Multimodal Deep Learning for a Global data set of COVID-19 Patients

Ioannis Bantzis
EPFL
ioannis.bantzis@epfl.ch

Manos Chatzakis
EPFL
emmanouil.chatzakis@epfl.ch

Maxence Hofer
EPFL
maxence.hofer@epfl.ch

***Abstract*—This paper presents an experimental evaluation of binary classifiers based on regression and deep learning models on a global data set of COVID-19 patients. The data comprises multimodal tabular data, comprising both static and time series entries. We compare different models including a novel modular algorithm and provide an extended experimental evaluation of the classifiers under different configurations. Our results demonstrate that different models have different performance w.r.t. the corresponding metric, and that generally the multilayer perceptron has significantly better performance in most of the settings.**

I. Introduction

Motivation. The COVID-19 pandemic has dominated the world in the recent years. The infectious disease data observatory (IDDO) made a major effort to better understand the disease by coordinating a large scale international collection of interoperable data gathered from patients. The data is multimodal, containing both static entries, such as age, gender or country, as well as time series, e.g. the temperature of a patient for every day of hospitalization. This multi-modal form of the data is challenging to handle as different models may be applicable to different types of data. For example, a logistic regression or an MLP model, that do not contain any memory about previous states, could be applicable for static data but for time series, it does not always guarantee good results. On the other hand, the long short-term memory of an LSTM network could easily handle time series[9], but using it for static data could be slow, as it requires more computational power while better results are not guaranteed. Additionally, the performance of an LSTM over time series is significantly connected to the type of the series, leading us to expect cases where an LSTM could give mediocre results in such type of data.

In this work, we present an experimental evaluation of multiple binary classification machine learning models that predict if a patient will survive COVID-19 infection, given the input data. We compare logistic regression, MLP and LSTM encoders. Our MLP and LSTM are implemented using [14], a modular deep learning framework that allows neural networks able to encode the features sequentially in a flexible order and combination, and most importantly, using a range of different encoders.

Challenge. Although this massive dataset of patients was collected in a coordinated effort, it still presents many inconsistencies and challenges that limit its use ([8] [7] [13]). Firstly, the large size of the data set could limit the training possibilities on conventional sources of computation. Secondly, missing data is a major issue and is often systematic w.r.t. the label of survival (i.e. patients who survive may be less intensely monitored). This systematic missingness especially occurs in time series observations, where values correspond per day measurements. Thus the processing, cleaning and sampling of the data must be done carefully to avoid bias.

Contributions. Our contributions can be summarized in the following:

- We present preprocessing and analysis methods for the clinical dataset of over 800,000 patients
- We implement several binary classifiers, including regression models, feature-wise modular encoding models and multi-modal combinations of encoders
- Finally, we present an experimental evaluation where we evaluate the performance of the models under different metrics and configurations, for a selected number of static and time series features extracted from the dataset.

II. Background

In this section we present the essential background and preliminary material.

Logistic Regression. Logistic regression (LR) is a binary classifier which fits a sigmoid function in order to classify the input data. Logistic regression is widely used in practice, and it is proven to give satisfying results in multiple settings . LR has been used extensively in COVID-19 prognostication and diagnosis, due to its ease, explainability and high performance in low-dimensional data([2][15]), . Some studies have had promising results in smaller data sets with validation accuracy up to 0.95[3]

MoDN. MoDN[14] (Modular Clinical Decision Support Networks) were proposed by our host group, intelligent Global Health. MoDN proposes a sequential combination of MLP encoders able to support a flexible feature-by-feature (feature-wise) encoding structure. It is able to handle systematically missing data as the model is never exposed to the entire dataset at once. This simplified implementation uses only MLP encoders.

MoMoNet. MoMoNet (multi-modal modular networks) is the general-purpose code adaption of MoDN able to combine various encoder types and thus handle multi-modal data. Relevant to our work, it offers support for MLP and LSTM encoders.

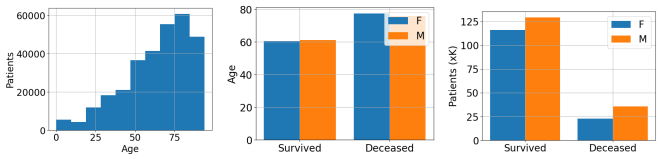
Appropriate baselines for MODN and MoMoNet are their ”monolithic” equivalents (i.e. an MLP or LSTM encoding all

features at once). We also compare various combinations of encoders.

III. Data

In this section we present the dataset we use for our test and training.

Description. Our data contain observations from 800,000 COVID-19 patients distributed across 15 countries in 103 hospitals and institutions. Some descriptive statistical insights about the data are shown in figure 1, where we present the age distribution of the patients, and comparison between the mean age and gender of the survived and deceased patients. Overall, a total of 19% patients were reported has having deceased.



(a) Age Dist. (b) Age Comp. (c) Gender Comp.

Fig. 1: Static feature analysis. We denote males as M and females as F. (a) Distribution of the age of the available patients (b) Comparison of the age of deceased and survived males and females (c) Comparison of the number of deceased and survived males and females

Feature selection. For our analysis, we selected a descriptive number of features, both static and continuous (time series), with a goal to predict if a patient is deceased. A complete list of our features (with their acronyms) and our label (L) are gathered in table I. There are many methods for feature extraction in literature ([6] [5] [4]), but we selected to extract the time series features of the dataset, which come from laboratory data measurements[1] and vital signs[11], and the static features that are included in the demographic part of the data[10].

Feature	Type	Acr	Feature	Type	Acr
Death	L	D	Oxygen Saturation	TS	OxS
Age	St	A	Heart Rate	TS	HR
Gender	St	G	Leukocytes	TS	Leuk
Temperature	TS	Tem	Platelets	TS	Pl
Systolic Blood Pressure	TS	SBP	Urea Nitrogen	TS	UN
Respiratory Rate	TS	RR	C Reactive Protein	TS	CRP
Diastolic Blood Pressure	TS	DBP	Lymphocytes	TS	Lym
Neutrophils	TS	Neut	Creatinine	TS	Cr
Potassium	TS	Pot	Sodium	TS	Sod

TABLE I: Selected dataset features. We denote L as the Label, St the Static Features and TS the time series. Acr. is the acronym we will use for our experiments.

Cohort selection. The dataset includes many inconsistencies, mostly due to missing data. Thus, we decided to select the subset of the patients that have the most complete data, i.e. the subjects that have all information about the static features (age, gender) and at least one observation of the time series features. For the missingness in the time series, we imputed the missing data with the mean of the corresponding feature across the dataset. As 91% of patients are from Europe, we decided to drop the patients from other continents in order to avoid bias. Thus, 10K patients were selected. In this selected population, 50% patients survived (balanced sample). The sample contains 5761 male patients and 4239 female patients. Among the males, the average age of the people who survived is 62, while for the deceased is 75 years. For the females, the average survival age is 62, while the average age for the deceased patients is 77 years.

For our training and testing procedures, we extracted a balanced sample of 10K patients. For reasons we will explain in later sections, we create two versions of the data, $data_{static}$ and $data_{ts}$. The $data_{static}$ contains (i) the static features and (ii) the time series features, represented as the mean of the time series. The $data_{ts}$ contains (ii) the dimensions of the time series of the features.

Example. We explain the difference through and example: Consider the temperature of a patient for 4 days (i.e. 4 dimension time-series), e.g. $t = [38.0, 38.2, 37.9, 37.6]$ for days 1, 2, 3, 4 of hospitalization. The $data_{static}$ version will represent this as one single feature, $temperature = mean(t)$. On the other hand, for the same patient, the $data_{ts}$ representation will have numerous features to represent this, $temperature_1$, $temperature_2$, $temperature_3$, $temperature_4$, for each day respectively.

IV. Models

In this section we present our classification models.

Baselines. Our baselines consist of "monolithic models" that take all features at once (i.e. not in sequence, as in MoMoNet). We have logistic regression models, MLPs and LSTMs. We define $logistic_{static}$ as the logistic regression model operating on the $data_{static}$ version of the dataset and $logistic_{ts}$ as the logistic regression model operating on $data_{ts}$. Similarly we define, MLP_{static}, MLP_{ts} and $LSTM_{ts}$. The MLP and LSTM models are identical to the modules used in MoMoNet, but encode all features at once. Exceptionally, the monolithic LSTM uses only time-series.

Unimodal MoMoNet We define $MLP_{featstatic}$, MLP_{featts}, which use the featurewise MLP encoders of MoMoNet, encoding each feature one by one. Similarly, we define $LSTM_{featts}$ which uses the LSTM featurewise encoder of MoMoNet.

Multimodal MoMoNet We combine the MLP and LSTM encoders in order to create a single model for both static and time series data. Here, we apply the MLP to the static features and LSTMs to time series features. An overview of this procedure is depicted in figure 2, where we show how the different encoders are applied in the model. We define this model as $MLP - LSTM$ and it operates on the $data_{ts}$ version of the dataset.

V. Experimental Evaluation

In this section we present our experimental evaluation.

Setup. We performed the experimental evaluation on a system with 10 cores and 16GBs of memory, running a MacOS operating system. All models and evaluations are implemented in

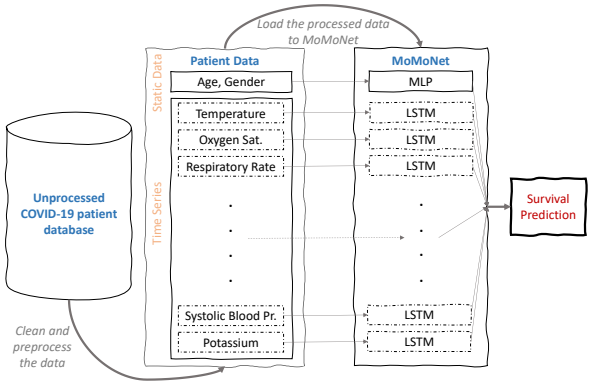


Fig. 2: Overview of Multimodal MoMoNet encoder combination

Python. The evaluation models are those described in section IV and we calculate their performance based on Accuracy, F1-Score, Precision and Recall.

Comparison. We compared the described models under several configurations, by varying the dimensionality (i.e. the data points) of the time series. We evaluated the models with selecting dimensionality of 3, 5, 10 and 19 (which is the max possible). We present the results for the evaluation metrics stated above, which are represented by the values on y-axis. X-axis is the corresponding feature. It is important to note two factors:

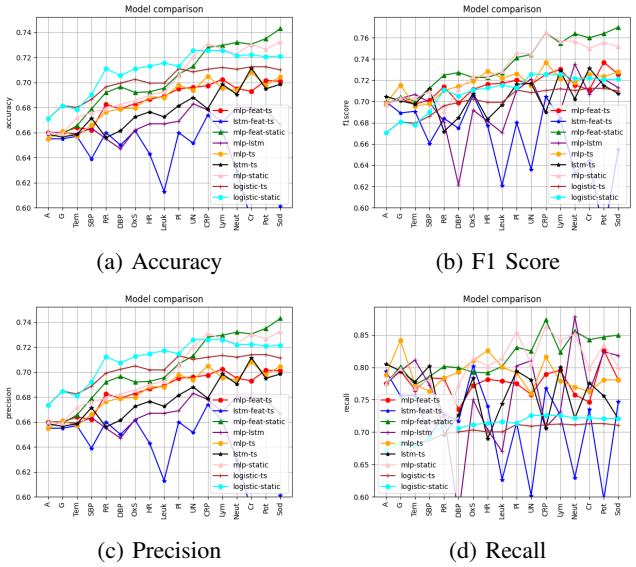
- A data point in a graph that corresponds to the Nth feature, corresponds to the evaluation of the model using all previous features of the graph
- The X-axis does not always correspond to the number of features used. For example, a static model will use one feature for the temperature, but a time-series model will use several features, depending on the selected dimensionality.

Figure 3 presents the evaluation of the models for 3 dimensions (i.e 3 time points for the models that encode time series). We see that overall, the $MLP_{featstatic}$ has good performance for all metrics, while for the rest of the models, the performance differs based on the selected metric. We also notice that the $MLP{-}LSTM$ model has the best performance for recall.

Figure 4 presents the evaluation of the models for 5 dimensions (for the models that encode time series). For this configuration we see that $MLP_{featstatic}$ has the best performance in all configurations and metrics, reaching 0.85 F1 score and 0.74 accuracy (not a significant difference from 3 dimensions).

Figure 5 presents the evaluation of the models for 10 dimensions (for the models that encode time series). Here, the best performance for Accuracy, F1 Score and Precision is the $MLP_{featstatic}$, while for Recall the best model is $MLP-LSTM$. Again, no significant improvement is seen compared to 3-dimensions.

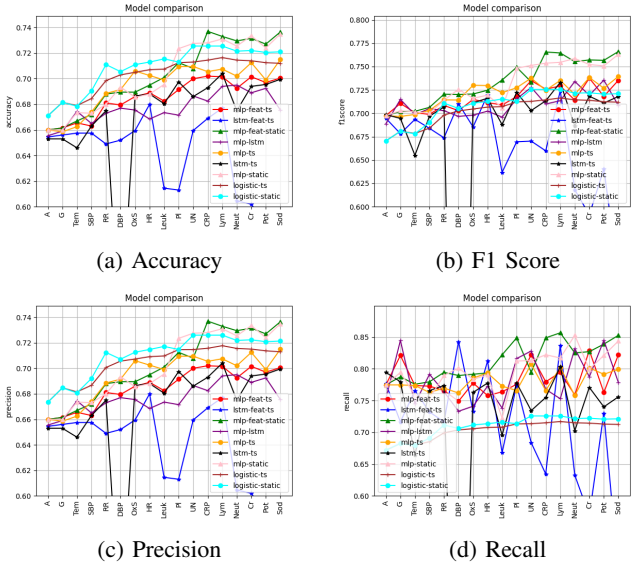
Figure 6 presents the evaluation of the models for 19



(a) Accuracy (b) F1 Score

(c) Precision (d) Recall

Fig. 3: Evaluation for 10K samples using 3 dimensions (i.e. 3 time points) for our models.



(a) Accuracy (b) F1 Score

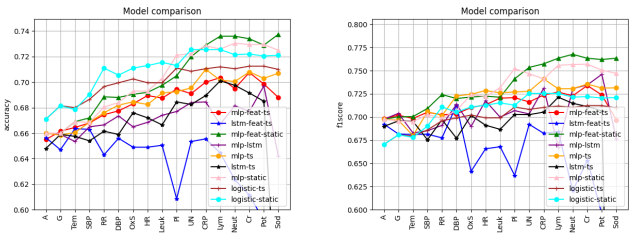
(c) Precision (d) Recall

Fig. 4: Evaluation for 10K samples using 5 dimensions (i.e. 5 time points) for our models.

dimensions (for the models that encode time series). Here, the best performance variate according to the metric. For Accuracy and Precision, the best model seems to be MLP_{static}, for F1 Score the best model is $MLP_{featstatic}$ while for Recall the best model is $MLP-LSTM$.

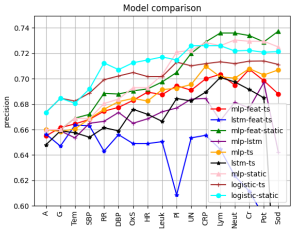
Table II gathers the results for the best models in every dimensionality configuration. We see that $MLP_{featstatic}$ is a dominant model for most experiments.

Generally, among all experiments with different dimensionality configuration we see that most of the models perform consistently, with scores around 0.65-0.75 for F1 Score. An in-

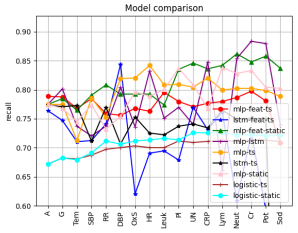


(a) Accuracy

(b) F1 Score

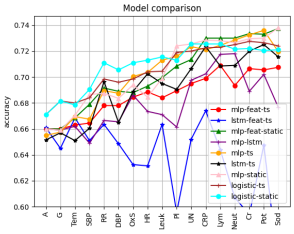


(c) Precision

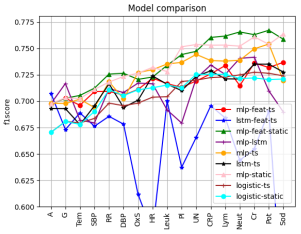


(d) Recall

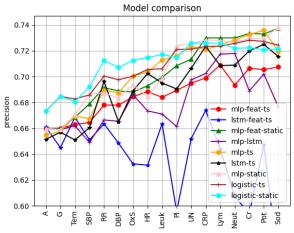
Fig. 5: Evaluation for 10K samples using 10 dimensions (i.e. 10 time points) for our models.



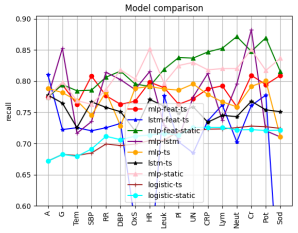
(a) Accuracy



(b) F1 Score



(c) Precision



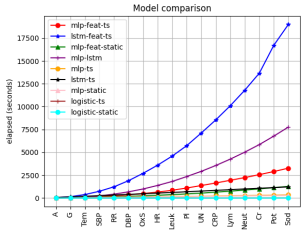
(d) Recall

Fig. 6: Evaluation for 10K samples using 19 dimensions (i.e. 19 time points) for our models.

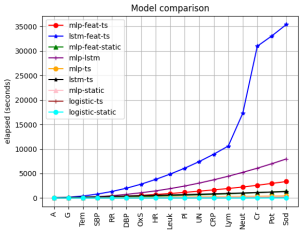
teresting case is the $LSTM_{featts}$ and $LSTM_{ts}$ model, which experiences high drops in the performance in several cases and configurations. LSTM is an applicable model when it comes to time series, and can give exceptional results when it is used properly [12]. In the experiments where the drops occur, we have used LSTMs as encoders for the whole feature list, which means that we don't exploit their memory attributes correctly, and the model can possibly correlate relations between data by mistake, leading to mediocre performance in several cases. For experiments that use higher number for dimensions, we see that the $MLP - LSTM$ model has improved performance,

TS Dimensions	Accuracy	F1 Score	Precision	Recall
3	$MLP_{featstatic}$	$MLP_{featstatic}$	$MLP_{featstatic}$	$MLP - LSTM$
5	$MLP_{featstatic}$	$MLP_{featstatic}$	$MLP_{featstatic}$	$MLP_{featstatic}$
10	$MLP_{featstatic}$	$MLP_{featstatic}$	$MLP_{featstatic}$	$MLP - LSTM$
19	MLP_{static}	$MLP - LSTM$	MLP_{static}	$MLP - LSTM$

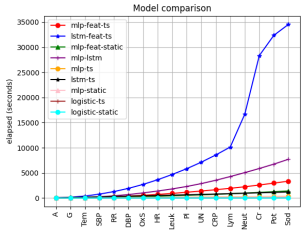
TABLE II: Evaluation Results.



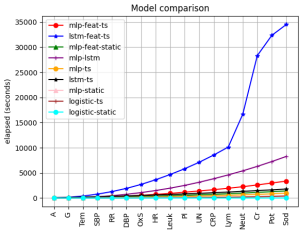
(a) 3 dimensions



(b) 5 dimensions



(c) 10 dimensions



(d) 19 dimensions

Fig. 7: Training time, 10K samples

because LSTMs are operated on specific time series (e.g. a single LSTM encoder for temperature, one for Heart Rate etc.), and there are enough dimensions for the LSTM to produce meaningful results.

Training Time. Figure 7 presents the needed training time for all presented models, measured in seconds, for all possible configurations of dimensionality. Not surprisingly, logistic regression based models require less time, while lstm featurewise and the combination of encoders require the most time, because of the need for multiple training of different encoders for each feature category. The MLP featurewise encoder is also expensive because of the feature-by-feature encoding, but it is still faster than the combination and the LSTM feature wise, because of the lack of memory in the MLP model.

VI. Conclusion

We presented an experimental evaluation of machine learning classification methods operated in multimodal COVID-19 patient data. We implemented and evaluated several models, starting from regression, proceeding to deep learning models, many types of MLP and LSTM encoders, and even combinations of them. Our experimental evaluation shows the different results and efficiency of these models under different metrics and configurations. As a future work, this evaluation could be extensively progressed, by tuning the different hyperparameters of the encoders, such as the hidden layers and state size, bigger or smaller samples of patients and by testing different preprocessing methods.

Acknowledgments. This work was performed in collaboration with the intelligent Global Health (iGH) laboratory of EPFL. We want to thank Mary-Anne Hartley and Hojat Karami for their supervision and guidance throughout the project, as well as Thierry Bossy for providing us with the MoMoNet code and helping us with technical issues.

References

[1] Talha Burak Alakus and Ibrahim Turkoglu. "Comparison of deep learning approaches to predict COVID-19 infection". In: *Chaos, Solitons & Fractals* 140 (2020), p. 110120.

[2] Abdullah M Almeshal et al. "Forecasting the spread of COVID-19 in Kuwait using compartmental and logistic regression models". In: *Applied Sciences* 10.10 (2020), p. 3402.

[3] Sudhir Bhandari et al. "Logistic regression analysis to predict mortality risk in COVID-19 patients from routine hematologic parameters". In: *Ibnosina Journal of Medicine and Biomedical Sciences* 12.02 (2020), pp. 123–129.

[4] Verónica Bolón-Canedo, Noelia Sánchez-Maroño, and Amparo Alonso-Betanzos. "A review of feature selection methods on synthetic data". In: *Knowledge and information systems* 34.3 (2013), pp. 483–519.

[5] Girish Chandrashekar and Ferat Sahin. "A survey on feature selection methods". In: *Computers & Electrical Engineering* 40.1 (2014), pp. 16–28.

[6] 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)*. 2015, pp. 1200–1205. DOI: 10.1109/MIPRO.2015.7160458.

[7] Shashi Kushwaha et al. "Significant applications of machine learning for COVID-19 pandemic". In: *Journal of Industrial Integration and Management* 5.04 (2020), pp. 453–479.

[8] Wei Tse Li et al. "Using machine learning of clinical data to diagnose COVID-19: a systematic review and meta-analysis". In: *BMC medical informatics and decision making* 20.1 (2020), pp. 1–13.

[9] Benjamin Lindemann et al. "A survey on long short-term memory networks for time series prediction". In: *Procedia CIRP* 99 (2021), pp. 650–655.

[10] Dhruv Patel et al. "Machine learning based predictors for COVID-19 disease severity". In: *Scientific reports* 11.1 (2021), pp. 1–7.

[11] Honnesh Rohmetra et al. "AI-enabled remote monitoring of vital signs for COVID-19: methods, prospects and challenges". In: *Computing* (2021), pp. 1–27.

[12] Ralf C Staudemeyer and Eric Rothstein Morris. "Understanding LSTM–a tutorial into long short-term memory recurrent neural networks". In: *arXiv preprint arXiv:1909.09586* (2019).

[13] Hafsa Bareen Syeda et al. "Role of machine learning techniques to tackle the COVID-19 crisis: systematic review". In: *JMIR medical informatics* 9.1 (2021), e23811.

[14] Cécile Trottet et al. "Modular Clinical Decision Support Networks (MoDN)—Updatable, Interpretable, and Portable Predictions for Evolving Clinical Environments". In: *medRxiv* (2022).

[15] Kandi Xu et al. "Application of ordinal logistic regression analysis to identify the determinants of illness severity of COVID-19 in China". In: *Epidemiology & Infection* 148 (2020).