ESANN 2025 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 23-25 April 2025, i6doc.com publ., ISBN 9782875870933. Available from http://www.i6doc.com/en/.

Investigating four deep learning approaches as candidates for unified models in time series forecasting and event prediction: application in anesthesia training

Q. Victor[1], I. Clavier[1], H. Boisaubert[1], F. Picarougne[1], C. Lejus-Bourdeau[2] and C. Sinoquet[1]

1 - Nantes University, CNRS, LS2N, UMR 6004, Nantes, France
2 - LE SiMU / Nantes University Hospital, Nantes University, France

Abstract. This paper explores deep learning architectures for the purposes of unsupervised representation learning of hybrid asynchronous data, and joint prediction tasks. We aim to forecast short-term multivariate time series contextualized by events and to predict events contextualized by time series. Our proof-of-concept examines a real-world case of digitally assisted training in anesthesia. We evaluate four different architectures, using two strategies to integrate both time series and event sequences in the models. We assess the prediction quality of the models, and demonstrate that only one of the four architectures achieves performance outcomes compatible with our application objective.

1 Introduction

Nowadays, systems and their environments increasingly generate event traces in parallel with multivariate time series. As a result, new opportunities are emerging to design frameworks for the joint modeling of event traces (ETs) and associated multivariate time series (MTS), thereby enhancing the pace and scope of research in this area. Despite this potential, research on joint ET-MTS modeling remains limited, with existing approaches primarily focused on specialized applications such as survival analysis and rare event prediction.

Our contribution to ET-MTS joint modeling, as discussed here, responds to healthcare professionals' demand for data-driven simulation training aimed at future anesthesia practitioners. Since 2000, Nantes University Hospital has been documenting anesthesia profiles for all surgical procedures performed. Each profile includes a multivariate time series and an event trace. The MTS records the evolution of physiological variables during the operation, while the ET tracks the medical actions performed throughout the surgery.

Our application objective is to predict the short-term evolution of a digital patient's physiological parameters in response to medical actions performed during surgery. As the only human agent of the medical team, the simulator user will perform these actions through the simulator's interface (*e.g.*, a laptop keyboard). In contrast, we need to forecast the actions performed by the rest of the medical team, which is virtual, taking into account the progression of the surgery.

ESANN 2025 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 23-25 April 2025, i6doc.com publ., ISBN 9782875870933. Available from http://www.i6doc.com/en/.

(a) (b)

Fig. 1: (a) Anesthetic profile. The event trace is shown in the upper part of the subfigure. (b) Piecewise constant time series derived from a sparse event trace.

We have identified no existing models that tackle the joint prediction of time series contextualized by events and events contextualized by time series. To bridge this gap, we have specifically tailored four deep learning architectures for the purposes of unsupervised representation learning of mixed asynchronous data and joint forecasting tasks.

2 Joint modeling of mixed asynchronous streams

2.1 Input data

The anesthetic profile of a patient consists of a multivariate time series z and a trace of events u (Figure 1 (a)). The MTS $z = \{z_1, z_2, \dots z_t, \dots z_{\ell_z}]$ of length ℓ_z describes n_v variables in parallel ($z_t \in \mathbb{R}^{n_v}$). The observations z_t are sampled at consistent intervals. We consider a mapping from the integer interval $[1, \ell_z]$ to $\mathbb{R}^+$, to assign the continuous timestamp τ_t to each observation z_t. In the event trace $u = \{(u_1, t_1), \ \cdots (u_{\ell_u}, t_{\ell_u})\}$, each event u_i belongs to a given set $\mathscr{C}$ of n_c event classes and is timestamped by $t_i \in \mathbb{R}^+$. We note that the same event may appear multiple times in u. These events are likely to impact the dynamics of z.

Synchronization and label encoding In any anesthetic profile (z,u), we synchronize the sparse event trace u with z by adding an event category, ϵ, which denotes the absence of an event. At most one event is assumed to occur in the interval $[\tau_t, \tau_{t+1}[$. If an event exists, it is aligned with t; otherwise, ϵ is aligned with t. This results in a univariate categorical time series v, defined on $\mathscr{C} \cup \{\epsilon\}$, with the same length as z. The next step is to randomly assign an integer from $[1, n_c]$ to each label in $\mathscr{C}$, assign 0 to ϵ, and map the events of v to $[0, n_c]$, resulting in the integer-valued univariate time series v_{num}. In this paper, we use two strategies for integrating the streams z and v_{num} into the four architectures under consideration.

Piecewise constant time series Figure 1 (b) shows how the PWC modality handles the sparsity issue by transforming v_{num} into a piecewise constant time series, v_{pwc}. Finally, the $n_v + 1$ variables in z and v_{pwc} are normalized to zero mean and unit standard deviation before being input into any of the four models.

ESANN 2025 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 23-25 April 2025, i6doc.com publ., ISBN 9782875870933. Available from http://www.i6doc.com/en/.

Learning-based embedding Neural networks have become the dominant framework for learning embeddings of categorical data. This prominence arises from their ability to map categorical data to points in a vector space, while safeguarding meaningful relationships between categories. In the EMB modality, we first normalize the n_v variables of z. The discretized data in v_{num} are passed through an embedding layer integrated into each model. Given the specified embedding dimension d_{ev}^{emb}, the embedding is learned simultaneously with the rest of the model, which is trained for the joint prediction tasks. Next, we concatenate the resulting embedded data with the normalized data from z. These data, with dimension $d_{ev}^{emb} + n_v$, are then input into each of the four models.

2.2 Investigated models

In the class of RNNs, we refer to ST-LSTM as a stack of LSTM layers, in which Long Short-Term Memory units recurrently update two latent states (c_t and h_t) to capture long-term dependencies in sequential data [1]. Next we selected the Multivariate Time-series Graph Neural Network (MTGNN) framework for its ability to learn the graph structure directly from the data, enabling it to capture the dependencies among variables in MTS [2]. Furthermore, MTGNN interweaves temporal and spatial convolutional modules, making it well-suited for modeling spatio-temporal relationships. Finally, we included the Vanilla Transformer [3], and one of its variants. The Informer was specifically designed to address the scalability issue in Transformers, for the time series forecasting task [4].

2.3 Model training

Each of the four models will be trained using pairs $\{x, y\}$, where x and y are sequences of dimension $n_v + 1$ (modality PWC) or $n_v + d_{ev}^{emb}$ (modality EMB), with lengths k and h. Parameter k specifies the length of past context from which to perform h-ahead prediction during training by minimizing a loss function. The trained model will be used for h-ahead prediction for any dynamic systems described by (z, u) (see Section 2.1). A composite loss controls training by combining mean squared error (MSE) and categorical cross-entropy (CCE): $\Phi = \text{MSE} + \alpha\ \text{CCE}$, where α accounts for the scale difference between MSE and CCE.

3 Experiments

3.1 Anesthesia dataset, implementation details, metrics

Our dataset includes 1,000 anesthetic profiles of men approximately 30 years old, with no prior medical history, who underwent laparoscopic inguinal hernia surgery. This procedure involves up to 37 distinct medical acts. The four physiological variables analyzed are detailed in Figure 1 (a). The dataset was split into training (60%), validation (20%), and test (20%) subsets.

We used PyTorch 2.5.0 to customize the ST-LSTM and Transformer architectures. The MvTs library was employed to tailor the MTGNN and Informer

ESANN 2025 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 23-25 April 2025, i6doc.com publ., ISBN 9782875870933. Available from http://www.i6doc.com/en/.

Table 1: Architecture-dependent hyperparameters considered in our work.

ST-LSTM	$\boldsymbol{n_\ell}$: number of stacked hidden layers, $\boldsymbol{n_n}$: number of nodes (*i.e*, LSTM units) common to the hidden layers; defining the dimension of both hidden state vectors h_t and c_t.
MTGNN	$\boldsymbol{n_\ell}$: total number of hidden layers, $\boldsymbol{n_{c\ell}}$: number of stacked convolutional layers, $\boldsymbol{n_{cc}}$: number of channels in the model's convolutional layers, $\boldsymbol{n_{rc}}$: number of channels used in the residual connections of the model.
Transformer and Informer	$\boldsymbol{n_\ell}$: total number of hidden layers in either the encoder or the decoder, $\boldsymbol{d_{t,i}^{emb}}$: embedding size used to represent the input data in a latent space.

models, as well as to manage data loading, training, validation, and testing [5] (https://github.com/MTS-BenchMark/MvTS).

We evaluated MTS prediction using four standard metrics: MSE, MAE, MAPE, and SMAPE. Event forecasting was assessed using global accuracy (GA), along with arithmetic (FSA) and geometric (FSG) F-score means.

3.2 Experimental settings

Table 1 lists the model-dependent hyperparameters adjusted in our experiments.

Preliminary study A NOE (No Event) modality was included for comparison with PWC and EMB modalities. To enable frugal exploration before intensive grid search, we varied one hyperparameter at a time from a reference configuration, keeping other parameters **constant** for each model and modality (see Table 2).

Grid search The preliminary study indicated that the EMB modality performed best. During the grid search, we varied the hyperparameters as shown in Table 3, and set the same values of common hyperparameters as in Table 2. For each of the n_{conf} hyperparameter configurations of a model, we ranked each of

Table 2: Hyperparameter configurations for the preliminary study. For instance, for ST-LSTM with NOE modality, we started from $(n_\ell, n_n) = (\mathbf{2}, \mathbf{64})$ and derived the configurations $(\mathbf{2}, \underline{128})$, $(\mathbf{2}, \underline{256})$, $(\underline{1}, \mathbf{64})$, and $(\underline{3}, \mathbf{64})$.

	NOE, PWC and EMB	**EMB**
ST-LSTM	$n_\ell \in \{1, \mathbf{2}, 3\}$; $n_n \in \{\mathbf{64}, 128, 256\}$	$d_{ev}^{emb} \in \{\mathbf{5}, 10, 15\}$; $\alpha \in \{0.25, \mathbf{0.50}, 0.75\}$
MTGNN	$n_\ell \in \{2, \mathbf{3}, 4\}$; $n_{c\ell} \in \{1, \mathbf{2}, 3\}$ $n_{cc} \in \{16, \mathbf{32}, 64\}$; $n_{rc} \in \{\mathbf{8}, 16, 32\}$	$d_{ev}^{emb} \in \{\mathbf{5}, 10, 15\}$; $\alpha \in \{0.25, \mathbf{0.50}, 0.75\}$
Transformer Informer	$n_\ell \in \{2, 4, \mathbf{6}, 8\}$; $d_{t,i}^{emb} \in \{64, 128, 256, \mathbf{512}, 1024\}$	$d_{ev}^{emb} \in \{\mathbf{5}, 10, 15, 20, 25\}$; $\alpha \in \{0.25, \mathbf{0.50}, 0.75\}$
Number of models trained	ST-LSTM: 5 models; MTGNN: 9 models; attention-based models: 8 models	ST-LSTM: 9 models; MTGNN: 13 models; attention-based models: 14 models
Hyperparameters common to the four models considered		
• $k = 30$ • $h = 10$ • Batch size: 32 • n_{epochs}= 200 (controled by early stopping) • Dropout: enabled • Gradient descent optimization algorithm: Adam • Network weight initialization: Glorot Uniform • Learning rate = 10^{-4}		

ESANN 2025 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 23-25 April 2025, i6doc.com publ., ISBN 9782875870933. Available from http://www.i6doc.com/en/.

Table 3: Range of hyperparameter variations for the grid search.

ST-LSTM	• $n_\ell \in \{2,3,4\}$ • $n_n \in \{128,256,512\}$ • $d_{ev}^{emb} \in \{10,15,20\}$ • $\alpha \in \{0.2,0.3,0.4\}$
MTGNN	• $n_\ell \in \{2,3,4\}$ • $n_{c\ell} \in \{2,3,4\}$ • $n_{cc} \in \{32,64,128\}$ • $n_{rc} \in \{8,16,32\}$ • $d_{ev}^{emb} \in \{8,10,12\}$ • $\alpha \in \{0.25,0.35,0.45\}$
Transformer	• $n_\ell \in \{3,4,5\}$ • $d_{t,i}^{emb} \in \{128,256,512\}$ • $d_{ev}^{emb} \in \{18,20,22\}$ • $\alpha \in \{0.4,0.5,0.6\}$
Informer	• $n_\ell \in \{5,6,7\}$ • $d_{t,i}^{emb} \in \{256,512,1024\}$ • $d_{ev}^{emb} \in \{3,5,7\}$ • $\alpha \in \{0.6,0.7,0.8\}$
MTGNN: 729 models trained; each of the three other architectures: 81 models trained	

the seven metrics within the range $[1,n_{conf}]$. Next, we computed the respective average ranks obtained over the 4 and 3 metrics dedicated to continuous and categorical variables. Each configuration was scored using r_{avg}, the average of these two quantities.

3.3 Results and analysis

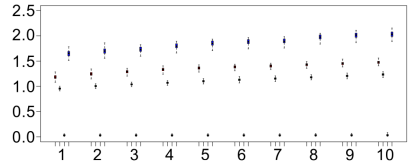
Preliminary study Table 4 highlights the performances measured with the MAPE metric and global accuracy. We outline four conclusions: (i) for all four models, the PWC and EMB modalities respectively deteriorate and improve the MTS forecasting performance with respect to the NOE modality; (ii) all models but the Transformer exhibit failures or poor performance in event prediction, even under the most favorable modality (EMB); (iii) in this modality, MTGNN is the second best model after the Transformer; (iv) in this experimental setting, no clear trend emerges, except for ST-LSTM, regarding the influence of hyperparameters on MTS prediction or event prediction performances.

Model selection and comparative analysis The grid search confirms the preliminary study's conclusions, except that this time ST-LSTM and the Informer exhibit similar poor event forecasting performances. Figures 2 (a) and 2 (b) compare the performance distributions across all prediction horizons while Figure 2 (c) focuses on the four best models obtained through score r_{avg}. The Transformer shows the highest performances. The MTGNN's spatial and temporal convolution modules fail to deliver acceptable performance for our future simulator. The Informer is unexpectedly the slowest to train of the four architectures (The averages are 24 hours, 2.9 days, 4.6 days and 6.1 days respectively for the ST-LSTM, MTGNN, Transformer and Informer). Notably, we observe that the reported ability of the Informer for long time series prediction does not necessarily translate to strong performance in the short term.

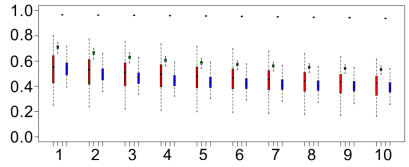
Table 4: Predictive performances at horizon 10 from the preliminary study.

	Modalities				
	NOE	**PWC**	**EMB**	**PWC**	**EMB**
	MAPE			Global accuracy	
ST-LSTM	1.6-1.8	2.5-3.4	1.4-1.6	0.25-0.33	0.04-0.26
MTGNN	1.3-1.4	1.5-1.7	1.2-1.3	0.21-0.27	0.74-0.81
Transformer	$\leq$0.2	0.4-2	0.01-0.1	0.01-0.99	0.96-0.99
Informer	1.5-1.6	1.8-3.7	1.9-2.0	0.16-0.29	0.66-0.83

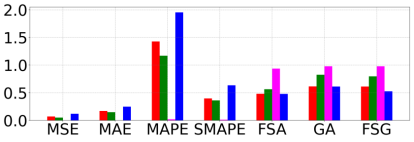
ESANN 2025 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 23-25 April 2025, i6doc.com publ., ISBN 9782875870933. Available from http://www.i6doc.com/en/.



(a) Time series prediction (MAPE)



(b) Event prediction (FSA)



ST-LSTM	n_ℓ=3, n_n=512, d_{ev}^{emb}=15, α=0.2
MTGNN	n_ℓ=4, $n_{c\ell}$=4, n_{cc}=128, n_{rc}=32, d_{ev}^{emb} = 10, α = 0.35
Transformer	n_ℓ=5, $d_{t,i}^{emb}$=128, d_{ev}^{emb}=20, α=0.6
Informer	n_ℓ=6, $d_{t,i}^{emb}$=256, d_{ev}^{emb}=3, α=0.7

(c) Description of the top four models and their performances

Fig. 2: (a) and (b) Distributions of performances across horizons and architectures. The boxplots are displayed in the order: ST-LSTM (red), MTGNN (green), Transformer (pink), Informer (blue). (c) Top four models.

4 Conclusion and future work

In this work, we have tailored four deep learning architectures for unsupervised representation learning of hybrid asynchronous data and joint forecasting tasks. We demonstrate the applicability of the Transformer model for joint prediction, with a specific focus on data-driven simulation of a patient under anesthesia. Future work will address surgeries with an increased number of variables and events. In personalized medicine, a Transformer-based solution will predict the risk of a medical action on a real patient's digital twin before surgery. Moreover, beyond simulating the next medical action not performed by the user, event prediction will help detect when the user deviates from the standard procedure.

References

[1] S. Hochreiter and J. Schmidhuber, Long Short-Term Memory, *Neural Computing*, 9(8):1735–1780 (1997).

[2] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang and C. Zhang, Connecting the dots: multivariate time series forecasting with graph neural networks. In *proceedings of the* 26^{th} *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (KDD), pages 753–763, 2020.

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones et al., Attention is all you need. *Proceedings of the Annual Conference on Neural Information Processing Systems* (NeurIPS), pages 5998–6008, 2017.

[4] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li et al., Informer: beyond efficient Transformer for long sequence time-series forecasting. In *proceedings of the* 35^{th} *AAAI Conference on Artificial Intelligence* (AAAI), pages 11106–11115, 2021.

[5] J. Ye, W. Li, Z. Zhang, T. Zhu, L. Sun and B. Du, MvTS-library: an open library for deep multivariate time series forecasting, *Knowledge-Based Systems*, 283:111170, 2024.