Dynamic Bayesian Networks and Transfer Learning Enable the Development of Deep Sequence-Based Models on Small-Sample Data

Enrico Longato[1], Erica Tavazzi[1], Adriano Chiò[2], Giovanni Sparacino[1], Barbara Di Camillo[1,3]
[1] *Department of Information Engineering, University of Padova, Padova, Italy.*
[2] *Neuroscience Department "Rita Levi Montalcini", University of Torino, Torino, Italy.*
[3] *Department of Comparative Biomedicine and Food Science, University of Padova, Padova, Italy.*

***Abstract*—The development of prognostic models for rare diseases is often limited by the latter's extremely low prevalence, which, in turn, affects the breadth of modelling techniques that may be applied. Notably, a severe lack of training data may hinder or prevent the implementation of high-capacity models, such as deep neural networks, despite their known effectiveness in predicting adverse outcomes and efficiency in making use of longitudinal data. To address this issue, in the present work, we propose a novel methodological pipeline where we first train a dynamic Bayesian network (DBN) on the few available data to simulate an adequate number of virtual patients whose variables are linked by the same probabilistic relationship over time as in the original data; then, we train a deep learning model based on recurrent neural networks on the simulated data; and, finally, we apply finetuning, a transfer learning (TL) technique, to adapt the model to the real data. To demonstrate the potential usefulness of our approach, we apply it to the prediction of 3-year mortality in amyotrophic lateral sclerosis (ALS), a rare (<0.01% prevalence), fatal neurodegenerative disease, starting from a population of 985 patients from the Piemonte and Valle d'Aosta ALS (PARALS) register. We show that our pipeline of DBN and TL effectively combines the simulated data it is able to generate and the few available real data, leading to an 8.2% AUROC improvement over a reference deep learning model trained only on the real data.**

***Keywords*—Data Augmentation, Deep Learning, Dynamic Bayesian Networks, Rare Diseases, Transfer Learning.**

I. Introduction

Prognostic model development for rare diseases presents an additional set of challenges compared to more frequent conditions. Notably, whereas, for many clinical applications, data scarcity can partially be remedied by extending the study to different types and sources of data (e.g., by making a secondary use of administrative or routinely acquired data), the inherently small number of people affected by rare diseases sets a hard limit to data availability. In turn, this reflects on the breadth of modelling approaches that can be applied with the reasonable expectation of obtaining robust results. Particularly, the usage of high-capacity models such as deep neural networks, which have been successful in tackling complex modelling tasks, especially using longitudinal data [7], seems difficult to justify in light of the disproportionately limited sample size of many rare-disease databases.

To address this issue, one possibility might be resorting to standard data augmentation techniques to artificially increase the number of available training examples via transformations of the data [8]. However, while some domains, such as image analysis, naturally lend themselves to the application of data augmentation techniques, the best way to augment typical clinical data, including patient characteristics, vital parameters, and, possibly, longitudinally-acquired timeseries, is still unclear. An alternative approach would be to leverage data generation techniques [16] to produce a sufficient number of virtual patients, and, then, train a model using the simulated data. Dynamic Bayesian networks (DBNs) are an effective tool to perform purely data-driven simulation of clinical data by, first, learning the joint probability distribution of patient covariates over time, and, then, resampling from the trained DBN to simulate the static and longitudinal data of an arbitrary number of virtual patients. While these simulated data can be (and have successfully been [14]) used for model training as if they were real patient data, there is no guarantee that the resulting model would then translate directly to real-life individuals. Challenges such as this are the focus of transfer learning (TL), a range of techniques aiming at transferring knowledge across tasks. A typical transfer learning workflow starts from a model that is known to solve a task on a given domain, the *source* domain, which is then adapted to solve a similar task on an adjacent domain, the *target* domain. This approach has shown great promise in different fields, especially in image recognition where, e.g., a deep neural network developed on generic image data (source domain) can be successfully adapted to radiology applications (target domain) without full retraining [5].

In the present work, we propose a novel approach combining DBNs and TL to develop deep, sequence-based models despite low data availability. Briefly, we first train a DBN on the training data and use it to simulate a large number of virtual patients, complete with their longitudinal data; then, we train a deep learning model based on recurrent neural networks on the simulated data (source domain); finally, we apply finetuning, a TL technique, to adapt the network trained on the simulated data to the target domain, i.e., the original, real data. Specifically, we focus on the case study of predicting 3-year mortality in amyotrophic lateral sclerosis (ALS), a rare (<0.01% prevalence) but fatal neurodegenerative

disease, based on the longitudinal data of 985 patients from the Piemonte and Valle d'Aosta ALS (PARALS) register [3], and show that our approach leads to a 8.2% performance improvement relative to an equivalent model trained from scratch on the real data only.

II. Data description and preparation

A. Database

We studied the data of a real-world cohort of 985 ALS patients extracted from the PARALS register. The data consisted of demographic and clinical information collected during routine screening visits, outlining the patients' status from diagnosis onwards. The information collected at diagnosis included sex, age at onset, site of onset (spinal or bulbar), diagnostic delay, body-mass index (BMI) both premorbid and at diagnosis, forced vital capacity (FVC) at diagnosis, the result of a genetic test on the main ALS-related genes (namely C9orf72, FUS, SOD1, and TARDBP), ALS familiality, and presence of frontotemporal dementia (FTD). Additionally, longitudinal information was also available both at the time of diagnosis and at each follow-up visit: namely, the use/administration of non-invasive ventilation (NIV) and percutaneous endoscopic gastrostomy (PEG), and the values of the Milano-Torino staging (MiToS) system [2], a scale comprising four binary variables reflecting impairment (or lack thereof) in the following four functional domains: breathing, walking/self-care, swallowing, and communicating. Finally, the date of death or tracheostomy (henceforth referred to simply as death, as tracheostomy-free survival is a conventional endpoint in ALS clinical trials) is also reported, if applicable.

B. Outcome definition and exclusion criteria

The prediction outcome for this study was 3-year mortality, i.e., the occurrence of death within 3 years after the 9-month longitudinal baseline that served as the input to the predictive model (see Section III-B).

Starting from the initial cohort of 985 patients, we applied the following exclusion criteria.

a) Subjects whose date of death preceded their first recorded visit.
b) Subjects with fewer than 2 visits before and 1 after the 9-month mark (the baseline length for prediction), to help the convergence of the DBN and ensure that all subjects had at least some longitudinal data.
c) Subjects for whom it was impossible to determine the prediction outcome, i.e., survivors censored before the 3-year mark.

This process resulted in the selection of 626 subjects, for a total of 7241 visits.

C. Data split and imputation

We split the data into a training, a validation, and a test sets, comprising 64%, 16%, and 20% of the subjects, respectively. The data characteristics are shown in Table I, with the continuous and categorical variables reported as means $\pm$ SD, and frequencies and proportions, respectively.

We then imputed the missing data by chained equations using the *mice* R package [15] with default parameters. To fit our modelling pipeline (see Section III) and avoid information leakage, we trained two imputers, one on the training data to impute missing values on the training and validation sets, and another on the union of the training and validation data to impute missing values on the entire database, including the test set. These steps resulted in a training set of 399 subjects and 4640 visits, a validation set of 98 subjects and 1128 visits, and a test set of 129 subjects and 1473 visits with no missing values.

TABLE I: DEMOGRAPHIC AND CLINICAL FEATURES OF THE STUDY POPULATION AFTER PREPROCESSING.

	Levels	**Training (n=399)**	**Validation (n=98)**	**Test (n=129)**
Sex	Female	197 (49.4%)	39 (39.8%)	60 (46.5%)
Familiality	Yes	44 (11.0%)	4 (4.1%)	10 (7.8%)
	<NA>	1 (0.3%)	0 (0%)	2 (1.6%)
Genetics	Mutated	44 (11.0%)	9 (9.2%)	8 (6.2%)
	<NA>	17 (4.3%)	6 (6.1%)	17 (13.2%)
FTD	Yes	42 (10.5%)	7 (7.1%)	19 (14.7%)
	<NA>	141 (35.3%)	31 (31.6%)	44 (34.1%)
Onset site	Bulbar	122 (30.6%)	26 (26.5%)	48 (37.2%)
Age at onset (years)		62.67 ± 11.17	62.78 ± 10.79	62.56 ± 10.34
Diagnostic delay (months)		12.00 ± 10.06	11.04 ± 10.28	12.00 ± 13.56
BMI premorbid (kg/m^2)		26.14 ± 4.39	26.17 ± 4.22	25.80 ± 3.73
BMI at diagnosis (kg/m^2)		25.19 ± 4.40	25.09 ± 4.71	24.32 ± 3.73
FVC at diagnosis (%)		93.41 ± 22.00	93.66 ± 23.13	90.69 ± 24.21
NIV	Administered	164 (41.1%)	42 (42.9%)	51 (39.5%)
PEG	Administered	134 (33.6%)	31 (31.6%)	49 (38.0%)
MiToS walking/self-care	Impaired	319 (79.9%)	77 (78.6%)	94 (72.9%)
MiToS swallowing	Impaired	142 (35.6%)	33 (33.7%)	50 (38.8%)
MiToS communication	Impaired	109 (27.3%)	31 (31.6%)	35 (27.1%)
MiToS breathing	Impaired	184 (46.1%)	44 (44.9%)	55 (42.6%)
Cumulative incidence of 36-month mortality		70.90%	68.40%	76.60%

III. Methodology

The proposed methodological pipeline is based on the cascade of a DBN and a TL approach known as finetuning [12]. Briefly, we performed the following sequence of actions.

i) Train a DBN on the training set and use it to simulate data from the training and validation sets.
ii) Develop a deep learning model based on recurrent neural networks on the simulated training data and use the simulated validation data for early stopping and hyperparameter optimisation. We call this model the encoder.
iii) Remove the output neuron from the encoder and attach a fully-connected subnetwork in its place to obtain the TL model.
iv) Train the TL model as per Section III-B on the real training data and use the real validation data for hyperparameter optimisation.

Finally, we compared the predictive performance of the TL model to that of a model with the same architecture as the encoder, but trained and optimised on the real data, on the real test set. The target metric was the cumulative/dynamic area under the receiver-operating characteristic curve (AUROC) [1] at 3 years.

A. DBN

DBNs are probabilistic graphical models capable of encoding the joint conditional probability distributions (CPDs) over time of a set of longitudinally-collected variables [10]. DBNs are well suited to modelling the evolution of diseases for descriptive and prognostic purposes (e.g., [13]). After a DBN has been trained, the probability relationships over time that it has encoded in the learning phase can be used to simulate,

starting from an initial (clinical) condition, the evolution over time of a population of subjects, thus obtaining an in silico population that preserves the trends seen in the real training population.

Here, we learnt a DBN starting from the training set using the *bnstruct* R package [4]. First, in order to better model the temporal evolution of ALS, we computed two derived variables for each visit: the distances of each visit from the onset (time since onset, TSO) and from the following visit (time between visits, TBV). Then, to allow the DBN to observe a more balanced occurrence of the death outcome, we coded survival at each visit as a binary variable answering the question "Will the subject die within the next 9 months?" We set the layering for the DBN, i.e., the rules that govern forbidden relationships, in accordance with literature knowledge (e.g., MiToS score values must depend on TSO, as they are expected to change as the disease progresses) and common sense (e.g., sex cannot depend on clinical values). The DBN structure was inferred using the Max-Min Hill-Climbing algorithm, using the Bayesian information criterion as the score function, while the CPDs' parameters were computed via maximum a posteriori estimation.

After learning the DBN on the training set, we simulated 100 synthetic repetitions of each real patient of both the training and validation sets starting from their first recorded visit, and ending after 40 virtual visits or on (simulated) death.

B. Transfer learning

1) Input data coding: The TL part of the proposed pipeline was based on recurrent neural networks. We considered a baseline window of 9 months to predict subsequent 3-year mortality. To leverage longitudinal information on the MiToS scores during the first 9 months, we used an input coding scheme that had previously proved successful for other applications [6]. Specifically, we built a (17×10) matrix where each row represented a variable and each column a visit within the 9-month baseline (up a maximum to 10). The first 6 rows were dedicated to dynamic data (the four MiToS scores, NIV, and PEG), while the remaining 11 to static variables (sex, age, familiality, genetics, onset site, FTD, diagnostic delay, time since onset, premorbid BMI, BMI at diagnosis, FVC at diagnosis). The data were right-aligned such that the latest visit within the baseline was in the tenth column. Static data were only reported on the tenth column as well. We used a masking value of "-1" to signal that either a visit was missing (which resulted in the 6 dynamic columns being masked at the missing row) or that the column referred to a static variable (which resulted in all rows except the tenth one being masked). Masked values were ignored during training.

2) Encoder and reference model: Developing an encoder is the first step to applying finetuning-based TL. In this context, an encoder is simply a model trained on the source domain (simulated data), which we will then extend to the target domain (real data) by attaching a fully-connected subnetwork to it and performing a second round of training on the real data. Particularly, for the encoder, we implemented a deep learning architecture based on recurrent layers as follows: the masked input matrix described in Section III-B1 enters a recurrent layer, possibly with dropout; then, the transformed sequence passes through a series of fully-connected layers with ReLU activation, of progressively halving size, finishing on a single output neuron with a sigmoid activation function, representing the predicted outcome, i.e., 3-year mortality. The hyperparameters of the architecture were: the type of recurrent layer (GRU or LSTM), its size (16, 32, or 64), its dropout (0 or 0.1) and recurrent dropout (0 or 0.1) levels, its activation function (ReLU or tanh), the number of fully-connected layers (2 or 3), and the number of nodes in the first fully-connected layer (16, 32, or 64). We used the simulated training data to tune the weights of the encoder via the ADAM algorithm (0.0001 learning rate) and the binary cross-entropy loss function. We used the simulated validation data to perform early stopping when the loss function on the simulated validation set stopped improving, and also to select the best hyperparameters as the ones maximising the (simulated) validation AUROC.

We followed the same procedure to train a reference model on the real data, i.e., a deep neural network with the same architecture, but trained and optimised on the real training and validation data, respectively. This model served as a comparison for the TL model.

3) Finetuned transfer learning model: To turn the encoder, i.e., the model able to solve the 3-year mortality prediction task on the source domain of simulated data, into the full TL model, we proceeded as follows. First, we removed the output neuron and retained the model, with its pre-trained weights, up until the last fully-connected layer, and we attached an untrained fully-connected subnetwork (hyperparameters: 2 or 3 progressively half-sized layers; 16, 32, or 64 nodes in the first layer) ending with a sigmoid-activated output neuron; then, keeping the encoder weights fixed for a number of pretraining epochs (0, 10, or 20), we trained only the newly-added fully-connected layers with the ADAM algorithm (learning rate 0.00005) on the real training set; finally, we finetuned the entire network, including the encoder weights, with a 100 times smaller learning rate. We used the real validation set to perform early stopping and hyperparameter optimisation. This procedure yielded the final TL model.

IV. Results and discussion

Figure 1 shows the DBN obtained on the training set, with nodes representing the variables and directed edges entering a node corresponding to the conditional dependency effect of the child node on its parents. Self-loops on the dynamic variables (blue nodes in the figure) represent the dependence of the variable at a time point from itself at the previous time point. Analysing the resulting relationships, we can see that the DBN effectively integrates the imposed literature relationships with others directly learned from the data. Numerous known relationships emerge, such as the relationship between ALS familiarity and age of onset [9], or the dependence of diagnostic delay on gender and site of onset [11]. The fact that the credibility of the relationships learnt by the DBN can

be checked via inspection of the graphical model allows for greater control over the expected quality of the simulated data compared to generic data augmentation techniques.

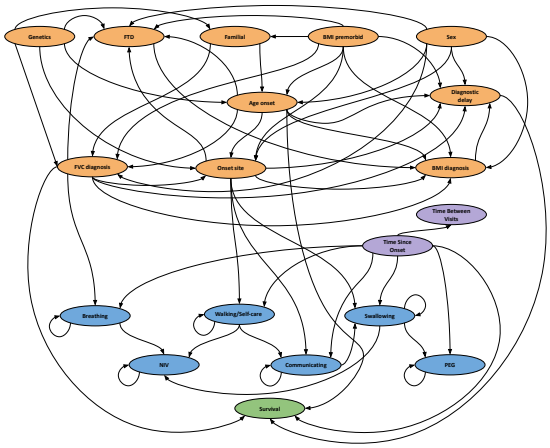

Fig. 1: DBN obtained on the training set. Dynamic variables are reported in blue (MiToS scores, NIV, and PEG), purple (TBV and TSO), and green (survival), while static variables are in orange.

Hyperparameter optimisation yielded the following results. The optimal reference model was made up of a size-32 ReLU-activated GRU with 0.1 dropout and 0.1 recurrent dropout, and two fully-connected layers of 32 and 16 neurons. The optimal TL model was the cascade of the optimal encoder (similar hyperparameters to the reference model, but different weights: size-32 GRU, ReLU activation, no dropout; two fully-connected layers of 32 and 16 neurons) and an optimal subnetwork of two fully-connected layers of sizes 16 and 8; the optimal number of pretraining epochs during which the encoder weights were fixed was 20.

TABLE II: PERFORMANCE EVALUATION ON THE REAL TEST SET (n=129).

Model	Training data	AUROC
Reference model	Real	0.649
Finetuned TL model	Real and simulated	0.702

Table II compares the reference model and the TL model on the task of predicting 3-year mortality after a 9-month baseline on the real data. The difference between the two was that the former only used the real data, whereas the latter was finetuned starting from an encoder learnt only on the data simulated by the DBN. We observed a substantial relative performance improvement of 8.2% between the reference model's AUROC of 0.649 and the TL model's AUROC of 0.702.

These results support the hypothesis that, while small-sample datasets might be insufficient, on their own, to develop deep sequence-based models, the opportune application of a cascade of DBNs (to simulate an adequate number of virtual subjects retaining the same conditional probability relationships over time as the original, real subjects) and TL (to adapt the model from the source domain of simulated data to the target domain of real data) may allow deep recurrent neural networks to learn a robust temporal representation of the data regardless of the original sample size. Having learnt such representation, a very limited number of real data are sufficient to finetune the model's parameters and improve on the reference model's performance. Future developments will pertain to the optimisation of the trade-off between the number of simulated data (computationally expensive) and model performance. This preliminary result is especially encouraging in the field of rare diseases, as it presents a novel, but effective way to implement high-capacity, sequence-based models despite the inevitably low cardinality of the available data.

Acknowledgement

This research was supported by the University of Padova project C94I19001730001, by the Italian Ministry of Health grant RF-2016-02362405, and by the Italian Ministry of Education, University and Research (PRIN) grant 2017SNW5MB. EL and ET were founded by the Department of Information Engineering, University of Padova (Research Grant B junior).

References

[1] Aasthaa Bansal and Patrick J Heagerty. A tutorial on evaluating the time-varying discrimination accuracy of survival models used in dynamic decision making. *Medical Decision Making*, 38(8):904–916, 2018.

[2] Adriano Chiò, Edward R Hammond, et al. Development and evaluation of a clinical staging system for amyotrophic lateral sclerosis. *Journal of Neurology, Neurosurgery & Psychiatry*, 86(1):38–44, 2015.

[3] Adriano Chiò, Gabriele Mora, et al. Secular Trends of Amyotrophic Lateral Sclerosis: The Piemonte and Valle d'Aosta Register. *JAMA Neurology*, 74(9):1097–1104, 2017.

[4] Alberto Franzin, Francesco Sambo, and Barbara Di Camillo. bnstruct: an R package for Bayesian Network structure learning in the presence of missing data. *Bioinformatics*, 33(8):1250–1252, 2017.

[5] Belal Hossain, SM Hasan Sazzad Iqbal, et al. Transfer learning with fine-tuned deep CNN ResNet50 model for classifying COVID-19 from chest X-ray images. *Informatics in Medicine Unlocked*, 30:100916, 2022.

[6] Enrico Longato, Barbara Di Camillo, et al. Time-resolved trajectory of glucose lowering medications and cardiovascular outcomes in type 2 diabetes: a recurrent neural network analysis. *Cardiovascular Diabetology*, 21(1):159, 2022.

[7] Enrico Longato, Gian Paolo Fadini, Giovanni Sparacino, Angelo Avogaro, et al. A deep learning approach to predict diabetes' cardiovascular complications from administrative claims. *IEEE Journal of Biomedical and Health Informatics*, 25(9):3608–3617, 2021.

[8] Kiran Maharana, Surajit Mondal, and Bhushankumar Nemade. A review: Data pre-processing and data augmentation techniques. *Global Transitions Proceedings*, 2022.

[9] Puja R Mehta, Ashley R Jones, Sarah Opie-Martin, et al. Younger age of onset in familial amyotrophic lateral sclerosis is a result of pathogenic gene variants, rather than ascertainment bias. *Journal of Neurology, Neurosurgery & Psychiatry*, 90(3):268–271, 2019.

[10] Kevin Patrick Murphy and Stuart Russell. Dynamic bayesian networks: representation, inference and learning. 2002.

[11] Hipolito Nzwalo, Daisy de Abreu, Michael Swash, Susana Pinto, and Mamede de Carvalho. Delayed diagnosis in als: the problem continues. *Journal of the Neurological Sciences*, 343(1-2):173–175, 2014.

[12] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1717–1724, 2014.

[13] Chiara Roversi, Erica Tavazzi, Martina Vettoretti, et al. A dynamic bayesian network model for simulating the progression to diabetes onset in the ageing population. In *2021 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI)*, pages 1–4. IEEE, 2021.

[14] Erica Tavazzi, Sebastian Daberdaku, et al. Predicting functional impairment trajectories in als: a probabilistic, multifactorial model of disease progression. *Journal of neurology*, pages 1–21, 2022.

[15] Stef van Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software, Articles*, 45(3):1–67, 2011.

[16] Marco Viceconti, Adriano Henney, and Edwin Morley-Fletcher. In silico clinical trials: how computer simulation will transform the biomedical industry. *International Journal of Clinical Trials*, 3(2):37–46, 2016.