

Communication

A Neural Probabilistic Graphical Model for Learning and Decision Making in Evolving Structured Environments

Edmondo Trentin 

Dipartimento di Ingegneria dell'Informazione e Scienze Matematiche (DIISM), Università di Siena, 53100 Siena, Italy; edmondo.trentin@gmail.com; Tel.: +39-0577-233609

Abstract: A difficult and open problem in artificial intelligence is the development of agents that can operate in complex environments which change over time. The present communication introduces the formal notions, the architecture, and the training algorithm of a machine capable of learning and decision-making in evolving structured environments. These environments are defined as sets of evolving relations among evolving entities. The proposed machine relies on a probabilistic graphical model whose time-dependent latent variables undergo a Markov assumption. The likelihood of such variables given the structured environment is estimated via a probabilistic variant of the recursive neural network.

Keywords: probabilistic graphical model; recursive neural network; density estimation; evolving environment; non-stationary environment; relational learning

MSC: 68T05



Citation: Trentin, E. A Neural Probabilistic Graphical Model for Learning and Decision Making in Evolving Structured Environments. *Mathematics* **2022**, *10*, 2646. <https://doi.org/10.3390/math10152646>

Academic Editors: Luca Pancioni, Giacomo Innocenti and Fernando Corinto

Received: 11 June 2022

Accepted: 25 July 2022

Published: 28 July 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The development of artificial intelligence (AI) systems capable of keeping up with evolving environments, just like “biological systems can adapt to new environments”, was posed as Challenge One for AI by Rodney Brooks [1]. To date, the challenge is still open. The relevance of machines capable of coping with evolving environments was further pointed out in the modern framework of evolving computation [2,3]. Evolving environments are met in most machine learning applications under real-world circumstances. This is the consequence of two main factors:

1. the world is not stationary, but it changes over time;
2. a learning machine may be applied under different conditions [4].

Examples are found in fault diagnosis for complex dynamical systems [5], modeling of climate and financial data [6,7], detection of fraud [8] or spam [9]. As pointed out by [10], coping with these evolving environments requires approaches that overcome a fundamental assumption that holds in traditional machine learning, namely that data are drawn from a fixed distribution [10]. In this respect, the headmost contribution of this communication is to cope for the first time in a formal manner with environments that change over time according to time-dependent, unknown probabilistic laws. To this aim, the machine sought is expected to realize a statistical model of the environment that probabilistically accounts for the environmental changes.

To this end, several learning machines suitable for sequential data processing may appear to be promising, e.g., hidden Markov models (HMM) [11] or recurrent neural networks [12]. Nonetheless, in our view, the general notion of a structured environment is better expressed in terms of a relation, or set of relations, over individual entities. Each entity may be represented as a variable-size, entity-specific set of attributes that capture certain characteristics, or perspectives, of the world. Attributes may be either discrete or real-valued. Given these premises, traditional learning machines are unfeasible. The second

contribution of the present communication is to present a new machine that is suitable for the aforementioned relational scenario. To the best of our knowledge, this is the first approach to modeling and learning structured environments that evolve over time.

An evolving structured environment is then expressed by means of a time-dependent graph process defined over the entities. The entities existing in the environment at time t are represented by the vertices of the graph at t . The edges of the graph encapsulate the binary relation(s) over the entities at time t . Labels in the form of real-valued attributes may, in turn, be attached to the edges. These labels account for the evolving type and characteristics of the relation(s) that hold within the environment. We assume that any decision made by the machine at time t does not depend only on the environment at time t , but also, the evolution of the environment that had occurred so far. To fix ideas, a toy example is given by an autonomous robotic platform that is required to move and assist human workers in setting up the booths and furniture within the exhibition spaces of a large trade fair pavilion. The partitioning of the building into booths, conference rooms, etc. changes from time to time, as do the amount, positioning, and type of furniture. The different rooms and pieces of furniture are the entities to be modeled in the present framework, and their mutual positioning is the relation among the entities. The features, or labels, to be associated with such entities may represent the geometry, size, weight, and specific function of the corresponding items or rooms. In order to be capable of navigating autonomously within such an evolving environment, the moving platform must be capable of keeping track of the changes, progressively adapting to the changeable spaces, and making suitable decisions accordingly.

We use a probabilistic framework, such that the environment can be described in terms of probabilistic laws over time-varying random quantities, e.g., the likelihood of the entities being related to each other at time t . Intrinsicly, the machine we are introducing is a novel ‘probabilistic graphical model [13] that represents a particular joint probability density, including a set of latent variables undergoing a first-order Markov assumption, and a set of observable, time-varying random graphs drawn from properly-defined probability density functions (pdf). In the light of our previous experience with neural networks for density estimation [14–16], the short-term modeling of these pdfs is realized via a probabilistic radial basis functions (RBF) [17] variant of recursive neural networks (RNN) [18,19], hereby denoted by RBF-RNN. Conversely, the latent variables and their conditional probabilistic dependencies are modeled via a long-term hidden Markov chain, as in HMMs. The potentialities of combinations of short-term predictors and long-term models of evolving environments were shown empirically by [20]. Moreover, suitable change detection tests based on the HMM likelihood were proven to be successful in detecting drifts in evolving environments [21]. The present approach can be seen as an instance of the online Bayesian learning setup for evolving environments [22] extended to graphical or relational data, where the time-dependent RBF-RNNs play the role of the parameterized probabilistic observation model, and the Markov chain encapsulates implicitly the probabilistic transition model.

The underlying assumption is that by observing the evolution of an environment for a long enough period of time, it is possible to learn the fundamental probabilistic laws that rule its behavior, as well as its evolution, such that suitable models of such laws may be later used to make statistically sound predictions, e.g., making educated decisions. As in Just-in-time classifiers for recurrent concepts drift in evolving environments [23], the present approach exploits the notion of recurrent, underlying concepts that re-present after an unpredictable amount of time, and that the machine shall be able to detect in order to react with the adequate probabilistic response. In the present framework, these concepts are the latent states of nature ruling the pdf of the observations.

Before introducing the machine formally, we need to extend the definitions of random variables and pdf to the present, structured setup. Eventually, learning (in terms of estimating proper statistical parametric models) will rely on a training sample of streams of environmental observations over time. Decision making among the actions $\mathcal{A}_1, \dots, \mathcal{A}_k$

in the environment \mathcal{E} at any future time t will take place within the Bayesian framework, relying on the action-conditional joint pdfs estimated at t via action-specific machines which observe the evolution of \mathcal{E} .

2. Materials and Methods

The present formalization requires assumptions on the type of evolving environments we can cope with. Still, it forms a well-defined ground on which to develop proper algorithms. First, the novel notion of the random environment is introduced, broadening the definition of random graph [24] to a large extent. Let Ω be a sample space, and let \mathcal{V} denote any discrete or continuous-valued universe. The latter is the set of entities. A random environment (RE) on \mathcal{V} and Ω is a function $\mathcal{E} : \Omega \rightarrow \{(V, R) | V \subseteq \mathcal{V}, R \subseteq V \times V\}$. Labels (namely, real-valued vectors associated with the entities or with the elements of R) can be accommodated in the definition in a straightforward manner. Let $\mathbb{E} = \{(V, R) | V \subseteq \mathcal{V}, R \subseteq V \times V\}$ be the space of RE outcomes. A pdf for REs over \mathcal{V} is defined as any function $p : \mathbb{E} \rightarrow \mathbb{R}$ such that (1) $p(\varepsilon) \geq 0, \forall \varepsilon \in \mathbb{E}$, and (2) $\int_{\mathbb{E}} p(\varepsilon) d\varepsilon = 1$. The Lebesgue-measurability of the space of labeled graphs defined on measurable domains is shown in [25], which makes the writing $\int_{\mathbb{E}} p(\varepsilon) d\varepsilon$ meaningful. It is seen that traditional notions of probability theory, such as conditional pdf, joint pdf, statistical independence, etc., are readily extended to REs. An *environment* is any outcome ε of a RE, i.e., drawn from the corresponding pdf $p(\varepsilon)$. This definition covers all cases of labeled/unlabeled entities, as well as environments having variable sizes (e.g., the number of entities making up the environment is not necessarily pre-defined, and it may change over time) and/or variable topology (e.g., two-by-two relations between pairs of entities may evolve, too).

Loosely speaking, a *stochastic environment process* $X(t)$ is any function that maps discrete or continuous time $t \in \mathcal{T}$ onto a corresponding RE. Since by definition REs are random variables, and since we can bijectively represent $X(t)$ as an indexed family $\{X(t) | t \in \mathcal{T}\}$, it is seen that a stochastic environment process is a special case of random process according to the classic definition of the latter (see, for instance, [26], Section 1.9, page 41). Consequently, an extension of the traditional hidden Markov model (HMM) to REs is given by proposing the novel definition of HMM over environments (ε -HMM) as a pair of stochastic processes: a *hidden* Markov chain, that is a traditional discrete-time random process, and an *observable* stochastic environment process which is a probabilistic function of the states of the former.

Let us assume that the evolution of an environment over time $t = 1, \dots, T$ has been observed, and represented as a sequence $Y = \varepsilon_1, \dots, \varepsilon_T$ generated by a hidden stochastic environment process. Also, let Y be the outcome of a sequence $\mathcal{W} = \omega_1, \dots, \omega_L$ of latent states of nature, e.g., drifting concepts. No prior segmentation of Y into subsequences Y_1, \dots, Y_L corresponding to the individual states of nature are known in advance. We propose a hybrid neural/Markovian realization [27] of ε -HMMs for the probabilistic graphical modelling of $p(Y | \mathcal{W})$. Using a standard HMM notation [28], an ε -HMM \mathcal{H} is formally defined as $\mathcal{H} = (S, \pi, \mathbf{A}, B_{\mathcal{E}})$ where: $S = \{S_1, \dots, S_Q\}$ is a set of states (namely, the Q different values that the hidden Markov chain can assume); $\pi = \{P(S_i | t = 0), S_i \in S\}$ is the probability distribution of the initial states (t being a discrete time index); \mathbf{A} is the $Q \times Q$ matrix of the transition probabilities, whose ij -th entry is $a_{ij} = P(S_j \text{ at time } t + 1 | S_i \text{ at time } t)$; finally, $B_{\mathcal{E}}$ is a set of pdfs over REs, called *emission* probabilities, describing the state-specific statistical distributions of the REs: $B_{\mathcal{E}} = \{b_i(\cdot) | b_i(\varepsilon) = p(\varepsilon | S_i), S_i \in S, \varepsilon \in \mathbb{E}\}$.

The observable component of the ε -HMM is realized via neural models of the emission probabilities [29]. For each state of the ε -HMM, a neural network is introduced that estimates the corresponding emission probability given the current status of the environment being observed. Relying on the formalism we introduced in [30], for each $\iota = 1, \dots, Q$ we assume the existence of $d \in \mathbb{N}$ and of the functions $\phi_{\iota} : \mathbb{E} \rightarrow \mathbb{R}^d$ and $p_{\iota} : \mathbb{R}^d \rightarrow \mathbb{R}$ such that $b_{\iota}(\varepsilon_t) = p_{\iota}(\phi_{\iota}(\varepsilon_t))$. It is seen that there are countless pairs of such functions $\phi(\cdot)$ and $\hat{p}(\cdot)$, the simplest choice being $\phi(\varepsilon) = p(\varepsilon), \hat{p}(x) = x$. The function $\phi_{\iota}(\cdot)$ is hereafter referred to as the *encoding* for ι -th state of the ε -HMM, whereas the function $p_{\iota}(\cdot)$ is called

the *emission* associated with the corresponding state. We assume parametric forms $\phi_i(\varepsilon|\theta_{\phi_i})$ and $p_i(\mathbf{x}|\theta_{p_i})$ for the encoding and for the emission, respectively, and we let $\theta_i = (\theta_{\phi_i}, \theta_{p_i})$. The function $\phi_i(\varepsilon|\theta_{\phi_i})$ is realized via an encoding network, suitable to mapping structured environments ε into real vectors \mathbf{x} , as it is found in supervised training of traditional RNNs over graphs [18,19]. Consequently, we identify the parameters θ_{ϕ_i} with the weights of the encoding network. An RBF network is then used to model the emission $p_i(\mathbf{x}_t|\theta_{p_i})$, where the parameters of the RBF play the role of θ_{p_i} . All the parameters in the ε -HMM can be estimated from examples by means of a global optimization algorithm, presented in the next Section, aimed at the maximization of the likelihood of the very ε -HMM given time-varying sequences of empirical observations of the evolving environment.

3. Results

The main result of the present communication is the following gradient-ascent global training technique. The proposed technique is based on the maximum likelihood (ML) criterion. Using standard notation [28], the likelihood \mathcal{L} of the ε -HMM given the sequence Y is given by $\mathcal{L} = \sum_{i \in \mathcal{F}} \alpha_{i,T}$, where $\alpha_{i,T}$ is the forward term for state i at time T . The sum is extended to the set \mathcal{F} of all final states [31] of the ε -HMM. Let us write $q_{i,t}$ in order to represent the fact that the Markov chain is in state S_i at time t . Then, the forward terms $\alpha_{i,t} = P(q_{i,t}, \varepsilon_1, \dots, \varepsilon_t)$ and the backward terms $\beta_{i,t} = P(\varepsilon_{t+1}, \dots, \varepsilon_T | q_{i,t})$ can be computed recursively as usual [28]. In turn, the forward-backward algorithm [28] can be used for ML estimation of the parameters of the underlying Markov chain, i.e., *initial* and *transition* probabilities. For a generic RBF-RNN parameter $\theta \in \theta_i$, instead, gradient-ascent over \mathcal{L} entails a learning rule in the form $\Delta\theta = \eta \frac{\partial \mathcal{L}}{\partial \theta}$, where $\eta \in \mathbb{R}^+$ is the learning rate. We can rewrite $\frac{\partial \mathcal{L}}{\partial \theta}$ as:

$$\frac{\partial \mathcal{L}}{\partial \theta} = \sum_{q=1}^Q \sum_{t=1}^T \beta_{q,t} \frac{\alpha_{q,t}}{b_q(\varepsilon_t)} \frac{\partial b_q(\varepsilon_t)}{\partial \theta} \tag{1}$$

The usual computations over the *trellis* of a standard HMM [28] can be used to obtain the quantities in the right-hand side of Equation (1), except for $\frac{\partial b_q(\varepsilon_t)}{\partial \theta}$ (where $b_q(\varepsilon_t)$ is the output from the corresponding RBF-RNN at time t). Hereafter, we focus on the computation of this partial derivative. Since (i) each state of the ε -HMM has its own RBF-RNN, (ii) in HMMs the emission probabilities for different states are mutually independent [28], and (iii) in HMMs the individual observations (i.e., environments) in the input sequence are assumed to be mutually independent given the state [28], we can simplify the notation by dropping the indexes i and t , and we resort to the calculation of the derivatives of the generic emission $p(\phi(\varepsilon|\theta_{\phi})|\theta_p)$. For any free parameter in the RBF-RNNs, say θ , an explicit formulation for $\frac{\partial p(\phi(\varepsilon|\theta_{\phi})|\theta_p)}{\partial \theta}$ can be obtained as follows. Eventually, the resulting formulation shall be put in place of $\frac{\partial b_q(\varepsilon_t)}{\partial \theta}$ into Equation (1) and the latter, in turn, into the overall learning rule $\Delta\theta = \eta \frac{\partial \mathcal{L}}{\partial \theta}$. There are three categories of parameters of the RBF-RNN that the algorithm shall estimate:

1. the hidden-to-output connection weights c_1, \dots, c_n of the RBF. In order to ensure the satisfaction of the axioms of probability, these weights must range over $(0, 1)$ and sum to one. This is guaranteed by introducing n unconstrained variables $\gamma_1, \dots, \gamma_n$ such that

$$c_i = \frac{\zeta(\gamma_i)}{\sum_{j=1}^n \zeta(\gamma_j)} \tag{2}$$

where $\zeta(x) = 1/(1 + e^{-x})$. The generic variable γ_i is estimated via ML. This guarantees the satisfaction of the axioms. The quantity $\frac{\partial p(\phi(\varepsilon|\theta_{\phi})|\theta_p)}{\partial c_i}$ is computed by applying the chain rule to $\frac{\partial p(\phi(\varepsilon|\theta_{\phi})|\theta_p)}{\partial \gamma_i}$ for $i = 1, \dots, n$.

2. The mean vector μ_i and covariance matrix Σ_i of the generic i -th Gaussian kernel $K_i(\mathbf{x}) = G(\mathbf{x}; \mu_i, \Sigma_i)$ in the RBF. For each component $j = 1, \dots, d$ of the encoding space,

ML parametric estimation of Gaussian mixture models [11] is applied in order to compute the quantities $\frac{\partial p(\phi(\varepsilon|\theta_\phi)|\theta_p)}{\partial \mu_{ij}}$ and $\frac{\partial p(\phi(\varepsilon|\theta_\phi)|\theta_p)}{\partial \sigma_{ij}}$.

3. The parameters $\mathcal{U} = \{v_1, \dots, v_u\}$ of the encoding network. For each $v \in \mathcal{U}$, the quantity $\frac{\partial p(\phi(\varepsilon|\theta_\phi)|\theta_p)}{\partial v}$ is obtained as follows: $\frac{\partial p(\phi(\varepsilon|\theta_\phi)|\theta_p)}{\partial v} = \frac{\partial p(\phi(\varepsilon|\theta_\phi)|\theta_p)}{\partial y} \frac{\partial y}{\partial v}$, where y represents the output from the neuron that is fed from v . The partial derivative $\frac{\partial y}{\partial v}$ is computed as usual. As for $\frac{\partial p(\phi(\varepsilon|\theta_\phi)|\theta_p)}{\partial y}$, the backpropagation through structures (BPTS) algorithm can be applied [18,19]. First, computation of the derivative $\frac{\partial p(\phi(\varepsilon|\theta_\phi)|\theta_p)}{\partial y}$ is straightforward for the connection weights v between hidden and output neurons. This yields the initialization of the δ 's to be backpropagated via BPTS. Then, let us turn our attention to the hidden weights $v = v_{m\ell}$, where ℓ and m are the hidden neurons connected by v . Relying on the aforementioned initialization of the δ 's, the derivative $\frac{\partial p(\phi(\varepsilon|\theta_\phi)|\theta_p)}{\partial v_{m\ell}}$ is finally obtained via standard BPTS.

After the training process has been completed, an estimation of the likelihood at time t of an ε -HMM given an evolving random environment \mathcal{E} can be obtained from the forward terms on the t -th column of the trellis, as in plain HMMs [28]. In turn, action-specific ε -HMMs shall be trained and applied for each possible action, say $\mathcal{A}_1, \dots, \mathcal{A}_k$, in those tasks that require to decide which action to undertake at any time t in the environment \mathcal{E} . In this setup, the i -th ε -HMM models the corresponding action-conditional pdf, namely $p(\varepsilon_1, \dots, \varepsilon_t | \mathcal{A}_i)$. Provided that the prior probability of any individual action is known, be estimated via the usual frequentist approach of Bayesian decision theory [32], the action is chosen to rely on Bayes' decision rule, as usual, as the action \mathcal{A}_{\max} that maximizes $p(\varepsilon_1, \dots, \varepsilon_t | \mathcal{A}_i)P(\mathcal{A}_i), i = 1, \dots, k$.

4. Summary and Conclusions

Most environments may be described as relations over entities whose attributes are random variables. In turn, evolving environments do implicitly entail time sequences of varying relations. The goal of this communication was the foundation of an ad-hoc paradigm for learning and modeling the statistical properties of such sequences. The goal was pursued by combining probabilistic graphical models, encoding neural networks, and constrained RBFs within a unifying framework. A ML adaptation algorithm for the parameters of the overall model was devised under an implicit assumption of recurrent concepts drift, where the concepts are represented by the latent variables. We expect applications in real-world scenarios, as well as extensions of the technique to the incremental real-time adaptation of the parameters.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The author gratefully acknowledges the valuable contributions from nobody.

Conflicts of Interest: The author declares no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---------|---|
| AI | artificial intelligence |
| HMM | hidden Markov model |
| pdf | probability density function |
| RBF | radial basis function |
| RNN | recursive neural network |
| RBF-RNN | probabilistic radial basis functions variant of recursive neural networks |

| | |
|-----------------|------------------------------------|
| RE | random environment |
| ϵ -HMM | HMM over environments |
| ML | maximum likelihood |
| BPTS | backpropagation through structures |

References

- Selman, B.; Brooks, R.A.; Dean, T.; Horvitz, E.; Mitchell, T.M.; Nilsson, N.J. Challenge problems for artificial intelligence. In Proceedings of the Thirteenth National Conference on Artificial Intelligence AAAI'96, Portland, OR, USA, 4–8 August 1996; Volume 2, pp. 1340–1345.
- Benuskova, L.; Kasabov, N. *Computational Neurogenetic Modeling*, 1st ed.; Springer Publishing Company, Incorporated: Berlin/Heidelberg, Germany, 2007.
- Angelov, P.; Filev, D.P.; Kasabov, N. *Evolving Intelligent Systems: Methodology and Applications*; Wiley-IEEE Press: Hoboken, NJ, USA, 2010.
- Shafiullah, N.M.; Pinto, L. One After Another: Learning Incremental Skills for a Changing World. *arXiv* **2022**, arXiv:2203.11176.
- Sayed Mouchaweh, M. *Fault Diagnosis of Hybrid Dynamic and Complex Systems*; Springer: Berlin/Heidelberg, Germany, 2018.
- Overpeck, J.; Meehl, G.; Bony, S.; Easterling, D. Climate data challenges in the 21st century. *Science* **2011**, *331*, 700–702. [[CrossRef](#)] [[PubMed](#)]
- Cont, R. Statistical Modeling of High-Frequency Financial Data. *IEEE Signal Process. Mag.* **2011**, *28*, 16–25. [[CrossRef](#)]
- Mohammadi, M.; Yazdani, S.; Khanmohammadi, M.H.; Ma ham, K. Financial Reporting Fraud Detection: An Analysis of Data Mining Algorithms. *Int. J. Financ. Manag. Account.* **2020**, *4*, 1–12.
- Dada, E.G.; Bassi, J.S.; Chiroma, H.; Abdulhamid, S.M.; Adetunmbi, A.O.; Ajibuwa, O.E. Machine learning for email spam filtering: review, approaches and open research problems. *Heliyon* **2019**, *5*, 180–192. [[CrossRef](#)]
- Polikar, R.; Alippi, C. Guest Editorial Learning in Nonstationary and Evolving Environments. *IEEE Trans. Neural Netw. Learn. Syst.* **2014**, *25*, 9–11. [[CrossRef](#)]
- Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: New York, NY, USA, 2006.
- Salem, F.M. *Recurrent Neural Networks—From Simple to Gated Architectures*, 1st ed.; Springer: Berlin/Heidelberg, Germany, 2022.
- Freno, A.; Trentin, E. *Hybrid Random Fields—A Scalable Approach to Structure and Parameter Learning in Probabilistic Graphical Models*; Springer: Berlin/Heidelberg, Germany, 2011; Volume 15.
- Trentin, E.; Lusnig, L.; Cavalli, F. Parzen neural networks: Fundamentals, properties, and an application to forensic anthropology. *Neural Netw.* **2018**, *97*, 137–151. [[CrossRef](#)]
- Trentin, E. Soft-Constrained Neural Networks for Nonparametric Density Estimation. *Neural Process. Lett.* **2018**, *48*, 915–932. [[CrossRef](#)]
- Trentin, E. Asymptotic Convergence of Soft-Constrained Neural Networks for Density Estimation. *Mathematics* **2020**, *8*, 572. [[CrossRef](#)]
- Ghosh, J.; Nag, A. An Overview of Radial Basis Function Networks. In *Radial Basis Function Networks 2: New Advances in Design*; Howlett, R.J., Jain, L.C., Eds.; Springer: Heidelberg, Germany, 2001; pp. 1–36.
- Sperduti, A.; Starita, A. Supervised neural networks for the classification of structures. *IEEE Trans. Neural Netw.* **1997**, *8*, 714–735. [[CrossRef](#)]
- Scarselli, F.; Gori, M.; Tsoi, A.C.; Hagenbuchner, M.; Monfardini, G. The graph neural network model. *IEEE Trans. Neural Netw.* **2009**, *20*, 61–80. [[CrossRef](#)] [[PubMed](#)]
- Salazar, D.S.P.; Adeodato, P.J.L.; Arnaud, A.L. Continuous Dynamical Combination of Short and Long-Term Forecasts for Nonstationary Time Series. *IEEE Trans. Neural Netw. Learn. Syst.* **2014**, *25*, 241–246. [[CrossRef](#)] [[PubMed](#)]
- Alippi, C.; Ntalampiras, S.; Roveri, M. A Cognitive Fault Diagnosis System for Distributed Sensor Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *24*, 1213–1226. [[CrossRef](#)] [[PubMed](#)]
- Nakada, Y.; Wakahara, M.; Matsumoto, T. Online Bayesian Learning With Natural Sequential Prior Distribution. *IEEE Trans. Neural Netw. Learn. Syst.* **2014**, *25*, 40–54. [[CrossRef](#)] [[PubMed](#)]
- Alippi, C.; Boracchi, G.; Roveri, M. Just-In-Time Classifiers for Recurrent Concepts. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *24*, 620–634. [[CrossRef](#)] [[PubMed](#)]
- Erdős, P.; Rényi, A. On Random Graphs. *Publ. Math. Debrecen* **1959**, *6*, 290–297.
- Hammer, B.; Micheli, A.; Sperduti, A. Universal Approximation Capability of Cascade Correlation for Structures. *Neural Comput.* **2005**, *17*, 1109–1159. [[CrossRef](#)]
- Ross, S. *Stochastic Processes*; Wiley: New York, NY, USA, 1996.
- Bongini, M.; Freno, A.; Laveglia, V.; Trentin, E. Dynamic Hybrid Random Fields for the Probabilistic Graphical Modeling of Sequential Data: Definitions, Algorithms, and an Application to Bioinformatics. *Neural Process. Lett.* **2018**, *48*, 733–768. [[CrossRef](#)]
- Rabiner, L.R. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proc. IEEE* **1989**, *77*, 257–286. [[CrossRef](#)]
- Castelli, I.; Trentin, E. Combination of supervised and unsupervised learning for training the activation functions of neural networks. *Pattern Recognit. Lett.* **2014**, *37*, 178–191. [[CrossRef](#)]

30. Bongini, M.; Rigutini, L.; Trentin, E. Recursive Neural Networks for Density Estimation Over Generalized Random Graphs. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 5441–5458. [[CrossRef](#)] [[PubMed](#)]
31. Bengio, Y. *Neural Networks for Speech and Sequence Recognition*; International Thomson Computer Press: London, UK, 1996.
32. Duda, R.O.; Hart, P.E.; Stork, D.G. *Pattern Classification*, 2nd ed.; John Wiley & Sons: New York, NY, USA, 2001.