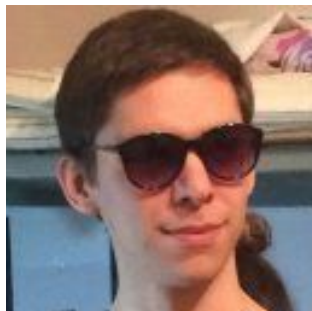




# On Embeddings for Numerical Features in Tabular Deep Learning

Рецензент: Петров Михаил  
НИС МОП, 18.01.2023

# Авторы статьи



**Yury Gorishniy,**  
Yandex Research,  
MIPT  
*115 citations*  
*h-index: 2*



**Ivan Rubachev,**  
Yandex Research,  
HSE  
*142 citations*  
*h-index: 3*



**Artem Babenko,**  
Yandex Research,  
HSE  
*4146 citations*  
*h-index: 18*

# Revisiting Deep Learning Models for Tabular Data

- Сравнительный анализ девяти разных DL-моделей в применении к табличным данным на 11 датасетах
- Показано, что transformer-based модели хорошо конкурируют с бустингами, а простые модели можно очень хорошо затюнить
- 103 цитирования

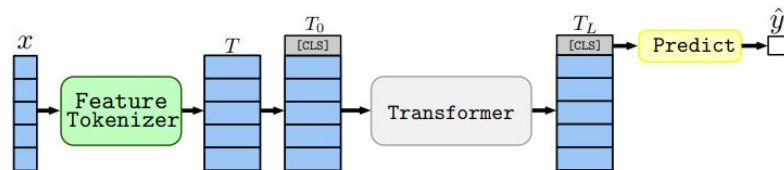


Figure 1: The FT-Transformer architecture. Firstly, Feature Tokenizer transforms features to embeddings. The embeddings are then processed by the Transformer module and the final representation of the [CLS] token is used for prediction.

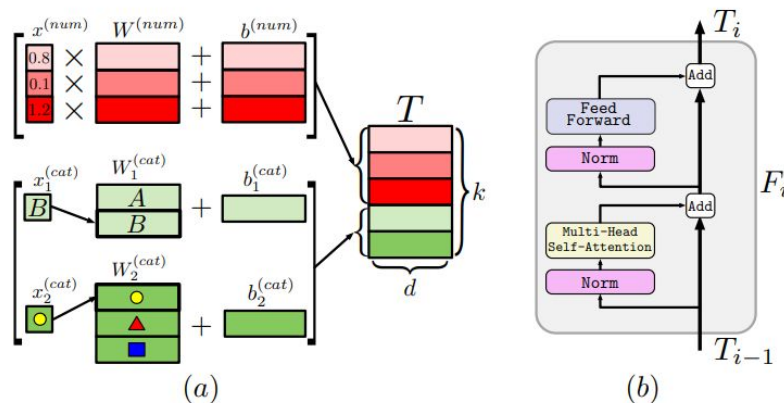


Figure 2: (a) Feature Tokenizer; in the example, there are three numerical and two categorical features; (b) One Transformer layer.

# Подходы к эмбедингам



- **Feature binning:** Dougherty et al. (1995). Supervised and unsupervised discretization of continuous features.
- **Periodic activations:**
  - *MLP:* Tancik et al. (2020). Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains.
  - *Transformers:* Vaswani et al. (2017). Attention is all you need.
  - *ResNet:* Li et al. (2021). Learnable fourier features for multi-dimensional spatial positional encoding.

# Revisiting Pretraining Objectives for Tabular Deep Learning

Разработанные модели используются дальше для исследования влияния предобучения на tabular DL-модели

	GE ↑	CH ↑	CA ↓	HO ↓	OT ↓	HI ↑	FB ↓	AD ↑	WE ↓	CO ↑	MI ↓
MLP											
no pretraining	0.635	0.849	0.506	3.156	0.479	0.801	5.737	0.908	1.909	0.963	0.749
contrastive	0.672	<b>0.855</b>	0.455	<b>3.056</b>	0.469	<b>0.813</b>	5.697	0.910	1.881	0.960	0.748
rec	0.662	0.853	<b>0.445</b>	<b>3.044</b>	<b>0.466</b>	0.805	<b>5.641</b>	<b>0.910</b>	<b>1.875</b>	<b>0.965</b>	<b>0.746</b>
mask	<b>0.691</b>	<b>0.857</b>	0.454	3.113	0.472	<b>0.814</b>	<b>5.681</b>	<b>0.912</b>	1.883	0.964	0.748
MLP-PLR											
no pretraining	0.668	<b>0.858</b>	0.469	<b>3.008</b>	0.483	0.809	<b>5.608</b>	0.926	1.890	0.969	0.746
rec	0.667	0.852	<b>0.439</b>	<b>3.031</b>	<b>0.472</b>	0.808	<b>5.571</b>	<b>0.926</b>	<b>1.877</b>	<b>0.971</b>	<b>0.745</b>
mask	<b>0.685</b>	<b>0.863</b>	<b>0.434</b>	<b>3.007</b>	0.477	<b>0.818</b>	<b>5.586</b>	<b>0.927</b>	1.911	<b>0.970</b>	0.748
MLP-T-LR											
no pretraining	0.634	<b>0.866</b>	0.444	3.113	0.482	0.805	5.520	0.925	1.897	0.968	0.749
rec	<b>0.652</b>	0.857	<b>0.424</b>	3.109	<b>0.472</b>	0.808	<b>5.363</b>	0.924	<b>1.861</b>	<b>0.969</b>	<b>0.746</b>
mask	<b>0.654</b>	<b>0.868</b>	<b>0.424</b>	<b>3.045</b>	<b>0.472</b>	<b>0.818</b>	5.544	<b>0.926</b>	1.916	<b>0.969</b>	0.748



## Сильные стороны

- “Простая” тема
- Обилие экспериментов
- Есть код

## Слабые стороны (слабые ли?)

- Почему бы не применить periodic препроцессинг к бустингам?
- Не хватает анализа