MULTIVARIATE TIME-SERIES SYNTHETIC DATA GENERATION

Alessandro Minutolo – Emanuele Ruoppolo

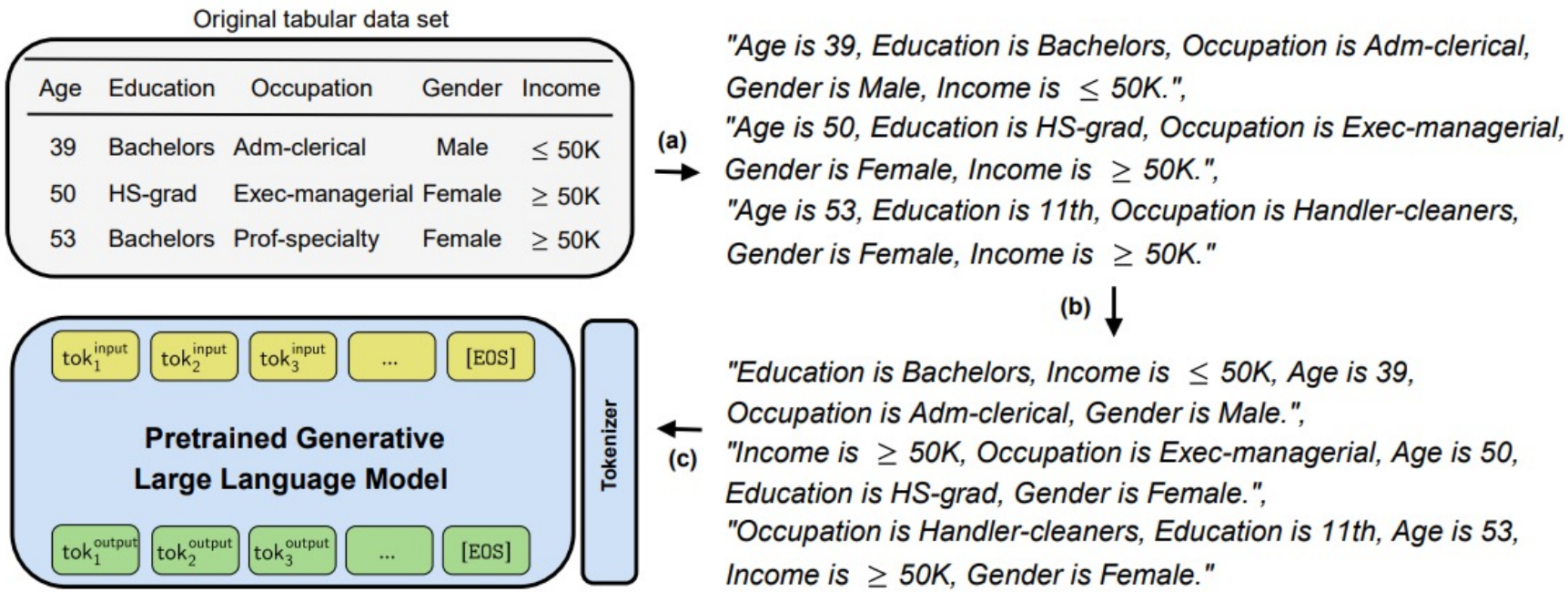
Probabilistic Machine Learning and Deep Learning

2023 – 2024

GREAT FRAMEWORK FOR SYNTHESIS

1. Converting tabular data to string format $'f_1 \text{ is } v_1, \ldots, f_n \text{ is } v_n'$
2. Shuffling strings ordering to lose positional patterns
3. Fine-tuning pretrained LLM
4. Generating strings with optional preconditioning (prompts)
5. Re-converting the generated strings in tabular data

GREAT PIPELINE FOR FINE-TUNING



GENERATING MODEL PIPELINE

- Input: a text T encoded into a sequence **t** of tokens $w \in W$
- Output: a text **S** of strings **s** decoded from a text **T'** of tokens **t'**
- The generation is done by maximizing the probability $\prod_{t \in T} p(t)$

$$p(\boldsymbol{t}) = p(w_1, \ldots, w_j) = \prod_{k=1}^{j} p(w_k | w_1, \ldots, w_{k-1})$$

SAMPLING

- **Output distribution**: $z = q(w_1, \ldots, w_{k-1})$
- Each next token **w** is sampled by weighted choice sampling with a temperature parameter $T > 0$:

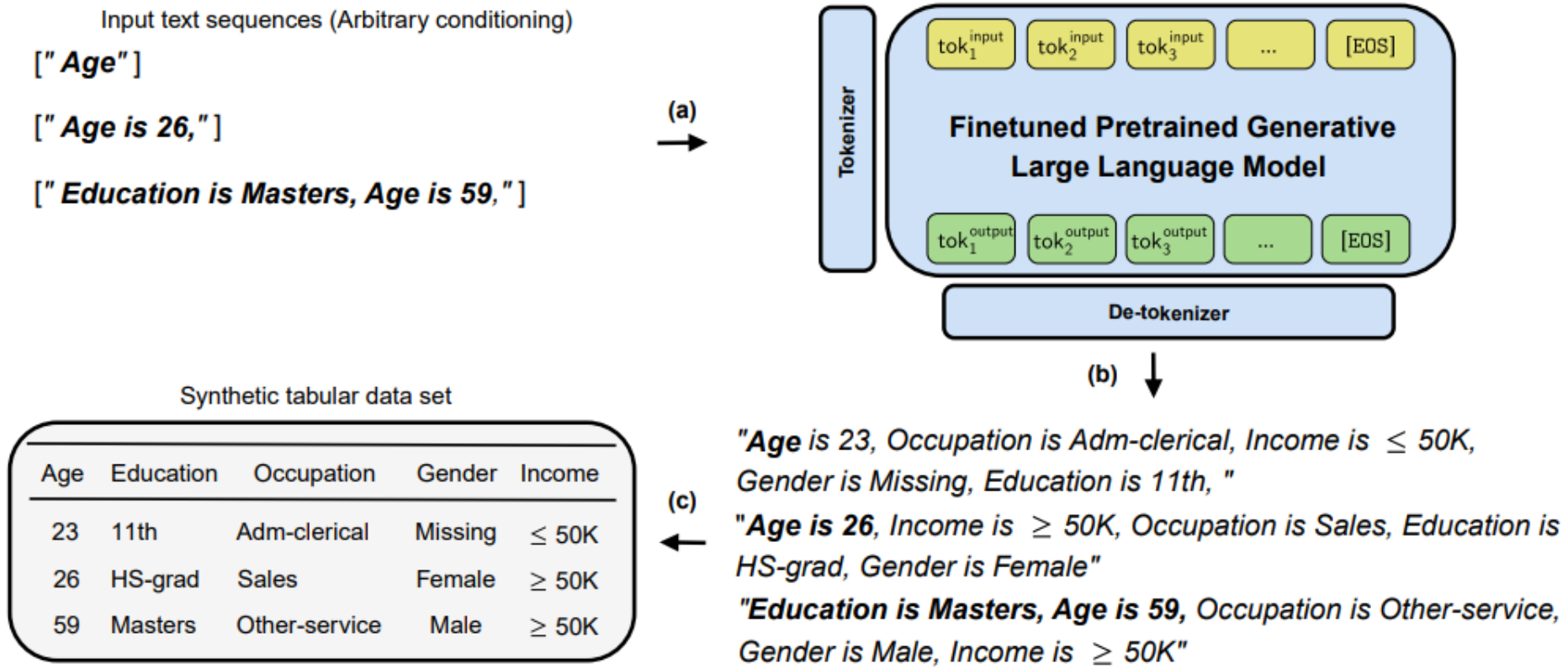
$$p(w_k | w_1, \ldots, w_{k-1}) = \frac{\exp\left(\frac{z_{w_k}}{T}\right)}{\sum_{w' \in W} \exp\left(\frac{z_{w'}}{T}\right)}$$

SAMPLING

- Prompts are given to the model to sample with preconditioning:
  - Multiple name-value pairs $\mathrm{V_{i_1}} = \mathrm{v_{i_1}}, \ldots, \mathrm{V_{i_k}} = \mathrm{v_{i_k}}$
  - Sampling is made from the distribution of the remaining features:

$$p\left(V_{\setminus\{i_1,\ldots,i_k\}} \middle| V_{i_1} = v_{i_1}, \ldots, V_{i_k} = v_{i_k}\right)$$

GREAT PIPELINE FOR SAMPLING



Age	Education	Occupation	Gender	Income
23	11th	Adm-clerical	Missing	≤ 50K
26	HS-grad	Sales	Female	≥ 50K
59	Masters	Other-service	Male	≥ 50K

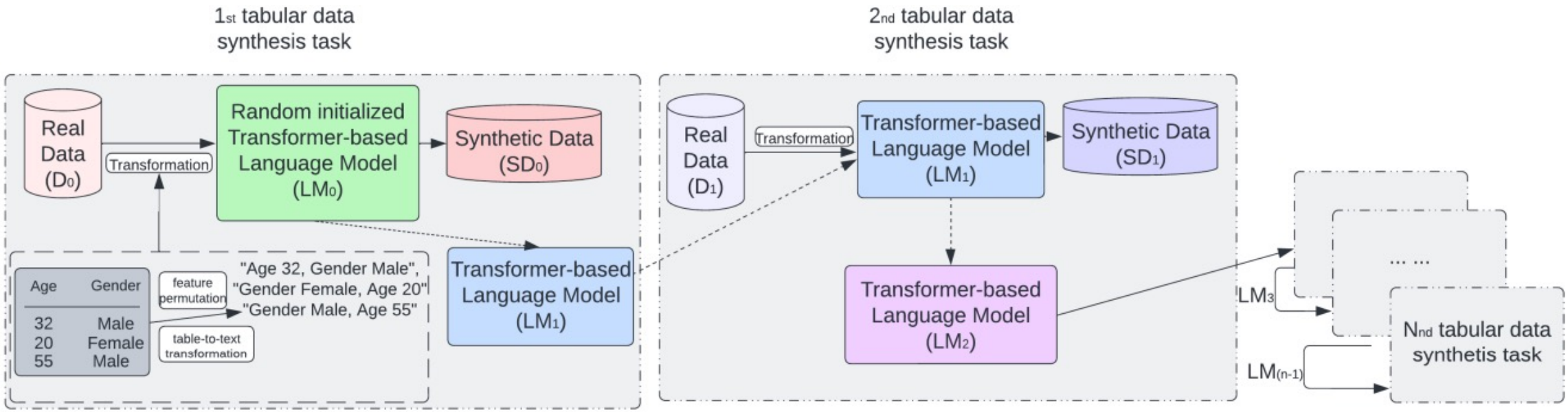
GREAT

- String: $'f_1$ is $v_1, \ldots, f_n$ is v_n'
- Expensive tokenization
- Pretrained LLM

TABULA

- String: $'f_1\ v_1, \ldots, f_n\ v_n'$
- Cheaper tokenization
- LLM from scratch

TABULA WORKFLOW



TABULA ON MULTIVARIATE TIME-SERIES DATA

- Main quests:
  - Understanding time dependancy between sequences of observations
  - Avoiding overfitting due to the timestamp

	date	demand	RRP	min_temperature	max_temperature	solar_exposure	school_day
0	2015-01-01	99635.03	25.63	13.3	26.9	23.6	N
1	2015-01-02	129606.01	33.14	15.4	38.8	26.8	N
2	2015-01-03	142300.54	34.56	20.0	38.2	26.5	N
3	2015-01-04	104330.72	25.01	16.3	21.4	25.2	N
4	2015-01-05	118132.20	26.72	15.0	22.0	30.7	N

STRATEGY: MODIFY TABULA FRAMEWORK

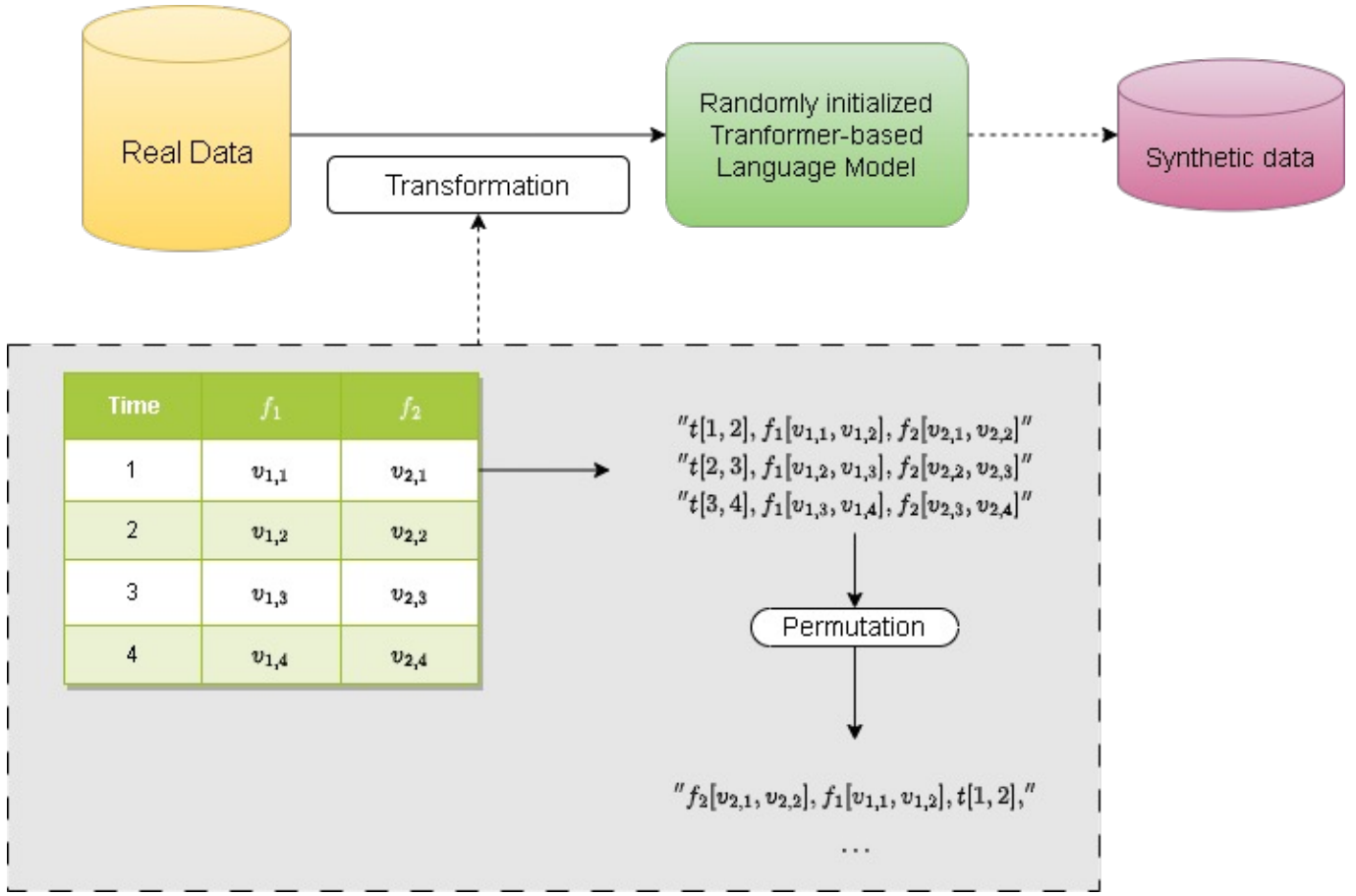
- Data preprocessing:
  - *Breaking timestamp into separate features*
  - *Introducing a temporal window to include values of more rows in a single string*

$$v = [v_{i+1}, v_{i+2}, \dots, v_{i+k}],\ k = \text{lenght of the window}$$

- Windows are selected through a stride (or *offset*)
- New enconded strings:

$$`f_1\ [v_{1,1}\ v_{1,2}\ \dots\ v_{1,k}]\ , \dots, f_n\ [v_{n,1}\ \dots\ v_{n,k}]'$$

PIPELINE


Real Data
Transformation
Randomly initialized
Tranformer-based
Language Model
Synthetic data
Time
f_1
f_2
1
2
3
4
$v_{1,1}$
$v_{2,1}$
$v_{1,2}$
$v_{2,2}$
$v_{1,3}$
$v_{2,3}$
$v_{1,4}$
$v_{2,4}$
$"t[1,2], f_1[v_{1,1}, v_{1,2}], f_2[v_{2,1}, v_{2,2}]"$
$"t[2,3], f_1[v_{1,2}, v_{1,3}], f_2[v_{2,2}, v_{2,3}]"$
$"t[3,4], f_1[v_{1,3}, v_{1,4}], f_2[v_{2,3}, v_{2,4}]"$
Permutation
$"f_2[v_{2,1}, v_{2,2}], f_1[v_{1,1}, v_{1,2}], t[1,2],"$
...

EXPERIMENTAL SET-UP

- Three datasets from Kaggle:
  - Energy consumption
  - London weather
  - Web visits
- Machine: GPU node V100 PCIe 32GiB on ORFEO cluster
- Modules: Tabula and Tabula modified with window/stride
- Comparison within the two frameworks

EXPERIMENTAL SET-UP

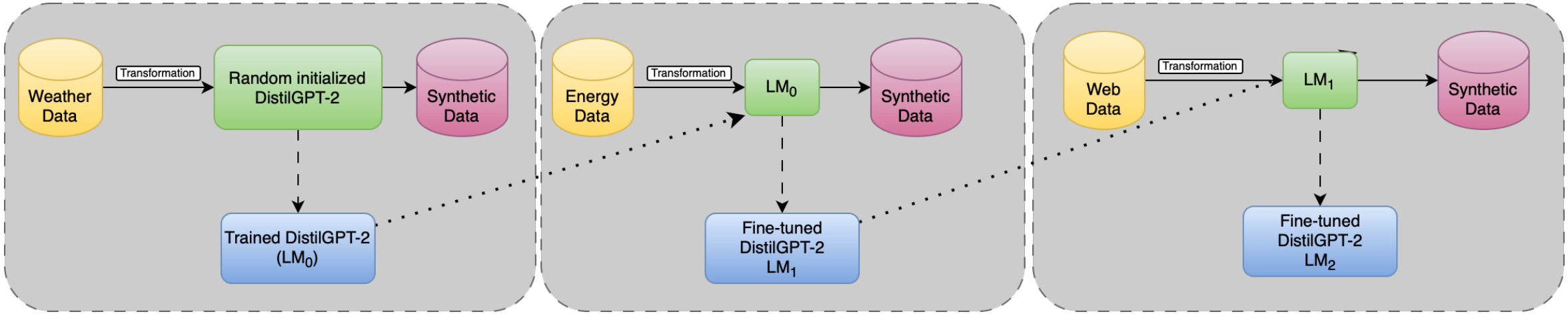
MODEL random initialized DistilGPT-2

ACTIVATION GeLU

LOSS cross-entropy

OPTIMIZER AdamW $[\mathrm{lr} = 5 \cdot 10^{-5}, \beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 1 \cdot 10^{-8}, \ \ \mathrm{decay} = 0]$

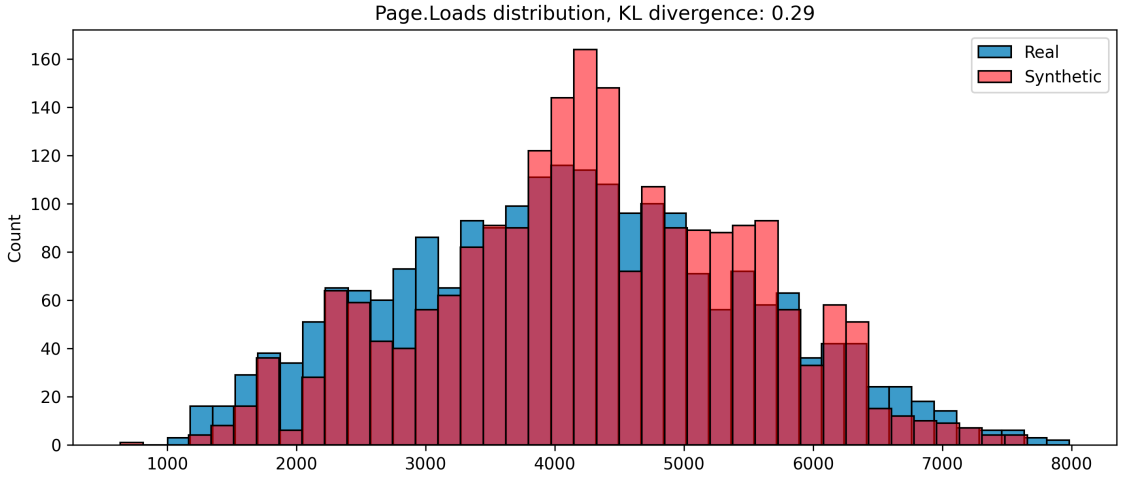
PIPELINE

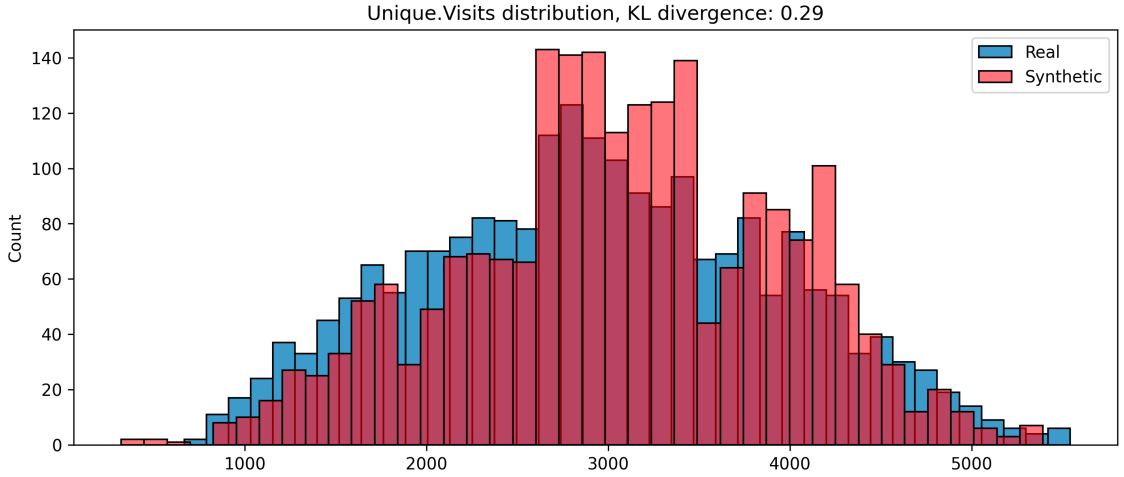

Weather Data
Transformation
Random initialized DistilGPT-2
Synthetic Data
Trained DistilGPT-2 (LM_0)
Energy Data
Transformation
LM_0
Synthetic Data
Fine-tuned DistilGPT-2 LM_1
Web Data
Transformation
LM_1
Synthetic Data
Fine-tuned DistilGPT-2 LM_2

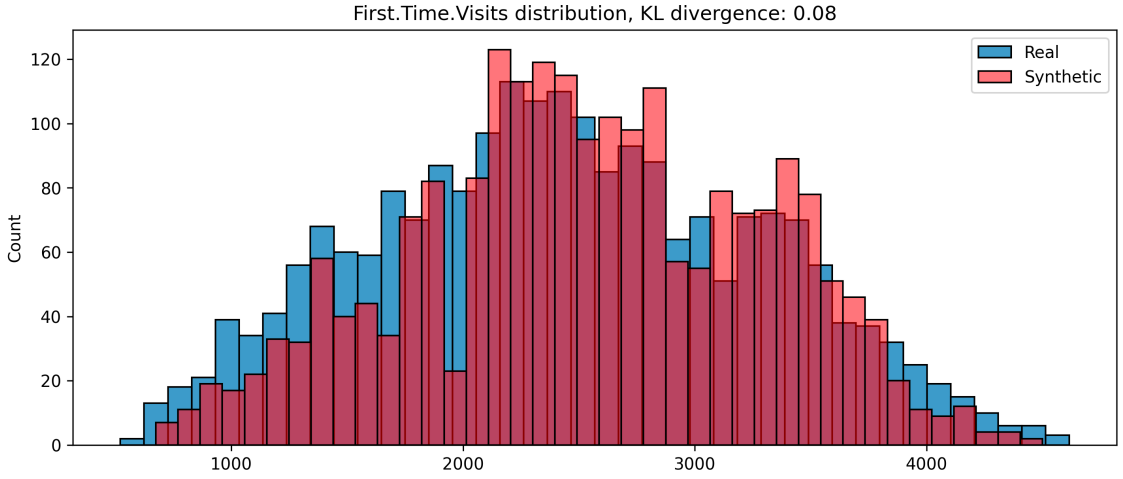
EVALUATION

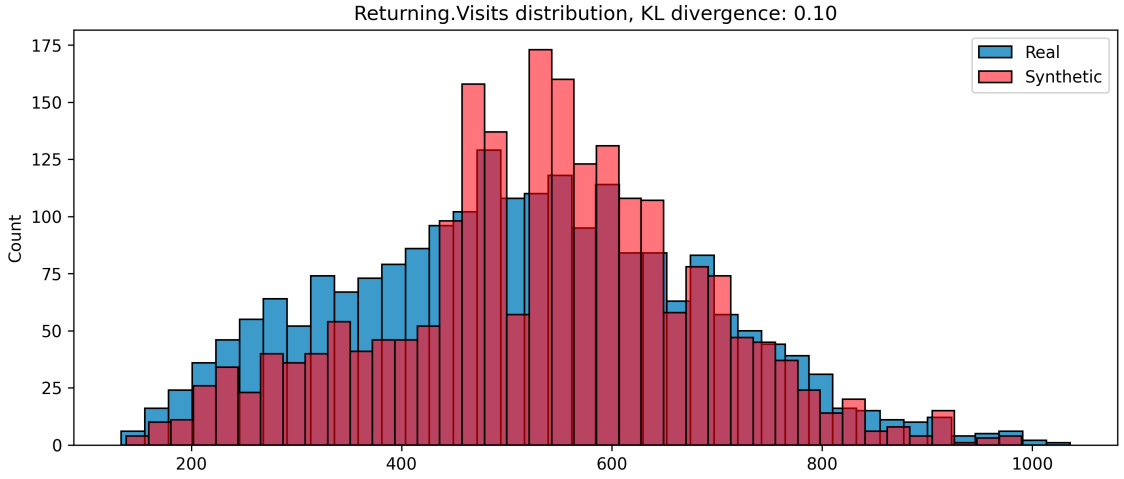
- Exploratory data analysis
  - *KL divergence on marginal distributions*
- Discriminator measure
- Distance to the closest record (DCR)
- Machine learning efficiency (MLE)

ANALYZING "DAILY WEB VISITS" DATASET RESULTS

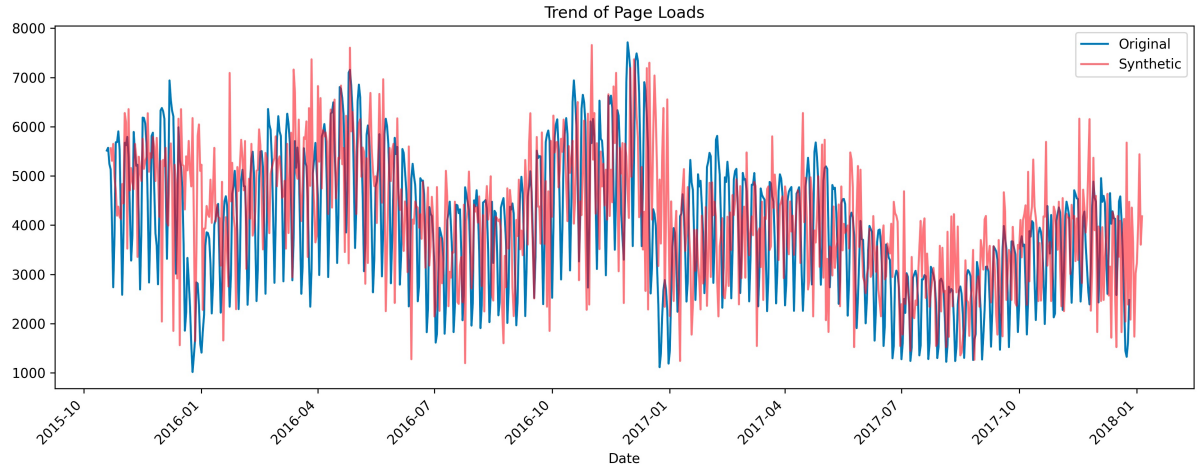


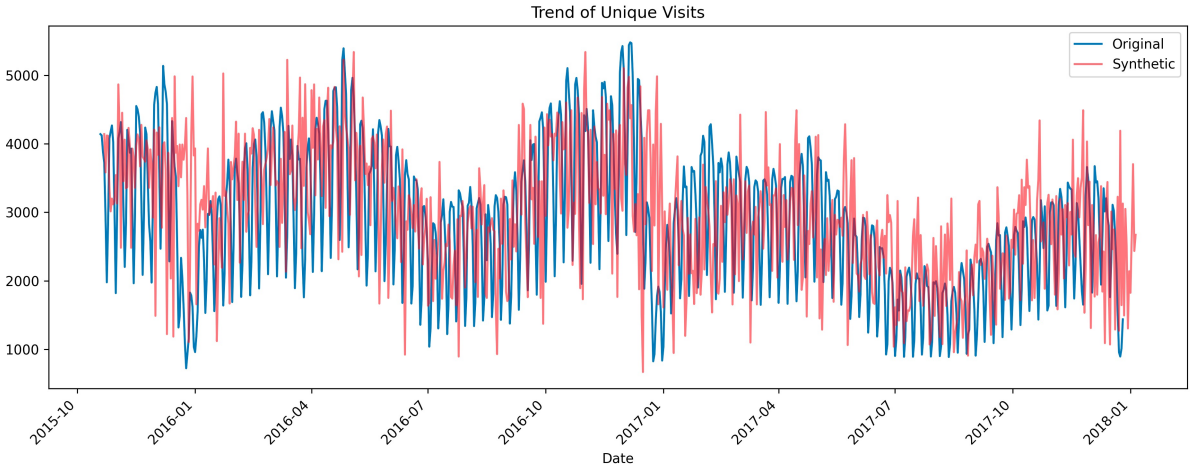


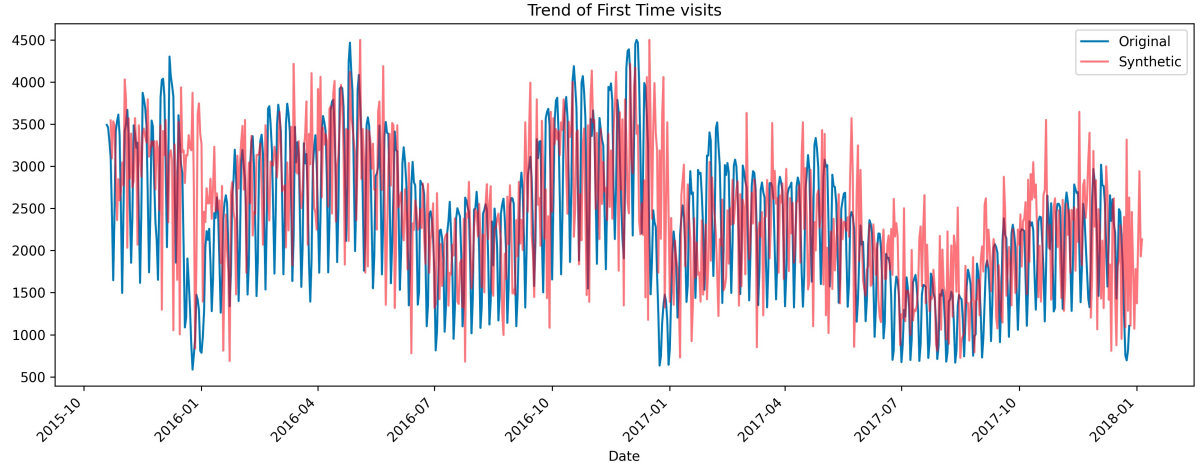


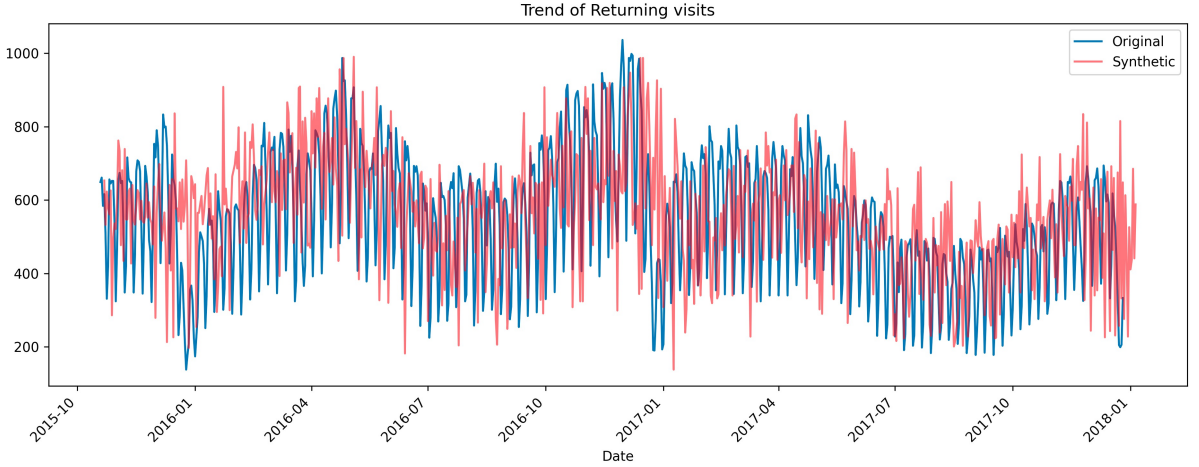


TRENDS DURING YEARS



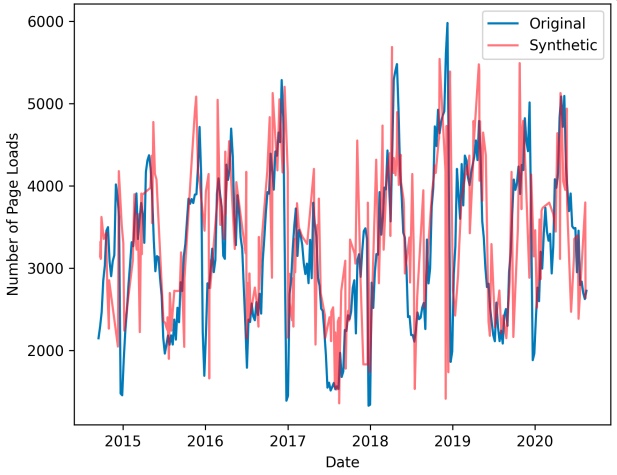


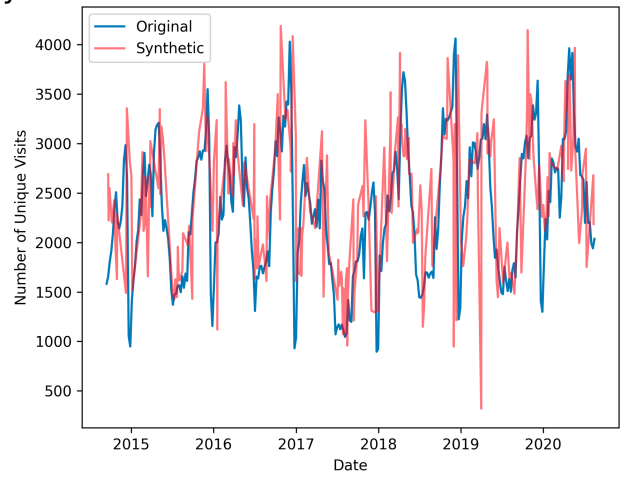


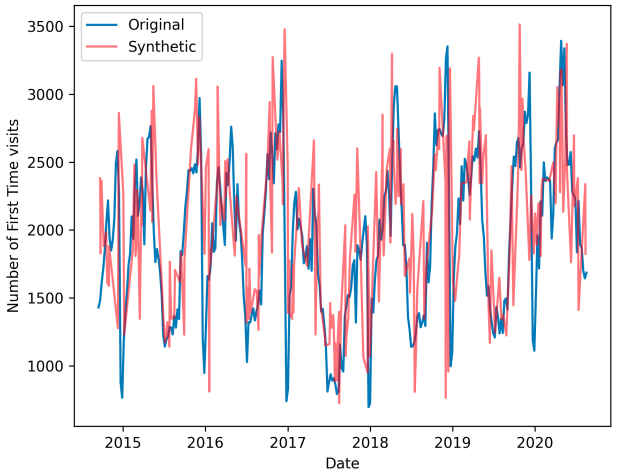


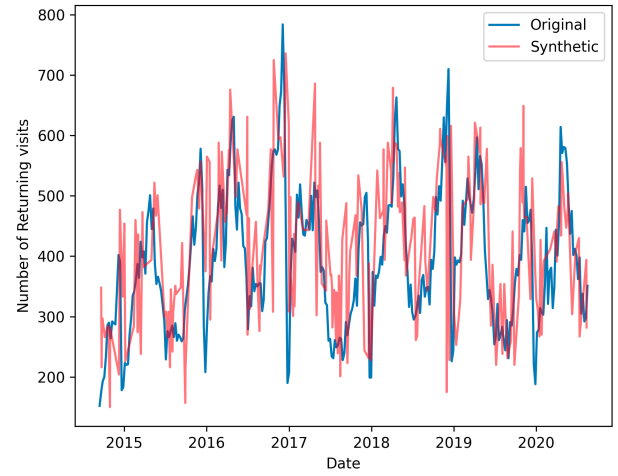
DAILY TRENDS

Monday Trend

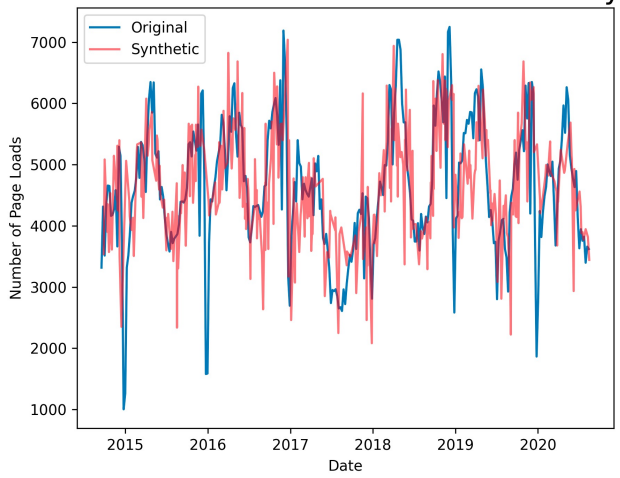


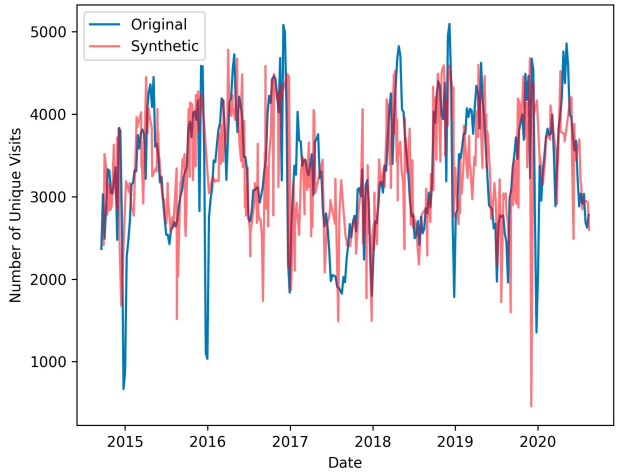


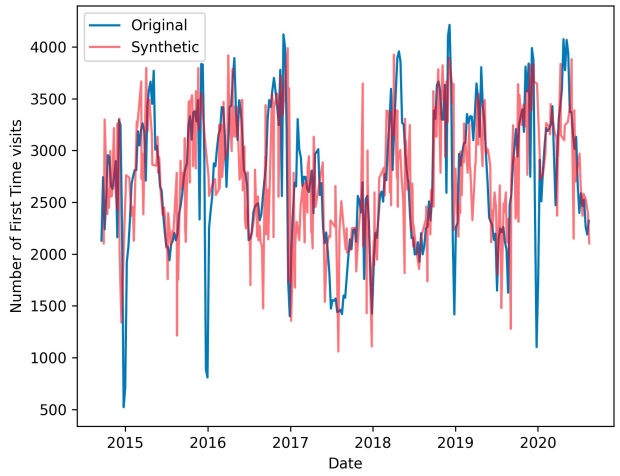


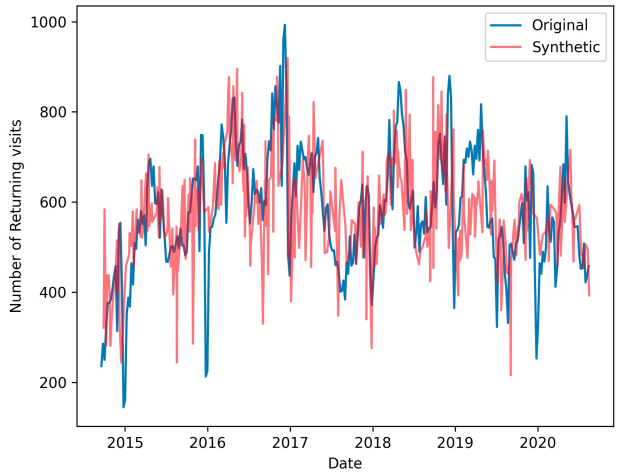


Friday Trend

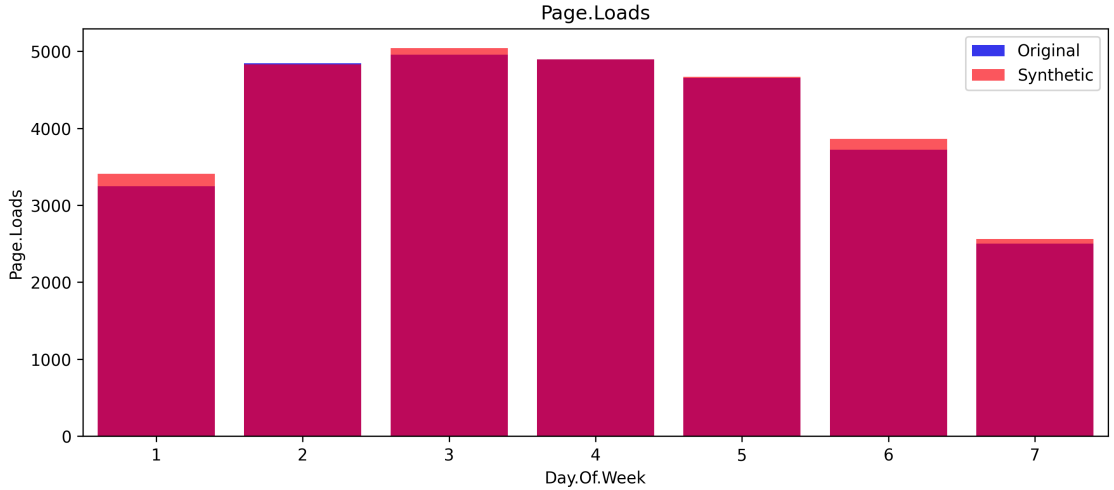


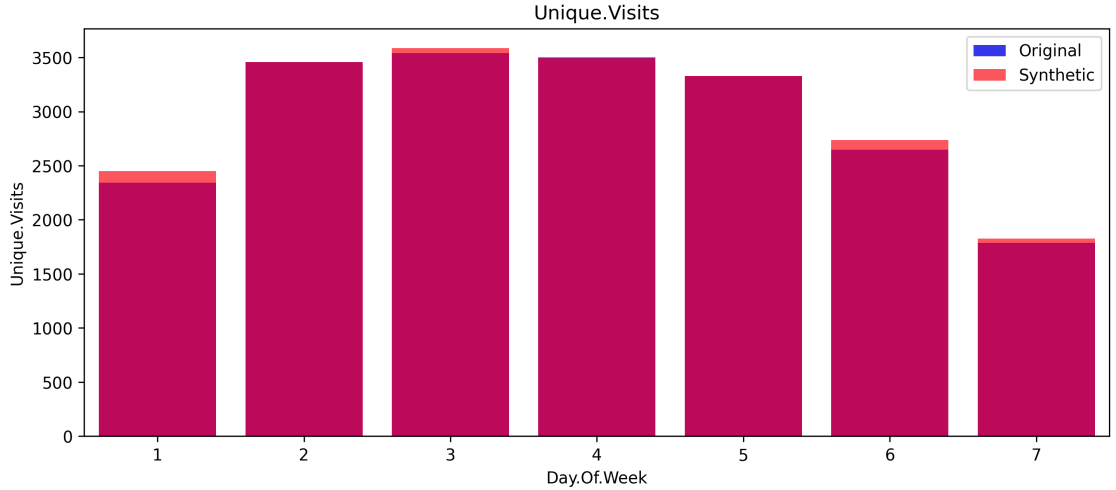


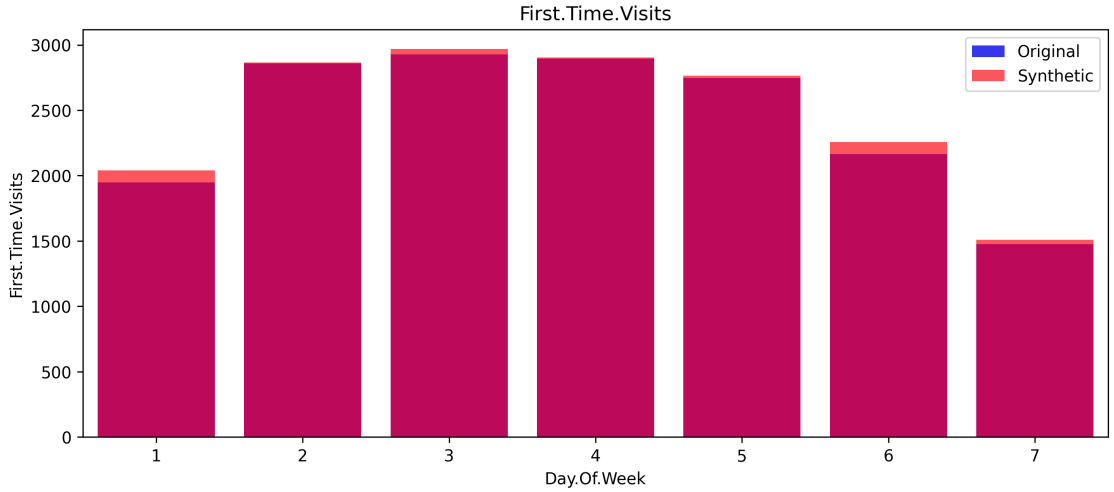


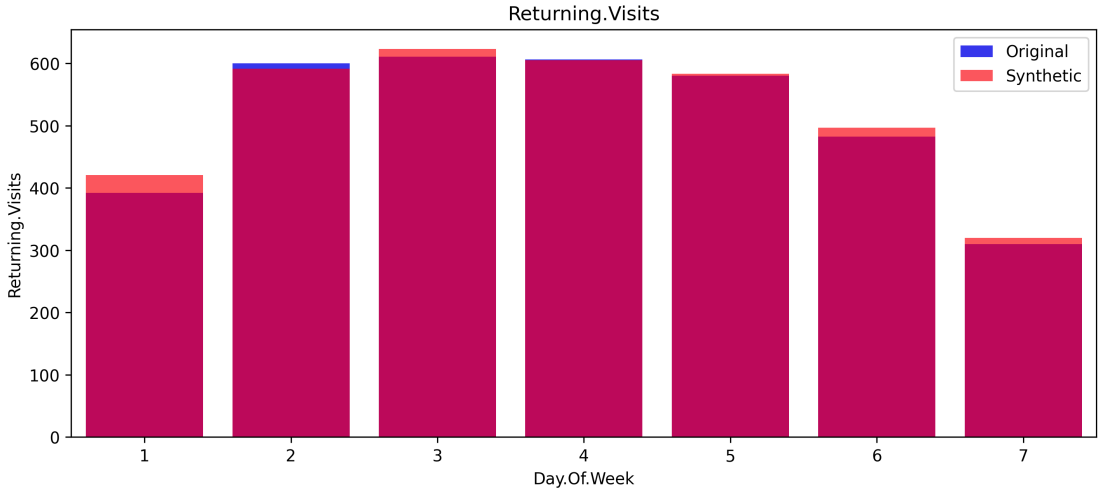


DAILY AVERAGE VALUES

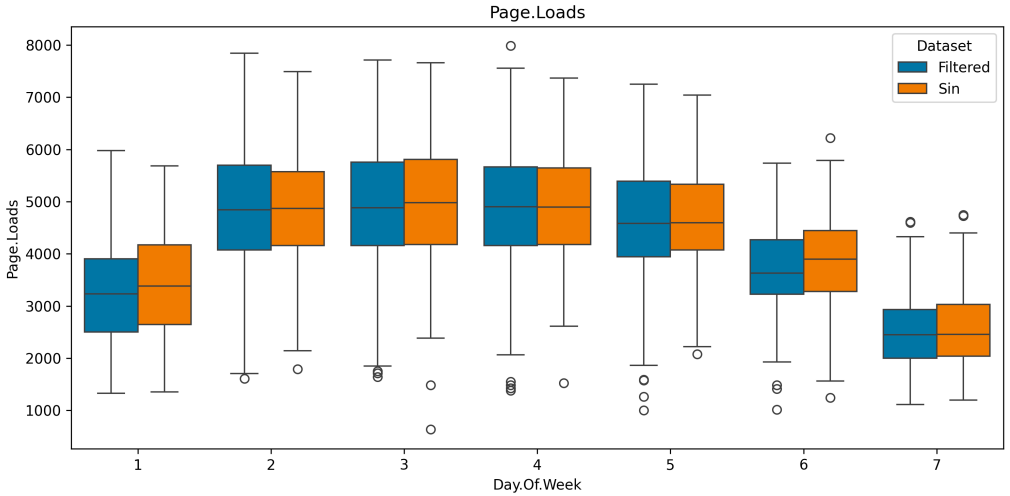


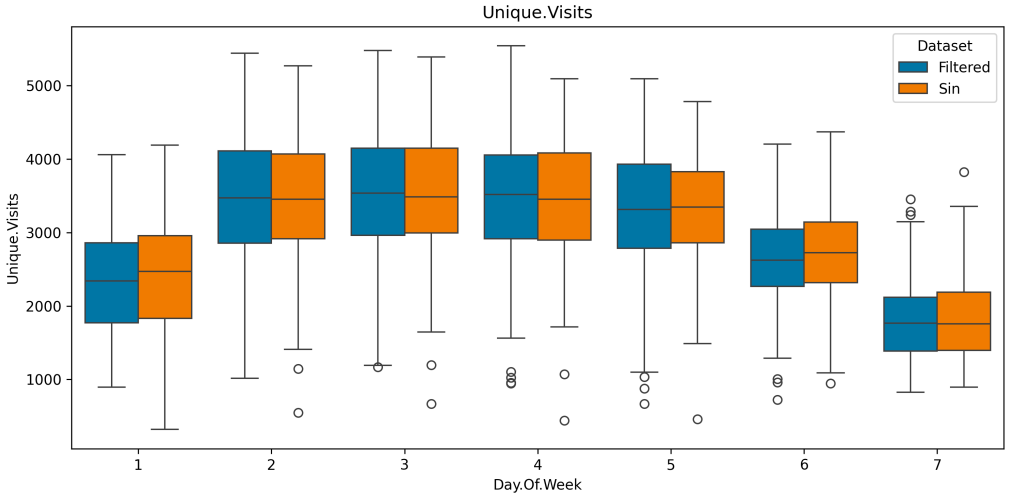


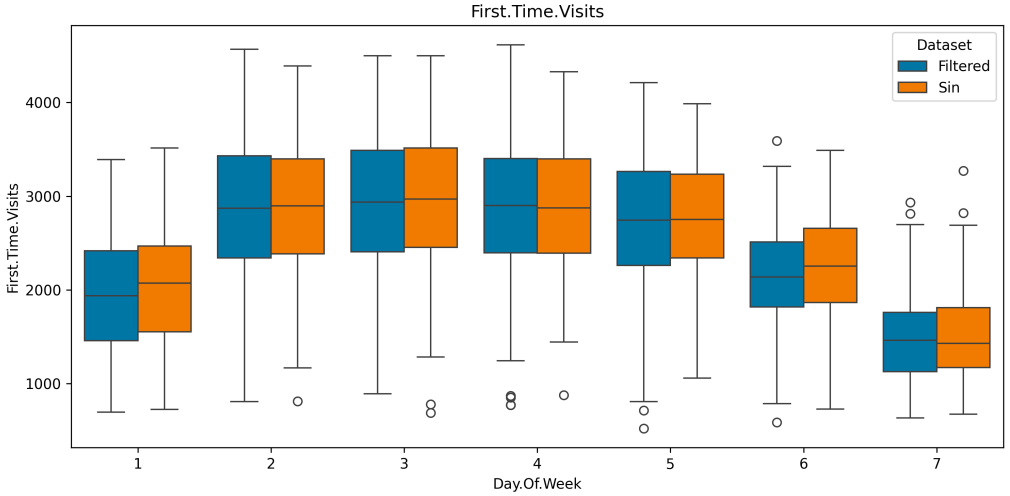


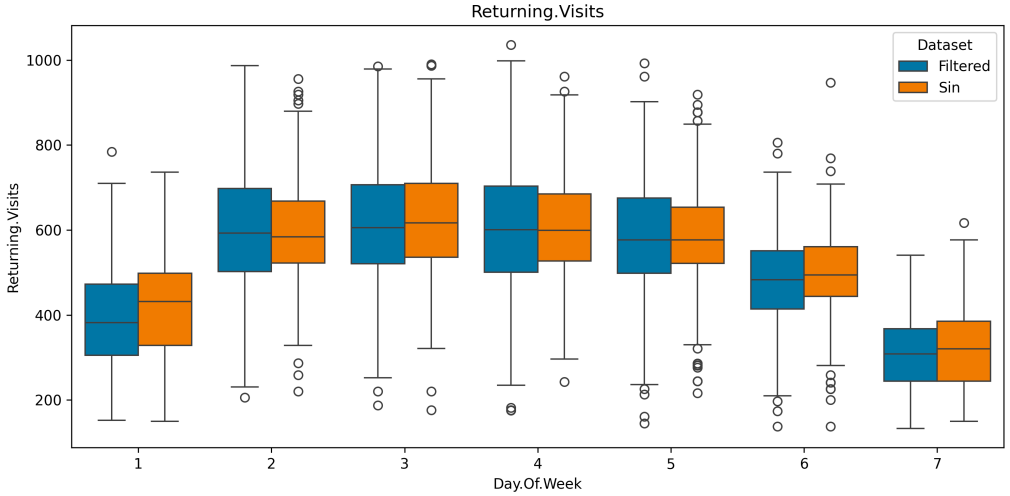


DAILY DISTRIBUTIONS

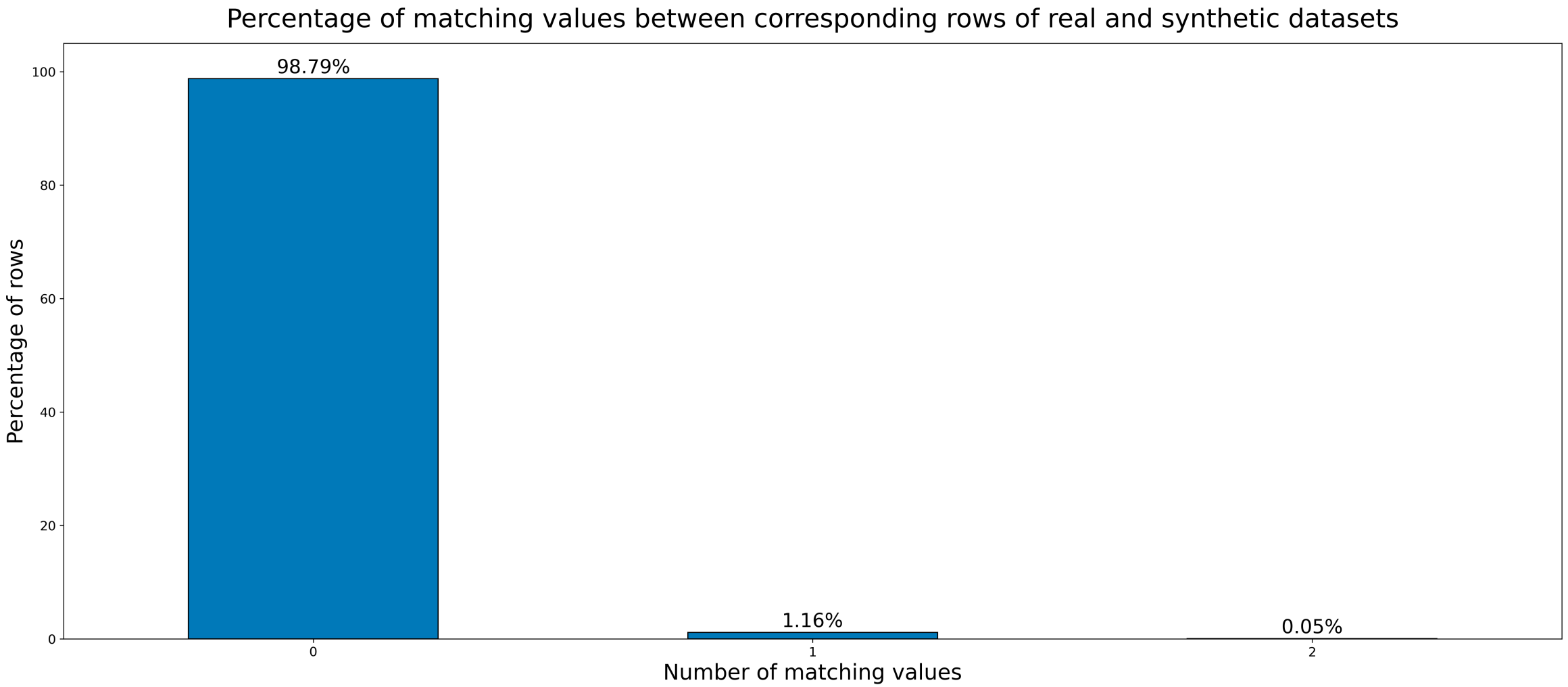




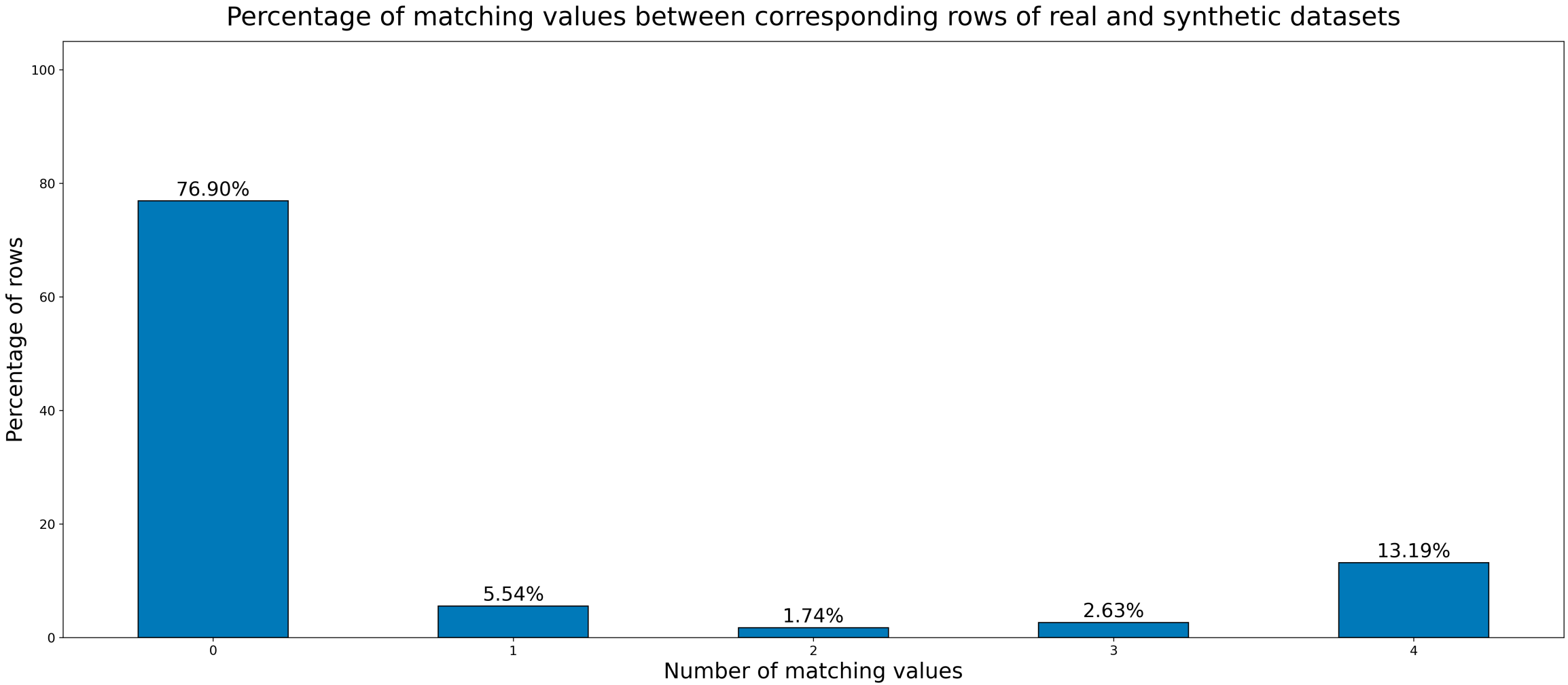




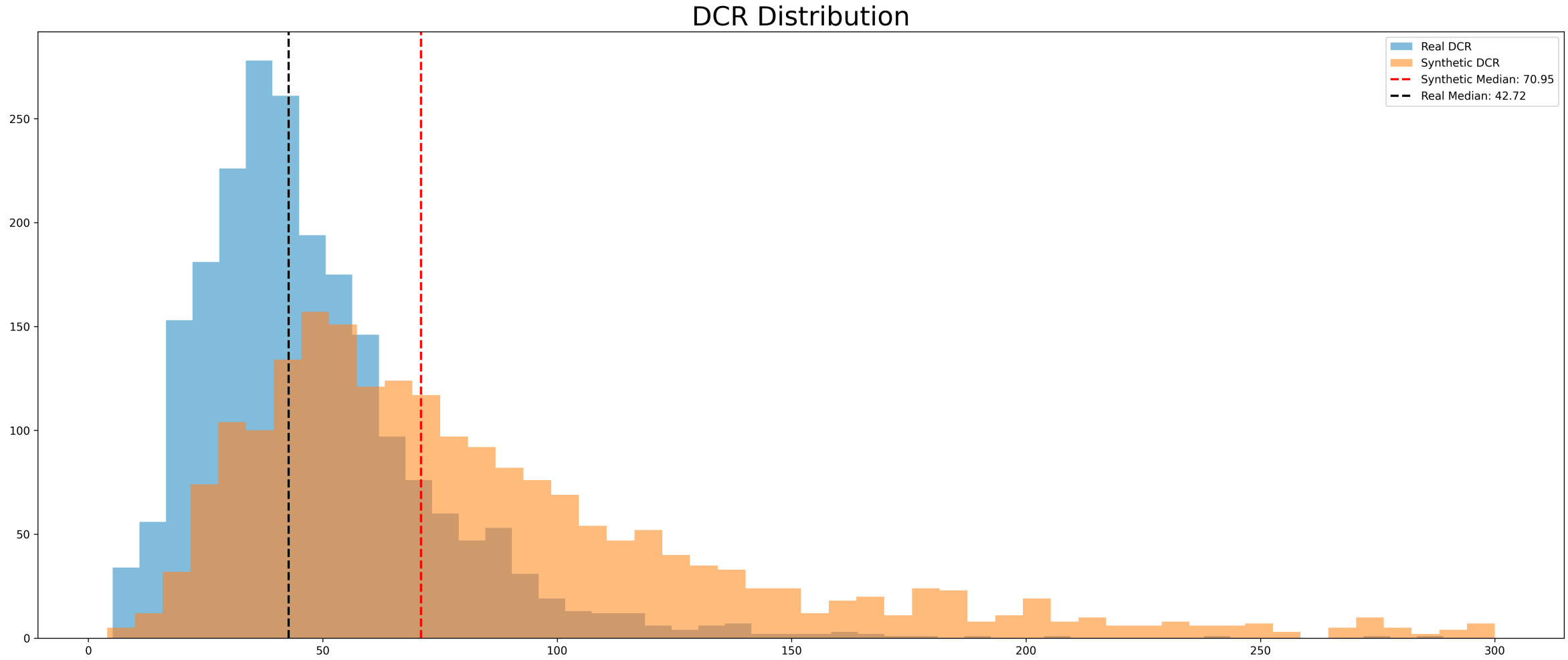
LOOKING FOR COPIED VALUES – TIME-WISE FRAMEWORK



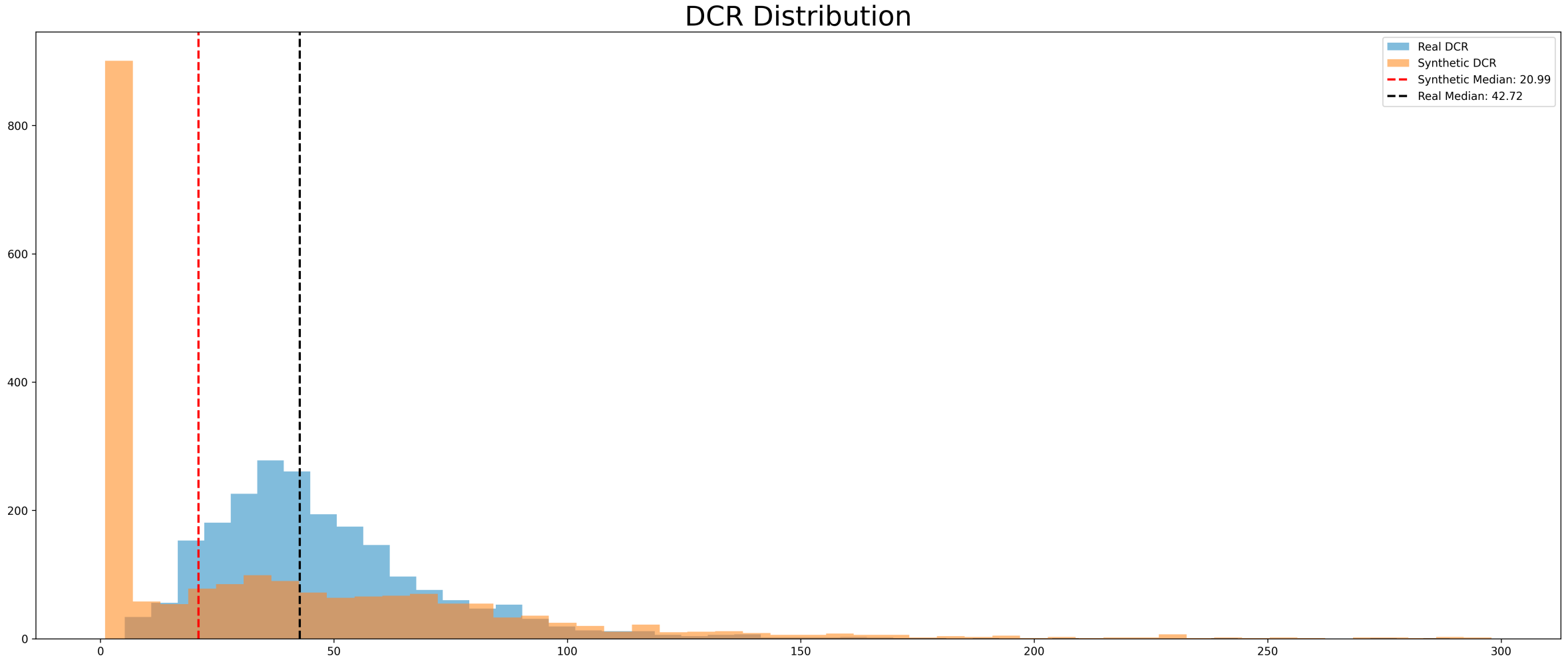
LOOKING FOR COPIED VALUES – TIME-WISE FRAMEWORK



DISTANCE TO THE CLOSEST RECORD – TIME-WISE FRAMEWORK



DISTANCE TO THE CLOSEST RECORD – ORIGINAL FRAMEWORK



DISCRIMINATOR AND MLE

TIME-WISE FRAMEWORK

	Weather	Energy	Web
Discriminator	0.59	0.56	0.63
MLE S-R	0.31	0.63	0.28
MLE R-R	0.29	0.73	0.28

ORIGINAL FRAMEWORK

	Weather	Energy	Web
Discriminator	0.42	0.34	0.37
MLE S-R	0.37	0.77	0.53
MLE R-R	0.29	0.73	0.28

FURTHER DEVELOPMENTS: AUTOCORRELATION WITHIN LAGS

- Let the model understand the real sequentiality of data

PAGE.LOADS

Lag	Real	Synthetic
0	1.00	1.00
1	0.74	0.30
2	0.35	0.33
3	0.14	0.29
4	0.13	0.29

FIRST.TIME.LOADS

Lag	Real	Synthetic
0	1.00	1.00
1	0.76	0.33
2	0.40	0.36
3	0.20	0.31
4	0.10	0.31

UNIQUE.VISITS

Lag	Real	Synthetic
0	1.00	1.00
1	0.76	0.31
2	0.37	0.32
3	0.17	0.28
4	0.16	0.28