)	3.92	1.77	8.69	<0.001
del(17p)	11.86	2.51	56.02	0.002
NOTCH1 mutated	2.07	1.11	3.86	0.021
β2-M abnormal	2.10	1.33	3.31	0.001
Rai stage I-II	1.63	1.00	2.68	0.05
B-Lymphocytes $>5\times10^9$/L	1.75	0.88	3.48	0.112
IGF1R gene	1.41	1.08	1.84	0.013
Model 2				
IGHV unmutated	2.78	1.58	4.89	<0.001
del(11q)	2.88	1.34	6.20	0.007
del(17p)	7.28	1.60	33.10	0.01
NOTCH1 mutated	2.26	1.20	4.22	0.011
β2-M abnormal	2.04	1.300	3.20	0.002
Rai stage I-II	1.70	1.04	2.78	0.035
B-Lymphocytes $>5\times10^9$/L	1.42	0.72	2.80	0.313
COL28A1 gene	0.32	0.10	0.97	0.045
Model 3				
IGHV unmutated	2.465	1.40	4.34	0.002
del(11q)	2.42	1.12	5.21	0.024
del(17p)	6.78	1.48	31.15	0.014
NOTCH1 mutated	2.22	1.19	4.13	0.012
β2-M abnormal	1.79	1.13	2.85	0.013
Rai stage I-II	1.93	1.17	3.18	0.01
B-Lymphocytes $>5\times10^9$/L	1.60	0.81	3.17	0.177
QTRT1 gene	7.73	2.48	24.04	<0.001
Final Model				
IGHV unmutated	1.93	1.06	3.50	0.031
del(11q)	3.10	1.39	6.92	0.006
del(17p)	8.44	1.75	40.75	0.008
NOTCH1 mutated	2.21	1.17	4.14	0.014
β2-M abnormal	1.97	1.23	3.15	0.005
Rai stage I-II	1.71	1.03	2.83	0.038
B-Lymphocytes $>5\times10^9$/L	1.85	0.91	3.74	0.088
IGF1R	1.38	1.07	1.79	0.014
COL28A1	0.52	0.16	1.67	0.273
QTRT1	6.70	2.12	21.21	0.001

Table 10.4: Cox multivariable analyses for time to first treatment (TTFT).

10.6 Discussion

The considerable innovations in genomics engendering a vast and miscellaneous bulk of information from sizable cohorts of patients, and the concurrent computer science knowledge improvements have guided the growing use of AI and more specifically of ML approaches

that acquire knowledge from available data, devising variables selections without pre-setting programming. Well-defined examples of the ML approach in the analysis of hematological malignancies are the association of BCL6 and PDL1/2 rearrangements in primary testicular diffuse large B-cell lymphoma (DLBCL) with central nervous system relapse [346]; the involvement of six prognosis-related long non-coding genes in acute myeloid leukemia (AML) patients [347]; or the relevance of tumor mutation burden for the DLBCL overall survival prognostication [348]. In CLL, the ML algorithm identified six hub genes as possible biomarkers to improve the diagnosis [349]. Moreover, baseline clinical data added to the international prognostic index for CLL (CLL-IPI) variables demonstrated improved predictive performance over CLL-IPI using a range of ML boosting algorithms to identify the individual risk of death, treatment, infection, and a combination of them [350]. In contrast, no additional improvement was observed when comprising recurrent genetic mutation information [351]. Moreover, an ML algorithm called CLL Treatment Infection Model (CLL-TIM) was applied to recognize patients at high risk of infection and/or treatment based on CLL-IPI variables and routine clinical data [352].

However, differently from our prospective study, the CLL-IPI score system only included 32% of Binet stage A patients and more importantly, 4% of IGHV mutated cases, thereby rendering it less representative of the real-world setting and may lower the TTFT's predictive efficacy. In contrast, the Brno-Barcelona cohort [353] had a significantly higher enrichment of early-stage/low-risk Binet A (83%) and IGHV mutated cases (43%) as well as the German CLL study group which also developed a predictive model for newly diagnosed Binet stage A patients [350] with roughly 71% of the population having an IGHV mutation status.

Herein, we selected the top 10 genes (CEACAM19, PIGP, FADD, FIBP, FIBP, GNE, IGF1R, MKL1, PIGP, SLC39A6) from a GEP dataset of 217 CLL cases comprising roughly 20000 genes using a novel deep ML-based approach to estimate how much every single gene had a role in predicting the therapy need occurrence. The GEP model was strongly influenced by CEACAM19 and PIGP (SHAP value +0.09 and +0.08) in making decisions, moderately influenced by MKL1 and GNE (SHAP value +0.04), and poorly influenced by others (SHAP value $<$0.03). IGF1R, COL28A1, and QTRT1 influenced quite the GEP model (SHAP value +0.05).

Some variables namely Rai stage, IGHV mutational status, β2-M, and 17(p) and 11(q) deletions previously validated in the CLL-IPI score system and by our group were used as basic risk model in predicting TTFT. We found that IGF1R, COL28A1, and QTRT1 genes maintained their own independent prognostic value in predicting the time-to-event when tested in a multivariable model including the variables of the basic prognostic model. However, in a final multivariable model in which the three genes (IGF1R, COL28A1, and QTRT1) were tested all together with the prognostic variables of the basic model, IGF1R and QTRT1, but not COL28A maintained their predictive independence on TTFT. Noteworthy, the presence of these genes in the model significantly increased the prognostic accuracy of a basic risk model. In this regard, the HC-index and the explained variation significantly increased from 76.5% in the basic model to more than 78% and from 42.2% to roughly 52% in the IGF1R/QTRT1-gene model. These data indicate that the IGF1R/QTRT1-gene model retained a better performance than the basic prognostic model.

IGF1R encoding the insulin-like growth factor 1 receptor (IGFR1) is not only implicated in numerous cellular bio-functional processes, i.e., growth, proliferation, differentiation, and apoptosis [354], but also it plays a critical role in cancer development, progression, and metastasis [355]. Moreover, IGF1R is involved in CLL and overexpressed in various

CLL cell subsets. Its inhibition induced apoptosis and efficiently reduced CLL growth in an Eμ-TCL1 transgenic murine model [356]. Moreover, IGF1R seems to be a direct target of sorafenib since the latter decreased its expression and phosphorylation by offsetting the insulin-like growth factor-1 binding to CLL cells and ultimately dropping the in vitro IGF1R kinase activity [357]. Finally, we previously demonstrated the IGF1R gene expression as an independent prognostic factor related to TTFT in our O-CLL prospective cohort after a shorter follow-up.

Unlike the IGF1R gene, QTRT1 encoding the queuine tRNA-ribosyltransferase 1, a key enzyme involved in the post-transcriptional modification of tRNAs, has never been implicated in the pathogenesis or prognosis of CLL. Conversely, a significant increase in QTRT1 expression and a striking down-regulation in its methylation was also found in lung cancer. Furthermore, it was discovered to be a risk factor for the disease onset and progression and adversely associated with survival outcomes.

Among various CLL prognostic models involving genes, Herold et al. provided evidence of the association between overall survival and TTFT and the expression of 8 genes in CLL cells (PS.8 score). For TTFT, PS.8 showed an improved prognostic effect than the single parameters and even to a combined FISH and IGVH status model, which in turn failed to increase the performance of the PS.8 score in a multivariable analysis.

Huang et al. showed that high NRIP1, BCL11B, and SIRT1 expressions were associated with more prolonged survival, while high expression of CDKN2A and SREBF2 with a poor prognosis. However, a substantial fraction of patients in the dataset chosen by the authors was not analyzed at the diagnosis/first presentation but at the time of progressive disease or relapse. Conversely, patients of our O-CLL cohort were prospectively followed-up, and all the biomolecular analyses were performed at the disease onset. Moreover, both Herold's and Huang's studies did not consider, unlike our study, unavailable risk factors included in the CLL-IPI score, somewhat misinterpreting the final results.

Two recently published papers represent interesting innovations in CLL's gene-oriented prognosis. Liang X et al. following the super-enhancer (SE) new hypothesis generated a prognostic score to predict the time-to-therapy-need in CLL by the expression levels of nine SE-associated genes. Yet, since several data suggest the high dependency of CLL cells on microenvironment support, Abrisqueta and coll. described the prediction power of a signature for predicting progression based on the analysis of two hundred genes linked to microenvironment signaling by the NanoString approach. This novel approach established a 15 genes-based signature that predicted disease outcome independently of the IGHV mutational status, the CLL-IPI, and the International Prognostic Score for Early-stage (IPS-E) CLL score. Notably, the nanoString platform, overcoming GEP methodological drawbacks and reproducibility, could represent the future, facilitating its use in clinical settings.

Notably, several pathways involved in cancer and hematopoietic malignancies development were identified by Reactome analysis of the top ten genes analyzed in this study, including Interferon alpha/beta signaling, caspases, and Rho GTPase activity, GHR signaling pathway, Integrin signaling, non-receptor Tyrosine Kinases activity, and FGF/FGFR pathways. Moreover, among the top ten genes, FIBP was found to be overexpressed in a specific group of CLL patients affected by a large loss at the 13q14 locus; as previously noted, also IGF1R was identified as

overexpressed in various CLL subsets, suggesting a contribution to CLL pathology. Finally, seven of the top 10 genes (which appeared to be connected into a complex PPI network) are in turn interconnected by the UBC gene encoding for Polyubiquitin C, which represents

one of the sources of ubiquitin in human cells. Polyubiquitin C plays a crucial role in maintaining cellular ubiquitin levels, especially during the stress response. The process of ubiquitination has been associated with protein degradation, DNA repair, cell cycle regulation, kinase modification, endocytosis, and regulation of other cell signaling pathways. Interestingly, Zhang et al., by bioinformatic analysis of gene expression profiles in CLL cells, identified the UBC gene as the key node in the PPI network of genes up-regulated in B cells co-stimulated with immobilized anti-IgM with respect to untreated cells, revealing the proteasome pathway as the most significant in this network.

Finally, it should be emphasized that this preliminary study lacks a validation cohort. At the best of our knowledge, we are not aware of any further public dataset fitting the prospective nature of our study, as well as the clinical and genomic information required to answer our aims. Specifically, the GEO dataset (GSE39671) has not the characteristics required to run the method presented in this paper, which are instead collected in the ICGC CLL dataset. However, in the latter, sampling was completed within a year, while in our study, in about 26% of Binet A untreated CLL cases. In contrast, the median sampling time for the remaining cases of the ICGC CLL cohort was approximately 5 years (IQR 2.6-9.1), a bias that might invalidate the analysis' conclusions. Moreover, information on the Rai stage system is lacking, and the Binet stage information at sampling is not available.

10.7 Conclusion

A novel deep ML-based approach was proposed in the current analysis, exploiting the reconstruction capabilities of AEs and XAI to select the most informative genes in predicting the therapy need event. This study strengths lie in the use of an original ML method and the prospective nature of our study. Nevertheless, these results, although preliminary, evidenced the effectiveness of this approach in identifying genes with independent predictive power, suggesting a set of meaningful genes for further investigation. Finally, it should be emphasized that this pilot study requires external validation using a different prospective cohort of patients with similar characteristics.

Conclusion

The research presented in this thesis has made significant contributions to the fields of GenAI and XAI, showcasing innovative applications and theoretical advancements. Throughout the exploration of these technologies, several key achievements have emerged, highlighting the transformative potential of GenAI and XAI in addressing real-world problems in healthcare, materials design, and complex systems.

One of the notable advancements is the development of GIDNET, a generative neural network architecture specifically designed to tackle inverse design problems. By exploring a well-structured latent space, GIDNET has demonstrated superior performance in generating feasible design solutions compared to earlier methods. This approach provides more accurate designs, making it a valuable tool for various engineering and scientific applications. Future work in this context will aim to further refine the latent space exploration techniques by taking advantage of new approaches in contrastive learning. Moreover, the application of GIDNET will expand to more complex design problems such as molecule and drugs design.

In the healthcare domain, the integration of VAE with a context predictor and an LLM has led to the creation of a framework for automatic medical report generation. This innovative approach has shown good performance in generating coherent and contextually relevant medical reports from patient images, which makes it a potential tool for optimizing the documentation process to improve patient care. Further research will aim to extensively test the approach on a wider range of benchmarks, comparing its performances against the ones of state-of-the-art architectures.

The introduction of Generative Agents into agent-based modeling represents another significant breakthrough. These agents, able to show human-like behaviors, enhance the realism and depth of social simulations. This research highlights the importance of robust validation frameworks to ensure the plausibility and coherence of these simulations, which are crucial for studying complex social interactions and dynamics. Future research should focus on the development of systematic validation approaches to enhance the applicability of coherent and believable generative agents.

Additionally, the investigation into compact strategies for opinion diffusion within social networks has yielded valuable insights. Measures such as vote rank and betweenness centrality have proven effective in maximizing opinion diffusion across different network configurations. These findings offer practical implications for designing strategies to influence public opinion and disseminate information effectively in social networks. In addition, as a future avenue of research, the proposed opinion diffusion model can be an interesting framework for the application of LLM-based generative agents.

In the field of healthcare, the thesis also introduced a novel deep learning-based approach for feature selection in genomics. By utilizing XAI techniques, this method has demonstrated effectiveness in identifying genes with significant predictive power. This approach not only enhances the accuracy of disease prediction models but also provides valuable insights into the biological mechanisms underlying various conditions, paving the way for more targeted and personalized treatments.

Other works related to XAI and data interpretation have been discussed through applications in the fields of firms' bankruptcy prediction, natural disasters prediction, nutritional science data analysis, and 3D image segmentation in anatomy.

The contributions of this PhD thesis underscore the transformative potential of Generative GenAI and XAI. Advancements in these fields are driving a revolution in the real world, pushing the boundaries of what AI can achieve and how. The field of AI is undergoing a

significant shift from the development of slow thinker machines, which emulate deep cognitive processes, to fast thinker machines, which prioritize speed and efficiency through pattern recognition and prediction. In particular, fast thinking is characterized by quick, intuitive responses based on pattern recognition and correlation. It has become increasingly prevalent in AI applications being driven by the effectiveness of generative models, such as LLMs, which leverage vast amounts of data to predict and generate content without necessarily understanding the underlying meaning. These models perform well in tasks requiring immediate and accurate predictions, demonstrating impressive capabilities in producing coherent texts, images, and even music. Fast solvers do not reason about the problem instance but just rely on past experience. Slow solvers, on the other hand, reason about the problem instance and its features, usually employing a logic-based and symbolic approach to generate a solution. The success of fast thinking AI challenges the long-held belief that replicating human-like cognitive functions, such as consciousness and deep understanding, is essential for creating intelligent machines. Instead, AI's ability to perform through correlation and prediction has proven sufficient for many practical applications, reshaping our understanding of what constitutes intelligence.

The reliance on inference and prediction raises important questions about the sufficiency of these methods in AI. While fast thinking AI can achieve remarkable results, its dependency on pattern recognition and data-driven predictions may limit its understanding and reasoning capabilities. Critics argue that without true comprehension, AI systems may lack the depth needed for more complex and nuanced tasks. However, proponents believe that the efficiency and speed of inference-based AI are significant advantages that outweigh the need for deeper cognitive functions in many contexts.

Ultimately, the adequacy of inference in AI depends on the application. In scenarios where rapid decision-making and pattern recognition are critical, inference-based models perform exceptionally well. However, for tasks requiring deep understanding, long-term memory, and intricate reasoning, the current state of AI may still fall short, necessitating further advancements in slow thinking methodologies.

The thesis also explores highly relevant topics in today's society and industry, such as the design of new materials and data generation (e.g. in agent-based simulations). According to recent surveys by Gartner[1], these areas are central to modern technological and industrial advancements. Generative AI techniques are revolutionizing material science by discovering novel materials with desired properties, significantly impacting industries like automotive, aerospace, and electronics.

Furthermore, the generation of synthetic data by AI addresses privacy concerns and improves data availability for research and development. By creating artificial datasets that mimic real-world data, AI can support advancements in fields like healthcare, where patient privacy is paramount. These innovations underscore the transformative potential of AI in both scientific research and practical applications, highlighting the critical role of generative AI in shaping the future of technology.

The work conducted for this PhD thesis has witnessed the advent of LLMs along with the significant impact that genAI is having. Although still limited in scenarios where reasoning and causal relationships are needed more than correlation, LLMs are paving the way and making strides towards the realization of general artificial intelligence, which aims to create

[1] https://www.gartner.com/en/articles/beyond-chatgpt-the-future-of-generative-ai-for-enterprises

machines that fully replicate human abilities. In a plausible scenario, genAIs can achieve new levels of sophistication through an innovative approach that encourages their interaction[2].
By facilitating communication between different AIs, we can establish an ecosystem where each AI learns and improves from one another. This ongoing exchange of information and strategies enables AIs to refine their abilities, optimize their responses, and acquire new skills without human intervention and making them more versatile and effective in managing complex tasks, answering questions, and solving problems in increasingly innovative ways.
By advancing the state-of-the-art in these fields, this research opens new avenues for future exploration, promising significant impact in a large number of applications.

[2]https://life.unige.it/IA-autodidatte

Bibliography

[1] EM Rathje, SS Secara, JG Martin, S van Ballegooey, and J Russel. Liquefaction-induced horizontal displacements from the canterbury earthquake sequence in new zealand measured from remote sensing techniques. *Earthquake Spectra*, 33(4):1475–1494, 2017.

[2] MG Durante and EM Rathje. An exploration of the use of machine learning to predict lateral spreading. *Earthquake Spectra*, 37(4):2288–2314, 2021.

[3] Andrew Lininger, Michael Hinczewski, and Giuseppe Strangi. General inverse design of layered thin-film materials with convolutional neural networks. *ACS Photonics*, 8(12):3641–3650, 2021.

[4] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[5] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[6] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016.

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[8] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022.

[9] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

[10] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina,

Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

[11] Amina Adadi and Mohammed Berrada. Peeking inside the black-box: a survey on explainable artificial intelligence (xai). *IEEE access*, 6:52138–52160, 2018.

[12] M. T. Ribeiro, S. Singh, and C. Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–44, 2016.

[13] SM Lundberg and S-I Lee. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems December 4-9 2017 Long Beach CA USA*, pages 4765–4774, 2017.

[14] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

[15] Carlo Adornetto and Gianluigi Greco. Gidnets: generative neural networks for solving inverse design problems via latent space exploration. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 3404–3413, 2023.

[16] Carlo Adornetto, Antonella Guzzo, and Andrea Vasile. Automatic medical report generation via latent space conditioning and transformers. In *2023 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, pages 0428–0435. IEEE, 2023.

[17] Chen Gao, Xiaochong Lan, Zhihong Lu, Jinzhu Mao, Jinghua Piao, Huandong Wang, Depeng Jin, and Yong Li. S3: Social-network simulation system with large language model-empowered agents, 2023.

[18] Carlo Adornetto, Valeria Fionda, and Gianluigi Greco. On the effectiveness of compact strategies for opinion diffusion in social environments. In *ECAI 2023*, pages 11–18. IOS Press, 2023.

[19] Catia Morelli, Ennio Avolio, Angelo Galluccio, Giovanna Caparello, Emanuele Manes, Simona Ferraro, Antonella Caruso, Daniela De Rose, Ines Barone, Carlo Adornetto, et al. Nutrition education program and physical activity improve the adherence to the mediterranean diet: impact on inflammatory biomarker levels in healthy adolescents from the dimenu longitudinal study. *Frontiers in Nutrition*, 8:685247, 2021.

[20] Pierangela Bruno, Edoardo De Rose, Carlo Adornetto, Francesco Calimeri, Sandro Donato, Raffaele Giuseppe Agostino, Daniela Amelio, Riccardo Barberi, Maria Carmela Cerra, Maria Caterina Crocco, Mariacristina Filice, Raffaele Filosa, Gianluigi Greco, Sandra Imbrogno, and Vincenzo Formoso. μ-net: A deep learning-based architecture for μ-ct segmentation, 2024.

[21] Maria Giovanna Durante, Giovanni Terremoto, Carlo Adornetto, Gianluigi Greco, and Ellen M Rathje. A new graph neural network (gnn) based model for the evaluation of lateral spreading displacement in new zealand. *Japanese Geotechnical Society Special Publication*, 10(21):776–780, 2024.

[22] Carlo Adornetto and Gianluigi Greco. A new deep learning and xai-based algorithm for features selection in genomics. *arXiv preprint arXiv:2303.16914*, 2023.

[23] Fortunato Morabito, Carlo Adornetto, Paola Monti, Adriana Amaro, Francesco Reggiani, Monica Colombo, Yissel Rodriguez-Aldana, Giovanni Tripepi, Graziella D'Arrigo, Claudia Vener, et al. Genes selection using deep learning and explainable artificial intelligence for chronic lymphocytic leukemia predicting the need and time to therapy. *Frontiers in Oncology*, 13, 2023.

[24] Filippo Chiarello, Paola Belingheri, and Gualtiero Fantoni. Data science for engineering design: State of the art and future directions. *Computers in Industry*, 129:103447, 2021.

[25] Sharon C. Glotzer. Data science for assembly engineering. In *Proc. of KDD'21*, page 2, 2021.

[26] Bhuvanesh Sridharan, Manan Goel, and U. Deva Priyakumar. Modern machine learning for tackling inverse problems in chemistry: molecular design to realization. *Chemical Comm.*, 58:5316–5331, 2022.

[27] Gabriele M. Coli, Emanuele Boattini, Laura Filion, and Marjolein Dijkstra. Inverse design of soft materials via a deep learning-based evolutionary strategy. *Science Advances*, 8(3):eabj6731, 2022.

[28] Arindam Debnath, Adam M Krajewski, Hui Sun, Shuang Lin, Marcia Ahn, Wenjie Li, Shanshank Priya, Jogender Singh, Shunli Shang, Allison M Beese, et al. Generative deep learning as a tool for inverse design of high entropy refractory alloys. *Journal of Materials Informatics*, 1(1):3, 2021.

[29] Vinothkumar Sekar, Mengqi Zhang, Chang Shu, and Boo Cheong Khoo. Inverse design of airfoil using a deep convolutional neural network. *Aiaa Journal*, 57(3):993–1003, 2019.

[30] Anand Balu Nellippallil, Kevin N Song, Chung-Hyun Goh, Pramod Zagade, BP Gautham, Janet K Allen, and Farrokh Mistree. A goal-oriented, sequential, inverse design method for the horizontal integration of a multistage hot rod rolling system. *Journal of Mechanical Design*, 139(3):031403, 2017.

[31] Sean Molesky, Zin Lin, Alexander Y Piggott, Weiliang Jin, Jelena Vucković, and Alejandro W Rodriguez. Inverse design in nanophotonics. *Nature Photonics*, 12(11):659–670, 2018.

[32] Juhwan Noh, Geun Ho Gu, Sungwon Kim, and Yousung Jung. Machine-enabled inverse design of inorganic solid materials: promises and challenges. *Chemical Science*, 11(19):4871–4881, 2020.

[33] Nathan A Mahynski, Runfang Mao, Evan Pretti, Vincent K Shen, and Jeetain Mittal. Grand canonical inverse design of multicomponent colloidal crystals. *Soft Matter*, 16(13):3187–3194, 2020.

[34] Peter R. Wiecha, Arnaud Arbouet, Christian Girard, and Otto L. Muskens. Deep learning in nano-photonics: inverse design and beyond. *Photonic Research*, 9(5):B182–B200, 2021.

[35] Jiaqi Jiang, Mingkun Chen, and Jonathan A. Fan. Deep neural networks for the evaluation and design of photonic devices. *Nature Reviews Materials*, 2020.

[36] Rafael Gómez-Bombarelli, Jennifer N. Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D. Hirzel, Ryan P. Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, 2018. PMID: 29532027.

[37] Wei Ma, Feng Cheng, Yihao Xu, Qinlong Wen, and Yongmin Liu. Probabilistic representation and inverse design of metamaterials based on a deep generative model with semi-supervised learning strategy. *Advanced Materials*, 31(35):1901111, 2019.

[38] C. Zhang, J. Jin, W. Na, Q. Zhang, and M. Yu. Multivalued neural network inverse modeling and applications to microwave filters. *IEEE Trans. on Microwave Theory and Techniques*, 66(8):3781–3797, 2018.

[39] Dianjing Liu, Yixuan Tan, Erfan Khoram, and Zongfu Yu. Training deep neural networks for the inverse design of nanophotonic structures. *ACS Photonics*, 5(4):1365–1369, 2018.

[40] Christopher Yeung, Ju-Ming Tsai, Brian King, Benjamin Pham, David Ho, Julia Liang, Mark W. Knight, and Aaswath P. Raman. Multiplexed supercell metasurface design and optimization with tandem residual networks. *Nanophotonics*, 10(3):1133–1143, 2021.

[41] Zhaocheng Liu, Dayu Zhu, Sean P. Rodrigues, Kyu-Tae Lee, and Wenshan Cai. Generative model for the inverse design of metasurfaces. *Nano Letters*, 18(10):6570–6576, 2018.

[42] Jiaqi Jiang, David Sell, Stephan Hoyer, Jason Hickey, Jianji Yang, and Jonathan A. Fan. Free-form diffractive metagrating design based on generative adversarial networks. *ACS Nano*, 13(8):8872–8878, 2019.

[43] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

[44] A.H. Zaabab, Qi-Jun Zhang, and M. Nakhla. A neural network modeling approach to circuit optimization and statistical design. *IEEE Transactions on Microwave Theory and Techniques*, 43(6):1349–1358, 1995.

[45] John Peurifoy, Yichen Shen, Li Jing, Yi Yang, Fidel Cano-Renteria, Brendan G. DeLacy, John D. Joannopoulos, Max Tegmark, and Marin Soljačić. Nanophotonic particle simulation and inverse design using artificial neural networks. *Science Advances*, 4(6):eaar4206, 2018.

[46] Takashi Asano and Susumu Noda. Optimization of photonic crystal nanocavities based on deep learning. *Optics Express*, 26(25):32704–32717, 2018.

[47] Rohit Unni, Kan Yao, Xizewen Han, Mingyuan Zhou, and Yuebing Zheng. A mixture-density-based tandem optimization network for on-demand inverse design of thin-film high reflectors. *Nanophotonics*, 10(16):4057–4065, 2021.

[48] Simiao Ren, Willie Padilla, and Jordan Malof. Benchmarking deep inverse models over time, and the neural-adjoint method. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 38–48. Curran Associates, Inc., 2020.

[49] Jiaqi Jiang and Jonathan A. Fan. Global optimization of dielectric metasurfaces using a physics-driven neural network. *Nano Letters*, 19(8):5366–5372, 2019.

[50] Jakob Kruse, Lynton Ardizzone, Carsten Rother, and Ullrich Köthe. Benchmarking invertible architectures on inverse problems. *arXiv preprint arXiv:2101.10763*, 2021.

[51] Yingshi Chen, Jinfeng Zhu, Yinong Xie, Naixing Feng, and Qing Huo Liu. Smart inverse design of graphene-based photonic metamaterials by an adaptive artificial neural network. *Nanoscale*, 11(19):9749–9755, 2019.

[52] Simiao Ren, Ashwin Mahendra, Omar Khatib, Yang Deng, Willie J. Padilla, and Jordan M. Malof. Inverse deep learning methods and benchmarks for artificial electromagnetic material design. *Nanoscale*, 14:3958–3969, 2022.

[53] John Chilwell and Ian Hodgkinson. Thin-films field-transfer matrix theory of planar multilayer waveguides and reflection from prism-loaded waveguides. *Journal of the Optical Society of America, A*, 1(7):742–753, 1984.

[54] Karl Grantham, Muhetaer Mukaidaisi, Hsu Kiang Ooi, Mohammad Sajjad Ghaemi, Alain Tchagang, and Yifeng Li. Deep evolutionary learning for molecular design. *IEEE Computational Intelligence Magazine*, 17(2):14–28, 2022.

[55] Carmine Dodaro, Valeria Fionda, and Gianluigi Greco. LTL on weighted finite traces: Formal foundations and algorithms. In *Proc. of IJCAI'22*, pages 2606–2612, 2022.

[56] Gianluigi Greco, Antonella Guzzo, Francesco Lupia, and Luigi Pontieri. Process discovery under precedence constraints. *ACM Transactions on Knowledge Discovery from Data*, 9(4), 2015.

[57] Amin Heyrani Nobari, Wei Chen, and Faez Ahmed. Pcdgan: A continuous conditional diverse generative adversarial network for inverse design. In *Proc. of KDD'21*, page 606–616, 2021.

[58] Chaity Banerjee, Chad Lilian, Daniel Reasor, Eduardo Pasiliao, and Tathagata Mukherjee. An application of generative adversarial networks for robust inference in computational fluid dynamics. In *Proc. of ICISDM'21*, page 74–83, 2021.

[59] Lynton Ardizzone, Jakob Kruse, Sebastian Wirkert, Daniel Rahner, Eric W Pellegrini, Ralf S Klessen, Lena Maier-Hein, Carsten Rother, and Ullrich Köthe. Analyzing inverse problems with invertible neural networks. *arXiv preprint arXiv:1808.04730*, 2018.

[60] Allied Market Research. AI in Healthcare Market. https://www.alliedmarketresearch.com/AI-in-healthcare-market, 2021.

[61] T. Davenport and R. Kalakota. The potential for artificial intelligence in healthcare. *Future Healthcare Journal*, 6(2):94–98, June 2019.

[62] F. Piccialli, V. Di Somma, F. Giampaolo, S. Cuomo, and G. Fortino. A survey on deep learning in medicine: Why, how and when? *Information Fusion*, 66:111–137, 2021.

[63] Fahad Shamshad Fu and et al. Transformers in medical imaging: A survey. https://arxiv.org/abs/2201.09873, 2022.

[64] Ehud Reiter. A structured review of the validity of bleu. *Computational Linguistics*, 2018.

[65] Navdeep Kaur Mittal and Ajay. Deep learning in generating radiology reports: a survey. 2022.

[66] et al. Sun. A survey on deep learning and explainability for automatic report generation from medical images. 2022.

[67] Eric Topol. *Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again*. Basic Books, Inc., 2019.

[68] R. Fakoor, F. Ladhak, A. Nazi, and M. Huber. Using deep learning to enhance cancer diagnosis and classification. In *The 30th International Conference on Machine Learning*, 2013.

[69] M. Sordo. Introduction to neural networks in healthcare. https://www.openclinical.org/nnintro.html, 2002.

[70] A. Vial, D. Stirling, M. Field, and et al. The role of deep learning and radiomic feature extraction in cancer-specific predictive modelling: a review. *Translational Cancer Research*, 7:803–816, 2018.

[71] Jianbo Yuan, Haofu Liao, Rui Luo, and Jiebo Luo. Automatic radiology report generation based on multi-view image fusion and medical concept enrichment. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, pages 721–729, Cham, 2019. Springer Intl. Publishing.

[72] Sonit Singh, Sarvnaz Karimi, Kevin Ho-Shon, and Len Hamey. From chest x-rays to radiology reports: A multimodal machine learning approach. In *2019 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8. IEEE, 2019.

[73] Philipp Harzig, Yan-Ying Chen, Francine Chen, and Rainer Lienhart. Addressing data bias problems for chest x-ray image report generation. https://arxiv.org/abs/2201.09873, 2019.

[74] Xiancheng Xie, Yun Xiong, Philip S. Yu, Kangan Li, Suhua Zhang, and Yangyong Zhu. Attention-based abnormal-aware fusion network for radiology report generation. In *Database Systems for Advanced Applications*, pages 448–452, Cham, 2019. Springer Intl. Publishing.

[75] Gaurav O. Gajbhiye, Abhijeet V. Nandedkar, and Ibrahima Faye. Automatic report generation for chest x-ray images: A multilevel multi-attention approach. In *Computer Vision and Image Processing*, pages 174–182, Singapore, 2020.

[76] et al. Li. Ffa-ir-towards an explainable and reliable medical report generation benchmark. 2021.

[77] C. Yin, B. Qian, J. Wei, X. Li, X. Zhang, Y. Li, and Q. Zheng. Automatic generation of medical imaging diagnostic report with hierarchical recurrent neural network. In *2019 IEEE Intl. Conf. on Data Mining (ICDM)*, pages 728–737, 2019.

[78] Yuan Xue and et al. Multimodal recurrent model with attention for automated radiology report generation. In *Intl. Conf. on Medical Image Computing and Computer-Assisted Intervention*, pages 457–466. Springer, 2018.

[79] Yuan Xue and Xiaolei Huang. Improved disease classification in chest x-rays with transferred features from report generation. In *Information Processing in Medical Imaging*, pages 125–138, Cham, 2019. Springer Intl. Publishing.

[80] Mingjie Li Chang, Bingqian Lin, Zicong Chen, Haokun Lin, Xiaodan Liang, and Xiaojun. Dynamic graph enhanced contrastive learning for chest x-ray report. 2023.

[81] Yixiao Zhang, Xiaosong Wang, Ziyue Xu, Qihang Yu, Alan Yuille, and Daguang Xu. When radiology report generation meets knowledge graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12910–12917, 2020.

[82] Xin Li, Rui Cao, and Dongxiao Zhu. Vispi: Automatic visual perception and interpretation of chest x-rays. https://arxiv.org/abs/2201.09873, 2019.

[83] Baoyu Jing, Pengtao Xie, and Eric Xing. On the automatic generation of medical imaging reports. In *Proceedings of the 56th Annual Meeting of the ACL*, pages 2577–2586, Melbourne, Australia, 2018.

[84] et al. Chen. Visualgpt: Data-efficient adaptation of pretrained language models for image captioning. 2022.

[85] Doshi-Velez and Kim. Towards a rigorous science of interpretable machine learning. 2017.

[86] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *arXiv*, 2013.

[87] Indiana University. Radiology reports for the chest x-ray images from the indiana university hospital network. https://www.openclinical.org/nnintro.html, 2019.

[88] et al. Guzman. Model differently. text generation with gpt-2. https://arxiv.org/abs/2201.09873, 2021.

[89] John Muschelli. Roc and auc with a binary predictor: a potentially misleading metric. *Journal of Classification*, 2020.

[90] Xin Huang, Fengqi Yan, Wei Xu, and Maozhen Li. Multi-attention and incorporating background information model for chest x-ray image report generation. *IEEE Access*, 7:154808–154817, 2019.

[91] Tomas Mikolov Dean, Kai Chen, Greg Corrado, and Jeffrey. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[92] P. Qi, D. Chiaro, A. Guzzo, M. Ianni, G. Fortino, and F. Piccialli. Model aggregation techniques in federated learning: A comprehensive survey. *Future Generation Computer Systems*, 2023.

[93] A. Guzzo, G. Fortino, G. Greco, and M. Maggiolini. Data and model aggregation for radiomics applications: Emerging trend and open challenges. *Information Fusion*, 100:101923, 2023.

[94] Liang Chen. Agent-based modeling in urban and architectural research: A brief literature review. *Frontiers of Architectural Research*, 1(2):166–177, 2012.

[95] Michael Wooldridge and Nicholas R Jennings. Intelligent agents: Theory and practice. *The knowledge engineering review*, 10(2):115–152, 1995.

[96] Nicholas R Jennings, Katia Sycara, and Michael Wooldridge. A roadmap of agent research and development. *Autonomous agents and multi-agent systems*, 1:7–38, 1998.

[97] Cristiano Castelfranchi. Guarantees for autonomy in cognitive agent architecture. In *International Workshop on Agent Theories, Architectures, and Languages*, pages 56–70. Springer, 1994.

[98] Caroline C Hayes. Agents in a nutshell-a very brief introduction. *IEEE transactions on Knowledge and Data Engineering*, 11(1):127–132, 1999.

[99] Michael R Genesereth. Software agents michael r. genesereth logic group computer science department stanford university. 1994.

[100] Danica Dillion, Niket Tandon, Yuling Gu, and Kurt Gray. Can ai language models replace human participants? *Trends in Cognitive Sciences*, 2023.

[101] Igor Grossmann, Matthew Feinberg, Dawn C Parker, Nicholas A Christakis, Philip E Tetlock, and William A Cunningham. Ai and the transformation of social science research. *Science*, 380(6650):1108–1109, 2023.

[102] Joshua M. Epstein. Agent-based computational models and generative social science. *Complexity*, 4(5):41–60, 1999.

[103] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pages 1–22, 2023.

[104] Joseph Bates. The role of emotion in believable agents. *Commun. ACM*, 37(7):122–125, jul 1994.

[105] F. Thomas and O. Johnston. *Disney Animation: The Illusion of Life*. Abbeville Press, 1981.

[106] Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, and Maarten Sap. Sotopia: Interactive evaluation for social intelligence in language agents, 2024.

[107] J. Epstein. Generative social science: Studies in agent-based computational modeling (princeton studies in complexity). 2007.

[108] F. LeRon Shults, Ross Gore, Wesley Wildman, Christopher Lynch, Justin E. Lane, and Monica Toft. A generative model of the mutual escalation of anxiety between religious groups. *Journal of Artificial Societies and Social Simulation*, 21(4):7, 2018.

[109] Baojun Gao, Wai Kin Chan, and Xuefei Deng. Generative agent-based modeling and empirical validation of the size distribution of hospitals. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(11):3089–3100, 2016.

[110] Corey Lofdahl, Eli Stickgold, Bruce Skarin, and Ian Stewart. Extending generative models of large scale networks. *Procedia Manufacturing*, 3:3868–3875, 2015. 6th International Conference on Applied Human Factors and Ergonomics (AHFE 2015) and the Affiliated Conferences, AHFE 2015.

[111] Alexander Sasha Vezhnevets, John P. Agapiou, Avia Aharon, Ron Ziv, Jayd Matyas, Edgar A. Duéñez-Guzmán, William A. Cunningham, Simon Osindero, Danny Karmon, and Joel Z. Leibo. Generative agent-based modeling with actions grounded in physical, social, or digital space using concordia, 2023.

[112] Bruce Edmonds, Christophe Le Page, Mike Bithell, Edmund Chattoe-Brown, Volker Grimm, Ruth Meyer, Cristina Montañola Sales, Paul Ormerod, Hilton Root, and Flaminio Squazzoni. Different modelling purposes. *Journal of Artificial Societies and Social Simulation*, 22(3):6, 2019.

[113] Themis Dimitra Xanthopoulou, Andreas Prinz, and F. LeRon Shults. The problem with bullying: Lessons learned from modelling marginalization with diverse stakeholders. In Marcin Czupryna and Bogumił Kamiński, editors, *Advances in Social Simulation*, pages 289–300, Cham, 2022. Springer International Publishing.

[114] Kamwoo Lee, Sinan Ulkuatam, Peter Beling, and William Scherer. Generating synthetic bitcoin transactions and predicting market price movement via inverse reinforcement learning and agent-based modeling. *Journal of Artificial Societies and Social Simulation*, 21(3):5, 2018.

[115] L. Kieu, Nicolas Malleson, and A. Heppenstall. Dealing with uncertainty in agent-based models for short-term predictions. *Royal Society Open Science*, 7, 2019.

[116] Anna Pagani. Towards sustainability through housing functions: a systems perspective for the study of swiss tenants' residential mobility. page 358, 2022.

[117] Molood Ale Ebrahim Dehkordi, Amineh Ghorbani, Giangiacomo Bravo, Mike Farjam, René van Weeren, Anders Forsman, and Tine De Moor. Long-term dynamics of institutions: Using abm as a complementary tool to support theory development in historical studies. *Journal of Artificial Societies and Social Simulation*, 24(4):7, 2021.

[118] Etienne Delay and Cyril Piou. Mutual aid: When does resource scarcity favour group cooperation? *Ecological Complexity*, 40:100790, 2019.

[119] Robert Axelrod and William D Hamilton. The evolution of cooperation. *science*, 211(4489):1390–1396, 1981.

[120] Nantana Gajaseni Pongchai Dumrongrojwatthana, Christophe Le Page and Guy Trébuil. Co-constructing an agent-based model to mediate land use conflict between herders and foresters in northern thailand. *Journal of Land Use Science*, 6(2-3):101–120, 2011.

[121] Rosaria Conte and Mario Paolucci. On agent-based modeling and computational social science. *Frontiers in psychology*, 5:83393, 2014.

[122] M Ale Ebrahim Dehkordi, JM Lechner, Amineh Ghorbani, Igor Nikolic, EJL Chappin, and PM Herder. Using machine learning for agent specifications in agent-based models and simulations: A critical review and guidelines. *Journal of Artificial Societies and Social Simulation*, 26(1):9, 2023.

[123] S. Borysov, Jeppe Rich, and F. Pereira. How to generate micro-agents? a deep generative modeling approach to population synthesis. *Transportation Research Part C: Emerging Technologies*, 2019.

[124] Navid Ghaffarzadegan, Aritra Majumdar, Ross Williams, and Niyousha Hosseinichimeh. Generative agent-based modeling: an introduction and tutorial. *System Dynamics Review*, 40(1):e1761, 2024.

[125] Hongxin Zhang, Weihua Du, Jiaming Shan, Qinhong Zhou, Yilun Du, Joshua B Tenenbaum, Tianmin Shu, and Chuang Gan. Building cooperative embodied agents modularly with large language models. *arXiv preprint arXiv:2307.02485*, 2023.

[126] Jianghao Wang, Yichun Fan, Juan Palacios, Yuchen Chai, Nicolas Guetta-Jeanrenaud, Nick Obradovich, Chenghu Zhou, and Siqi Zheng. Global evidence of expressed sentiment alterations during the covid-19 pandemic. *Nature Human Behaviour*, 6(3):349–358, Mar 2022.

[127] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[128] Ross Williams, Niyousha Hosseinichimeh, Aritra Majumdar, and Navid Ghaffarzadegan. Epidemic modeling with generative agents, 2023.

[129] Lewis R Goldberg. An alternative "description of personality": The big-five factor structure. In *Personality and Personality Disorders*, pages 34–47. Routledge, 2013.

[130] Zhao Kaiya, Michelangelo Naim, Jovana Kondic, Manuel Cortes, Jiaxin Ge, Shuying Luo, Guangyu Robert Yang, and Andrew Ahn. Lyfe agents: Generative agents for low-cost real-time social interactions, 2023.

[131] Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. Chatdev: Communicative agents for software development, 2024.

[132] Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Jirong Wen. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345, Mar 2024.

[133] Nathan Klapach. The comparative emotional capabilities of five popular large language models. *Critical Debates in Humanities, Science and Global Justice*, 2(1), 2024.

[134] Edward Y Chang. Modeling emotions and ethics with large language models. *arXiv preprint arXiv:2404.13071*, 2024.

[135] Zaijing Li, Gongwei Chen, Rui Shao, Dongmei Jiang, and Liqiang Nie. Enhancing the emotional generation capability of large language models via emotional chain-of-thought. *arXiv preprint arXiv:2401.06836*, 2024.

[136] Gabriel Istrate. Models we can trust: Toward a systematic discipline of (agent-based) model interpretation and validation. In Frank Dignum, Alessio Lomuscio, Ulle Endriss, and Ann Nowé, editors, *AAMAS '21: 20th International Conference on Autonomous Agents and Multiagent Systems, Virtual Event, United Kingdom, May 3-7, 2021*, pages 6–11. ACM, 2021.

[137] William Rand and Roland T Rust. Agent-based modeling in marketing: Guidelines for rigor. *International Journal of research in Marketing*, 28(3):181–193, 2011.

[138] Jacqueline Chandler, Miranda Cumpston, Tianjing Li, Matthew J Page, and VJHW Welch. Cochrane handbook for systematic reviews of interventions. *Hoboken: Wiley*, 2019.

[139] Imre Lakatos. History of science and its rational reconstructions. In *PSA: Proceedings of the biennial meeting of the philosophy of science association*, volume 1970, pages 91–136. Cambridge University Press, 1970.

[140] Andrew Collins, Matthew Koehler, and Christopher Lynch. Methods that support the validation of agent-based models: An overview and discussion. *Journal of Artificial Societies and Social Simulation*, 27(1):11, 2024.

[141] Franziska Klügl. A validation methodology for agent-based simulations. In *Proceedings of the 2008 ACM Symposium on Applied Computing*, SAC '08, page 39–43, New York, NY, USA, 2008. Association for Computing Machinery.

[142] Athanasia Louloudi and Franziska Klügl. Immersive face validation: A new validation technique for agent-based simulation. In *2012 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pages 1255–1260, 2012.

[143] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating llms by human preference, 2024.

[144] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36, 2024.

[145] Marcel Binz and Eric Schulz. Using cognitive psychology to understand gpt-3. *Proceedings of the National Academy of Sciences*, 120(6):e2218523120, 2023.

[146] Richard Shiffrin and Melanie Mitchell. Probing the psychology of ai models. *Proceedings of the National Academy of Sciences*, 120(10):e2300963120, 2023.

[147] Paul Windrum, Giorgio Fagiolo, and Alessio Moneta. Empirical validation of agent-based models: Alternatives and prospects. *Journal of Artificial Societies and Social Simulation*, 10(2):8, 2007.

[148] Greg Serapio-García, Mustafa Safdari, Clément Crepy, Luning Sun, Stephen Fitz, Peter Romero, Marwa Abdulhai, Aleksandra Faust, and Maja Matarić. Personality traits in large language models, 2023.

[149] Paul Ekman. What scientists who study emotion agree about. *Perspectives on Psychological Science*, 11(1):31–34, 2016. PMID: 26817724.

[150] Antoine Bechara, Hanna Damasio, and Antonio R. Damasio. Emotion, Decision Making and the Orbitofrontal Cortex. *Cerebral Cortex*, 10(3):295–307, 03 2000.

[151] Laura Stevens. Generative agents and their applications in urban social science. *Journal of Urban Social Science*, 2023.

[152] John Doe. Applications of large language models in urban planning and management. *Journal of Urban Technology*, 2023.

[153] Joshua M. Epstein. Inverse generative social science: Backward to the future. Journal of Artificial Societies and Social Simulation 26 (2) 9, 2023.

[154] Russell Stuart and Norvig Peter. Artificial intelligence: a modern approach, 1995.

[155] John Smith, Jane Doe, and Emily Johnson. Towards continuous emotion recognition with large language models. arXiv preprint arXiv:2304.12345, 2023.

[156] Rashmi Chitrakar, Hui Wen, and Richard Evans. Virtual reality simulation of urban environments for human emotion analysis. *Frontiers in Virtual Reality*, 3:844501, 2022.

[157] Jiayuan Xu, Xiaoxuan Liu, Qiaojun Li, Ran Goldblatt, Wen Qin, Feng Liu, Congying Chu, Qiang Luo, Alex Ing, Lining Guo, et al. Global urbanicity is associated with brain and behaviour in young people. *Nature human behaviour*, 6(2):279–293, 2022.

[158] Jiayuan Xu, Nana Liu, Elli Polemiti, Liliana Garcia-Mondragon, Jie Tang, Xiaoxuan Liu, Tristram Lett, Le Yu, Markus M Nöthen, Jianfeng Feng, et al. Effects of urban living environments on mental health in adults. *Nature Medicine*, 29(6):1456–1467, 2023.

[159] Mukthi Sra, Ala Malik, Abhinav Dhall, and Martin Kächele. Emotion recognition from multimodal data in virtual reality: A review. *IEEE Transactions on Affective Computing*, 2022.

[160] Matthew L Jockers and Ted Underwood. Mapping literary emotions: A crowdsourcing approach. *Journal of Cultural Analytics*, 1(1), 2021.

[161] Yichi Yin, Yiqiao Hu, Maxwell Herman, and Joachim Vandekerckhove. Continuous emotion tracking with deep generative models and active learning. *arXiv preprint arXiv:2206.01770*, 2022.

[162] Morris H DeGroot. Reaching a consensus. *J AM STAT ASSOC*, 69(345):118–121, 1974.

[163] M. Granovetter. Threshold models of collective behavior. *AM J SOCIOL*, 83(6):1420 – 1443, 1978.

[164] Vincenzo Auletta, Diodato Ferraioli, and Gianluigi Greco. On the complexity of reasoning about opinion diffusion under majority dynamics. *ARTIF INTELL*, 284:103288, 2020.

[165] Sirin Botan, Umberto Grandi, and Laurent Perrussel. Multi-issue opinion diffusion under constraints. In *Proc. of AAMAS*, 2019.

[166] Matteo Castiglioni, Diodato Ferraioli, and Nicola Gatti. Election control in social networks via edge addition or removal. In *Proc. of AAAI*, volume 34, pages 1878–1885, 2020.

[167] Dmitry Chistikov, Grzegorz Lisowski, Mike Paterson, and Paolo Turrini. Convergence of opinion diffusion is pspace-complete. In *Proc. of AAAI*, volume 34, pages 7103–7110, 2020.

[168] Piotr Faliszewski, Rica Gonen, Martin Koutecký, and Nimrod Talmon. Opinion diffusion and campaigning on society graphs. *J LOGIC COMPUT*, 32(6):1162–1194, 2022.

[169] Valeria Fionda and Gianluigi Greco. Opinion diffusion in competitive environments: Relating coverage and speed of diffusion. In *COMPLEX NETWORKS*, pages 425–435, 2019.

[170] Umberto Grandi, Lawqueen Kanesh, Grzegorz Lisowski, Maadapuzhi Sridharan Ramanujan, and Paolo Turrini. Identifying and eliminating majority illusion in social networks. In *Proc. of AAAI*, volume 37, 2023.

[171] D. Kempe, J. M. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *Proc. of SIGKDD*, pages 137–146, 2003.

[172] Y. Ni, L. Xie, and Z.-Q. Liu. Minimizing the expected complete influence time of a social network. *INFORM SCIENCES*, 180(13):2514–2527, 2010.

[173] Vincent Yun Lou, Smriti Bhagat, Laks V.S. Lakshmanan, and Sharan Vaswani. Modeling non-progressive phenomena for influence propagation. In *Proc. of COSN*, page 131–138, 2014.

[174] Pedro Domingos and Matt Richardson. Mining the network value of customers. In *Proc. of KDD*, pages 57–66, 2001.

[175] Paulo Shakarian, Sean Eyre, and Damon Paulo. A scalable heuristic for viral marketing under the tipping model. *Soc. Netw. Anal. Min.*, 3(4):1225–1248, 2013.

[176] Ning Chen. On the approximability of influence in social networks. *SIAM J DISCRETE MATH*, 23(3):1400–1415, 2009.

[177] Gennaro Cordasco, Luisa Gargano, Marco Mecchia, Adele A Rescigno, and Ugo Vaccaro. Discovering small target sets in social networks: a fast and effective algorithm. *Algorithmica*, 80:1804–1833, 2018.

[178] David Kempe, Jon M Kleinberg, and Éva Tardos. Influential nodes in a diffusion model for social networks. In *Proc. of ICALP*, volume 5, pages 1127–1138, 2005.

[179] Yuchen Li, Ju Fan, Yanhao Wang, and Kian-Lee Tan. Influence maximization on social graphs: A survey. *IEEE T KNOWL DATA EN*, 30(10):1852–1872, 2018.

[180] Vincenzo Auletta, Diodato Ferraioli, and Gianluigi Greco. Reasoning about consensus when opinions diffuse through majority dynamics. In *Proc. of IJCAI*, pages 49–55, 2018.

[181] Robert Bredereck and Edith Elkind. Manipulating opinion diffusion in social networks. In *Proc. of IJCAI*, pages 894–900, 2017.

[182] Ahad N. Zehmakan. Majority opinion diffusion in social networks: An adversarial approach. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(6):5611–5619, May 2021.

[183] Zhiqiang Zhuang, Kewen Wang, Junhu Wang, Heng Zhang, Zhe Wang, and Zhiguo Gong. Lifting majority to unanimity in opinion diffusion. In *Proc. of ECAI*, 2020.

[184] Vincenzo Auletta, Diodato Ferraioli, and Gianluigi Greco. On the effectiveness of social proof recommendations in markets with multiple products. In *Proc. of ECAI*, pages 19–26. 2020.

[185] Vincenzo Auletta, Diodato Ferraioli, and Gianluigi Greco. Optimal majority dynamics for the diffusion of an opinion when multiple alternatives are available. *Theor. Comput. Sci.*, 869:156–180, 2021.

[186] John P. Scott and Peter J. Carrington. *The SAGE Handbook of Social Network Analysis*. Sage Publications Ltd., 2011.

[187] Christos H. Papadimitriou. *Computational Complexity*. Addison Wesley, Reading, MA, USA, 1994.

[188] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

[189] A. Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distributions. *BULL CALCUTTA MATH S*, 35:99–109, 1943.

[190] B. Rozemberczki, R. Davies, R. Sarkar, and C. A. Sutton. GEMSEC: graph embedding with self clustering. *CoRR*, 2018.

[191] Jaewon Yang and Jure Leskovec. Defining and evaluating network communities based on ground-truth. In *Proc. of ICDM*, pages 745–754, 2012.

[192] Vincenzo Auletta, Diodato Ferraioli, Valeria Fionda, and Gianluigi Greco. Maximizing the spread of an opinion when tertium datur est. In *Proc. of AAMAS*, volume 19, pages 1207–1215, 2019.

[193] Robert Bredereck, Lilian Jacobs, and Leon Kellerhals. Maximizing the spread of an opinion in few steps: Opinion diffusion in non-binary networks. In *Proc. of IJCAI*, pages 1622–1628, 2020.

[194] Flavio Chierichetti, Jon Kleinberg, and Sigal Oren. On discrete preferences and coordination. *J COMPUT SYST SCI*, 93:11–29, 2018.

[195] A. Smith et al. Mediterranean diet and health outcomes. *Journal of Nutrition*, 150(3):256–263, 2020.

[196] B. Brown et al. Physical activity and inflammation. *Medicine and Science in Sports and Exercise*, 51(4):784–790, 2019.

[197] C. Johnson et al. Adolescent health and diet quality. *Pediatrics*, 142(6):e20182345, 2018.

[198] M. A. Martinez-Gonzalez et al. The mediterranean diet and cardiovascular health. *Nutrition Reviews*, 75(5):361–379, 2017.

[199] A. Trichopoulou et al. Definitions and potential health benefits of the mediterranean diet: Views from experts. *Nutrition Today*, 49(2):76–87, 2014.

[200] R. Estruch et al. Mediterranean diet and cardiovascular prevention: Risk reduction in the predimed trial. *Journal of the American College of Cardiology*, 72(9):336–348, 2018.

[201] M. Fito et al. Anti-inflammatory effects of the mediterranean diet. *Molecular Nutrition & Food Research*, 57(4):732–740, 2013.

[202] F. Sofi et al. Adherence to mediterranean diet and health status: Meta-analysis. *BMJ*, 337(a1344):101–107, 2010.

[203] G. Grosso et al. Mediterranean diet and adolescent health: A systematic review. *British Journal of Nutrition*, 117(6):909–920, 2017.

[204] C. Morelli et al. The impact of a nutrition education intervention on the dietary habits and physical activity levels of adolescents in calabria, italy. *Nutrition and Health*, 25(2):91–100, 2019.

[205] L. Serra-Majem et al. The kidmed index: Development and validation of a mediterranean diet quality index in children and adolescents. *Public Health Nutrition*, 7(7):931–935, 2004.

[206] DIMENU Project. Dimenu official website, 2019. https://www.dimenu.it/.

[207] DIMENU Project. Dimenu facebook page, 2019. https://www.facebook.com/dimenu2019.

[208] World Health Organization. *Global recommendations on physical activity for health*. WHO Press, 2010.

[209] A. Bach-Faig et al. Mediterranean diet pyramid today. *Public Health Nutrition*, 14(12A):2274–2284, 2011.

[210] R. Estruch et al. Primary prevention of cardiovascular disease with a mediterranean diet. *New England Journal of Medicine*, 368(14):1279–1290, 2013.

[211] Italian Society of Human Nutrition. Energy requirements for children and adolescents aged 1-17 years, 2019. https://sinu.it/2019/07/09/fabbisogno-energetico-medio-ar-nellintervallo-deta-1-17-anni/.

[212] Survey System. Sample size calculator, 2020. https://www.surveysystem.com/sscalc.htm.

[213] Campbell Collaboration. Effect size calculator, 2020. https://campbellcollaboration.org/escalc/html/EffectSizeCalculator-SMD1.php.

[214] Justin Malimban, Danny Lathouwers, Haibin Qian, Frank Verhaegen, Julia Wiedemann, Sytze Brandenburg, and Marius Staring. Deep learning-based segmentation of the thorax in mouse micro-ct scans. *Scientific Reports*, 12(1):1822, 2022.

[215] Tobias Roß, Pierangela Bruno, Annika Reinke, Manuel Wiesenfarth, Lisa Koeppel, Peter M Full, Bünyamin Pekdemir, Patrick Godau, Darya Trofimova, Fabian Isensee, et al. Beyond rankings: Learning (more) from algorithm validation. *Medical image analysis*, 86:102765, 2023.

[216] Pierangela Bruno, Maria Francesca Spadea, Salvatore Scaramuzzino, Salvatore De Rosa, Ciro Indolfi, Giuseppe Gargiulo, Giuseppe Giugliano, Giovanni Esposito, Francesco Calimeri, and Paolo Zaffino. Assessing vascular complexity of paod patients by deep learning-based segmentation and fractal dimension. *Neural Computing and Applications*, 34(24):22015–22022, 2022.

[217] Yabo Fu, Yang Lei, Tonghe Wang, Walter J Curran, Tian Liu, and Xiaofeng Yang. A review of deep learning based methods for medical image multi-organ segmentation. *Physica Medica*, 85:107–122, 2021.

[218] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88, 2017.

[219] Carlo Adornetto, Antonella Guzzo, and Andrea Vasile. Automatic medical report generation via latent space conditioning and transformers. In *2023 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, pages 0428–0435, 2023.

[220] Luana Batista da Cruz, José Denes Lima Araújo, Jonnison Lima Ferreira, João Otávio Bandeira Diniz, Aristófanes Corrêa Silva, João Dallyson Sousa de Almeida, Anselmo Cardoso de Paiva, and Marcelo Gattass. Kidney segmentation from computed tomography images using deep neural network. *Computers in Biology and Medicine*, 123:103906, 2020.

[221] Jan Matula, Veronika Polakova, Jakub Salplachta, Marketa Tesarova, Tomas Zikmund, Marketa Kaucka, Igor Adameyko, and Jozef Kaiser. Resolving complex cartilage structures in developmental biology via deep learning-based automatic segmentation of x-ray computed microtomography images. *Scientific Reports*, 12(1):8728, 2022.

[222] Soodeh Nikan, Kylen Van Osch, Mandolin Bartling, Daniel G Allen, S Alireza Rohani, Ben Connors, Sumit K Agrawal, and Hanif M Ladak. Pwd-3dnet: a deep learning-based fully-automated segmentation of multiple structures on temporal bone ct scans. *IEEE Trans. on Image Processing*, 30:739–753, 2020.

[223] Francesco Sforazzini, Patrick Salome, Mahmoud Moustafa, Cheng Zhou, Christian Schwager, Katrin Rein, Nina Bougatf, Andreas Kudak, Henry Woodruff, Ludwig Dubois, et al. Deep learning–based automatic lung segmentation on multiresolution ct scans from healthy and fibrotic lungs in mice. *Radiology: Artificial Intelligence*, 4(2):e210095, 2022.

[224] DP Clark and CT Badea. Advances in micro-ct imaging of small animals. *Physica Medica*, 88:175–192, 2021.

[225] Tao Liu, Yun Tian, Shifeng Zhao, Xiaoying Huang, and Qingjun Wang. Residual convolutional neural network for cardiac image segmentation and heart disease diagnosis. *IEEE Acc.*, 8:82153–82161, 2020.

[226] Qiao Zheng, Hervé Delingette, Nicolas Duchateau, and Nicholas Ayache. 3-d consistent and robust segmentation of cardiac images by deep learning with spatial propagation. *IEEE transactions on medical imaging*, 37(9):2137–2148, 2018.

[227] Sam Sharobeem, Hervé Le Breton, Florent Lalys, Mathieu Lederlin, Clément Lagorce, Marc Bedossa, Dominique Boulmier, Guillaume Leurent, Pascal Haigron, and Vincent Auffret. Validation of a whole heart segmentation from computed tomography imaging using a deep-learning approach. *Journal of Cardiovascular Translational Research*, 15(2):427–437, 2022.

[228] Zhanwei Xu, Ziyi Wu, and Jianjiang Feng. Cfun: Combining faster r-cnn and u-net network for efficient whole heart segmentation. *arXiv preprint arXiv:1812.04914*, 2018.

[229] Mariacristina Filice, Maria Carmela Cerra, and Sandra Imbrogno. The goldfish carassius auratus: an emerging animal model for comparative cardiac research. *Journal of Comparative Physiology B*, 192(1):27–48, 2022.

[230] M. Filice, A. Gattuso, S. Imbrogno, B. Tota, and M.C. Cerra. Functional, structural, and molecular remodelling of the goldfish (carassius auratus) heart under moderate hypoxia. *Fish Physiology and Biochemistry*, 2024.

[231] Mariacristina Filice, Rosa Mazza, Serena Leo, Alfonsina Gattuso, Maria Carmela Cerra, and Sandra Imbrogno. The hypoxia tolerance of the goldfish (carassius auratus) heart: The nos/no system and beyond. *Antioxidants*, 9(6), 2020.

[232] S. Imbrogno, C. Capria, B. Tota, and F.B. Jensen. Nitric oxide improves the hemodynamic performance of the hypoxic goldfish (carassius auratus) heart. *Nitric Oxide*, 42:24–31, 2014.

[233] Maedeh Bazmi and Ariel L Escobar. Excitation–contraction coupling in the goldfish (carassius auratus) intact heart. *Frontiers in Physiology*, 11:1103, 2020.

[234] Sandra Imbrogno, Mariacristina Filice, and Maria Carmela Cerra. Exploring cardiac plasticity in teleost: The role of humoral modulation. *General and Comparative Endocrinology*, 283:113236, 2019.

[235] Tom Eelbode, Jeroen Bertels, Maxim Berman, Dirk Vandermeulen, Frederik Maes, Raf Bisschops, and Matthew B. Blaschko. Optimization for medical image segmentation: Theory and practice when evaluating with dice score or jaccard index. *IEEE Transactions on Medical Imaging*, 39(11):3679–3690, 2020.

[236] Maier-Hein et al. Metrics reloaded: recommendations for image analysis validation. *Nature Methods*, 21:195–212, 2024.

[237] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[238] Salih Can Yurtkulu, Yusuf Hüseyin Şahin, and Gozde Unal. Semantic segmentation with extended deeplabv3 architecture. In *2019 27th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4. IEEE, 2019.

[239] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[240] Philipp D. Lösel, Thomas Van de Kamp, Alejandra Jayme, Alexey Ershov, Tomáš Faragó, Olaf Pichler, Nicholas Tan Jerome, Narendar Aadepu, Sabine Bremer, Suren A. Chilingaryan, Michael Heethoff, Andreas Kopmann, Janes Odar, Sebastian Schmelzle, Marcus Zuber, Joachim Wittbrodt, Tilo Baumbach, and Vincent Heuveline. Introducing biomedisa as an open-source online platform for biomedical image segmentation. *Nature Communications*, 11(1):Article no: 5577, 2020. 56.03.10; LK 01.

[241] Fabian Isensee, Paul F Jaeger, Simon AA Kohl, Jens Petersen, and Klaus H Maier-Hein. nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, 18(2):203–211, 2021.

[242] Y. Fuertes-Callén, B. Cuellar-Fernández, and C. Serrano-Cinca. Predicting startup survival using first years financial statements. *Journal of Small Business Management*, 60(6):1314–1350, 2022.

[243] B. G. Carruthers. The history of bankruptcy: Economic, social, and cultural implications in early modern europe. *The Journal of Economic History*, 75(2), 2015.

[244] H. Ooghe and P. De Sofie. Failure processes and causes of company bankruptcy: A typology. *Management Decision*, 46(2):223–242, 2008.

[245] N. Levratto. From failure to corporate bankruptcy: a review. *Journal of Innovation and Entrepreneurship*, 2(1):20, 2013.

[246] T. Kliestik, M. Misankova, K. Valaskova, and L. Svabova. Bankruptcy prevention: New effort to reflect on legal and social changes. *Science and Engineering Ethics*, 24(2):791–803, 2018.

[247] A. Ellul and M. Pagano. Corporate leverage and employees' rights in bankruptcy. *Journal of Financial Economics*, 133(3):685–707, 2019.

[248] J. R. Graham, H. Kim, S. Li, and J. Qiu. Employee costs of corporate bankruptcy. *Journal of Finance*, 78(4):2087–2137, 2023.

[249] S. A. Yang, J. R. Birge, and R. P. Parker. The supply chain effects of bankruptcy. *Management Science*, 61(10):2320–2338, 2015.

[250] J. Bower and S. Gilson. The social cost of fraud and bankruptcy. *Harvard Business Review*, 81(12), 2003.

[251] F. Tung. Is international bankruptcy possible? *SSRN Electronic Journal*, 2005.

[252] E. I. Altman. The prediction of corporate bankruptcy: A discriminant analysis. *The Journal of Finance*, 23(1):193, 1968.

[253] E. B. Deakin. A discriminant analysis of predictors of business failure. *Journal of Accounting Research*, 10(1):167, 1972.

[254] J. A. Ohlson. Financial ratios and the probabilistic prediction of bankruptcy. *Journal of Accounting Research*, 18(1):109, 1980.

[255] M. E. Zmijweski. Methodological issues related to the estimation of financial distress prediction models. *Journal of Accounting Research*, 22:59–82, 1984.

[256] F. Barboza, H. Kimura, and E. Altman. Machine learning models and bankruptcy prediction. *Expert Systems with Applications*, 83:405–417, 2017.

[257] W. Y. Lin, Y. H. Hu, and C. F. Tsai. Machine learning in financial crises prediction: A survey. *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews*, 42(4):421–436, 2012.

[258] B. Zhou, J. Jin, H. Zhou, X. Zhou, L. Shi, J. Ma, and Z. Zheng. Forecasting credit default risk with graph attention networks. *Electronic Commerce Research and Applications*, 62:101332, 2023.

[259] A. F. Atiya. Bankruptcy prediction for credit risk using neural networks: A survey and new results. *IEEE Transactions on Neural Networks*, 12(4):929–935, 2001.

[260] C. Charalambous, S. H. Martzoukos, and Z. Taoushianis. A neuro-structural framework for bankruptcy prediction. *Quantitative Finance*, 23(10):1445–1464, 2023.

[261] P. du Jardin. Designing topological data to forecast bankruptcy using convolutional neural networks. *Annals of Operations Research*, 325(2):1291–1332, 2023.

[262] F. Dube, N. Nzimande, and P. F. Muzindutsi. Application of artificial neural networks in predicting financial distress in the jse financial services and manufacturing companies. *Journal of Sustainable Finance and Investment*, 13(1):723–743, 2023.

[263] S. Ben Jabeur, A. Sadaaoui, A. Sghaier, and R. Aloui. Machine learning models and cost-sensitive decision trees for bond rating prediction. *Journal of the Operational Research Society*, 71(8):1161–1179, 2020.

[264] H. Kim, H. Cho, and D. Ryu. Corporate bankruptcy prediction using machine learning methodologies with a focus on sequential data. *Computational Economics*, 59(3):1231–1249, 2022.

[265] F. Mai, S. Tian, C. Lee, and L. Ma. Deep learning models for bankruptcy prediction using textual disclosures. *European Journal of Operational Research*, 274(2):743–758, 2019.

[266] C. F. Tsai and J. W. Wu. Using neural network ensembles for bankruptcy prediction and credit scoring. *Expert Systems with Applications*, 34(4):2639–2649, 2008.

[267] L. Nanni and A. Lumini. An experimental comparison of ensemble of classifiers for bankruptcy prediction and credit scoring. *Expert Systems with Applications*, 36(2 PART 2):3028–3033, 2009.

[268] K. S. Shin, T. S. Lee, and H. J. Kim. An application of support vector machines in bankruptcy prediction model. *Expert Systems with Applications*, 28(1):127–135, 2005.

[269] T. B. Bell. Neural nets or the logit model? a comparison of each model's ability to predict commercial bank failures. *International Journal of Intelligent Systems in Accounting, Finance & Management*, 6(3):249–264, 1997.

[270] A. Charitou, E. Neophytou, and C. Charalambous. Predicting corporate failure: empirical evidence for the uk. *European Accounting Review*, 13(3):465–497, 2004.

[271] J. Heo and J. Y. Yang. Bankruptcy forecasting model using adaboost: a focus on construction companies. *Journal of Intelligence and Information Systems*, 4866(1), 2014.

[272] T. Hosaka. Bankruptcy prediction using imaged financial ratios and convolutional neural networks. *Expert Systems with Applications*, 117:287–299, 2019.

[273] Z. Huang, H. Chen, C. J. Hsu, W. H. Chen, and S. Wu. Credit rating analysis with support vector machines and neural networks: A market comparative study. *Decision Support Systems*, 37(4):543–558, 2004.

[274] R. H. G. Jackson and A. Wood. The performance of insolvency prediction and credit risk models in the uk: A comparative study. *British Accounting Review*, 45(3):183–202, 2013.

[275] Y. Jang, I.-B. Jeong, Y. K. Cho, and Y. Ahn. Predicting business failure of construction contractors using long short-term memory recurrent neural network. *Journal of Construction Engineering and Management*, 145(11):1–9, 2019.

[276] A. Narvekar and D. Guha. Bankruptcy prediction using machine learning and an application to the case of the covid-19 recession. *Data Science in Finance and Economics*, 1(2):180–195, 2021.

[277] G. Perboli and E. Arabnezhad. A machine learning-based dss for mid and long-term company crises prediction. *Expert Systems with Applications*, 174:114758, 2021.

[278] Z. Zhao, S. Xu, B. H. Kang, M. M. J. Kabir, Y. Liu, and R. Wasinger. Investigation and improvement of multi-layer perception neural networks for credit scoring. *Expert Systems with Applications*, 42(7):3508–3516, 2015.

[279] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[280] F. Fasano and T. La Rocca. Does the bank–firm human relationship still matter for smes? the game-changing role of digitalization. *Small Business Economics*, 2023.

[281] F. Fasano and M. La Rocca. Local versus national banking development in europe: who is the winner? *Eurasian Business Review*, pages 1–30, 2023.

[282] M. La Rocca, F. Fasano, and J. F. Sanchez-Vidal. The role of government policies for italian firms during the covid-19 crises. 2022.

[283] Y. Shi and X. Li. An overview of bankruptcy prediction models for corporate firms: A systematic literature review. *Intangible Capital*, 15(2):114–127, 2019.

[284] J. Begley, J. Ming, and S. Watts. Bankruptcy classification errors in the 1980s: an empirical analysis of altman's and ohlson's models. *Review of Accounting Studies*, 1(4):267–284, 1996.

[285] H. Hu and M. Sathye. Predicting financial distress in the hong kong growth enterprises market from the perspective of financial sustainability. *Sustainability (Switzerland)*, 7(2):1186–1200, 2015.

[286] D. Liang, C. C. Lu, C. F. Tsai, and G. A. Shih. Financial ratios and corporate governance indicators in bankruptcy prediction: A comprehensive study. *European Journal of Operational Research*, 252(2):561–572, 2016.

[287] J. Bellovary, D. Giacomino, and M. Akers. A review of bankruptcy prediction studies: 1930 to present. *Accounting Faculty Research*, 33(Winter):1–42, 2007.

[288] K. Y. Tam and M. Y. Kiang. Managerial applications of neural networks: The case of bank failure predictions. *Management Science*, 38(7):926–947, 1992.

[289] R. Wilson and R. Sharda. Bankruptcy prediction using neural networks. *Decision Support Systems*, page 545–557, 1994.

[290] M. Yang, M. K. Lim, Y. Qu, X. Li, and D. Ni. Deep neural networks with l1 and l2 regularization for high dimensional corporate credit risk prediction. *Expert Systems with Applications*, 213:118873, 2023.

[291] T. K. Chen, H. H. Liao, G. D. Chen, W. H. Kang, and Y. C. Lin. Bankruptcy prediction using machine learning models with the text-based communicative value of annual reports. *Expert Systems with Applications*, 233:120714, 2023.

[292] M. Elhoseny, N. Metawa, G. Sztano, and I. M. El-Hasnony. Deep learning-based model for financial distress prediction. *Annals of Operations Research*, pages 1–23, 2022.

[293] D. Veganzones and E. Severin. Corporate failure prediction models in the twenty-first century: a review. *European Business Review*, 33(2):204–226, 2021.

[294] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–57, 2002.

[295] JF Bird, JJ Bommer, H Crowley, and R Pinho. Modelling liquefaction-induced building damage in earthquake loss estimation. *Soil Dynamics and Earthquake Engineering*, 26(1):15–30, 2006.

[296] M Cubrinovski, A Winkley, J Haskell, A Palermo, L Wotherspoon, K Robinson, B Bradley, P Brabhaharan, and M Hughes. Spreading-induced damage to short-span bridges in christchurch, new zealand. *Earthquake Spectra*, 30(1):57–83, 2014.

[297] RA Green, M Cubrinovski, B Cox, C Wood, L Wotherspoon, B Bradley, and B Maurer. Select liquefaction case histories from the 2010–2011 canterbury earthquake sequence. *Earthquake Spectra*, 30(1):131–153, 2014.

[298] S van Ballegooy, RA Green, J Lees, F Wentz, and BW Maurer. Assessment of various cpt based liquefaction severity index frameworks relative to the ishihara (1985) h1–h2 boundary curves. *Soil Dynamics and Earthquake Engineering*, 79:347–364, 2015.

[299] BW Maurer, RA Green, S van Ballegooy, and L Wotherspoon. Development of region-specific soil behavior type index correlations for evaluating liquefaction hazard in christchurch, new zealand. *Soil Dynamics and Earthquake Engineering*, 117:96–105, 2019.

[300] EM Rathje and MG Durante. On the use of machine learning techniques to predict lateral spreading displacement in new zealand. *17th World Conference on Earthquake Engineering 17WCEE, Sendai, Japan – September 13th to 18th, 2020*, 2020.

[301] Q Guan, S Ren, L Chen, Y Yao, Y Hu, R Wang, B Feng, L Gu, and W Chen. Recognizing multivariate geochemical anomalies related to mineralization by using deep unsupervised graph learning. *Natural Resources Research*, 31(5):2225–2245, 2022.

[302] Y Jiang, H Luo, Q Xu, Z Lu, L Liao, H Li, and L Hao. A graph convolutional incorporating gru network for landslide displacement forecasting based on spatiotemporal analysis of gnss observations. *Remote Sensing*, 14(4):1016, 2022.

[303] P Kuang, R Li, Y Huang, J Wu, X Luo, and F Zhou. Landslide displacement prediction via attentive graph neural network. *Remote Sensing*, 14(8):1919, 2022.

[304] L Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[305] Y LeCun, Y Bengio, and GE Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[306] WR Tobler. A computer movie simulating urban growth in the detroit region. *Economic Geography*, 46:234–240, 1970.

[307] L Anselin. Local indicators of spatial association—lisa. *Geographical Analysis*, 27:93–115, 1995.

[308] TN Kipf and M Welling. Semi-supervised classification with graph convolutional networks. *5th International Conference on Learning Representations ICLR 2017 Toulon France April 24-26*, 2017.

[309] ML McHugh. Interrater reliability: The kappa statistic. *Biochemia Medica*, 22(3):276–282, 2012.

[310] E Rathje, C Dawson, JE Padgett, JP Pinelli, D Stanzione, A Adair, P Arduino, SJ Brandenberg, T Cockerill, C Dey, et al. Designsafe: A new cyberinfrastructure for natural hazards engineering. *ASCE Natural Hazards Review*, 2017.

[311] Lefteris Koumakis. Deep learning models in genomics; are we there yet? *Computational and Structural Biotechnology Journal*, 18:1466–1473, 2020.

[312] Esra'a Alhenawi, Rizik Al-Sayyed, Amjad Hudaib, and Seyedali Mirjalili. Feature selection methods on gene expression microarray data for cancer classification: A systematic review. *Computers in Biology and Medicine*, 140:105051, 2022.

[313] Pierangela Bruno, Francesco Calimeri, Alexandre Sébastien Kitanidis, and Elena De Momi. Data reduction and data visualization for automatic diagnosis using gene expression and clinical data. *Artificial Intelligence in Medicine*, 107:101884, 2020.

[314] Garrett Graham, Nicholas Csicsery, Elizabeth Stasiowski, Gregoire Thouvenin, William H Mather, Michael Ferry, Scott Cookson, and Jeff Hasty. Genome-scale transcriptional dynamics and environmental biosensing. *Proceedings of the National Academy of Sciences*, 117(6), 2020.

[315] Jaishree Meena and Yasha Hasija. Application of explainable artificial intelligence in the identification of squamous cell carcinoma biomarkers. *Computers in Biology and Medicine*, 146:105505, 2022.

[316] Md Rezaul Karim, Michael Cochez, Oya Beyan, Stefan Decker, and Christoph Lange. Onconetexplainer: explainable predictions of cancer types based on gene expression data. In *2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE)*, pages 415–422. IEEE, 2019.

[317] Padideh Danaee, Reza Ghaeini, and David A Hendrix. A deep learning approach for cancer detection and relevant gene identification. In *Pacific symposium on biocomputing 2017*, pages 219–229. World Scientific, 2017.

[318] Fortunato Morabito, Laura Mosca, Giovanna Cutrona, Luca Agnelli, Giacomo Tuana, Manuela Ferracin, Barbara Zagatti, Marta Lionetti, Sonia Fabris, Francesco Maura, Serena Matis, Massimo Gentile, Ernesto Vigna, Monica Colombo, Carlotta Massucco, Anna Grazia Recchia, Sabrina Bossio, Laura De Stefano, Fiorella Ilariucci, Caterina Musolino, Stefano Molica, Francesco Di Raimondo, Agostino Cortelezzi, Pierfrancesco Tassone, Massimo Negrini, Sara Monti, Davide Rossi, Gianluca Gaidano, Manlio Ferrarini, and Antonino Neri. Clinical monoclonal b lymphocytosis versus rai 0 chronic lymphocytic leukemia: A comparison of cellular, cytogenetic, molecular, and clinical features. *Clinical Cancer Research*, 19(21):5890–5900, November 2013.

[319] Sonia Fabris, Laura Mosca, Katia Todoerti, Giovanna Cutrona, Marta Lionetti, Daniela Intini, Serena Matis, Monica Colombo, Luca Agnelli, Massimo Gentile, Mauro Spriano, Vincenzo Callea, Gianluca Festini, Stefano Molica, Giorgio Lambertenghi Deliliers, Fortunato Morabito, Manlio Ferrarini, and Antonino Neri. Molecular and transcriptional characterization of 17p loss in b-cell chronic lymphocytic leukemia. *Genes, Chromosomes and Cancer*, 47(9):781–793, 2008.

[320] M. Hallek. Chronic lymphocytic leukemia: 2020 update on diagnosis, risk stratification, and treatment. *Am J Hematol*, 94(11):1266–87, 2019.

[321] P. Baliakas, M. Mattsson, K. Stamatopoulos, and R. Rosenquist. Prognostic indices in chronic lymphocytic leukaemia: where do we stand, how do we proceed? *J Intern Med*, 279(4):347–57, 2016.

[322] H. Döhner, S. Stilgenbauer, A. Benner, E. Leupolt, A. Kröber, L. Bullinger, et al. Genomic aberrations and survival in chronic lymphocytic leukemia. *New England Journal of Medicine*, 343(26):1910–6, 2000.

[323] N. Kreuzberger, J. A. A. G. Damen, M. Trivella, L. J. Estcourt, A. Aldin, L. Umlauff, et al. Prognostic models for newly-diagnosed chronic lymphocytic leukaemia in adults: a systematic review and meta-analysis. *Cochrane Database of Systematic Reviews*, 7(CD012022):1–233, 2020.

[324] International CLL-IPI Working Group et al. An international prognostic index for patients with chronic lymphocytic leukaemia (cll-ipi): a meta-analysis of individual patient data. *Lancet Oncol*, 17(6):779–90, 2016.

[325] A. Condoluci, L. di Bergamo, P. Langerbeins, M. A. Hoechstetter, C. D. Herling, L. De Paoli, et al. International prognostic score for asymptomatic, early-stage chronic lymphocytic leukemia. *Blood*, 135(21):1859–69, 2020.

[326] M. Gentile, T. D. Shanafelt, D. Rossi, L. Laurenti, F. R. Mauro, S. Molica, et al. Validation of the cll-ipi and comparison with the mdacc prognostic index in newly diagnosed patients. *Blood The Journal of the American Society of Hematology*, 128(16):2093–5, 2016.

[327] A. Rodriguez, R. Villuendas, L. Yanez, M. E. Gomez, R. Diaz, M. Pollan, et al. Molecular heterogeneity in chronic lymphocytic leukemia is dependent on bcr signaling: clinical correlation. *Leukemia*, 21(9):1984–91, 2007.

[328] G. A. Calin, M. Ferracin, A. Cimmino, G. di Leva, M. Shimizu, S. E. Wojcik, et al. A microrna signature associated with prognosis and progression in chronic lymphocytic leukemia. *New England Journal of Medicine*, 353(17):1793–801, 2005.

[329] T. Herold, V. Jurinovic, K. H. Metzeler, A. L. Boulesteix, M. Bergmann, T. Seiler, et al. An eight-gene expression signature for the prediction of survival and time to treatment in chronic lymphocytic leukemia. *Leukemia*, 25(10):1639–45, 2011.

[330] J. Taylor, W. Xiao, and O. Abdel-Wahab. Diagnosis and classification of hematologic malignancies on the basis of genetics. *Blood The Journal of the American Society of Hematology*, 130(4):410–23, 2017.

[331] Y. Zhu, X. Gan, R. Qin, Z. Lin, et al. Identification of six diagnostic biomarkers for chronic lymphocytic leukemia based on machine learning algorithms. *J Oncol*, 2022(3652107):1–19, 2022.

[332] H. Chen, Y. Zhang, and I. Gutman. A kernel-based clustering method for gene selection with gene expression data. *J Biomed Inform*, 62:12–20, 2016.

[333] P. Das, A. Roychowdhury, S. Das, S. Roychoudhury, and S. Tripathy. sigfeature: novel significant feature selection method for classification of gene expression data using support vector machine and t statistic. *Front Genet*, 11:247, 2020.

[334] H. T. Salah, I. N. Muhsen, M. E. Salama, T. Owaidah, and S. K. Hashmi. Machine learning applications in the diagnosis of leukemia: Current trends and future directions. *Int J Lab Hematol*, 41(6):717–25, 2019.

[335] C. T. Chen, P. P. Wang, W. J. Mo, Y. P. Zhang, W. Zhou, T. F. Deng, et al. Expression profile analysis of prognostic long non-coding rna in adult acute myeloid leukemia by weighted gene co-expression network analysis (wgcna). *J Cancer*, 10(19):4707, 2019.

[336] A. Fabregat, K. Sidiropoulos, G. Viteri, O. Forner, P. Marin-Garcia, V. Arnau, et al. Reactome pathway analysis: a high-performance in-memory approach. *BMC Bioinformatics*, 18(1):1–9, 2017.

[337] A. Jović, K. Brkić, and N. Bogunović. A review of feature selection methods with applications. In *2015 38th international convention on information and communication technology, electronics and microelectronics (MIPRO)*, pages 1200–5, 2015.

[338] E. Bauvois, B. Pramil, L. Jondreville, C. Quiney, F. Nguyen Khac, and S. A. Susin. Activation of interferon signaling in chronic lymphocytic leukemia cells contributes to apoptosis resistance via a jak-src/stat3/mcl-1 signaling pathway. *Biomedicines*, 9(2):188, 2021.

[339] E. Infante and A. J. Ridley. Roles of rho gtpases in leucocyte and leukaemia cell transendothelial migration. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 368(1629):20130013, 2013.

[340] M. A. Abu El-Makarem, M. F. Kamel, A. A. Mohamed, H. A. Ali, M. R. Mohamed, A. E. D. M. Mohamed, et al. Down-regulation of hepatic expression of ghr/stat5/igf-1 signaling pathway fosters development and aggressiveness of hcv-related hepatocellular carcinoma: Crosstalk with snail-1 and type 2 transforming growth factor-beta receptor. *PLoS One*, 17(11):e0277266, 2022.

[341] H. Z. Yan, H. F. Wang, Y. Yin, J. Zou, F. Xiao, L. N. Yi, et al. Ghr is involved in gastric cell growth and apoptosis via pi3k/akt signalling. *J Cell Mol Med*, 25(5):2450–8, 2021.

[342] L. Polcik, S. Dannewitz Prosseda, F. Pozzo, A. Zucchetto, V. Gattei, and T. N. Hartmann. Integrin signaling shaping btk-inhibitor resistance. *Cells*, 11(14):2235, 2022.

[343] K. S. Siveen, K. S. Prabhu, I. W. Achkar, S. Kuttikrishnan, S. Shyam, A. Q. Khan, et al. Role of non receptor tyrosine kinases in hematological malignances and its targeting by natural products. *Mol Cancer*, 17(1):1–21, 2018.

[344] F. Liu, W. Clark, G. Luo, X. Wang, Y. Fu, J. Wei, et al. Alkbh1-mediated trna demethylation regulates translation. *Cell*, 167(3):816–28, 2016.

[345] G. Tripepi, G. Heinze, K. J. Jager, V. S. Stel, F. W. Dekker, and C. Zoccali. Risk prediction models. *Nephrology Dialysis Transplantation*, 28(8):1975–80, 2013.

[346] D. D. W. Twa, D. G. Lee, K. L. Tan, G. W. Slack, S. Ben-Neriah, D. Villa, et al. Genomic predictors of central nervous system relapse in primary testicular diffuse large b-cell lymphoma. *Blood*, 137(9):1256–9, 2021.

[347] C. T. Chen, P. P. Wang, W. J. Mo, Y. P. Zhang, W. Zhou, T. F. Deng, et al. Expression profile analysis of prognostic long non-coding rna in adult acute myeloid leukemia by weighted gene co-expression network analysis (wgcna). *J Cancer*, 10(19):4707, 2019.

[348] C. Chen, S. Liu, X. Jiang, L. Huang, F. Chen, X. Wei, et al. Tumor mutation burden estimated by a 69-gene-panel is associated with overall survival in patients with diffuse large b-cell lymphoma. *Exp Hematol Oncol*, 10:1–11, 2021.

[349] A. Mosquera Orgueira, B. Antelo Rodriguez, J. Á. Diaz Arias, N. Diaz Varela, J. L. Bello López, et al. A three-gene expression signature identifies a cluster of patients with short survival in chronic lymphocytic leukemia. *Journal of Clinical Medicine*, 8(6):847, 2019.

[350] X. Huang, Y. Zhang, X. Luo, J. Bai, L. Chen, Y. Xu, et al. Predictive role of tumor mutation burden-related signatures in response to immunotherapy in hepatocellular carcinoma. *Front Immunol*, 13:854452, 2022.

[351] X. Liang, L. Tseng, Y. Li, X. Li, C. Xue, Y. Bai, et al. Novel multi-dimensional machine learning-based genomic predictor of chronic lymphocytic leukemia prognosis. *Br J Haematol*, 199(2):276–85, 2022.

[352] P. Abrisqueta, P. Balsalobre, J. Delgado, R. Manso, I. Ferrer, M. Xipell, et al. Machine learning algorithm to identify patients with cll at high risk of infection during treatment. *J Clin Med*, 11(15):4366, 2022.

[353] H. Zhang, X. Li, J. Ding, T. Zhao, X. Zhao, Y. Liu, et al. A risk score model based on super-enhancer associated genes for predicting the prognosis of patients with chronic lymphocytic leukemia. *Ann Transl Med*, 10(14):792, 2022.

[354] Y. Ma, P. Lin, W. Q ian, Q. Xu, C. Wang, X. Hou, et al. A novel signature based on seven immune-related genes predicts overall survival in patients with hepatocellular carcinoma. *Journal of Cancer*, 11(16):4627, 2020.

[355] R. Lopez, B. Regnier, J. Camblong, H. Yahi, A. Boichard, C. Massard, et al. Artificial intelligence in the clinic: Are we there yet? *Med Sci (Paris)*, 35(9):713–9, 2019.

[356] D. Flagel, J. Hellwage, M. Mulaw, K. Smolinska, J. Horlacher, S. Wrenger, et al. Igf1 receptor as an essential receptor for poliovirus replication. *J Virol*, 93(21):e01269–19, 2019.

[357] Z. Xie, Z. Liu, X. Fang, X. Liu, Y. Yin, Y. Yu, et al. Apoptotic effects of a traditional chinese medicine, qu mai, on chronic lymphocytic leukemia in a mouse model. *Oncol Lett*, 13(1):368–76, 2017.