SotA для TS forecasting, все ли так однозначно?

Дмитрий Симаков
Sber AI Lab
02 2025

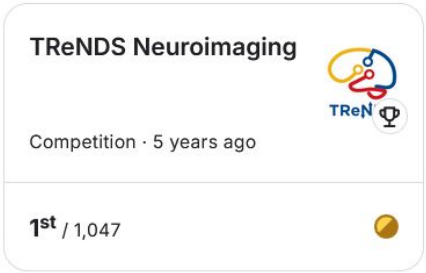
О лекторе:

- Team Lead at Sber AI Lab
- Kaggle Master
- Иногда пишу статьи

Области интересов:

- Tabular, TS, Seq
- DL (NLP / LLM, Metric Learning)

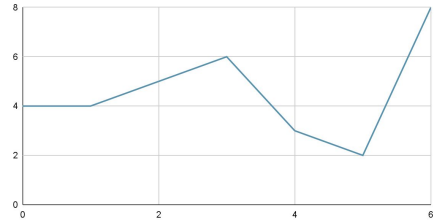
AUTOML GRAND PRIX POINTS WINNER



TS -> ML

Выбираем логику составления плоского датасета: **горизонт** предсказания, длину **истории**, размер пересечения окна
Индексы – составляем в соответствие наблюдению в плоской таблице группу наблюдений из длинной.

Временной ряд



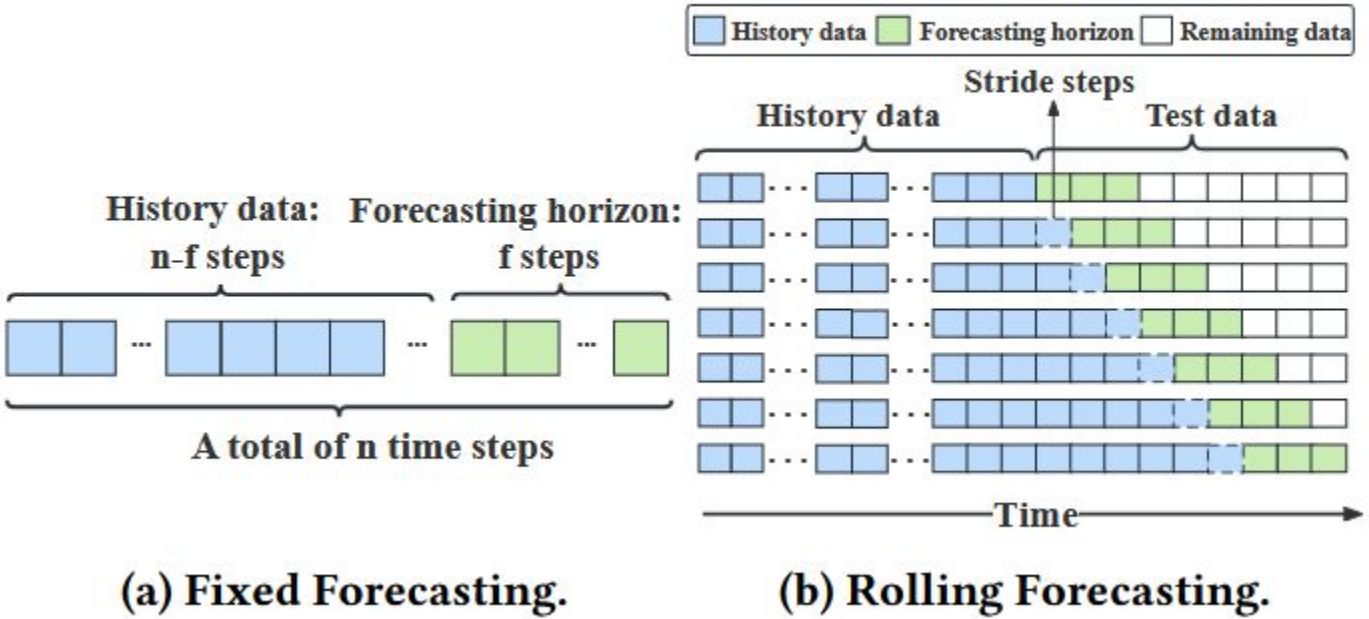
1. ряд представлен в виде **длинной** таблицы
2. “нарезаем” окна из ряда
3. для каждого окна считаем агрегаты
4. получаем **широкий** датасет с X и y

time	value	Additional features
0	4	…
1	4	…
2	5	…
3	6	…
4	3	…
5	2	…
6	8	…

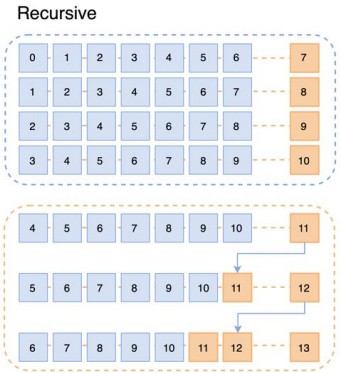
id	ids history	Ids horizon
0	[0, 1, 2]	[3, 4]
1	[1, 2, 3]	[4, 5]
2	[2, 3, 4]	[5, 6]

base ts	lag 3	lag 2	lag 1	Max	target 1	target 2
2	4	4	5	5	6	3
3	4	5	6	6	3	2
4	5	6	3	6	2	8

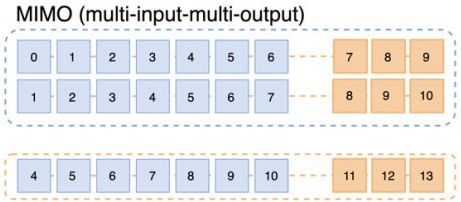
TS -> ML: как оценивать качество?



Стратегии прогнозирования на несколько точек вперед

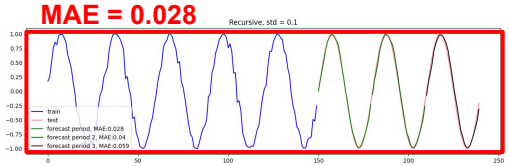
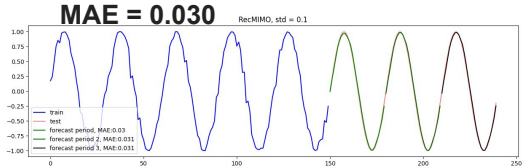
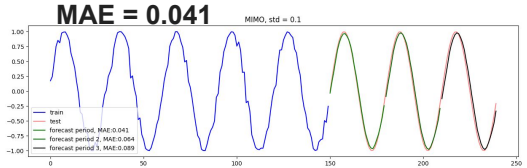
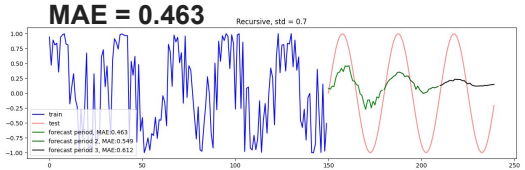
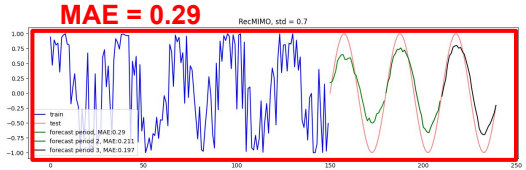
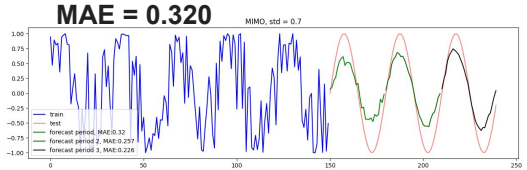
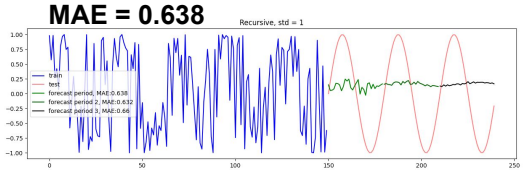
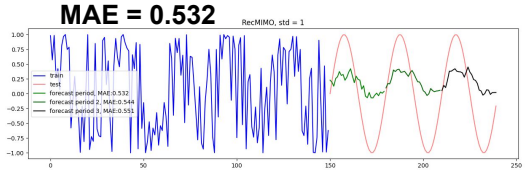
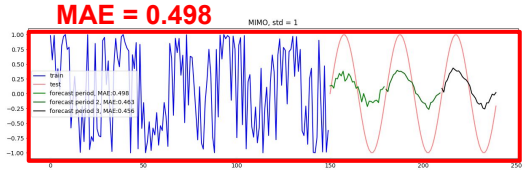


Recursive - авторегрессионная
Плюсы: быстрое обучение, хорошая метрика на короткие горизонты, требует малой длины ряда, можно получить прогноз на любой горизонт
Минусы: медленный инференс на длинные горизонты, накопление ошибок, сложность использования экзогенных признаков
Встречается: NLP (Causal LM), RecSys (next item prediction)



MIMO - множественный выход
Плюсы: быстрое обучение и инференс, легко добавлять признаки
Минусы: не подойдет для коротких рядов, длинный горизонт возможен только для длинных рядов
Встречается: NLP (multitoken prediction)

Зачем использовать разные стратегии?

std / strategy	Recursive	RecMIMO 10	MIMO
0.1	MAE = 0.028 	MAE = 0.030 	MAE = 0.041 
0.7	MAE = 0.463 	MAE = 0.29 	MAE = 0.320 
1	MAE = 0.638 	MAE = 0.532 	MAE = 0.498 

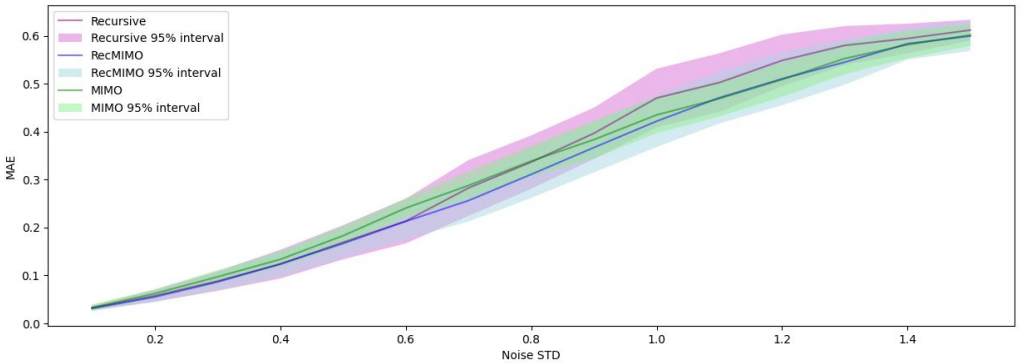
https://www.kaggle.com/code/simakov/tsururu-demo-images-rec-mimo-std

Зачем использовать разные стратегии?

Не все так однозначно, но:

- Recursive хуже с ростом шума, выше волатильность метрик
- MIMO сглаживает шум, более стабильна
- RecMIMO берет лучшее из двух подходов

Многое зависит от шума в конце истории



mean MAE

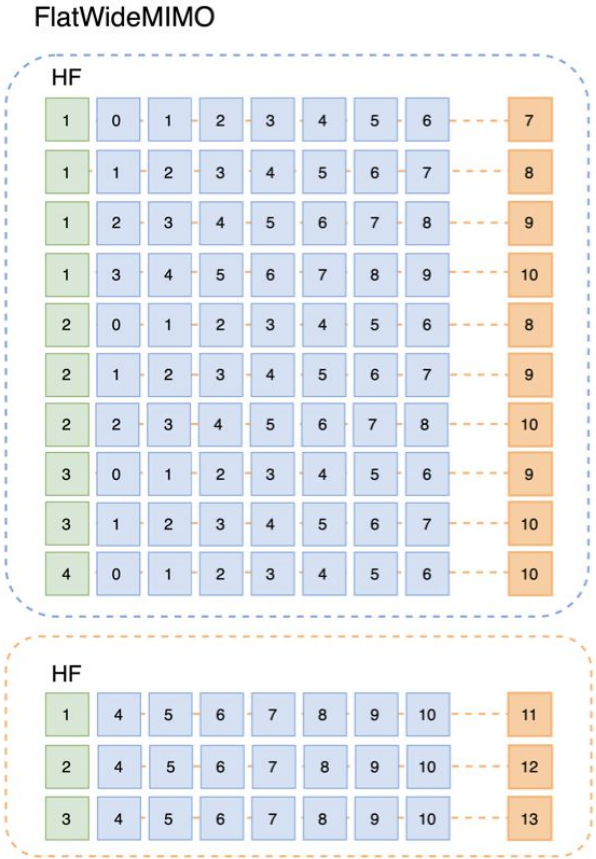
std	Recursive	RecMIMO	MIMO
0.100000	0.032035	0.030644	0.032427
0.200000	0.057227	0.054676	0.061837
0.300000	0.087785	0.086059	0.096687
0.400000	0.123184	0.123513	0.133176
0.500000	0.168716	0.166207	0.182431
0.600000	0.213032	0.212531	0.240233
0.700000	0.282204	0.255867	0.287457
0.800000	0.336254	0.310685	0.338512
0.900000	0.396425	0.366704	0.383235
1.000000	0.470196	0.421737	0.434884
1.100000	0.502446	0.470753	0.468884
1.200000	0.548604	0.510278	0.508815
1.300000	0.580037	0.545105	0.552802
1.400000	0.594543	0.583606	0.581455
1.500000	0.611975	0.599659	0.601766

std MAE

Recursive	RecMIMO	MIMO
0.008071	0.008666	0.010568
0.020120	0.016745	0.014805
0.032698	0.028295	0.022476
0.048445	0.043769	0.025141
0.057462	0.047758	0.033503
0.075231	0.059821	0.032925
0.093897	0.070482	0.046723
0.089965	0.079553	0.051420
0.086491	0.083359	0.063326
0.098842	0.086882	0.061166
0.098628	0.086665	0.059936
0.087271	0.088852	0.058375
0.065129	0.076102	0.052443
0.049789	0.053381	0.045624
0.035126	0.050098	0.036095

https://www.kaggle.com/code/simakov/tsururu-demo

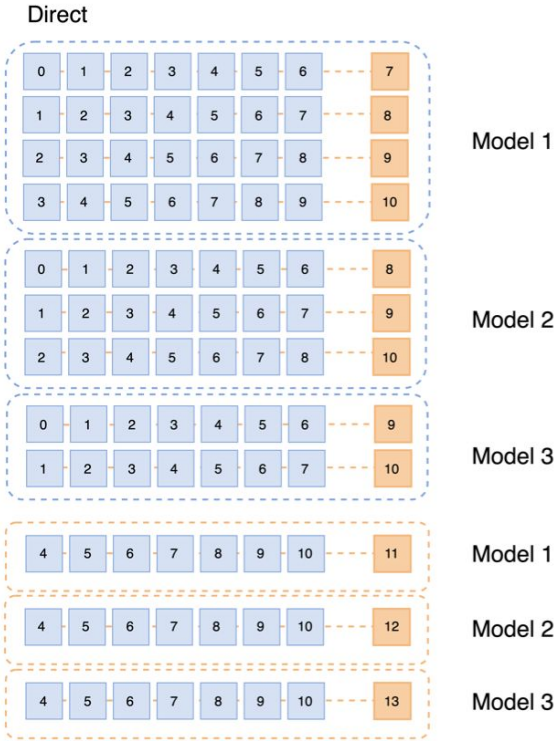
Стратегии прогнозирования на несколько точек вперед

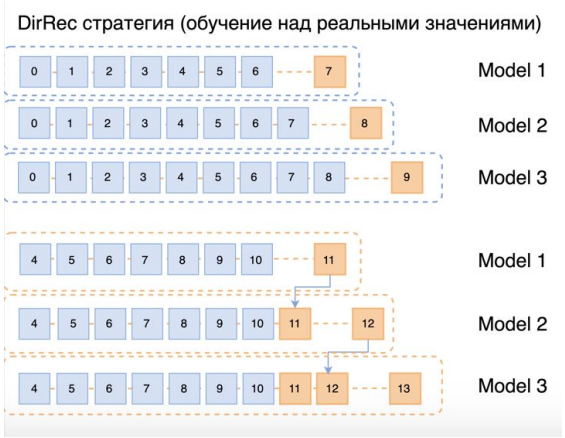


FlatWideMIMO - развернутая по строкам MIMO
Плюсы: нативная передача признаков из будущего (даты, прогнозные величины) без накопления ошибки, инференс быстрее Recursive, выборка для обучения синтетически больше
Минусы: долгое обучение (в horizon раз)
Встречается:

- Промежуточный вариант между регрессией и прогнозом
- Мультилейбл классификация, когда у каждого таргета (товара) есть свои признаки.

Стратегии прогнозирования на несколько точек вперед



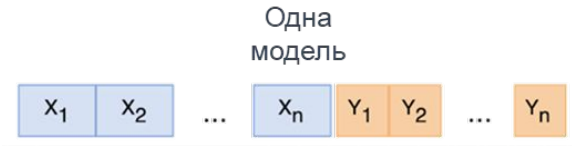


Подходы к работе с несколькими рядами

Local

Много моделей, по одной для каждого временного ряда.

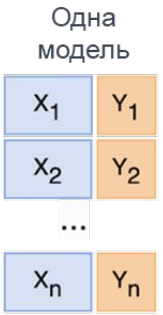
Multivariate



Одна модель для всех одномерных временных рядов.

Признаки наблюдений, относящихся к одной временной точке, **объединяются**

Global

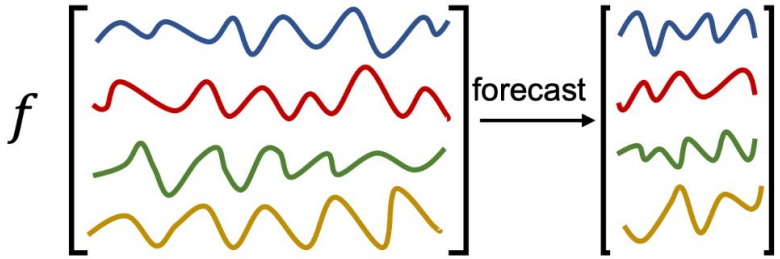


Одна модель для всех одномерных временных рядов.

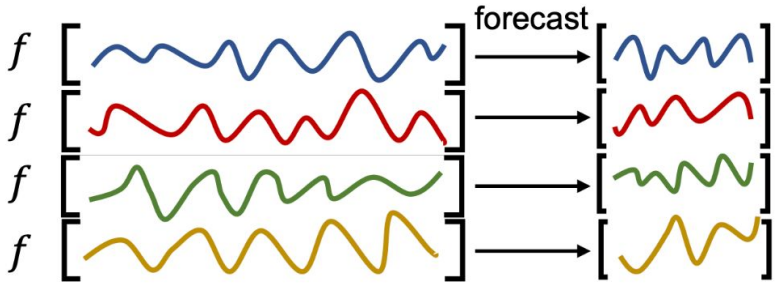
Признаки отдельных наблюдений **не пересекаются** между рядами.

https://arxiv.org/abs/2008.00444

Multivariate: CI and CM



(a) Channel Dependent (CD) Strategy



(b) Channel Independent (CI) Strategy

Chanel Independence на примере PatchTST:

[bs, nvars, seq_len] - batch size, число рядов, history

[bs x nvars x **patch_num x patch_len**] - patching

[bs x nvars x patch_num x **d_model**] - linear projection

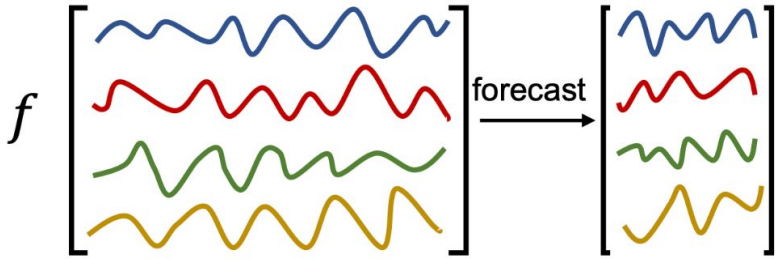
[**(bs * nvars)** x patch_num x d_model] - combine bs and nvars

[(bs * nvars) x patch_num x d_model] - **backbone forward**

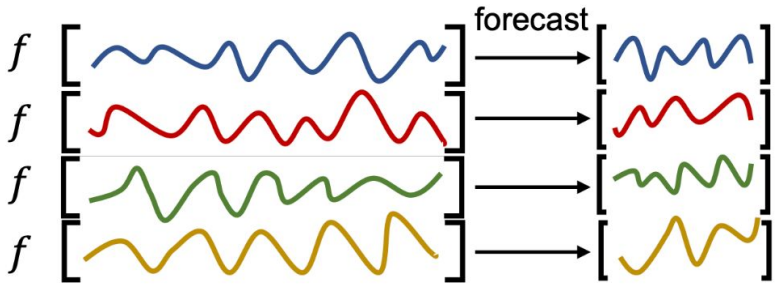
[**bs x nvars** x patch_num, d_model] - split bs and nvars

[bs x nvars x **horizon**] - after head

Multivariate: CI and CM



(a) Channel Dependent (CD) Strategy



(b) Channel Independent (CI) Strategy

Chanel Independence на примере PatchTST:

[bs, nvars, seq_len] - batch size, число рядов, history

[bs x nvars x **patch_num x patch_len**] - patching

[bs x nvars x patch_num x **d_model**] - linear projection

[**(bs** * **nvars)** x patch_num x d_model] - combine bs and nvars

[(bs * nvars) x patch_num x d_model] - **backbone forward**

[**bs x nvars** x patch_num, d_model] - split bs and nvars

[bs x nvars x **horizon**] - after head

Chanel Independence **- shared веса** для каждой компоненты, по сути эквивалентен **global**

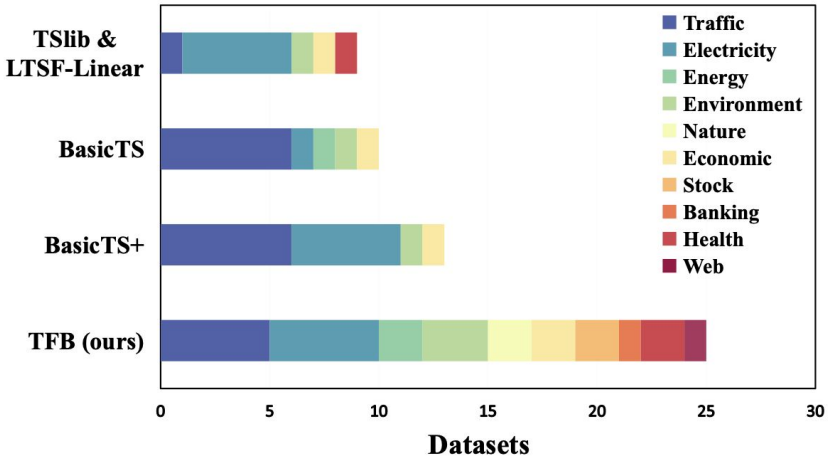
SotA TS forecasting и его проблемы

- Недостаточная представленность различных доменов => возможный оверфит на бенчмарк (**Issue 1**)
- Слабые бейзлайны, отсутствие классических бейзлайнов (**Issue 2**)
- Непонятно, от чего конкретно появляется прирост по качеству в статьях, проблема неконсистентных сравнений (**Issue 3**)
- Не исследуется вопрос целесообразности (скорость / качество) (**Issue 4**)

Issues 1, 2, 3: https://arxiv.org/abs/2403.20150v3
Issue 3, 4: https://arxiv.org/abs/2310.06119

SotA TS forecasting и его проблемы

- Для старых DL моделей - использование финансовых данных, исключение некоторых точек из сравнения [1]
- Для новых DL моделей - маленькая длина входа, единые гиперпараметры для всех методов [1], drop last trick [2]
- Для фундаментальных моделей - на соревнованиях проигрывают [3], решения на базе LLM работают хуже чем без них [4], но требуют больше вычислительных ресурсов
- Бенчмарки нерепрезентативны, для использования на конкретной задаче всё равно нужно проверять на своих данных - проблема подбора контекста в фундаментальных моделях, правильная подача контекста и дополнительной информации улучшает качество [1, 5]

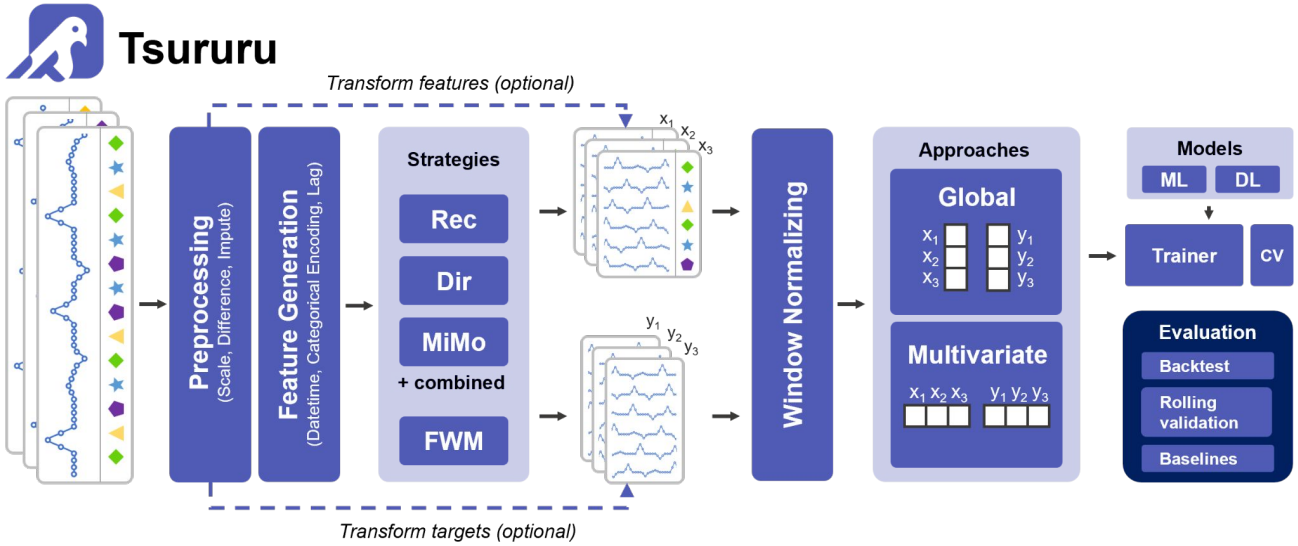


[1] https://youtu.be/vNul_AjRPFw?si=Ya71_JxtnDyBXKYd
[2] https://arxiv.org/abs/2403.20150v3
[3] https://cbergmeir.com/papers/Bergmeir2024LLMs.pdf
[4] https://arxiv.org/abs/2406.16964
[5] https://arxiv.org/abs/2410.18959

Подобные проблемы в других областях ML

- **RecSys**: многие SotA методы при честном сравнении хуже более простых бейзлайнов [6]
- **RL**: бейзлайн с фичами из статей работает лучше, чем модели из самих статей [7]
- **Optimization**: Новые алгоритмы оптимизации (LARS, LAMB) оказываются хуже, чем стандартные (вроде Adam) при тюнинге параметров [8]
- **Graphs**: В графовых задачах [9] оказывается, что некоторые датасеты для тестирования алгоритмов на гетерофильных графов имеют серьезные проблемы, например утечку данных
- **Metric learning:** в статьях утверждалось улучшение метрик за счёт новых лоссов, но на самом деле оно было за счёт более продвинутых моделей и отсутствия тюнинга гиперпараметров в сравнениях [10]
- **Tabular**: многие методы автоматической генерации признаков используют тестовую выборку для подбора оптимальных признаков [11]

[6] https://arxiv.org/abs/1907.06902
[7] https://arxiv.org/abs/2305.09836
[8] https://openreview.net/forum?id=E9e18Ms5TeV
[9] https://arxiv.org/abs/2302.11640
[10] https://arxiv.org/abs/2003.08505
[11] https://arxiv.org/abs/2211.12507



Hyperparameter	Value	NN models Rank	NN models Median MAE	Boosting Rank	Boosting Median MAE	Overall Rank	Overall Median MAE
Datetime Features	False	1.3819	1.0087	1.3778	1.7026	1.3810	1.0671
	True	1.6181	1.1323	1.6222	1.8036	1.6190	1.2209
ID Features	False	1.7262	1.0780	1.6286	1.8237	1.6975	1.1698
	True	1.2738	1.0024	1.3714	1.7486	1.3025	1.1151
Mode	Global	1.5476	1.0056	1.0857	1.7048	1.5476	1.1213
	Multivariate CI	2.2619	1.1217	—	—	2.2619	1.1217
	Multivariate CM	2.1905	1.1319	1.9143	1.8426	2.1905	1.2380
Prediction Strategy	FlatWideMIMO	3.9375	1.3080	2.7333	1.7906	3.6508	1.3770
	MIMO	1.7500	1.0280	2.7333	1.7210	1.9841	1.0779
	Recursive ($MH = 1$)	2.4167	1.0314	2.5333	1.8373	2.4444	1.1092
	Recursive ($MH = 6$)	1.8958	1.0228	2.0000	1.7144	1.9206	1.0877

https://github.com/sb-ai-lab/tsururu

Архитектуры: Supervised

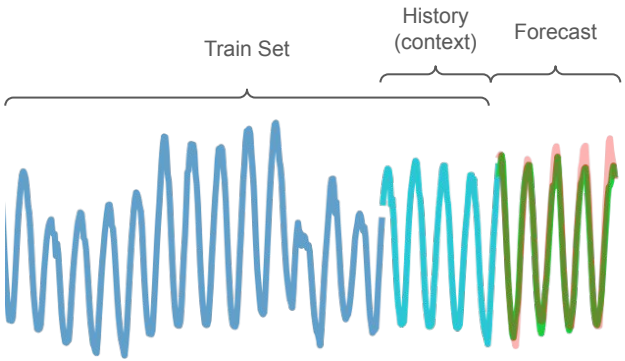
Stat & ML

Stat

- (Seasonal) Naive
- ARIMA
- ETS
- Theta

ML

- sklearn-like models
- CatBoost / PyBoost



Supervised: Train the model to predict labels for new data, based on patterns identified in the train set.

DL

Linear

- DLinear
- CycleNet

CNN

- TimesNet

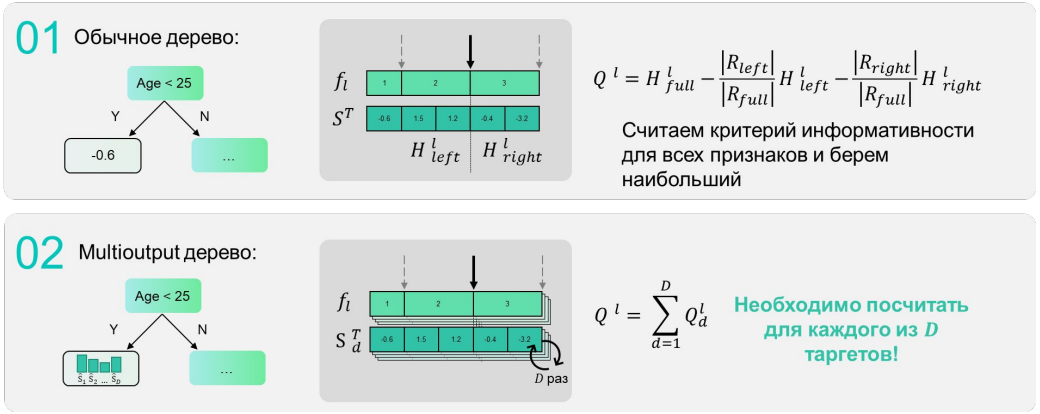
Transformers

- GPT4TS
- PatchTST
- Crossformer
- iTransformer

GBDT: PyBoost (2022, 13 citations)

Проблема: для MIMO стратегии прогнозирования необходим эффективный алгоритм предсказания MultiOutput таргета. Сложность стандартных алгоритмов бустинга растет линейно по числу выходов.

Идея: считать Information Gain не для всех таргетов



01 Top Outputs

Выбираем k выходов с наибольшей нормой градиентов

$$\underbrace{\|g_{i_1}\| \geq \|g_{i_2}\| \geq \cdots \geq \|g_{i_d}\|}_{Top\ k}$$

02 Random Sampling

Сэмплируем k выходов с оптимальными вероятностями p_i

$$p_i = \|g_i\|^2 / \sum_{j=1}^{D} \|g_j\|^2$$

Новый выход нормируем:

$$\hat{g} = \frac{1}{\sqrt{k p_i}} g_i$$

03 Random Projections

Сэмплируем k случайных комбинаций выходов

$$G_k = G\Pi$$

$\Pi \in \mathbb{R}^{dxk}$ - случайная матрица
С ячейками i.i.d. $\mathcal{N}(0, k^{-1})$

Code: https://github.com/sb-ai-lab/Py-Boost
Paper: https://openreview.net/forum?id=WSxarC8t-T

DLinear (2023, 1842 citations)

Идея: использовать простые линейные модели

$$\hat{X}_i = W X_i \quad W \in \mathbb{R}^{T \times L}$$

Предобработка:

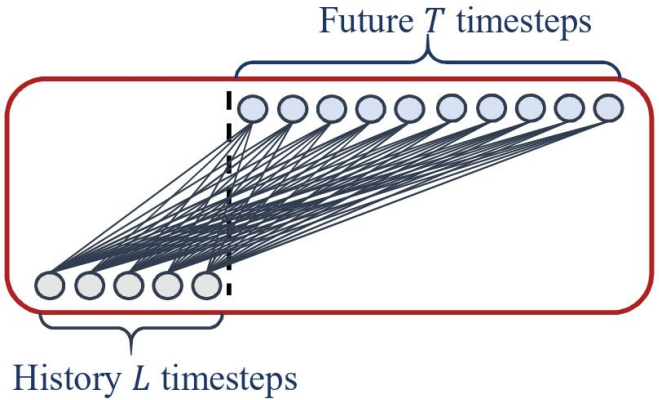
- Декомпозиционный модуль из Autoformer
  Извлечение тренда скользящим средним:

$$\mathcal{X}_{\mathrm{t}} = \mathrm{AvgPool}(\mathrm{Padding}(\mathcal{X}))$$
$$\mathcal{X}_{\mathrm{s}} = \mathcal{X} - \mathcal{X}_{\mathrm{t}},$$

отдельные модели

Задачи: forecasting

Постановка: multivariate (CI)



Code: https://github.com/vivva/DLinear
Paper: https://arxiv.org/abs/2205.13504

CycleNet (2024, 6 citations)

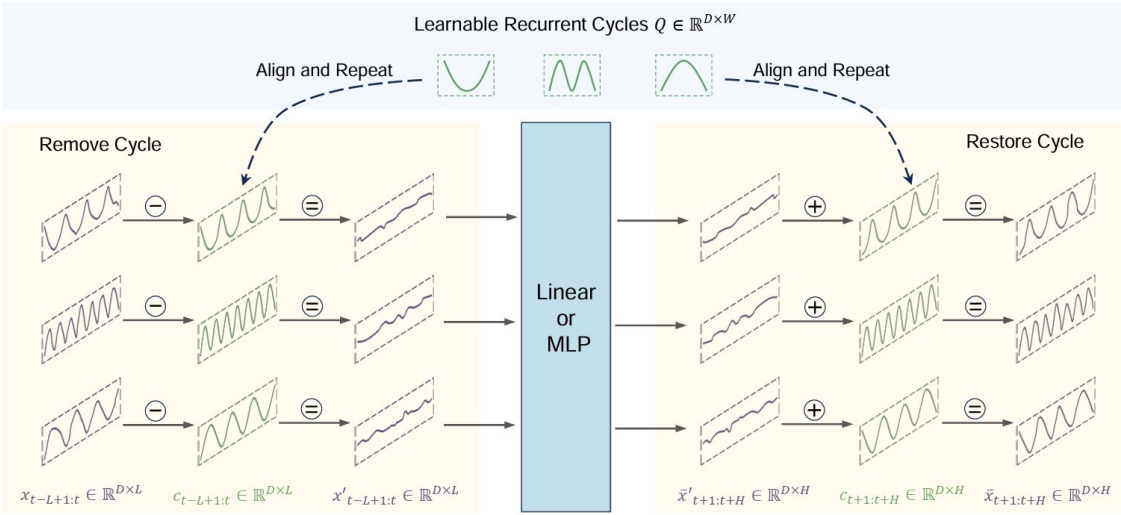
Идея: inductive bias для периодических паттернов.

Архитектура:

- Извлечение циклической компоненты обучаемыми шаблонами
- Моделирование остатков линейной моделью или MLP
- Возвращение циклической части

Задачи: forecasting
Постановка: multivariate (CI)

- Модуль цикла легко может быть встроен в любую другую модель



Code: https://github.com/ACAT-SCUT/CycleNet
Paper: https://arxiv.org/abs/2409.18479

TimesNet (2023, 1043 citations)

Идея: использовать 2D представления временных рядов в разрезе разных периодичностей

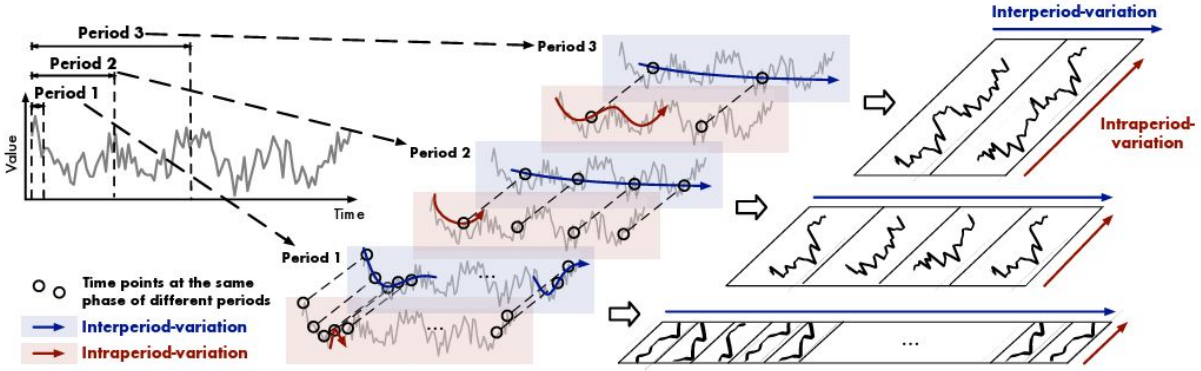
Архитектура:

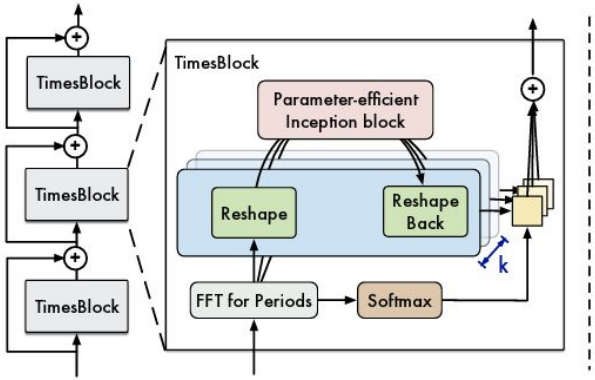
- FFT блок для определения top-k частот с наибольшей амплитудой
- 2D CNN

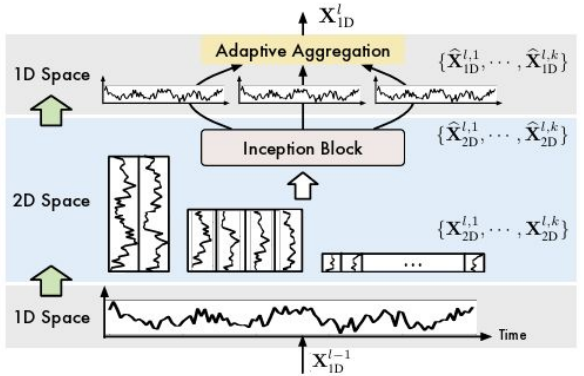
Задачи: forecasting, imputation, classification, anomaly detection
Постановка: multivariate (CD)

- Учет intra-period & inter-period взаимосвязей между рядами

Code: https://github.com/thuml/Time-Series-Library
Paper: https://arxiv.org/abs/2210.02186







PatchTST (2022, 1340 citations)

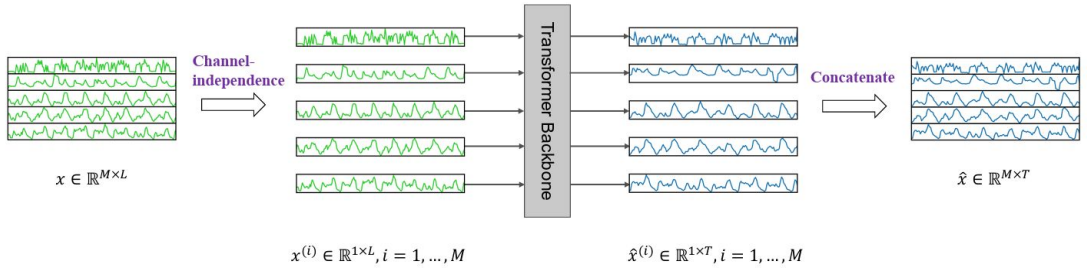
Идея: одно наблюдение в конкретный момент времени несет мало информации (как один пиксель в картинке) - лучше рассматривать “патчи”

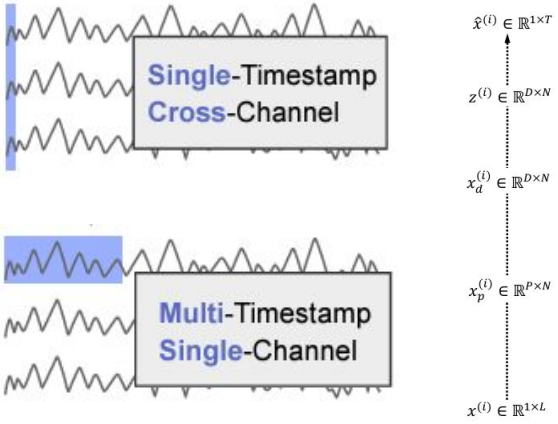
Архитектура:

- Transformer

Задачи: forecasting, self-supervised
Постановка: multivariate (CI)

- Число патчей меньше длины ряда - сокращение вычислительной сложности механизма внимания





Code: https://github.com/yuqinie98/PatchTST
Paper: https://arxiv.org/abs/2211.14730

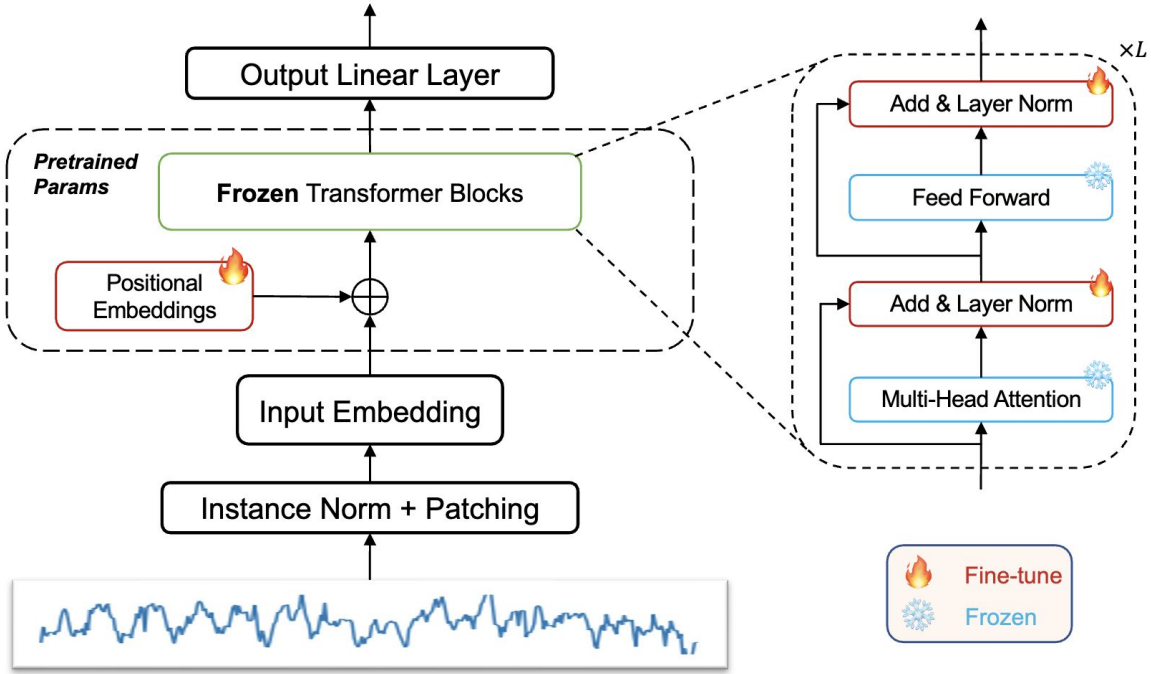
GPT4TS (2023, 343 citations)

Идея: использовать предобученные трансформерные модели на других доменах (NLP/CV).

Архитектура:

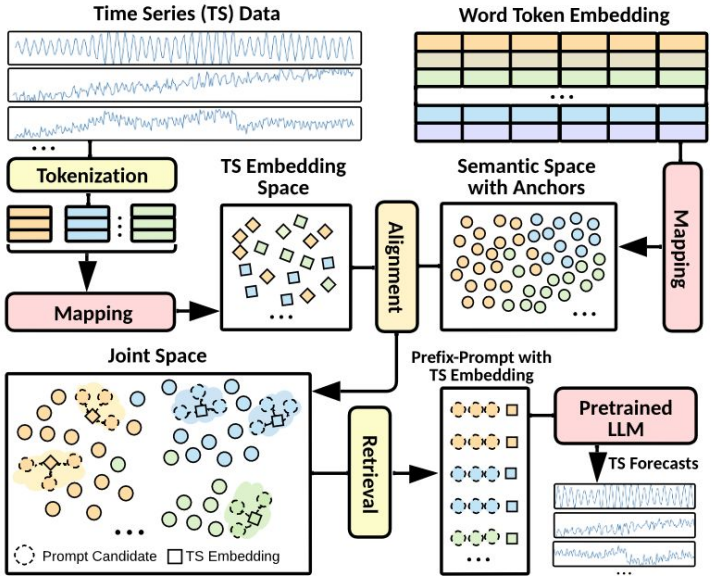
- Предобученный Transformer (GPT2, BERT)
- FF и MHA слои заморожены, так как в них хранится основная информация с предобучения

Задачи: forecasting, anomaly detection, imputation, zero/few-shot, classification
Постановка: global

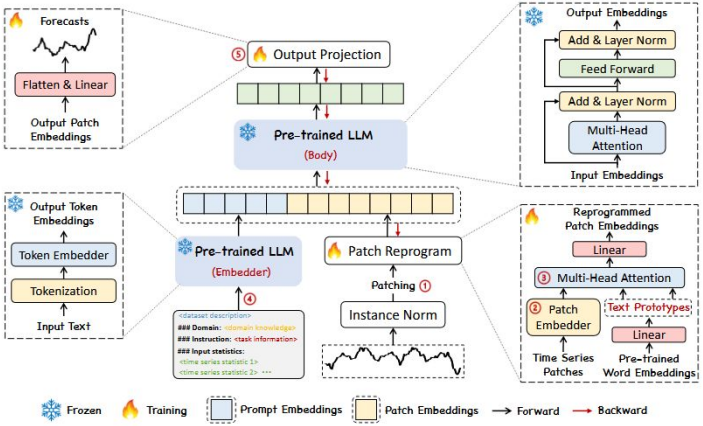


Code: https://github.com/DAMO-DI-ML/NeurIPS2023-One-Fits-All
Paper: https://arxiv.org/abs/2302.11939

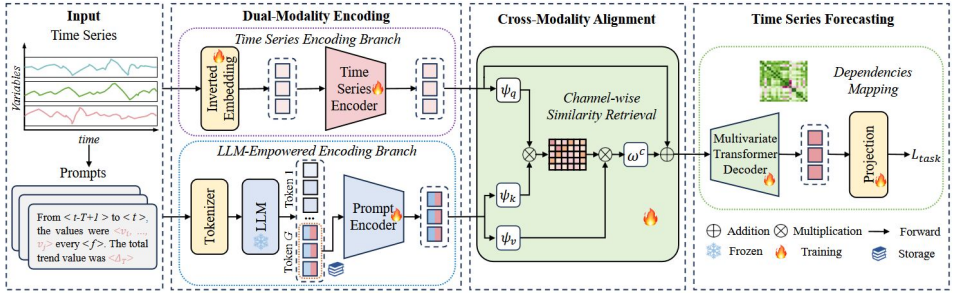
Issue 3: Странная SotA



https://openreview.net/forum?id=qwQVV5R8Y7



https://arxiv.org/abs/2310.01728



https://arxiv.org/abs/2406.01638

Issue 1: переменчивая SotA

PERFORMANCE OF ADVANCED TRANSFORMER MODELS AND BASIC LINEAR MODELS ACROSS HETEROGENEOUS MTS DATASETS.

Methods	PEMS04			PEMS08			ETTh2			ETTm2		
	MAE	RMSE	WAPE	MAE	RMSE	WAPE	MAE	RMSE	WAPE	MAE	RMSE	WAPE
Informer	27.94	44.74	12.84%	26.92	43.79	11.63%	7.12	6.87	47.44%	5.84	7.90	38.97%
Autoformer	34.72	50.33	14.81%	33.75	51.23	14.13%	3.33	**4.91**	22.17%	2.74	4.58	18.27%
FEDformer	**26.89**	**41.46**	**12.39%**	**25.14**	**39.17**	**10.87%**	**3.27**	4.93	**21.78%**	**2.70**	**4.54**	**17.99%**
Linear	37.42	**62.14**	17.22%	**34.04**	**57.07**	**14.71%**	3.18	5.04	21.19%	2.52	4.24	16.80%
DLinear	**37.51**	62.21	**17.26%**	34.15	57.18	14.76%	**3.13**	**5.00**	**20.85%**	**2.49**	4.23	16.63%
NLinear	37.62	62.38	17.31%	34.11	57.26	14.74%	3.16	5.06	21.09%	**2.49**	**4.21**	**16.60%**
Gap	**39.49%↓**	**49.87%↓**	**39.30%↓**	**35.40%↓**	**45.69%↓**	**35.32%↓**	**4.28%↑**	**1.83%↑**	**4.26%↑**	**7.78%↑**	**7.27%↑**	**7.72%↑**

https://arxiv.org/abs/2310.06119

Архитектуры: Zero-shot

LLM

Non-adapted LLM

- LSTPrompt
- PromptCast
- LLMTime

Adapted LLM

- Time-LLM
- FPT
- Chronos
- UniTS
- DAM

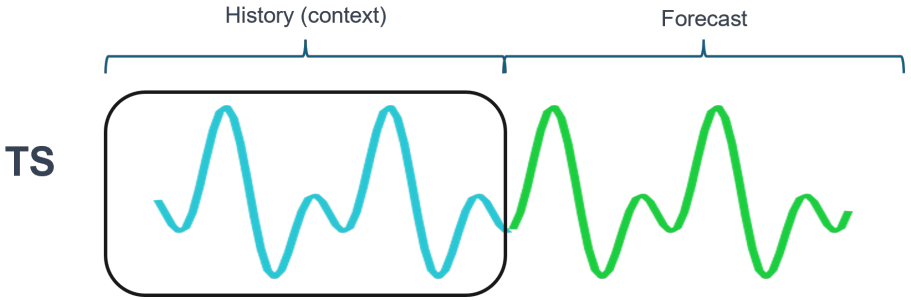
Specialized

Using synthetic data

- ForecastPFN

Using real data

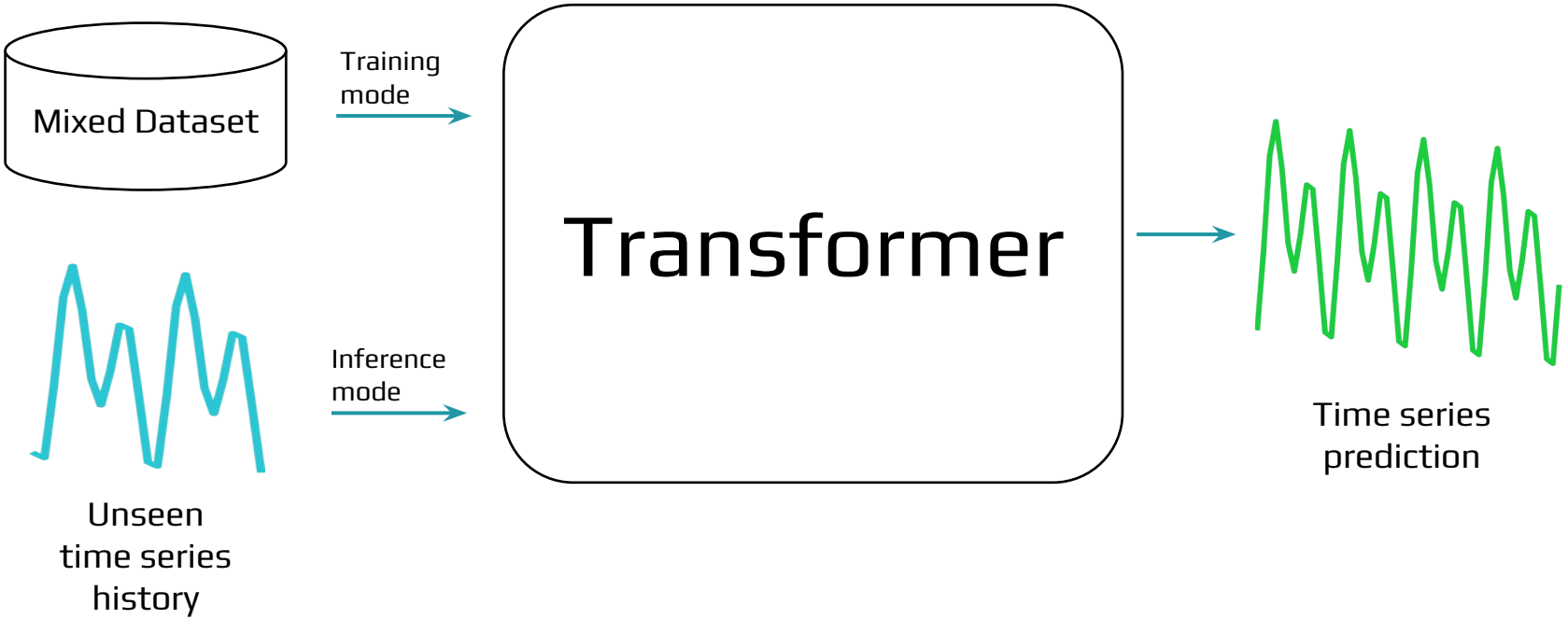
- GPHT
- MOIRAI
- Moment



Zero-shot: Train the model to predict labels for new data **without training on the target dataset**, based on patterns identified in the unrelated data.

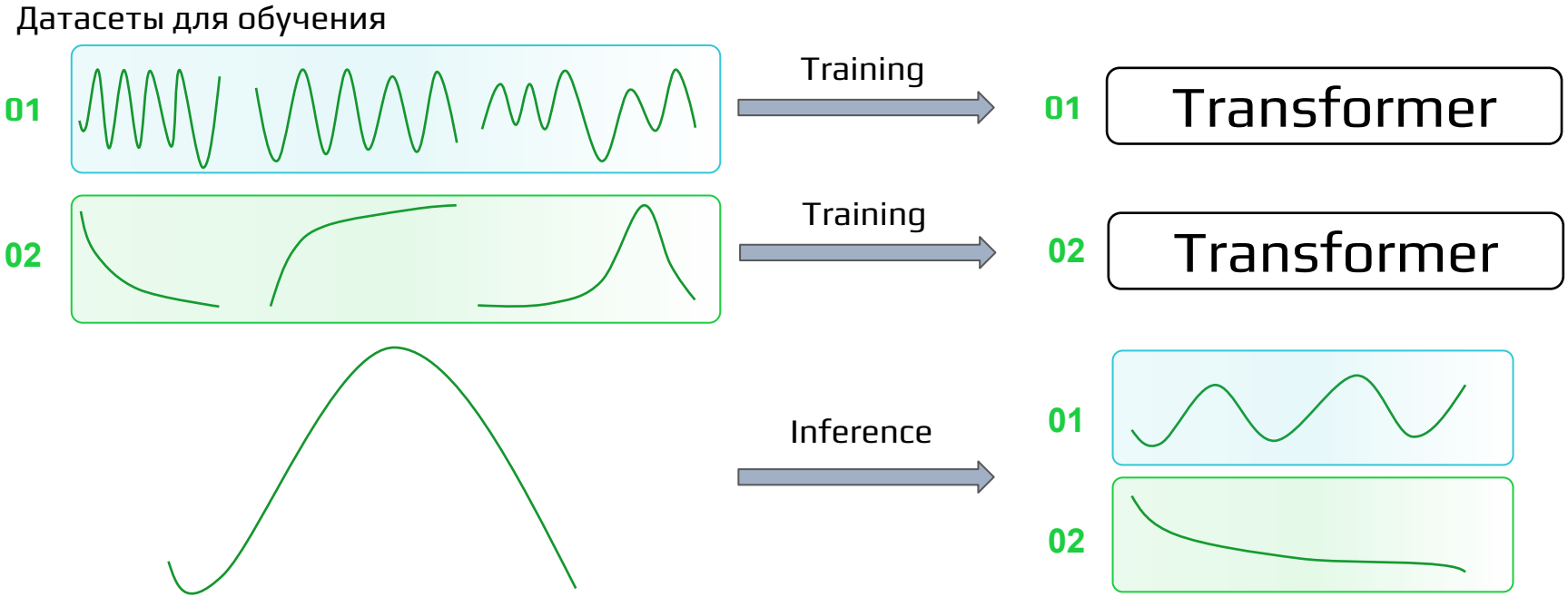
Обучение Zero-shot models for time series

Zero-shot модели для временных рядов – это в основном трансформеры.
Также они требуют большой разнообразный датасет для предобучения.

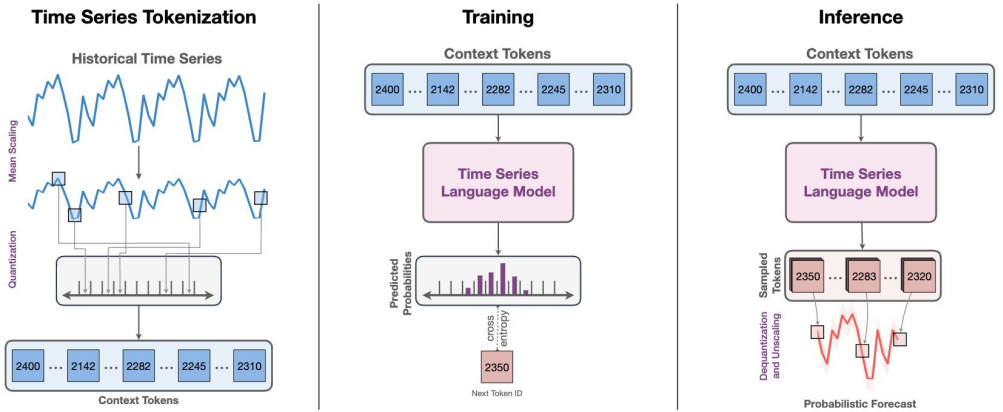


Данные для обучения zero-shot модели

Модель предсказывает те особенности (длина периода, тренд), которые заложены в данных



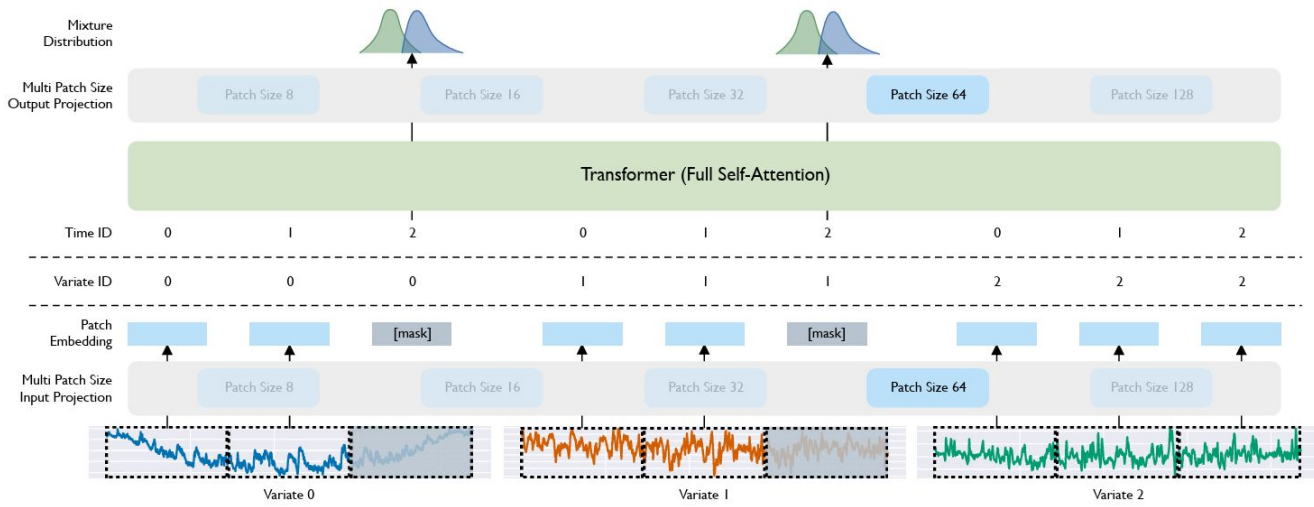
CHRONOS (2024, 190 citations)



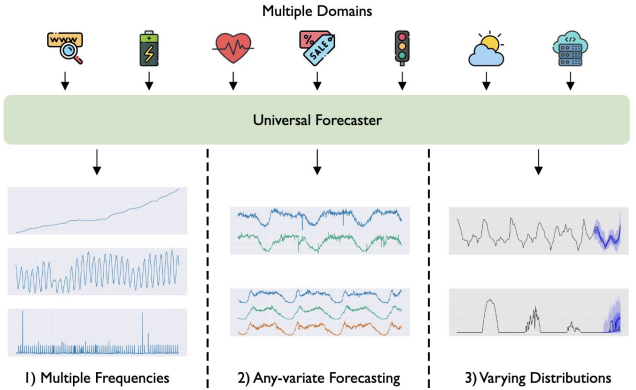
Идея: преобразование рядов в последовательность дискретных токенов для применения языковых моделей. Это позволяет унифицировать подход без необходимости к адаптации под конкретные данные.
Архетектура: используются стандартные трансформенные архитектуры, например, T5 без модификаций для временных рядов
Данные: Генерация данных через гауссовские процессы (KernelSynth) и 15 и 27 датасетов, разделенные в разные бенчмарки для in-domain и zero-shot предсказаний. Используются аугментация TSMixup, похожая на Mixup из картинок: сигналы из разных датасетов суммируются в один ряд с разными коэффициентами.
Задачи: probabilistic forecasting (uncertainty estimation), zero-shot прогнозы
Постановка: zero-shot

Code: https://github.com/amazon-science/chronos-forecasting
Paper: https://arxiv.org/abs/2403.07815

Moirai (2024, 116 citations)



Идея: фундаментальная модель для временных рядов, предобученная на рядах с различной грануляцией.
Архетектура: Маскированная encoder-only архитектура. RoPE эмбеддинг, flatten для multivariate данных, благодаря чему может обрабатывать данные с произвольной размерностью.
Данные: 27 млрд наблюдений из 9 доменов (энергетика, финансы, здравоохранение и др.), сагригированны из Monash, GluonTS, CloudOps TSF, LibCity и других датасетов. Включают различную гранулярность (от годовых до секундных наблюдений).
Задачи: probabilistic forecasting
Постановка: zero-shot



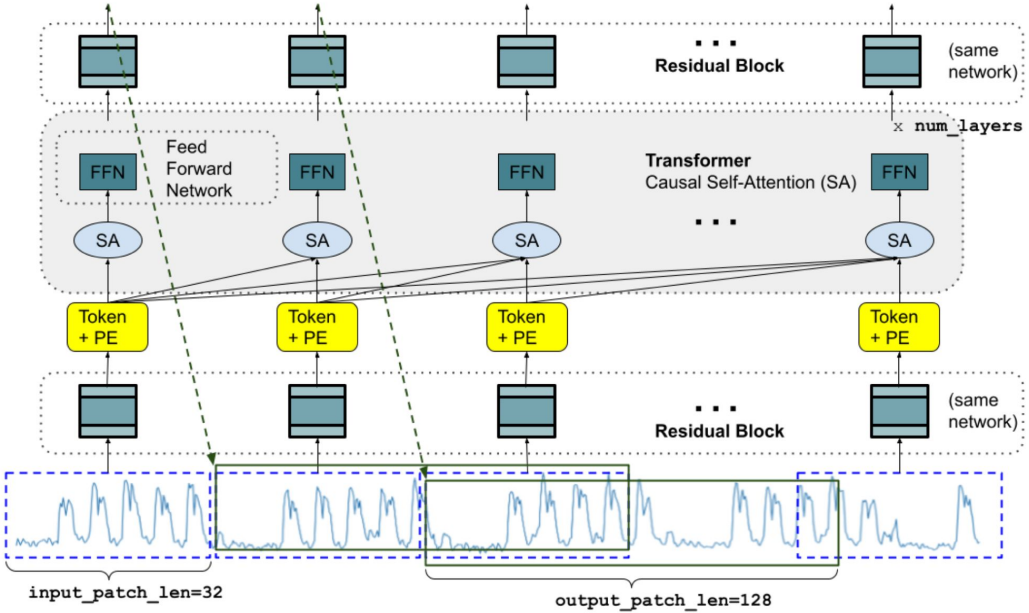
Code: https://github.com/SalesforceAIResearch/uni2ts
Paper: https://arxiv.org/abs/2402.02592

TimesFM (2023, 185 citations)

Идея: патчинг рядов, смешивание реальных данных и синтетики, форкаст как генеративная задача, разные размеры входного и выходного патча
Архитектура: трансформер декодер
Задачи: forecasting (+quantile)
Постановка: univariate (+ exogenous features)
Данные: синтетика + реальные (~370млрд наблюдений)

Table 1: Composition of TimesFM pretraining dataset.

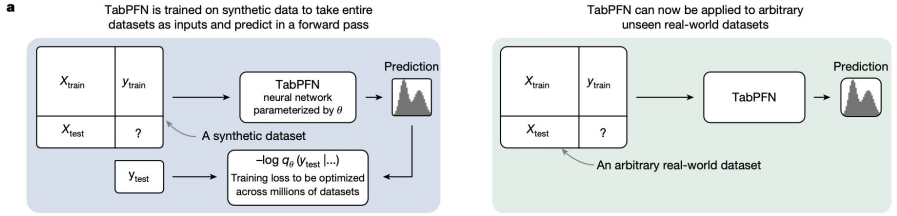
Dataset	Granularity	# Time series	# Time points
Synthetic		3,000,000	6,144,000,000
Electricity	Hourly	321	8,443,584
Traffic	Hourly	862	15,122,928
Weather [ZZP+21]	10 Min	42	2,213,232
Favorita Sales	Daily	111,840	139,179,538
LibCity [WJJ+23]	15 Min	6,159	34,253,622
M4 hourly	Hourly	414	353,500
M4 daily	Daily	4,227	9,964,658
M4 monthly	Monthly	48,000	10,382,411
M4 quarterly	Quarterly	24,000	2,214,108
M4 yearly	Yearly	22,739	840,644
Wiki hourly	Hourly	5,608,693	239,110,787,496
Wiki daily	Daily	68,448,204	115,143,501,240
Wiki weekly	Weekly	66,579,850	16,414,251,948
Wiki monthly	Monthly	63,151,306	3,789,760,907
Trends hourly	Hourly	22,435	393,043,680
Trends daily	Daily	22,435	122,921,365
Trends weekly	Weekly	22,435	16,585,438
Trends monthly	Monthly	22,435	3,821,760

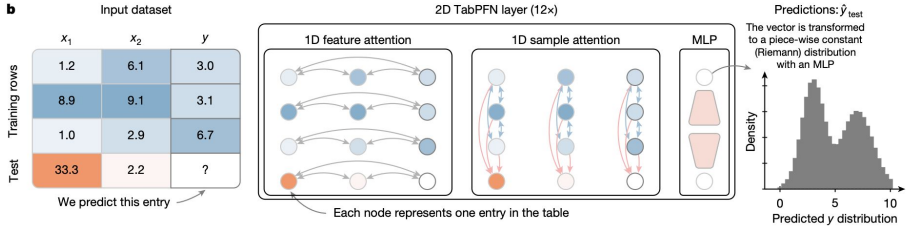


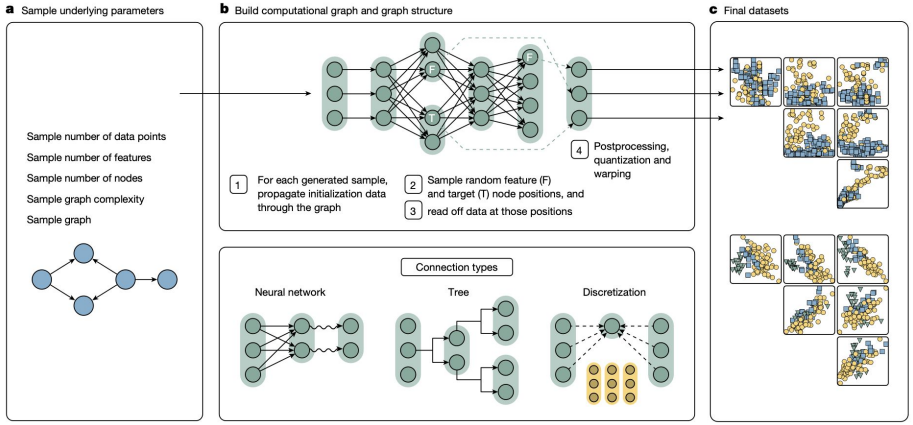
Code: https://github.com/google-research/timesfm
Paper: https://arxiv.org/abs/2310.10688v2

TabPFN (2025, 9 citations (287 для первой версии))

Идея: in-context learning для табличных данных, предобученный на реализации каузальных моделей
Архитектура: трансформер энкодер с двойным attention (feature-wise + sample-wise)
Задачи: регрессия, классификация для таблиц
Постановка: univariate with exogenous features
Данные: синтетика (130млн датасетов)







Пример генерации синтетики - DAG в виде MLP.

1. Сэмплируем ноды (слои) из графа (MLP) (количество фичей) + 1 нода для таргета - получаем наблюдаемые ноды MLP.
2. Датасет получается как выходы соответствующих нод (слоёв) с помощью форварда n (количество сэмплов) случайных векторов

Code: https://github.com/PriorLabs/TabPFN (без генерации синтетики и обучения), https://github.com/PriorLabs/TabPFN/tree/tabpfn_v1 (первая версия с кодом обучения и части синтетики). Можно пользоваться функционалом типа fit-predict.
Paper: https://www.nature.com/articles/s41586-024-08328-6

TabPFNv2 for time series

FeatureTransformer before TabPFNv2:

- running_index: порядковый номер для каждой точки в ряде
- calendar_features: фича года + добавление сезонных фичей от месяца, недели, дня, часа

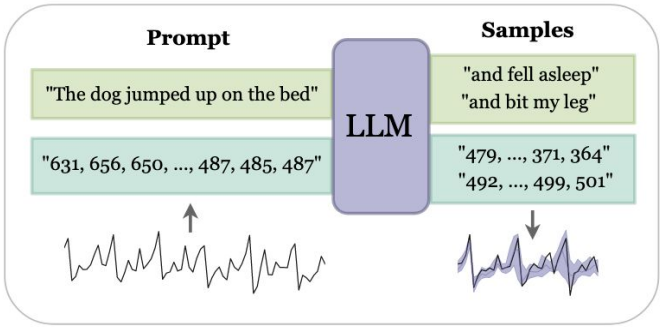
$$\sin\left(\frac{2\pi \times \text{feature}}{\text{seasonality} - 1}\right)$$

$$\cos\left(\frac{2\pi \times \text{feature}}{\text{seasonality} - 1}\right)$$

https://arxiv.org/abs/2501.02945

LLMTime (2023, 337 citations)

Идея и Архитектура: кодируем последовательность как текст и используем любую из доступных LLM.
Задачи: probabilistic forecasting
Постановка: univariate



Упрощенно, задача прогноза — это та же задача продолжения последовательности

Code: https://github.com/ngruver/llmtime
Paper: https://arxiv.org/abs/2310.07820

Как кодировать временной ряд?

Rescaling

$$x_t \to \frac{x_t - b}{a} \begin{cases} b = \min_t x_t - \beta \cdot (\max_t x_t - \min_t x_t) \\ a = a\text{-percentile}(x_1 - b, x_2 - b, \ldots, x_T - b) \end{cases}$$

Type-changing 0.123, 1.23, 12.3, 123.0 → "12,123,1230,12300"

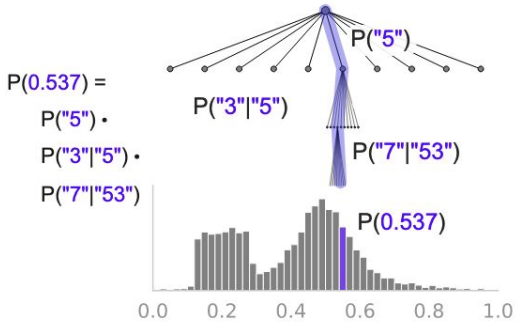
Tokenization

"151,167,...,267"
"151,167,...,267"

alpha=0.95, beta=0.3

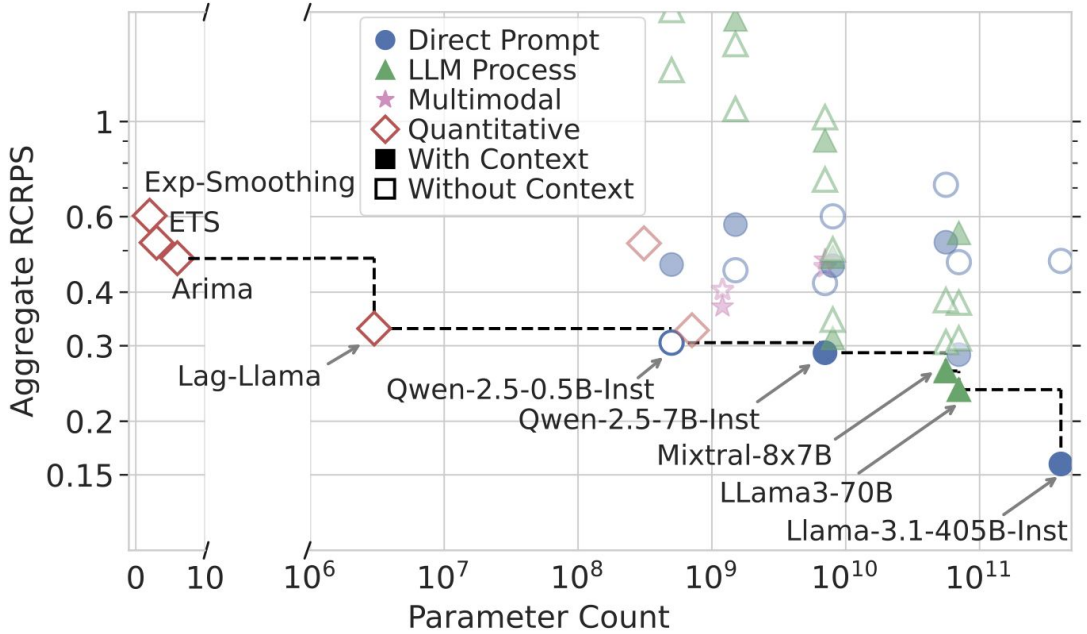
Как моделировать распределение?

Иерархический softmax — собираем вероятность числа из вероятностей входящих в него токенов. Каждая последовательность цифр соответствует интервалу ~ U.



Issue 4: LLM для forecasting

Хотите ли вы использовать 400b модель для предсказаний?



Paper: https://arxiv.org/abs/2410.18959

Задача tourism monthly

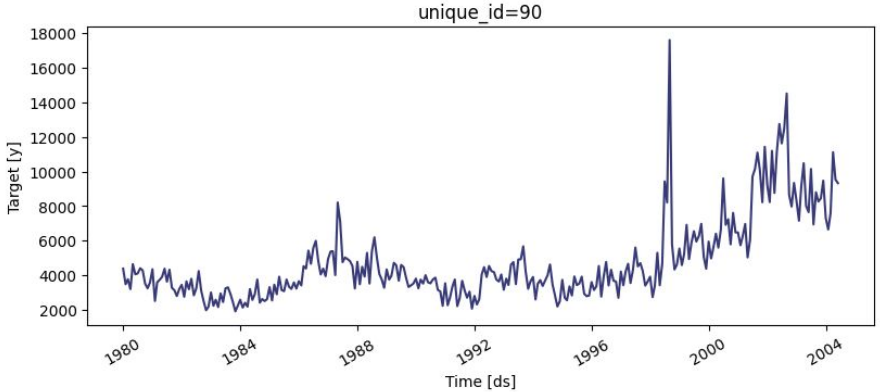
Данные: 366 помесячных рядов, некоторые не выровнены — нельзя multivariate

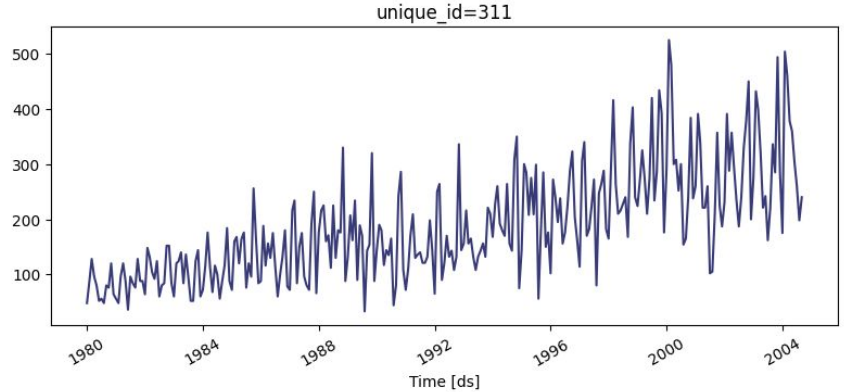
Длина: 91 - минимальная, 333 - максимальная, 298 - средняя

Постановка: горизонт 36, история 24 (supervised) или весь ряд (zero-shot)

Разбиение: тест — последние 36 точек, валидация — предпоследние 36 точек

Метрика: MAE на всех рядах на тесте, MAE на случайных 10 рядах





DEMO Kaggle

- Baseline (SeasonalNaive)
- PatchTST
- Zero-shot: Chronos Bolt Small

https://www.kaggle.com/code/simakov/ts-demo-notebook

Результаты

model	All MAE	R10 MAE
PyBoost MIMO	3874,589	2095,231
CatBoost Rec	5017,375	2451,885
DLinear 23	3377,582	1771,220
CycleNet 24	3020,283	1171,221
TimesNet 23	3009,491	1244,206
PatchTST **22**	2912,750	1146,443
GPT4TS 23	3184,911	1436,651

Disclaimer: метрики - подходы as is без тюна

Supervised:

- https://www.kaggle.com/code/simakov/tsururu-models-example
- https://www.kaggle.com/code/simakov/tsururu-models-example-timesnet
- https://www.kaggle.com/code/simakov/tsururu-models-example-cyclenet

Результаты

Disclaimer: метрики - подходы as is без тюна и подбора признаков

Supervised:

- https://www.kaggle.com/code/simakov/tsururu-models-example
- https://www.kaggle.com/code/simakov/tsururu-models-example-timesnet
- https://www.kaggle.com/code/simakov/tsururu-models-example-cyclenet

Zero-shot:

- https://www.kaggle.com/code/simakov/zero-shot-models-example
- https://www.kaggle.com/code/elineii/qwen-lllmtime/

model	All MAE	R10 MAE
PyBoost MIMO	3874,589	2095,231
CatBoost Rec	5017,375	2451,885
DLinear	3377,582	1771,220
CycleNet	3020,283	1171,221
TimesNet	3009,491	1244,206
PatchTST	2912,750	1146,443
GPT4TS	3184,911	1436,651
Chronos small	3222,315	1770,393
Chronos base	3563,093	1455,220
Chronos large	3286,125	2079,411
Chronos Bolt small	2610,310	1405,167
Chronos Bolt base	2643,248	1278,269
TimesFM	3192,440	791,175
Moirai Small	4054,879	2674,172
Moirai Base	3485,167	2724,329
Moirai Large	3342,392	2103,918
Moirai MoE small	3015,249	1926,300
TabPFNv2 TS	3178,334	1361,444
LLMTime Qwen2,5 3b	4539,963	2185,284

Issue 2: Результаты

Disclaimer: метрики - подходы as is без тюна и подбора признаков

Supervised:

- https://www.kaggle.com/code/simakov/tsururu-models-example
- https://www.kaggle.com/code/simakov/tsururu-models-example-timesnet
- https://www.kaggle.com/code/simakov/tsururu-models-example-cyclenet

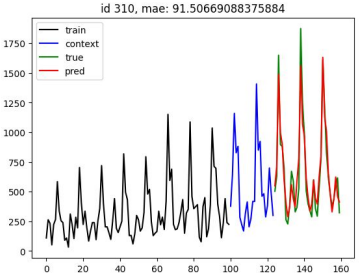
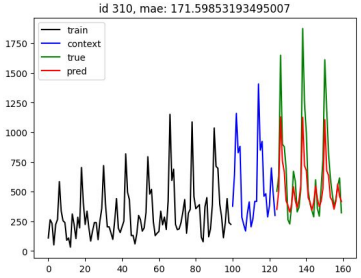
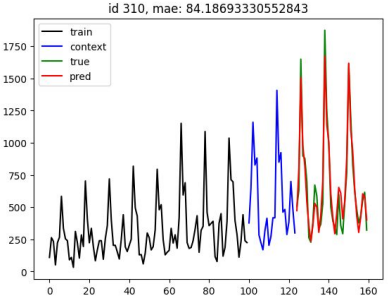
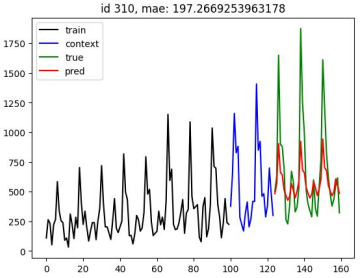
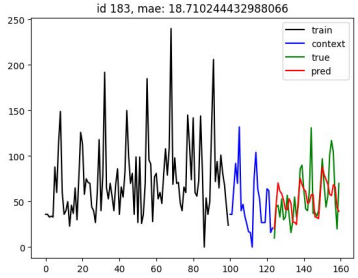
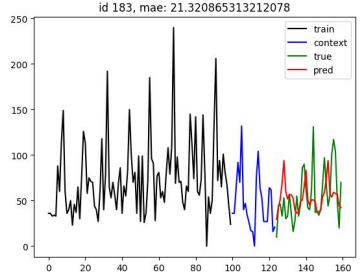
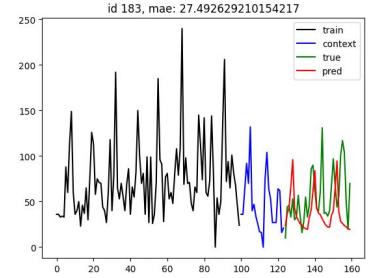
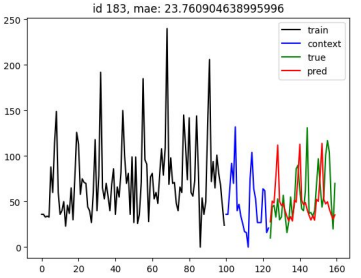
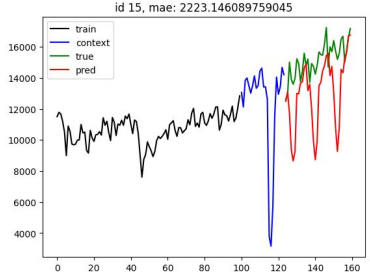
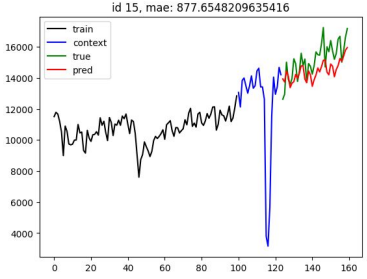
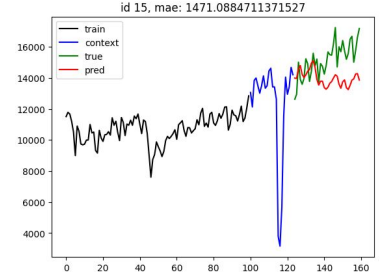
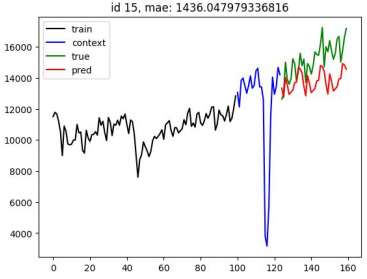
Zero-shot:

- https://www.kaggle.com/code/simakov/zero-shot-models-example
- https://www.kaggle.com/code/elineii/qwen-lllmtime/

Baselines:

- https://www.kaggle.com/code/elineii/tsururu-baselines

model	All MAE	R10 MAE
PyBoost MIMO	3874,589	2095,231
CatBoost Rec	5017,375	2451,885
DLinear	3377,582	1771,220
CycleNet	3020,283	1171,221
TimesNet	3009,491	1244,206
PatchTST	2912,750	1146,443
GPT4TS	3184,911	1436,651
Chronos small	3222,315	1770,393
Chronos base	3563,093	1455,220
Chronos large	3286,125	2079,411
Chronos Bolt small	2610,310	1405,167
Chronos Bolt base	2643,248	1278,269
TimesFM	3192,440	791,175
Moirai Small	4054,879	2674,172
Moirai Base	3485,167	2724,329
Moirai Large	3342,392	2103,918
Moirai MoE small	3015,249	1926,300
TabPFNv2 TS	3178,334	1361,444
LLMTime Qwen2,5 3b	4539,963	2185,284
Last	5615,765	3070,491
SeasonalNaive	3812,564	2211,820
AutoARIMA	2888,172	1549,064
AutoETS	2573,537	1605,357
AutoTheta	2484,090	858,028

PatchTST	TimesFM	Chronos Bolt	AutoTheta
			
			
			

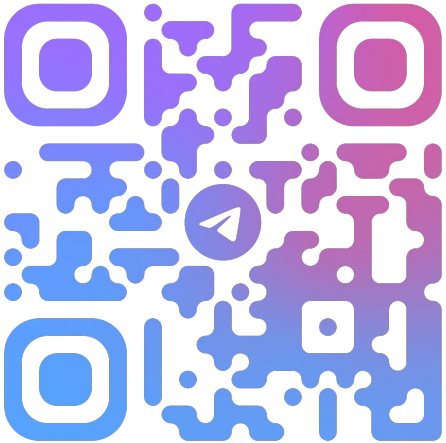
Что вы можете для себя вынести:

- Задача DS — правильно формулировать постановку ML
- Начиная задачу, не забудь сильные бейзлайны
- Статьи часто оторваны от практики, но идеи можно переиспользовать
- Zero-shot — перспективное направление, но ему еще нужно время

Выводы из экспериментов выше на другом датасете и при наличии экзогенных признаков могут измениться

Спасибо за внимание!

@SB_AI_LAB



@DMITRYSIMAKOV