CLINICAL DATA ENCODER: Encoding Clinical Data with Null Values Imputation

Angelo Nardone
Giordano Scerra
Davide Borghini
Davide Marchi

Università di Pisa

Introduction

Digital Health Twin

- Clinical data often **limited**: it is difficult to create accurate and robust ML models.

- Collecting more data is obviously a solution, but….
- … more interesting to find better ways to manage the little data available!

- Overcome this problem: creation of a **digital health twin** [3][4]**.** Allows us to:
    - Compare data between similar patients.
    - Speed up the diagnostic process.
    - Reduce the likelihood of errors.
    - Identify effective treatments more quickly.

Introduction

Encoding Clinical Data

- Clinical data are data with:
  - a large number of features;
  - several **null values.**

- Create a model that **encodes clinical data** in a **latent space.**
  - Easier to recognise the similarity of patients.
  - Able to handle null values well and ...
  - ...able to do **imputation** on null values

- Evaluated the model by comparing the results of a classification task applied on the original and encoded data.

Table of Contents

Dataset

Overview

- Worked with **real clinical data** provided by the Pisa Hospital.

- First time used: it is an extension of the dataset used in [2].

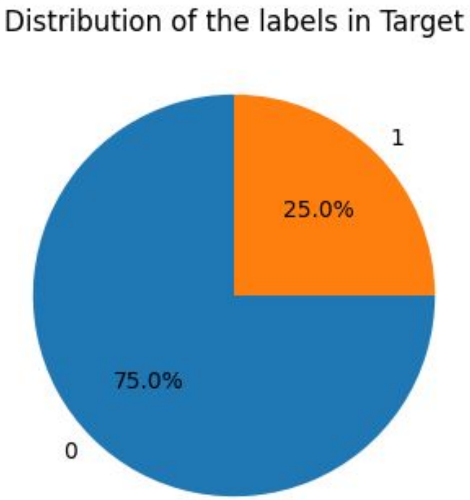
- A data total of about 8000 patients.
- Extracted 68 independent variables for each patient:
  - Each variable represented a clinical data.
  - 46 binary variables.
  - 22 continuous variables.

Dataset

Extract Target

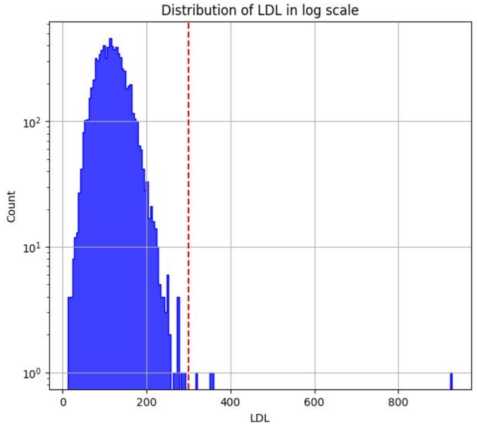
Target

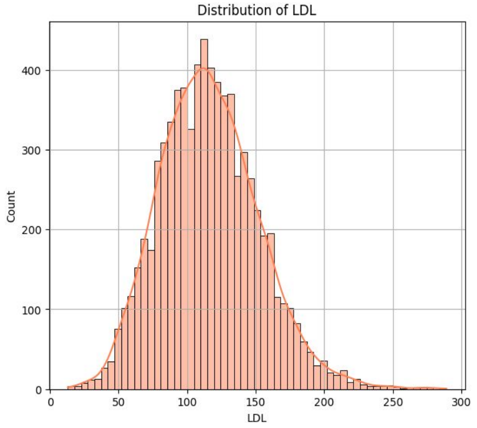
- Extracted the target using additional information.
- Assigned **1** to patients who **died within 8 years** from first admission, **0 otherwise**.
- Good tradeoff: **class balance** and **meaningful timeframe**.



Dataset

Remove Outliers





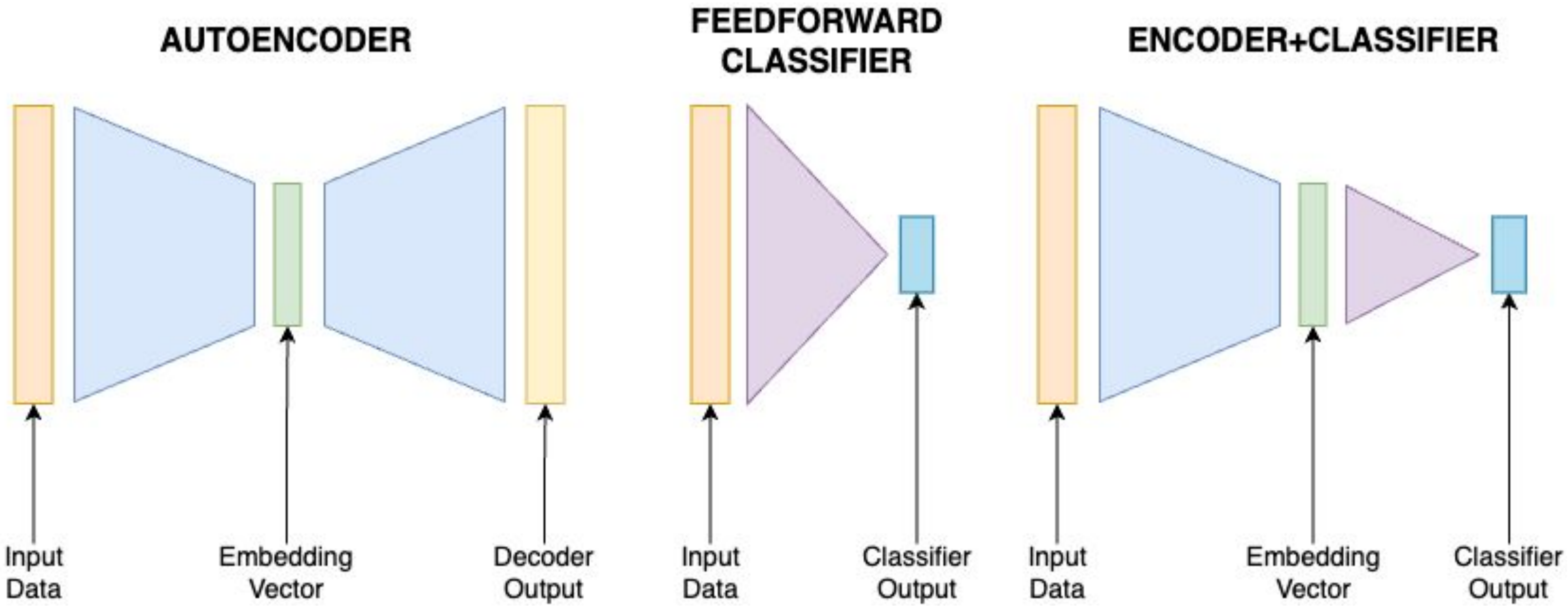
Cleaning

- Creatinine and Vessels contained several null values as in [2].
- Other columns had outliers: global and local detection approaches.
- The dataset had **2.1% null data**.

Table of Contents

Models

Overview

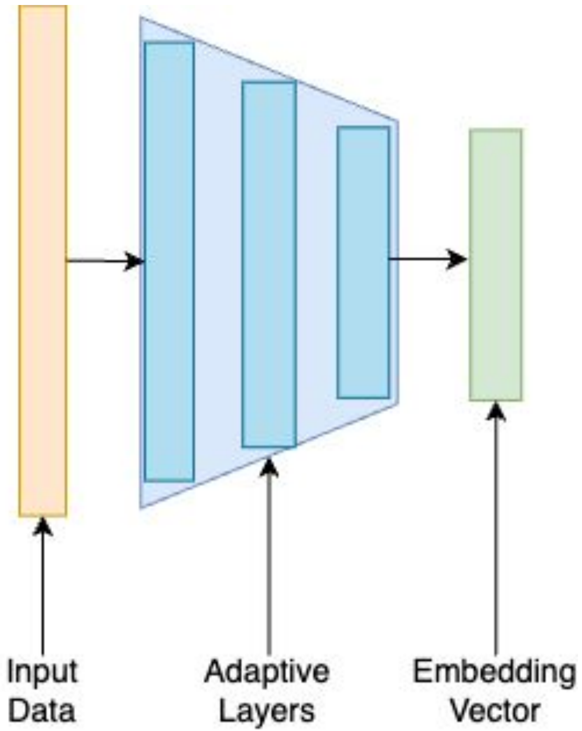


MIEO (Masked Input Encoded Output)

Our Personalized Autoencoder

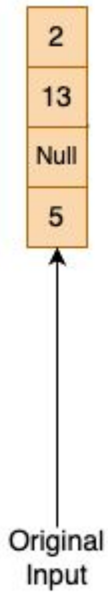
- How is different from classical autoencoder:
    - Handle null values.
    - Enforce imputation.
    - Handle binary columns.

- Three **adaptive layers** according to the size of the **embedding output**.



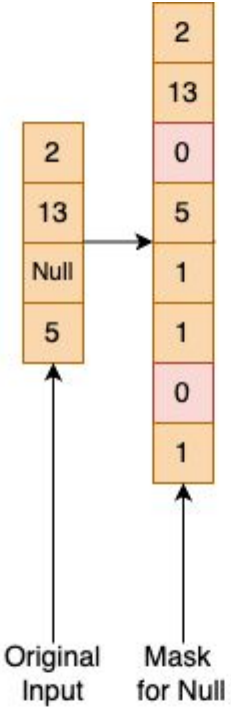
MIEO: How it Works

Starting Input



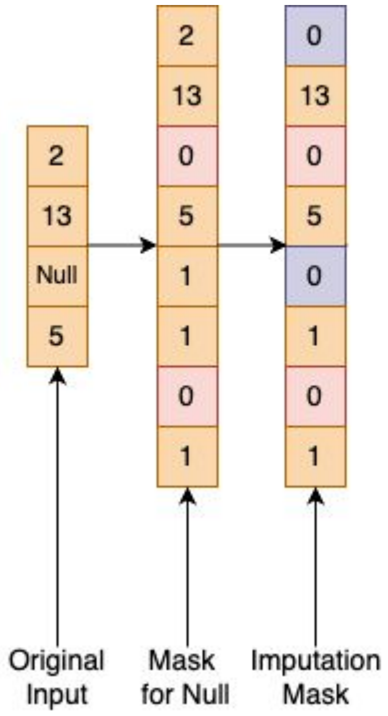
MIEO: How it Works

Masking Null Data



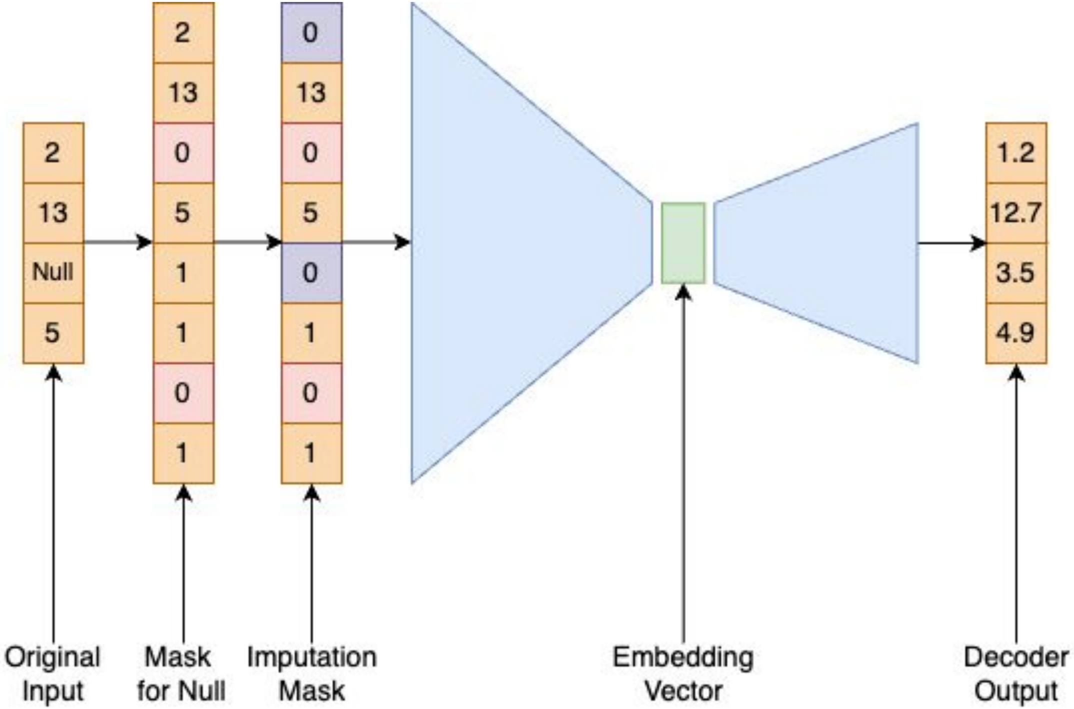
MIEO: How it Works

Masking for Imputation



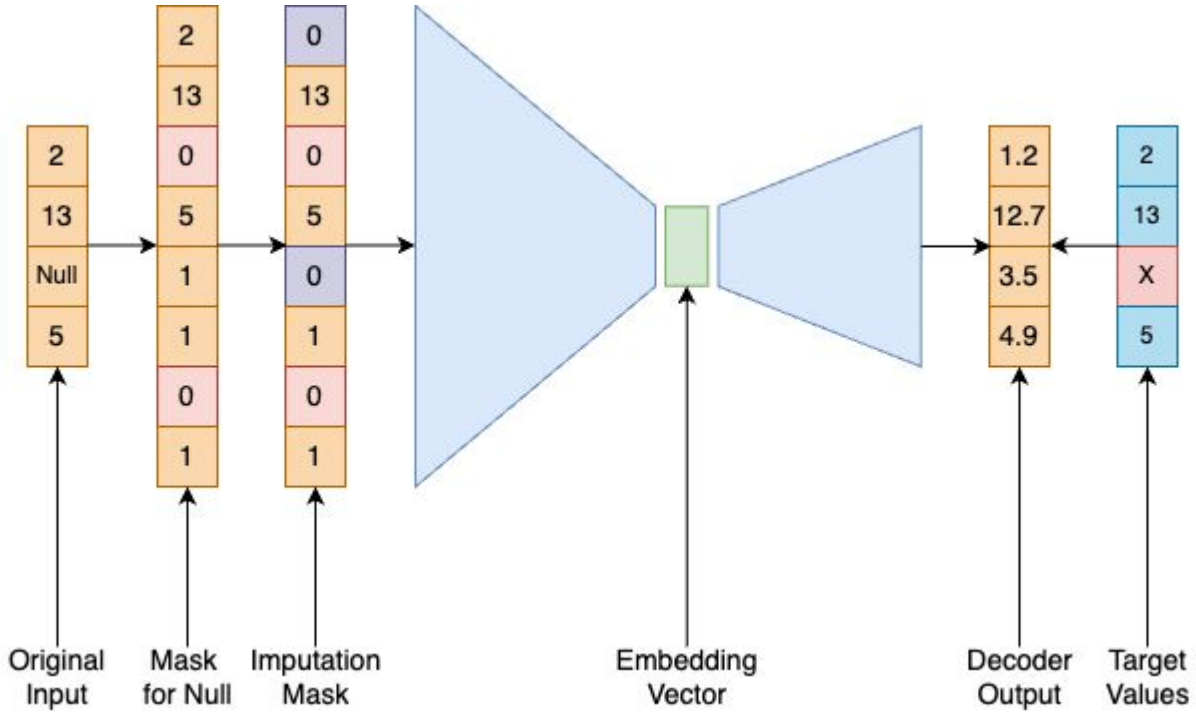
MIEO: How it Works

Produce the Output



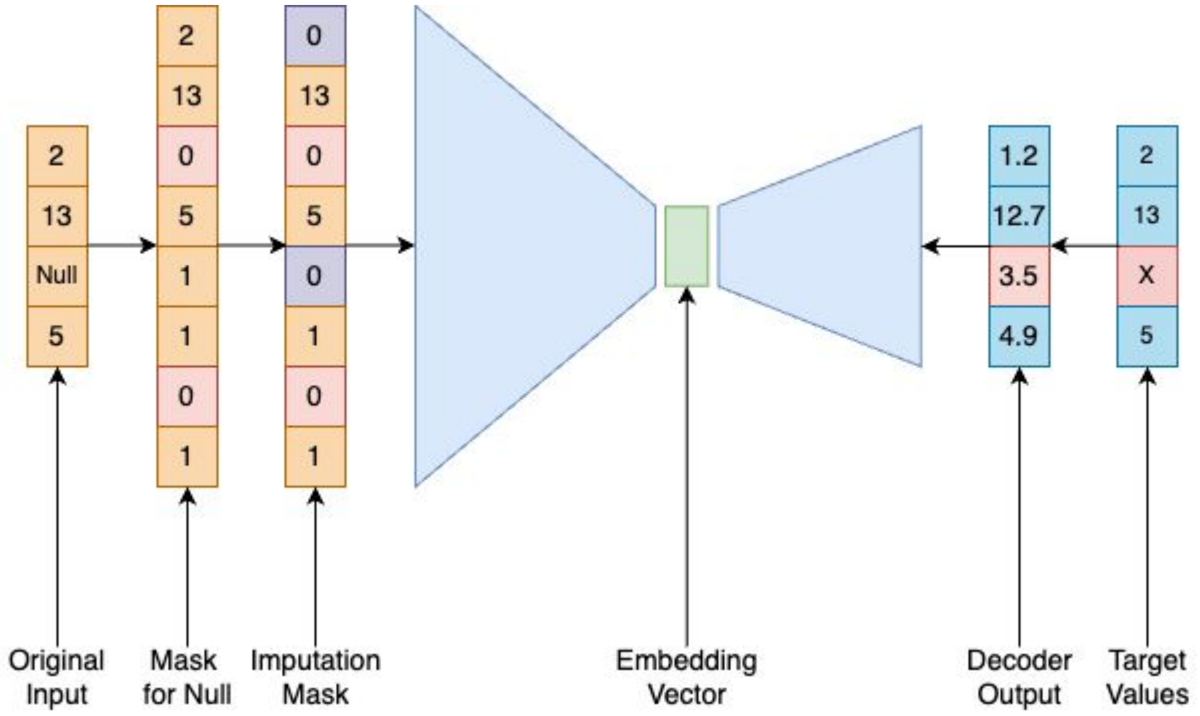
MIEO: How it Works

Target Values for Loss



MIEO: How it Works

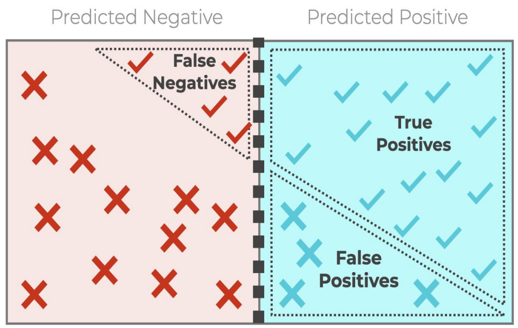
Compute Loss



How to Evaluate?

Classification Task

- **Classification task** using the target column.
- **Two models**: one applied to the original data and one to the encoded data.
- Tried few different models:
  - Random Forest
  - Feedforward Classifier
- Better results with **feedforward** using **macro F1 score.**



$$\text{F1 Score} = \frac{2 * (\text{True Positives})}{2 * (\text{True Positives}) + (\text{False Positives}) + (\text{False Negatives})}$$

How to Evaluate?

Feedforward Classifiers

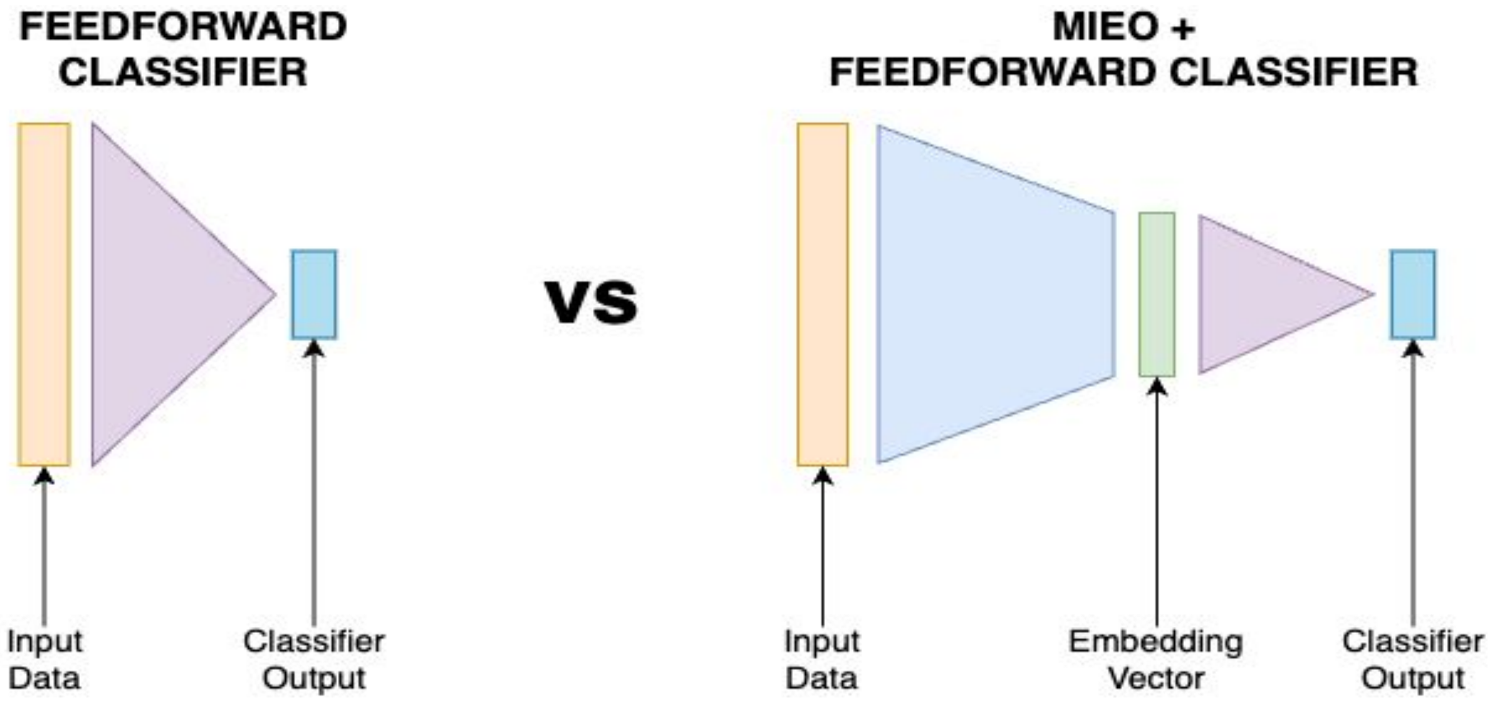


Table of Contents

3 - Experiments

Experiments

Overview

- Models had a large number of **hyperparameters**.
- Performed **grid search** to choose the best model.
- Used embedding percentages greater than 1 using the idea of [5].
    - To get a better imputation.

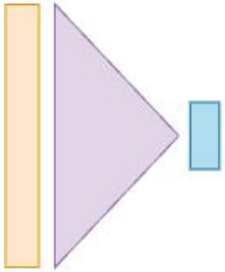
Experiments

Grid Search

CLASSIFIER PARAMETERS

Loss Weight	[(0.3, 0.7)]
Batch Size	[75]
Learning Rate	[5e-04, 3e-04, 1e-04]
Weight Decay	[0.03, 0.05, 0.06]
N. Epochs	[150, 200, 250]
Gamma	[0, 0.001]
Step Size	[75, 85, 100]

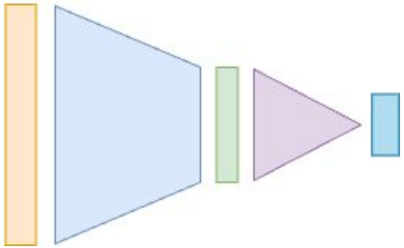
FEEDFORWARD CLASSIFIER



Experiments

Grid Search

MIEO +
FEEDFORWARD CLASSIFIER



MIEO PARAMETERS

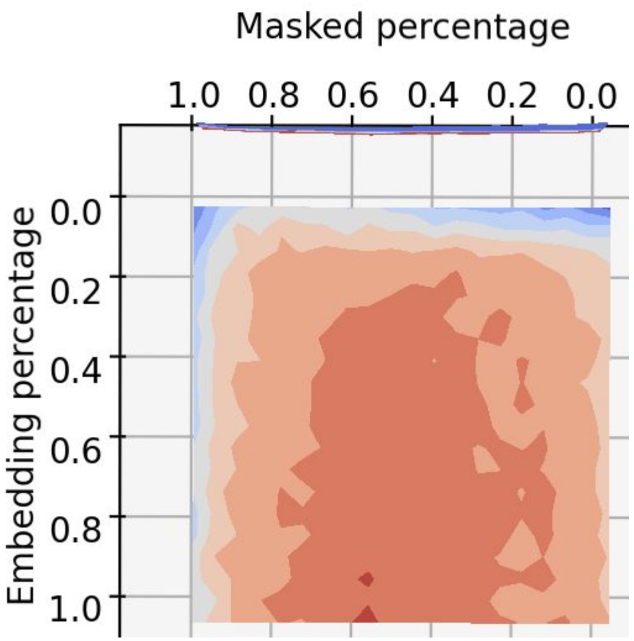
Embedding Percentage	[0.05 - 3] (every 0.05)
Masked Percentage	[0 - 0.95] (every 0.05)
Binary Loss Weight	[None, 0.5]
Batch Size	[200]
Learning Rate	[0.0015, 0,002]
Weight Decay	[5e-07, 2e-06]
N. Epochs	[250]
Patience	[10]

CLASSIFIER PARAMETERS

Loss Weight	[(0.25, 0.75), (0.3, 0.7), (0.5, 0.5)]
Batch Size	[200]
Learning Rate	[0.0001, 0.0002, 0.0004]
Weight Decay	[2e-06, 5e-06]
N. Epochs	[50]
Patience	[5]

Experiments

3D Visualization of Grid Search



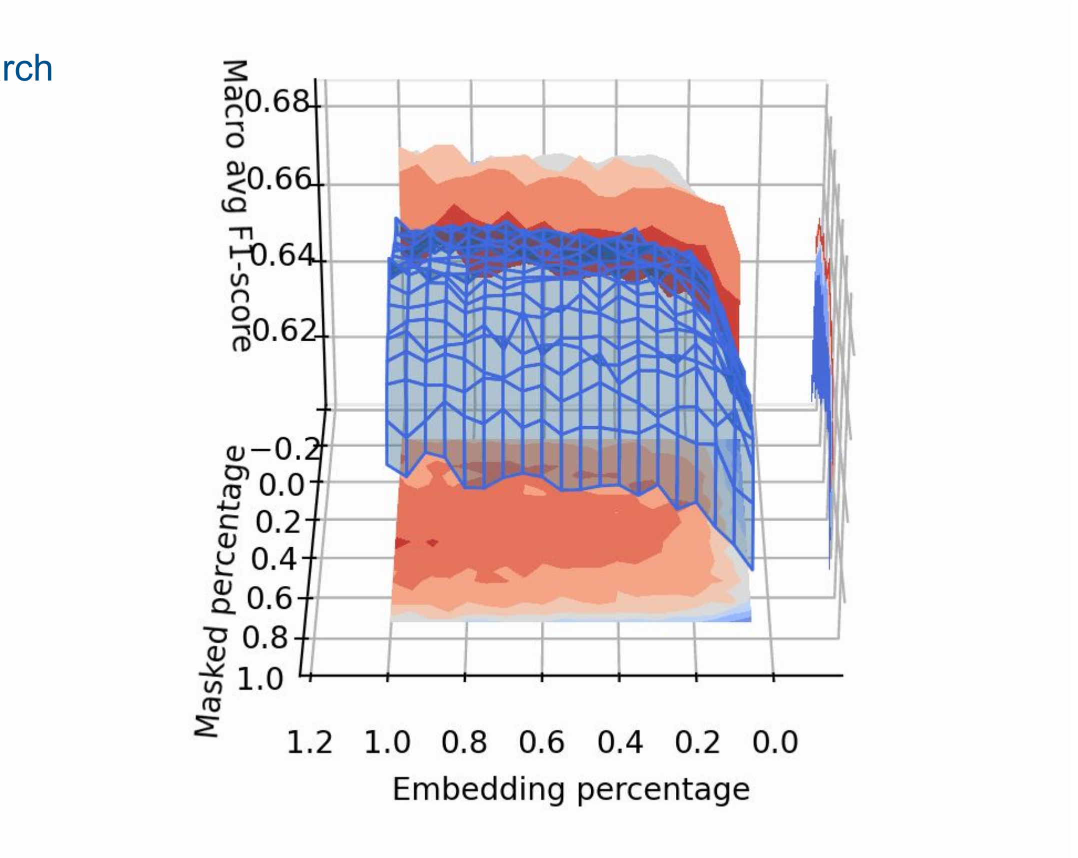


Table of Contents

Experiments

Classifier with Original Data Best Model

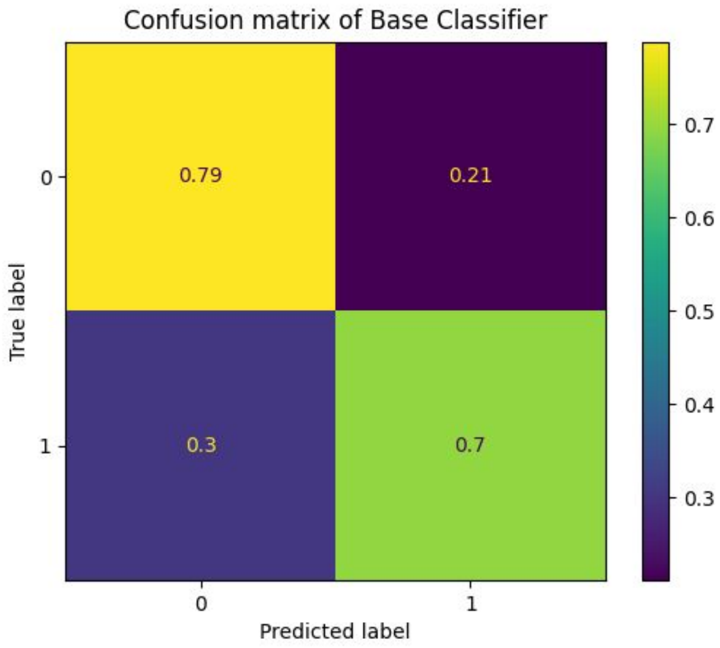
CLASSIFIER PARAMETERS

Binary Loss Weight	(0.3, 0.7)
Batch Size	75
Learning Rate	5e-04
Weight Decay	0.05
N. Epochs	150
Gamma	0.001
Step Size	75

- F1-score on validation: 0.75

Results

Classifier with Original Data on Test Set



	precision	recall	f1-score	support
0	0.89	0.79	0.83	1212
1	0.52	0.70	0.60	401
accuracy			0.77	1613
macro avg	0.70	0.74	**0.72**	1613
weighted avg	0.80	0.77	0.78	1613

Results

MIEO + Classifier best model

- F1-score on validation: 0.7

MIEO PARAMETERS

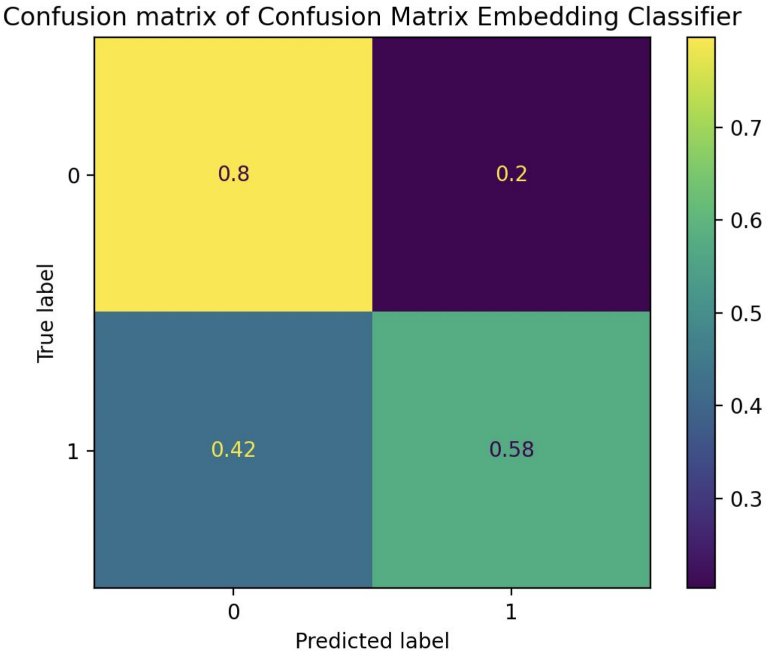
Embedding Percentage	0.35
Masked Percentage	0.35
Binary Loss Weight	None
Batch Size	200
Learning Rate	0,002
Weight Decay	5e-07
N. Epochs	250
Patience	10

CLASSIFIER PARAMETERS

Loss Weight	(0.3, 0.7)
Batch Size	200
Learning Rate	0.0001
Weight Decay	5e-06
N. Epochs	50
Patience	5

Results

MIEO + Classifier on Test Set



	precision	recall	f1-score	support
0	0.85	0.80	0.82	1212
1	0.49	0.58	0.53	401
accuracy			0.74	1613
macro avg	0.67	0.69	**0.68**	1613
weighted avg	0.76	0.74	0.75	1613

Table of Contents

5 - Conclusion

Conclusion

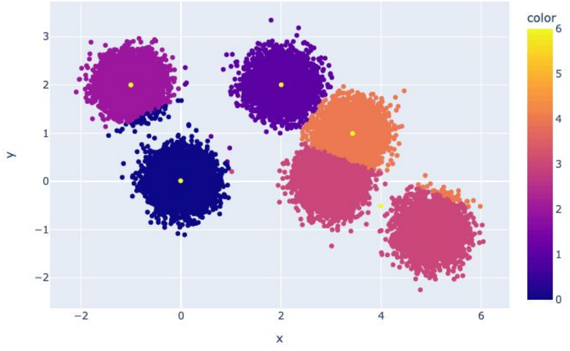
Obtained results

- Developed **MIEO** to deal with problems regarding encoding clinical data
- Delineated **limits and tradeoffs** to both compress and impute
- Found a model that **effectively compress** the informations to 35% of the original size while keeping a comparable macro f1 score

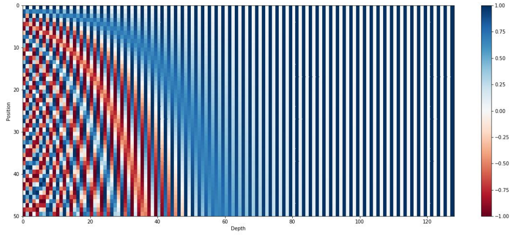
Conclusion

Future Developments

- Evaluate embeddings with **clustering** techniques.



- **Transformers** with clinical tests in time.



- New **dataset** opportunity!

References

[1] D. Borghini, D. Marchi, A. Nardone and G. Scerra. “Clinical-data encoding.” In *GitHub repository:* https://github.com/davide-marchi/clinical-data-encoding

[2] A. Pingitore, C. Zhang et al. “Machine Learning to identify a composite indicator to predict cardiac death in ischemic heart disease". In *International Journal of Cardiology, 404, 131981.* (2024)

[3] A. Vallée. "Digital twin for healthcare systems." In *Frontiers in Digital Health 5: 1253050.* (2023)

[4] T. Sun, H. Xiwang and L. Zhonghai. "Digital twin in healthcare: Recent updates and challenges." In *Digital Health 9: 20552076221149651.* (2023)

[5] L. Gondara, and K. Wang. "Mida: Multiple imputation using denoising autoencoders." In *Advances in Knowledge Discovery and Data Mining: 22nd Pacific-Asia Conference, PAKDD 2018, Melbourne, Springer International Publishing.* (2018)

Thank you for your attention!

UNIVERSITÀ DI PISA