ReMasker: Imputing Tabular Data with Masked Autoencoding

Tianyu Du Luca Melis Ting Wang

ICLR 2024

Background

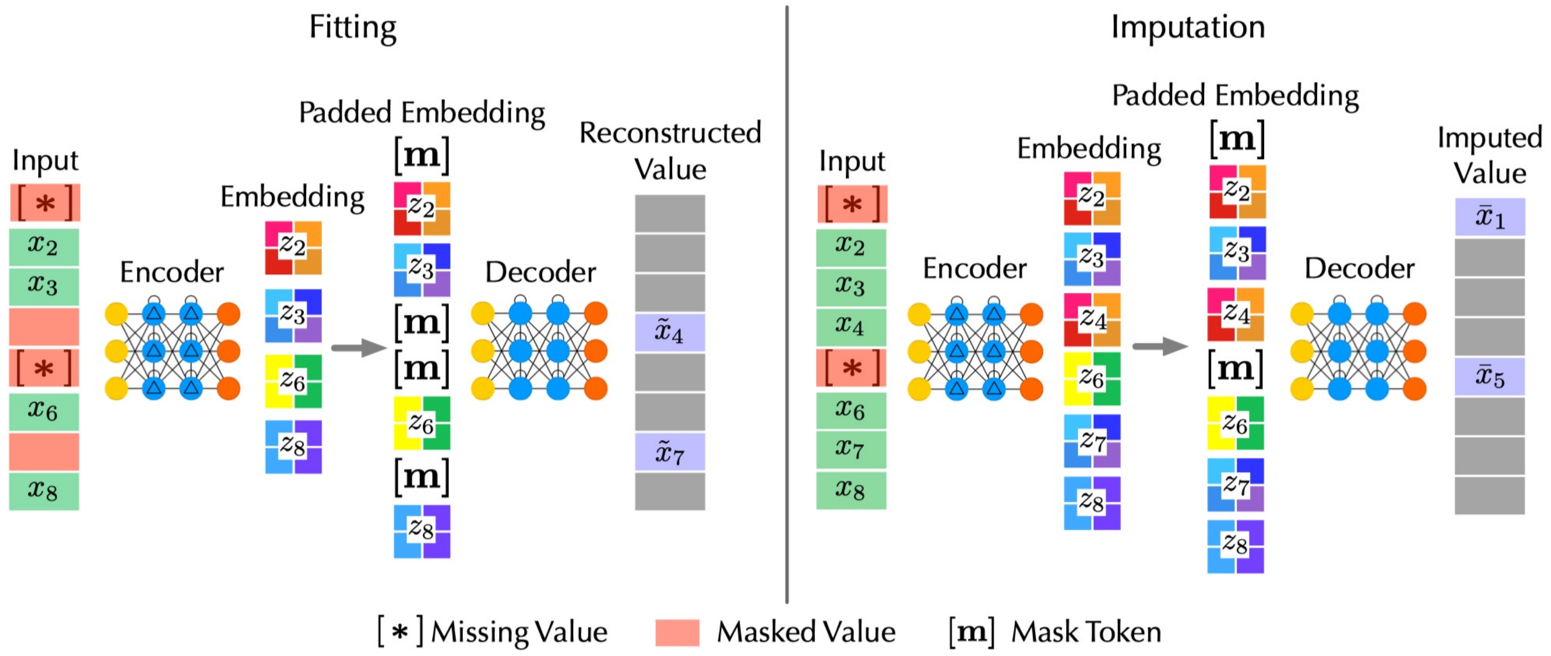
- Missing values are **ubiquitous** in **real-world** tabular data
- **Imputation**: estimate missing values based on observed data
- **Challenges**: imputing missing values in tabular data with **high fidelity and utility**
  - the **intricate correlation** across different features
  - the **variety** of missingness scenarios
  - the **scarce** amount of available data with respect to the number of missing values

Related Works

- **Tabular Data Imputation**
  - **discriminative** methods: MissForest, MICE, and MIRACLE, etc.
  - **generative** methods: GAIN, MIWAE, GAMIN, HI-VAE, etc.
- **Limitations**
  - GAN-based methods require a **large amount** of training data and suffer the difficulties of **adversarial training**
  - VAE-based methods often face the limitations of training through **variational bounds**
  - require **complete** data during **training,** operate on the **assumptions** of specific missingness patterns

To our best knowledge, this represents the first work to explore an extended MAE approach (with Transformer as the backbone) in the task of tabular data imputation.

Overall Framework



Experimental Setting

Datasets

- **12** datasets from UCI ML repo

Missing mechanisms

- **MCAR**
- **MAR**
- MNAR

Baselines

- **13** methods: HyperImpute, MIWAE, EM, GAIN, SoftImpute, MissForest, ICE, MICE, MIRACLE, Mean, Median, Frequent, Sinkhorn

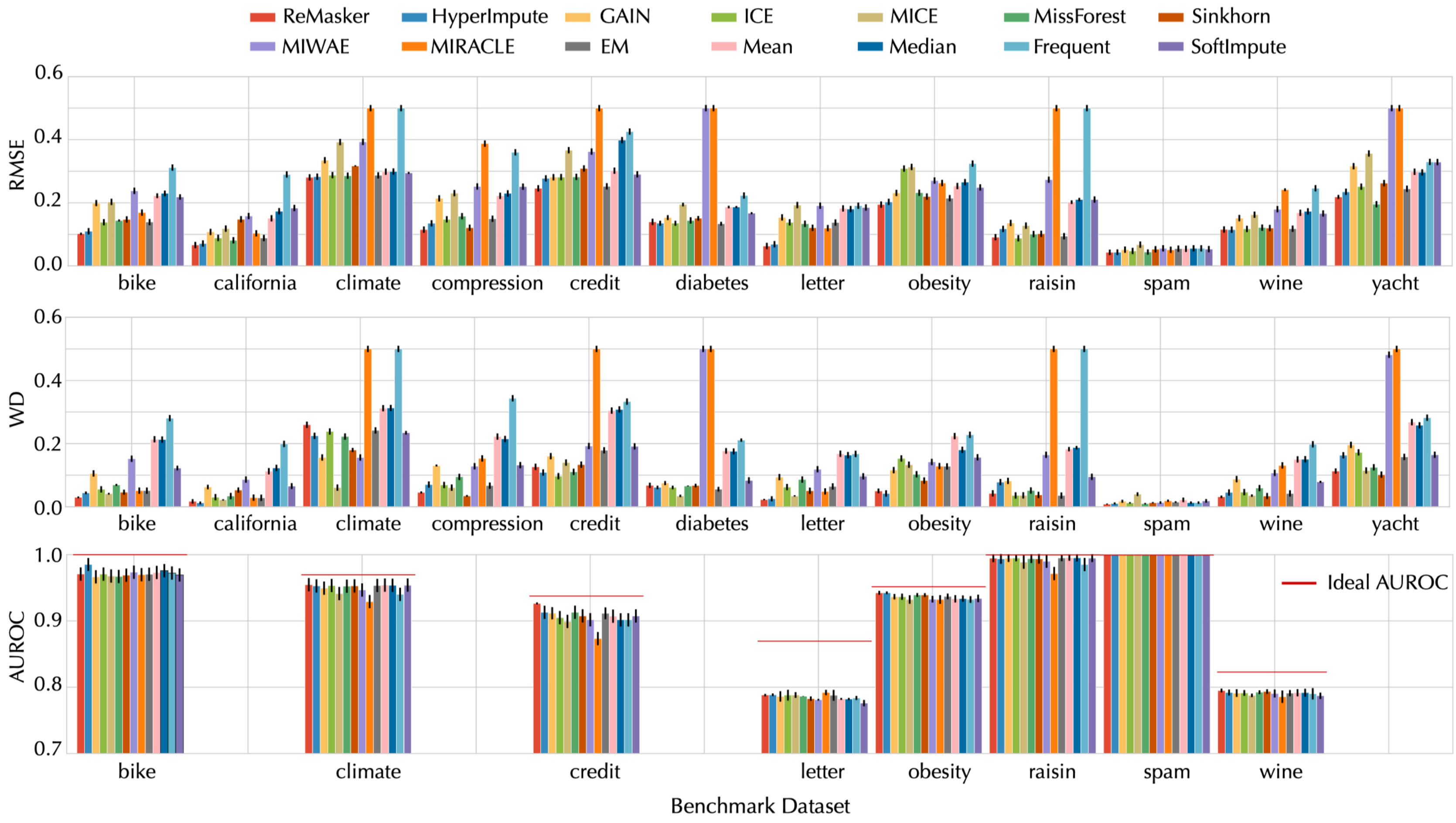
Metrics

- Fidelity: RMSE, WD
- Utility: AUROC

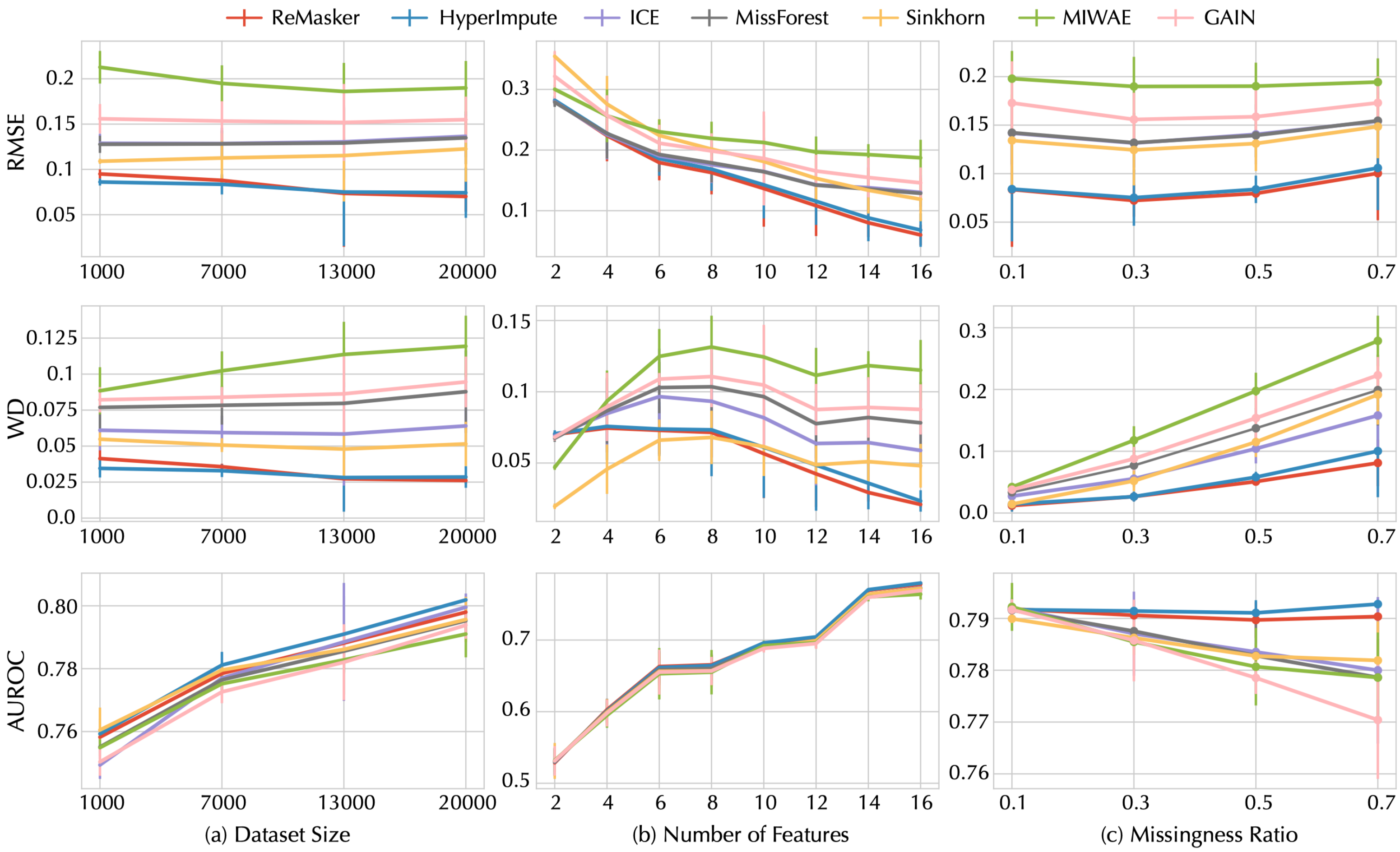
Key Questions

- Does ReMakser work?
- How does it work?
- What is the best way of using ReMakser?

Overall Performance



Sensitivity Analysis



Ablation Study

- **Model Design**
  - encoder depth, embedding width, decoder depth, backbone
  - model complexity needs to fit the given dataset

(a) Decoder depth

depth	RMSE	WD	AUROC
2	0.0729	0.0263	0.7898
4	0.0636	0.0228	0.7903
6	0.0616	0.0219	0.7909
8	0.0611	0.0217	0.7892
10	0.0673	0.0245	0.7879

(b) Embedding width

width	RMSE	WD	AUROC
16	0.0902	0.0379	0.7902
32	0.0714	0.0289	0.7885
64	0.0616	0.0219	0.7909
128	0.0795	0.0305	0.7845
256	0.1040	0.0403	0.7868

(c) Encoder depth

depth	RMSE	WD	AUROC
2	0.0637	0.0239	0.7887
4	0.0625	0.0236	0.7877
6	0.0644	0.0239	0.7889
8	0.0616	0.0219	0.7909
10	0.0637	0.0227	0.7878

Table 1. Ablation study of REMASKER on the `letter` dataset. The default setting is as follows: encoder depth = 8, decoder depth = 6, embedding width = 64, masking ratio = 50%, and training epochs = 600.

backbone	letter			california	
	RMSE	WD	AUROC	RMSE	WD
Transformer	0.0611	0.0217	0.7892	0.0663	0.0172
Linear	0.1732	0.1604	0.7821	0.1786	0.1329
Convolutional	0.1694	0.1582	0.7836	0.1715	0.1286

Table 2. Performance with different backbones. (note: AUROC is inapplicable to the `california` dataset)

- **Reconstruction loss**
  - using the reconstruction of unmasked values only is insufficient

loss	letter			california	
	RMSE	WD	AUROC	RMSE	WD
$\mathcal{I}_{\text{mask+}} \cup \mathcal{I}_{\text{unmask}}$	0.0616	0.0219	0.7909	0.0663	0.0172
$\mathcal{I}_{\text{mask+}}$	0.0629	0.0237	0.7890	0.0840	0.0311
$\mathcal{I}_{\text{unmask}}$	0.2079	0.1129	0.7901	0.1932	0.1906

Table 3. Performance of REMASKER with reconstruction loss w/ or w/o unmasked values.

Practice of ReMasker

Training Regime

- **Terminate early** (e.g., 600 epochs) for efficient training
- The quickly converging loss demonstrates the **trainability** of ReMakser

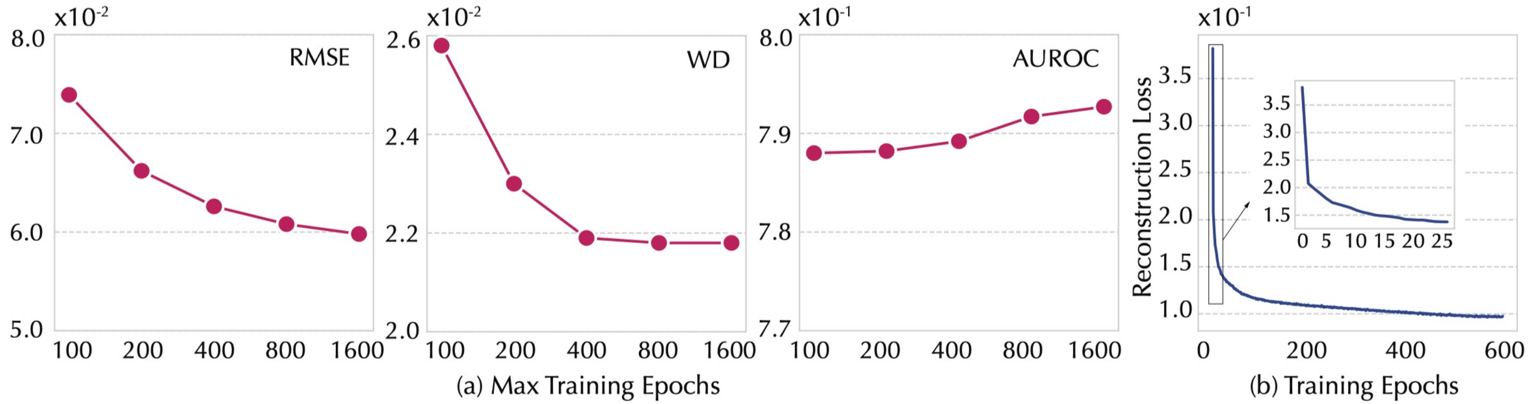


Figure 4: (a) REMASKER performance with respect to the maximum number of training epochs; (b) Convergence of REMASKER's reconstruction loss. Performed on `letter` under MAR with 0.3 missingness ratio.

Practice of ReMasker

- **Masking Ratio**
  - a **larger** number of features affords a **higher** masking ratio

masking ratio	letter			california	
	RMSE	WD	AUROC	RMSE	WD
0.1	0.0668	0.0215	0.0789	0.0888	0.0230
0.3	0.0562	0.0207	0.7897	0.0654	0.0151
0.5	0.0554	0.0212	0.7935	0.0663	0.0172
0.7	0.0906	0.0366	0.7878	0.1320	0.0650

Table 4. Performance with varying masking ratio. The results are evaluated on `letter` and `california` under MAR with 0.3 missingness ratio.

- **Standalone vs. Ensemble**
  - use ReMakser as the **base imputer** of HyperImpute **improves** the imputation performance

base imputer	letter			california	
	RMSE	WD	AUROC	RMSE	WD
default	0.0564	0.0215	0.7899	0.0722	0.0134
REMASKER	0.0554	0.0212	0.7935	0.0702	0.0115

Table 5. REMASKER as the base imputer within HyperImpute. The results are evaluated on `letter` and `california` under 0.3 MAR.

Discussion

- **Q1: What is ReMasker learning?**
  - missingness-invariant representations of input data
- **Q2: How is ReMasker's performance influenced by the missingness mechanism?**
  - better performance under MAR and MCAR compared with MNAR
- **Q3: Why is Transformer effective for tabular data imputation?**
  - multi-head self-attention (MSA) mechanism
- **Q4: What are ReMasker's limitations?**
  - biased towards re-constructing individual missing values
  - may be suboptimal when downstream tasks are unknown

Conclusion

- ✓ A pilot study exploring the masked autoencoding approach for tabular data imputation
- ✓ Developing and evaluating ReMasker, a novel imputation method for tabular data
- ✓ Reveal that masked tabular modeling represents a promising direction for future research

Q&A

zjradty@zju.edu.cn