Исследование влияния модульной гибридизации на точность и эффективность трансформеров для долгосрочного прогнозирования временных рядов *

Очкин Н.В.
email@email

Зубарев К.М.
email@email

Аннотация

Задача долгосрочного прогнозирования многомерных временных рядов остается сложной: в реальных данных наблюдается ряд проблемных эффектов - среди них, краткосрочные мотивы (скачки, ступени), крайне долгосрочные и межрядовые зависимости, выраженная нестационарность (сдвиги тренда/уровня, мультисезонность, гетероскедастичность). В данной работе мы предлагаем расширить слой эмбеддинга в модели Informer [14] компактным двухслойным сверточным блоком с целью повышения эффективности извлечения локальных паттернов, внедрить модуль декомпозиции ряда из Autoformer [12] для явного разделения трендовых и сезонных компонент и заменить механизм ProbSparse на линейное внимание FAVOR+ (Performer [2]) для учёта глобальных зависимостей при низких вычислительных затратах. Эксперименты на стандартных бенчмарках демонстрируют снижение ошибок на длинных горизонтах прогнозирования в ряде настроек. Абляционные исследования указывают на вклад каждого из модулей, хотя эффект не универсален для всех горизонтов. На рассмотренных конфигурациях наблюдается благоприятный компромисс между точностью и затратами времени/памяти.

Keywords LSTF · Long Sequence Time-series Forecasting · Transformer · Informer · Performer · Autoformer

1 Введение

Долгосрочное прогнозирование временных рядов является одной из ключевых задач во многих областях - от энергетики и финансов до транспорта и здравоохранения. Однако реальные многомерные ряды редко обладают простой структурой: среди прочего, они сочетают краткосрочные локальные паттерны, очень длинные и межрядовые зависимости, а также выраженную нестационарность, включая сдвиги тренда, мультисезонность и гетероскедастичность. Эти свойства затрудняют обучение моделей и ухудшают их способность к экстраполяции на длительных горизонтах.

С момента публикации, архитектура Трансформера [11] завоевала широкое признание. Однако у нее есть несколько серьезных проблем, которые усложняют работу с длинными временными последовательностями (LSTF). Последующие исследования предложили различные методы решения данных и связанных с ними проблем (Informer [14], Autoformer [12], Performer [2]). Тем не менее, существующие архитектуры сталкиваются с рядом ограничений: необходимость одновременного учёта локальных и глобальных закономерностей, высокая чувствительность к нестационарности и ограниченная масштабируемость по длине входной последовательности.

*Исходный код, конфигурации и скрипты обучения доступны: https://github.com/namenick91/Convformer

В данной работе мы предлагаем архитектуру, которая объединяет три взаимодополняющих индуктивных смещения, специально ориентированных на решение этих вызовов: **(i)** расширенный сверточный входной блок, способный эффективно кодировать краткосрочные локальные паттерны (скачки, ступени, импульсные всплески). Такой модуль выполняет роль фильтра низкого уровня и одновременно стабилизирует статистики входных данных, что повышает устойчивость модели к локальной нестационарности; **(ii)** механизм внимания FAVOR+ (Performer [2]), обеспечивающий линейные $O(Lr)$ затраты времени и памяти; параметр ранга r задаёт настраиваемый компромисс между скоростью и точностью. Это позволяет обрабатывать очень длинные последовательности и улавливать отложенные зависимости без взрыва затрат, что критично для прогнозов на больших горизонтах. **(iii)** явная декомпозиция ряда в стиле Autoformer [12] после каждого блока self-attention: низкочастотный тренд выводится через остаточные соединения, а очищенный от тренда стационаризированный остаток подаётся в механизм внимания. Такой приём снижает влияние сдвигов уровня и многосезонности, улучшая обобщающую способность модели.

Такая композиция локальной фильтрации, глобального внимания и явной декомпозиции улучшает точность прогнозирования на длинных горизонтах, ускоряет и стабилизирует оптимизацию, а также снижает затраты памяти по сравнению с базовыми трансформер-подходами. Основной вклад данной работы можно резюмировать следующим образом:

- Вводим расширенный сверточный входной блок для кодирования краткосрочных мотивов и стабилизации статистик нестационарных входов.
- Для моделирования глобальных связей применяем FAVOR+ внимание с линейными затратами по времени и памяти, что обеспечивает масштабируемое и не зависящее от распределения моделирование глобальных зависимостей;
- Встраиваем Autoformer-подобную [12] декомпозицию после каждого блока self-attention, которая пропускает низкочастотный тренд по остаточным соединениям и передаёт более стационаризированный поток данных в механизм внимания.
- Эксперименты на стандартных бенчмарках демонстрируют устойчивое снижение ошибок на длительных горизонтах прогнозирования; абляционные исследования подтверждают вклад каждого модуля, а анализ масштабируемости показывает благоприятный баланс между точностью и вычислительными затратами.

2 Методология

Повышение эффективности извлечения локальных паттернов Для усиления способности модели к распознаванию краткосрочных закономерностей мы предлагаем заменить стандартный механизм представления значений (TokenEmbedding) на сверточный блок ConvStem. Данный блок сочетает проекцию входных признаков через точечную свёртку с последующими операциями широкой и глубинной свёрток, дополненных нормализацией и нелинейностью. Такое построение позволяет модели фиксировать повторяющиеся локальные мотивы и вариации формы сигналов непосредственно на этапе встраивания, ещё до применения механизмов внимания. В результате глобальное внимание может быть сфокусировано преимущественно на долговременных зависимостях, тогда как локальная динамика эффективно извлекается специализированным сверточным модулем.

Масштабируемое моделирование глобальных зависимостей В качестве механизма внимания предлагается использовать FAVOR+ (Performer [2]) вместо ProbSparse (оригинально применяемого в Informer [14]). Подобно тому, как в оригинальном Informer [14] механизм FullAttention [11] был заменен на ProbSparse исключительно для вычисления self-attention (механизм самовнимания) в слоях кодировщика и декодера, в нашей модификации ProbSparse заменяется на FAVOR+ в тех же местах, тогда как cross-attention останется реализованным через полное внимание (FullAttention). Данную замену мы мотивируем тем, что FAVOR+ обеспечивает несмещённую аппроксимацию softmax-ядра с линейной по времени и памяти сложностью [2], что гарантирует предсказуемую и масштабируемую работу на очень длинных последовательностях в условиях ограниченных ресурсов GPU. При этом точность аппроксимации управляется числом случайных признаков r: при меньших значениях достигается высокая скорость, при больших - более точное восстановление распределения внимания. Такая настраиваемость делает механизм универсальным инструментом, позволяющим варьировать баланс

между эффективностью и качеством. Кроме того, отказ от ручной выборки top-u запросов позволяет модели учитывать любые глобальные зависимости, а не только наиболее сильные периодические, что критично для учёта непериодичных скачков в реальных временных рядах.

Явное разделение тренда и сезонности Чтобы явно разделить тренд и сезонную компоненту, мы интегрировали механизм декомпозиции временных рядов, заимствованный из архитектуры Autoformer [12]. После каждого блока self-attention скрытое представление разлагается на два канала: низкочастотный тренд и остаточную компоненту, содержащую более стационарные сезонные и случайные колебания. Тренд передаётся по остаточному пути, обеспечивая стабильность прогнозов при сдвигах уровня, тогда как внимание применяется к сезонной компоненте, где наиболее выражены периодические и локальные закономерности. Такая стратегия снижает вариативность обучения, повышает способность к экстраполяции за пределами обучающего диапазона и улучшает интерпретируемость результатов за счёт явного выделения трендовой и сезонной составляющих.

Схема механизма декомпозиции ряда [12]

Input: X

(1) [SeriesDecomp] $\rightarrow$ Trend, Seasonal

$$\text{where} \quad \begin{aligned} \text{Trend} &= \text{AvgPool}(\text{Padding}(\text{X})) \\ \text{Seasonal} &= \text{X} - \text{Trend} \end{aligned}$$

(2) [self-attention mechanism] on Seasonal
$\rightarrow$ Seasonal$'$ (+ residual connection, normalization, etc.)

(3) [Feed-Forward] on Seasonal$'$ $\rightarrow \widetilde{\text{Seasonal}}$

(4) Re-compose $\text{X} = \widetilde{\text{Seasonal}} + \text{Trend}$

В итоге архитектура объединяет сильные стороны трёх подходов: локальное кодирование паттернов через ConvStem, линейное глобальное внимание FAVOR+ и явную декомпозицию ряда из Autoformer [12], сохраняя при этом полную совместимость с остальными гиперпараметрами оригинального Informer [14].

3 Эксперимент

Датасет Ниже приводится описание набора данных экспериментов: *ETT* [14] датасет содержит информацию, собранную с электрических трансформаторов, включая нагрузку и температуру масла, которые регистрировались каждые 15 минут в период с июля 2016 года по июль 2018 года.

Детали реализации Обучение проводилось с использованием функции потерь MSE (L2) и оптимизатора Adam [8] с начальной скоростью обучения 10^{-4}. Размер батча - 32. Процесс обучения досрочно останавливается в пределах 10 эпох. Результаты экспериментов были получены в ходе усреднения по трем отдельным запускам, реализованы в PyTorch [10] и проводились на одной графической карте NVIDIA GTX 1660 SUPER с 6 ГБ видеопамяти. Архитектура модели включает 2 слоя энкодера и 1 слой декодера. Полный перечень гиперпараметров и конфигураций приведён в репозитории: https://github.com/namenick91/Convformer

3.1 Абляционные исследования

ConvStem вместо TokenEmbedding Сравнивалась эффективность оригинальной модели Informer [14] и её модификации с расширенным слоем эмбеддинга 2.

Встраивание компактного сверточного блока на вход приводит к заметному улучшению качества на коротких (24-48) и особенно длинных горизонтах (336), при этом затраты по времени остаются сопоставимыми. Однако на отдельных горизонтах (например, 168) Informer [14] сохраняет преимущество. Это подтверждает, что сверточная фильтрация локальных паттернов в среднем повышает устойчивость к краткосрочной нестационарности, хотя её вклад не универсален.

	Horizon	**Convformer**			Informer			Performer			Autoformer		
		MSE	MAE	t	MSE	MAE	t	MSE	MAE	t	MSE	MAE	t
ETTh1	24	**0.388**	0.428	3m50s/4s	0.524	0.527	**2m5s/3s**	0.598	0.570	2m50s/4s	0.401	**0.425**	3m58s/7s
	48	0.435	0.451	3m55s/4s	0.631	0.601	**2m24s/4s**	0.765	0.673	2m35s/4s	**0.430**	**0.445**	4m43s/7s
	168	**0.435**	**0.459**	8m39s/6s	0.825	0.705	**3m14s/5s**	0.918	0.768	3m42s/6s	0.478	0.473	6m45s/11s
	336	**0.469**	**0.490**	7m55s/8s	1.310	0.937	9m19s/6s	1.024	0.823	**5m36s/7s**	0.516	0.497	10m19s/17s
	720	**0.510**	**0.528**	**12m51s/10s**	1.205	0.879	14m36s/9s	1.107	0.843	14m9s/9s	-	-	-
ETTh2	24	**0.248**	**0.345**	2m38s/4s	1.284	0.891	**1m58s/3s**	1.296	0.907	2m49s/4s	0.290	0.365	4m7s/6s
	48	**0.298**	**0.373**	2m49s/5s	1.559	1.008	**1m58s/4s**	1.568	1.008	2m35s/4s	0.324	0.382	3m31s/6s
	168	0.539	0.509	4m6s/6s	7.587	2.335	**2m49s/5s**	8.487	2.569	3m40s/6s	**0.451**	**0.456**	5m15s/9s
	336	0.684	0.600	7m9s/8s	4.369	1.773	**4m19s/7s**	8.158	2.456	6m16s/7s	**0.478**	**0.480**	7m3s/13s
	720	**0.662**	**0.595**	13m1s/11s	2.977	1.467	**6m1s/9s**	3.707	1.640	7m39s/9s	-	-	-

Таблица 1: Результаты многомерных предсказаний на 2 датасетах ETT с горизонтами предсказаний: { 24, 48, 168, 336, 720 }. Мы фиксируем входную длину последовательностей у моделей как 96. Символ "–" обозначает выход за пределы памяти (OOM). Время указано как train/inference. Полный бенчмарк см. в табл. 5. Примеры визуализации приведены на рис. B.1.

Horizon	Informer			**ConvStem**		
	MSE	MAE	t	MSE	MAE	t
24	0.524	0.527	**2m5s/3s**	**0.469**	**0.480**	2m11s/4s
48	0.631	0.601	2m24s/4s	**0.587**	**0.558**	**2m23s/4s**
168	**0.825**	**0.705**	**3m14s/5s**	0.861	0.733	3m34s/5s
336	1.310	0.937	9m19s/6s	**1.094**	**0.843**	**7m22s/7s**
720	**1.205**	**0.879**	14m36s/9s	1.261	0.913	**9m58s/9s**

Таблица 2: Результаты многомерных предсказаний на датасете ETTh1 с горизонтами предсказаний: { 24, 48, 168, 336, 720 }. Входная длина последовательности фиксирована: 96. Время указано как train/inference. Примеры визуализации приведены в табл. 6.

ProbSparse → FAVOR+ Сравнивалась эффективность Informer [14] с исходным механизмом ProbSparse и модифицированной версии, в которой ProbSparse заменён на FAVOR+ (Performer [2]).

Horizon	Informer			**Informer w/ FAVOR+**		
	MSE	MAE	t	MSE	MAE	t
24	0.524	0.527	**2m5s/3s**	**0.494**	**0.505**	2m30s/4s
48	**0.631**	**0.601**	2m24s/4s	0.710	0.640	**2m20s/4s**
168	**0.825**	**0.705**	**3m14s/5s**	0.864	0.746	3m24s/5s
336	1.310	0.937	9m19s/6s	**1.088**	**0.846**	**6m22s/7s**
720	1.205	0.879	14m36s/9s	**1.065**	**0.831**	**9m54s/9s**

Таблица 3: Результаты многомерных предсказаний на датасете ETTh1 с горизонтами предсказаний: { 24, 48, 168, 336, 720 }. Входная длина последовательности фиксирована: 96. Время указано как train/inference. Примеры визуализации приведены в табл. 7.

Линейное внимание обеспечивает выигрыш по качеству на длинных горизонтах (336, 720) при сопоставимых или меньших вычислительных затратах, в то время как на коротких горизонтах улучшение выражено слабее или отсутствует. Таким образом, FAVOR+ [2] даёт наибольший эффект именно в режимах, где глобальные зависимости становятся критичными, подтверждая его ценность для масштабируемого долгосрочного прогнозирования.

Informer с модулем декомпозиции Сравнивалась эффективность оригинальной модели Informer [14] и модификации, дополненной механизмом декомпозиции ряда из Autoformer [12].

Horizon		Informer			**Informer w/ s.decomp**		
		MSE	MAE	t	MSE	MAE	t
ETTh1	24	0.524	0.527	**2m5s/3s**	**0.478**	**0.494**	3m36s/4s
	48	0.631	0.601	**2m24s/4s**	**0.561**	**0.547**	3m4s/4s
	168	0.825	0.705	**3m14s/5s**	**0.649**	**0.597**	7m2s/5s
	336	1.310	0.937	**9m19s/6s**	**0.952**	**0.748**	10m55s/7s
	720	1.205	0.879	**14m36s/9s**	**1.059**	**0.775**	16m46s/10s

Таблица 4: Результаты многомерных предсказаний на датасете ETTh1 с горизонтами предсказаний: { 24, 48, 168, 336, 720 }. Входная длина последовательности фиксирована: 96. Время указано как train/inference. Примеры визуализации приведены в табл. 8.

Встраивание операции разделения на тренд и остаток после каждого блока self-attention приводит к систематическому снижению ошибок на всех горизонтах ETTh1, хотя ценой становится рост времени обучения и инференса (inference). Это подтверждает гипотезу о том, что явная стабилизация нестационарности улучшает обобщающую способность модели, особенно при наличии сдвигов уровня и мультисезонности.

4 Заключение

Предложенная комбинация ConvStem + FAVOR+ + Autoformer-подобная декомпозиция демонстрирует улучшение качества на длинных горизонтах при сопоставимых вычислительных затратах на рассматриваемых настройках. Абляционные исследования подтверждают вклад модулей на большинстве горизонтов: ConvStem - выделение локальных мотивов; FAVOR+ - масштабируемые дальние зависимости; декомпозиция - устойчивость к тренду/сезонности. На ETT наибольшие выигрыши достигаются при длительных горизонтах (336-720), тогда как на средних горизонтах Autoformer остается сопоставимым.

При этом исследование имеет ряд ограничений: в качестве бенчмарка был рассмотрен один набор данных (ETT), cross-attention сохраняет квадратичную сложность, чувствительность MAPE/MSPE к масштабам признаков, отсутствие оценки неопределённости. Перспективными направлениями являются линеаризация cross-attention, адаптивный выбор ранга r в FAVOR+, расширение набора датасетов, включение калибровки и методов, устойчивых к дрейфу распределений.

Список использованных источников

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In International Conference on Learning Representations (ICLR), 2015.

[2] Krzysztof Choromanski, Valentin Likhosherstov, David Dohan, Xingyou Song, Alex Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Łukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. Rethinking attention with performers. In Proceedings of the International Conference on Learning Representations (ICLR), 2020. arXiv:2009.14794.

[3] Gregory C. Reinsel George E. P. Box, Gwilym M. Jenkins and Greta M. Ljung. Time Series Analysis: Forecasting and Control. Wiley, fifth edition edition, 2015.

[4] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep Learning. MIT Press, 2016. `http://www.deeplearningbook.org`.

[5] James D. Hamilton. Time Series Analysis. Princeton University Press, 1994.

[6] Rob J. Hyndman and George Athanasopoulos. Forecasting: principles and practice. OTexts, 2013.

[7] Brockwell P. J. and Davis R. A. Introduction to Time Series and Forecasting. Springer, third edition edition, 2016.

[8] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. International Conference on Learning Representations (ICLR), 2015.

[9] Kevin P. Murphy. Probabilistic Machine Learning: An introduction. MIT Press, 2022.

[10] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Advances in Neural Information Processing Systems (NeurIPS), 2019.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Advances in Neural Information Processing Systems (NeurIPS), volume 30, 2017.

[12] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with Auto-Correlation for long-term series forecasting. In Advances in Neural Information Processing Systems, 2021.

[13] Haoyi Zhou, Jianxin Li, Shanghang Zhang, Shuai Zhang, Mengyi Yan, and Hui Xiong. Expanding the prediction capacity in long sequence time-series forecasting. Artificial Intelligence, 318:103886, 2023.

[14] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In The Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Virtual Conference, volume 35, pages 11106–11115. AAAI Press, 2021.

A Полный бенчмарк

Horizon		**Convformer**						Informer					
		MSE	MAE	RMSE	MAPE	MSPE	t	MSE	MAE	RMSE	MAPE	MSPE	t
ETTh1	24	**0.388**	0.428	**0.622**	11.030	50292.405	3m50s/4s	0.524	0.527	0.724	11.117	**43860.164**	**2m5s/3s**
	48	0.435	0.451	0.660	10.864	49359.185	3m55s/4s	0.631	0.601	0.794	11.973	50739.564	**2m24s/4s**
	168	**0.435**	**0.459**	**0.659**	11.585	48605.939	8m39s/6s	0.825	0.705	0.908	9.711	31104.102	**3m14s/5s**
	336	**0.469**	**0.490**	**0.685**	12.928	62006.074	7m55s/8s	1.310	0.937	1.143	17.267	108984.891	9m19s/6s
	720	**0.510**	**0.528**	**0.714**	**12.617**	**56880.186**	**12m51s/10s**	1.205	0.879	1.098	17.282	111811.141	14m36s/9s
ETTh2	24	**0.248**	**0.345**	**0.498**	**1.511**	**282.651**	2m38s/4s	1.284	0.891	1.133	8.751	4080.943	**1m58s/3s**
	48	**0.298**	**0.373**	**0.546**	1.569	**244.634**	2m49s/5s	1.559	1.008	1.247	9.655	4751.785	**1m58s/4s**
	168	0.539	0.509	0.733	2.148	**315.627**	4m6s/6s	7.587	2.335	2.754	13.709	9135.792	**2m49s/5s**
	336	0.684	0.600	0.826	2.552	646.271	7m9s/8s	4.369	1.773	2.088	10.857	8606.445	**4m19s/7s**
	720	**0.662**	**0.595**	**0.811**	**2.721**	**630.013**	13m1s/11s	2.977	1.467	1.725	11.613	8735.043	**6m1s/9s**
Horizon		Performer						Autoformer					
		MSE	MAE	RMSE	MAPE	MSPE	t	MSE	MAE	RMSE	MAPE	MSPE	t
ETTh1	24	0.598	0.570	0.773	13.036	59454.181	2m50s/4s	0.401	**0.425**	0.633	**10.876**	47465.651	3m58s/7s
	48	0.765	0.673	0.873	**9.637**	**27418.091**	2m35s/4s	**0.430**	**0.445**	**0.656**	10.304	46080.775	4m43s/7s
	168	0.918	0.768	0.957	**9.107**	**22036.375**	3m42s/6s	0.478	0.473	0.691	12.045	54136.419	6m45s/11s
	336	1.024	0.823	1.011	**11.928**	**42467.367**	**5m36s/7s**	0.516	0.497	0.717	12.388	57259.240	10m19s/17s
	720	1.107	0.843	1.052	20.032	149892.094	14m9s/9s	-	-	-	-	-	-
ETTh2	24	1.296	0.907	1.130	8.613	4234.640	2m49s/4s	0.290	0.365	0.539	1.539	298.584	4m7s/6s
	48	1.568	1.008	1.252	9.667	4570.592	2m35s/4s	0.324	0.382	0.569	**1.509**	261.721	3m31s/6s
	168	8.487	2.569	2.904	14.583	13909.365	3m40s/6s	**0.451**	**0.456**	**0.672**	**1.946**	370.939	5m15s/9s
	336	8.158	2.456	2.852	11.938	9109.486	6m16s/7s	**0.478**	**0.480**	**0.691**	**2.045**	**478.931**	7m3s/13s
	720	3.707	1.640	1.925	11.051	10487.951	7m39s/9s	-	-	-	-	-	-

Таблица 5: Результаты многомерных предсказаний на 2 датасетах ETT с горизонтами предсказаний {24, 48, 168, 336, 720}. Входная длина последовательности фиксирована: 96. Символ "–" обозначает выход за пределы памяти (OOM). Время указано как train/inference.

B Дополнение к основным результатам

B.1 Визуализация абляционных исследований

Для оценки предсказаний различных моделей приводится визуализация результатов прогнозирования на тестовой выборке набора данных ETTh1 в рамках абляционных исследований, что позволяет провести качественное сравнение.

Таблица 6: Примеры прогнозирования из набора данных ETTh1 в ходе абляционных исследований *ConvStem вместо TokenEmbedding* (см. раздел 3.1) в настройках `input-96-predict-horizon`. Синие линии - истинные значения, оранжевые линии - предсказания модели, зеленые - входные данные длиной 96.

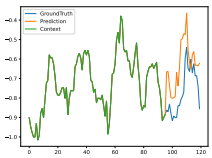
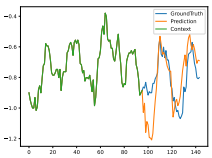
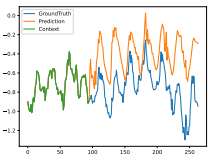
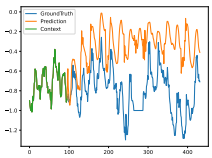
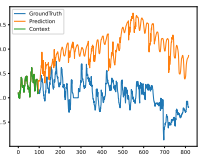
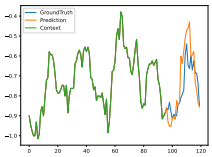
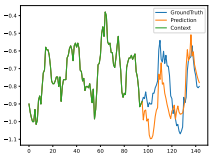
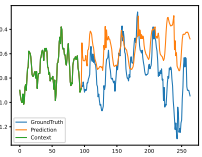
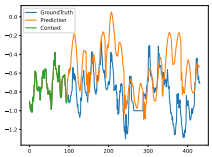
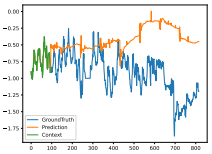
Model	Horizon				
	24	48	168	336	720
Informer					
ConvStem					

Таблица 7: Примеры прогнозирования из набора данных ETTh1 в ходе абляционных исследований *ProbSparse* $\rightarrow$ *FAVOR+* (см. раздел 3.1) в настройках `input-96-predict-horizon`.

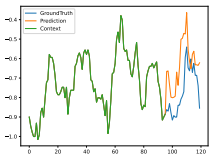
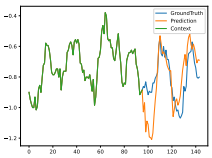
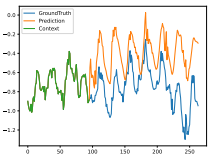
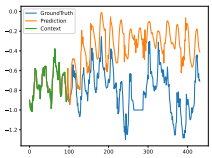
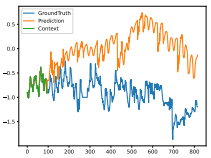
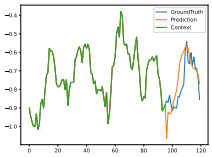
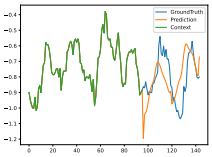
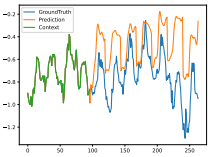
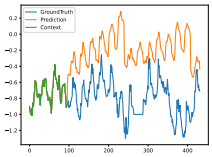
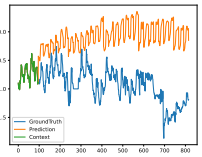
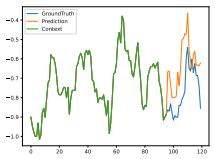
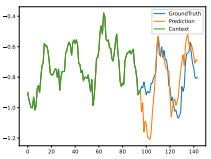
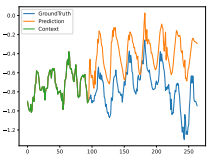
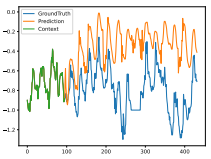
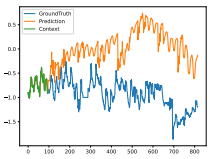
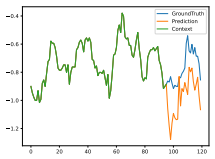
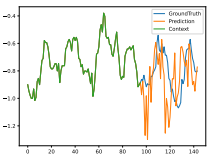
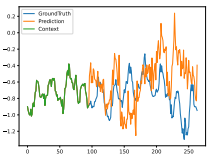
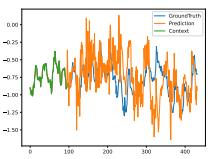
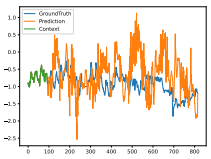
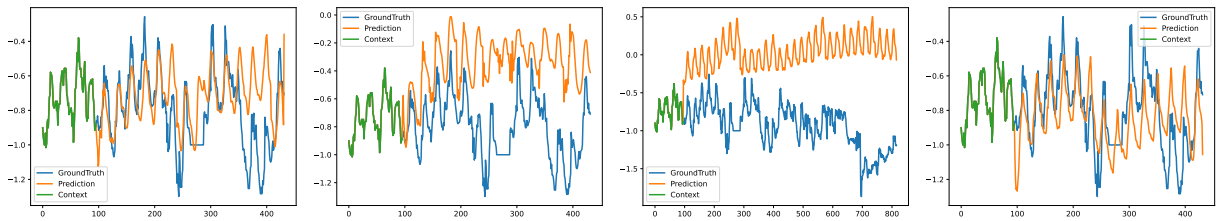
Model	Horizon				
	24	48	168	336	720
Informer					
w/ FAVOR+					

Таблица 8: Примеры прогнозирования из набора данных ETTh1 в ходе абляционных исследований *Informer с модулем декомпозиции* (см. раздел 3.1) в настройках `input-96-predict-horizon`.

Model	Horizon				
	24	48	168	336	720
Informer					
w/ s.decomp					

B.2 Визуализация основных результатов



Convformer Informer Performer Autoformer

Рис. B.1: Примеры прогнозирования из набора данных ETTh1 в настройках `input-96-predict-336`.