

Применение моделей трансформеров для прогнозирования цен акций.

Введение

Прогнозирование цен акций – сложная задача вследствие высокой волатильности и шумов финансовых рядов. Традиционные подходы (ARIMA, регрессионные модели) и классические нейросети (например, LSTM) имеют ограничения в улавливании сложных нелинейных зависимостей и долгосрочных трендов ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)) ([Predicting Stock Market Trends with Data Science: Part 4 — ARIMA ...](#)). В последние годы всё больше внимания уделяется моделям типа Transformer, добившимся успехов в обработке последовательностей в NLP и других областях, и их адаптации для временных рядов. За последние ~2 года (2022–2024) появилось множество исследований, применяющих трансформеры к финансовым данным, в том числе для прогнозирования цен акций. В данном отчёте проводится анализ методов трансформеров в контексте фондового рынка, обзор актуальной литературы (с упором на модель **MASTER: Market-Guided Stock Transformer** и ряд других работ), обсуждаются технические аспекты (вычислительные требования, данные, проблемы), сравнивается эффективность трансформеров с традиционными методами (ARIMA, LSTM, XGBoost и др.), а также формулируются выводы о текущем состоянии исследований и рекомендации по дальнейшему развитию.

Обзор литературы

Применение трансформеров в финансовых исследованиях. В 2022–2024 гг. опубликован ряд работ, демонстрирующих потенциал трансформеров в прогнозировании акций. Ранние эксперименты показывали, что базовый Transformer способен уловить сложные зависимости и иногда превосходит классические методы. Так, Wang и соавт. (2022) применили «глубокий» трансформер для прогнозирования фондовых индексов и получили точность, **существенно превосходящую традиционные методологии** ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Muhammad и соавт. (2022) первыми протестировали трансформер на данных рынка Бангладеш (Dhaka Stock Exchange), используя технику **Time2Vec** для представления временных признаков, что позволило эффективно адаптировать модель к финансовым рядам ([\[2208.08300\] Transformer-Based Deep Learning Model for Stock Price Prediction: A Case Study on Bangladesh Stock Market](#)). В их экспериментах трансформер успешно

предсказывал движения цен восьми акций с приемлемой точностью (RMSE), демонстрируя применимость даже на развивающемся рынке ([\[2208.08300\] Transformer-Based Deep Learning Model for Stock Price Prediction: A Case Study on Bangladesh Stock Market](#)) ([\[2208.08300\] Transformer-Based Deep Learning Model for Stock Price Prediction: A Case Study on Bangladesh Stock Market](#)).

Интеграция текстовых и новостных данных. Одно из направлений – комбинирование трансформеров с анализом новостей и соцмедиа. Например, Zhang и соавт. (2022) **интегрировали тексты из соцсетей** (финансовые новости, твиты) в архитектуру на основе Transformer, предложив модель с механизмом внимания для прогнозирования тренда акций ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Их **Transformer-encoder** сеть учитывает настроение новостей, что улучшает прогноз движения акций по сравнению с моделью, использующей только ценовые ряды. Подобные подходы показывают, что многофакторные модели (цены + новости) на базе трансформеров способны улавливать рыночные настроения и давать более точные прогнозы направления рынка ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)).

Специализированные архитектуры для акций (MASTER). Отдельного внимания заслуживает работа Tong Li и др. – **MASTER: Market-Guided Stock Transformer** (AAAI 2024). Эта модель специально спроектирована для совместного прогнозирования группы акций с учётом их взаимодействий ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)) ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)). В отличие от предыдущих подходов, где сначала обучаются индивидуальные модели на каждую акцию, а затем их признаки смешиваются, MASTER сразу **моделирует взаимосвязи между акциями** на разных временных масштабах. Авторы отмечают, что корреляции между акциями могут быть **мгновенными и межвременными**, а значимость различных признаков меняется в зависимости от рыночных условий ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)). Чтобы учесть это, MASTER чередует два вида механизмов внимания: **внутри каждой акции (intra-stock)** – для выявления ее временных паттернов, и **между акциями (inter-stock)** – для выявления корреляций между разными акциями ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)). Дополнительно вводится “market-guided” механизм – использование агрегированной рыночной информации для **автоматического выбора признаков**,

наиболее релевантных текущей фазе рынка ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)). Эксперименты показали превосходство MASTER над предшественниками, а визуализация весов внимания выявила осмысленные экономические зависимости между акциями ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)). Авторы выложили открытый код модели ([GitHub - SJTU-DMTai/MASTER: This is the official code and supplementary materials for our AAAI-2024 paper: MASTER: Market-Guided Stock Transformer for Stock Price Forecasting. MASTER is a stock transformer for stock price forecasting, which models the momentary and cross-time stock correlation and guide feature selection with market information.](#)), что позволяет воспроизвести результаты и использовать MASTER в прикладных задачах. MASTER демонстрирует, как включение знаний о рынке и структурирование внимания помогает трансформеру лучше предсказывать цены акций.

Гибридные модели (CNN+Transformer). Другая группа работ сосредоточена на сочетании трансформеров с свёрточными нейросетями для улучшения извлечения локальных паттернов. Xie и соавт. (2024) предложили архитектуру **DCT (Deep Convolutional Transformer)** – гибрид CNN и трансформера для прогнозирования направлений движения акций ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)) ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). В DCT входной ряд сначала проходит через специальный блок **Inception-Conv** (несколько сверточных фильтров разного масштаба), превращающий ценовые ряды в “токены”, содержащие как глобальные, так и локальные признаки ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Далее применяются **разделённые полносвязные слои** для повышения выразительности признаков при меньшей сложности, и стандартный multi-head self-attention для улавливания взаимосвязей во времени ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)) ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). За счёт CNN-компоненты модель лучше улавливает краткосрочные колебания и шумы рынка, с которыми чистый Transformer справляется хуже ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)) ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Авторы показали на индексах NASDAQ, HSI, SSE, что DCT даёт наименьшие ошибки (MAE, MSE, MAPE) среди всех сравниваемых моделей и самую высокую MCC (коэф. Маттьюса) для задачи классификации движения ([Deep](#)

[Convolutional Transformer Network for Stock Movement Prediction](#)) ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Похожий подход предложили Ying Li и соавт. (2024) – их модель **CNN-Trans-SPP** использует сверточный блок для локальных признаков и компактный Transformer-энкодер для временных зависимостей ([CNN-Trans-SPP: A small Transformer with CNN for stock price prediction](#)). В эксперименте на акциях из разных секторов (финансы, технологии, промышленность, сельское хоз.) их гибрид смог **сократить ошибку прогноза на 32–45%** по сравнению с LSTM, attention-LSTM и даже чистым Transformer ([CNN-Trans-SPP: A small Transformer with CNN for stock price prediction](#)). Это подтверждает, что комбинация трансформера с CNN существенно повышает точность на финансовых данных.

Графовые и дифференциальные трансформеры. Для учёта структурных отношений между акциями (например, принадлежность к сектору, корреляции доходностей) используются **графовые нейросети с механизмом внимания**. Свежий пример – модель **Differential Graph Transformer (DGT)** (Reddy et al., 2025), совмещающая **темпоральное внимание** по временной оси с **графовым вниманием** по связям между акциями ([Stock Market Forecasting with Differential Graph Transformer: A Novel Approach to Temporal and Spatial Stock Data Analysis | Journal of Information Systems Engineering and Management](#)). DGT вычисляет глобальные и локальные матрицы корреляций акций (например, на основе взаимной информации и коэффициентов Пирсона) и использует их как весовые коэффициенты для межакционного внимания ([Stock Market Forecasting with Differential Graph Transformer: A Novel Approach to Temporal and Spatial Stock Data Analysis | Journal of Information Systems Engineering and Management](#)). В результате модель эффективно захватывает как **долгосрочные тренды**, так и **краткосрочные межакционные зависимости** ([Stock Market Forecasting with Differential Graph Transformer: A Novel Approach to Temporal and Spatial Stock Data Analysis | Journal of Information Systems Engineering and Management](#)). На данных S&P500 DGT превзошёл базовую модель GRU: RMSE ниже на ~13.5%, MAE – на 12.2% ([Stock Market Forecasting with Differential Graph Transformer: A Novel Approach to Temporal and Spatial Stock Data Analysis | Journal of Information Systems Engineering and Management](#)). Это заметное улучшение, показывающее силу **дифференциального внимания** для моделирования отношений между акциями.

Отметим также работу Szydłowski & Chudziak (2024), которые проанализировали новый вариант трансформера **Hidformer** применительно к финансовым рядам ([\[2412.19932\] Hidformer: Transformer-Style Neural Network in Stock Price Forecasting](#)) ([\[2412.19932\] Hidformer: Transformer-Style Neural Network in Stock Price Forecasting](#)). Их результаты подтверждают **перспективность трансформеров** для алгоритмического трейдинга и приводят дополнительные инсайты по улучшению точности предсказаний за счёт модификации базовой архитектуры. Ещё одно направление – **Temporal Fusion Transformer (TFT)**, изначально предложенный для многофакторных временных рядов. Islam et al. (2023) протестировали TFT на данных индонезийского рынка с историческими техническими индикаторами и показали, что TFT хорошо захватывает краткосрочные паттерны, **превосходя LSTM и классический Transformer** по точности выделения локальных пиков и переходных фаз ([Temporal Fusion Transformers for Enhanced Multivariate Time Series Forecasting of Indonesian Stock Prices](#)) ([Temporal Fusion Transformers for Enhanced Multivariate Time Series Forecasting of Indonesian Stock Prices](#)). TFT объединяет рекуррентный seq2seq и механизм внимания, а также учитывает статические признаки, обеспечивая более интерпретируемое прогнозирование (например, вывод вклад разных индикаторов) ([Temporal Fusion Transformers for Enhanced Multivariate Time Series Forecasting of Indonesian Stock Prices](#)) ([Temporal Fusion Transformers for Enhanced Multivariate Time Series Forecasting of Indonesian Stock Prices](#)).

В совокупности, современная литература демонстрирует, что трансформеры могут успешно применяться для прогнозирования цен акций. Ключевые тенденции – **обогащение входных данных** (новости, многомерные показатели), **специализированные механизмы внимания** (по активам, по признакам), а также **гибридные архитектуры**, устраняющие слабые места базового Transformer. Далее мы рассмотрим устройство этих моделей и особенности их применения более подробно.

Методология

Принцип работы трансформеров на временных рядах. Модели типа Transformer изначально разработаны для последовательностей слов, однако их базовый механизм – **самовнимание (self-attention)** – универсален и может применяться к любым последовательным данным. В задачах прогнозирования

временных рядов трансформер просматривает последовательность прошлых значений и **параллельно** вычисляет веса внимания, оценивая вклад каждого временного шага в прогноз. Это позволяет ему улавливать как ближние, так и дальние зависимости во времени (long-term dependencies), чего трудно достичь традиционным RNN, страдающим от затухания градиентов. Тем не менее, прямое применение трансформера к ряду сталкивается с проблемой представления позиций во времени – ведь модель не имеет встроенного понятия порядка. Поэтому используются специальные **временные кодировки** (positional encoding). В ранних работах применялись фиксированные синусоидальные кодировки по индексу дня, однако для финансовых данных этого недостаточно. В 2022 году получила распространение обучаемая кодировка **Time2Vec**, представляющая каждый временной шаг набором периодических функций с обучаемыми параметрами. Muhammad et al. отметили, что внедрение Time2Vec фактически **открывает трансформеру путь в задачи прогнозирования акций**, позволяя ему оперировать датами/временами как дополнительными признаками ([\[2208.08300\] Transformer-Based Deep Learning Model for Stock Price Prediction: A Case Study on Bangladesh Stock Market](#)). Помимо позиции во временном ряде, вектора признаков на входе трансформера могут включать технические индикаторы (скользящие средние, объемы и т.д.), макроэкономические показатели или другие признаки. Например, одно исследование явно **добавляло признак скользящей средней** цен вместе с time embedding, чтобы сгладить ряд и помочь модели видеть тренд ([Transformer-Based Deep Learning Model for Stock Price Prediction](#)).

Адаптации архитектуры для финансовых данных. Стандартный Transformer состоит из энкодера и, при необходимости, декодера с многоголовым вниманием. Для прогнозирования цен обычно достаточно энкодера (берём фиксированное окно прошлых цен и выдаём прогноз на шаг вперёд или несколько шагов). Однако прямое применение «как есть» выявило ряд проблем. Во-первых, финансовые ряды часто **нестационарны** – статистика меняется со временем (волатильность, тренды). Это приводит к тому, что трансформер может переобучиваться на исторические шаблоны, не актуальные в будущем. В 2022 году Liu и соавт. предложили концепцию **Non-stationary Transformer**, в которой нормализация входов и механизм внимания адаптируются под локальные статистики, что снизило MSE ошибок почти на 50% по сравнению с обычным Transformer ([Non-stationary Transformers: Exploring the Stationarity in Time ...](#)). Таким

образом, для финансовых данных важно встроить в модель способы бороться с изменяющимся распределением – от скользящего нормирования данных до обучения на расширяющемся окне с переобучением модели под новые режимы рынка.

Во-вторых, трансформеры склонны **фокусироваться на глобальных зависимостях** и могут пропускать краткосрочные локальные паттерны ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Для рынков же локальные всплески или краткие аномалии (например, резкое движение в ответ на новость) критически важны. Решение, предложенное в ряде работ, – комбинировать трансформер с *свёртками*. Свёрточные слои acting as **Token Embedding** извлекают из сырых временных данных локальные признаки (тенденции на малых интервалах) прежде, чем данные поступают в блок внимания ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Например, Inception-Conv модуль в DCT обрабатывает ценовой ряд фильтрами разной ширины, получая представления и для мелких, и для крупных колебаний ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Другая идея – разбить входной ряд на *patches* (отрезки) и применять внимание на уровне этих патчей (подход PatchTST, 2022), что уменьшает длину последовательности и выделяет значимые сегменты.

Третья особенность – необходимость моделировать **взаимосвязи между множеством временных рядов**. В отличие от языкового текста, где последовательность одна, в биржевой задаче мы часто имеем сразу несколько активов, влияющих друг на друга. Архитектурно это решается либо увеличением размерности ввода (например, матрица цен всех акций за дни), либо введением двухуровневого внимания. MASTER реализует второй подход: чередуются слои **внимания по времени для каждой акции** и слои **внимания между акциями**, благодаря чему модель ловит как индивидуальные паттерны, так и одновременные корреляции между разными акциями ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)). Альтернативный подход – представить отношения акций в виде графа (акции = узлы, связи = корреляции) и применять модифицированный трансформер, совмещающий **Self-Attention с Graph Attention**. Так, в DGT вычисляются матрицы связи (граф сходства) и далее через специальный слоеный механизм внимания модель **одновременно обрабатывает временную последовательность и взаимосвязи между узлами графа** ([Stock Market](#)

[Forecasting with Differential Graph Transformer: A Novel Approach to Temporal and Spatial Stock Data Analysis | Journal of Information Systems Engineering and Management](#)). Такие улучшения архитектуры позволяют учитывать структуру фондового рынка (сектора, индексы, лидирующие или отстающие акции и пр.), что повышает точность прогноза.

Наконец, адаптация касается и **обучения** трансформеров на финансовых данных. Чтобы избежать переобучения, исследователи используют регуляризацию (например, **Dropout 0.2** на каждом слое внимания ([Deep Convolutional Transformer Network for Stock Movement Prediction](#))), небольшое число слоёв в модели (так, CNN-Trans-SPP назван «small transformer» – содержит минимально достаточное число параметров ([CNN-Trans-SPP: A small Transformer with CNN for stock price prediction](#))) и специальные техники вроде *early stopping*. Кроме того, из-за динамичности рынков актуальна **инкрементальная тренировка**: модель дообучается по мере поступления новых данных, не забывая старые (для этого применяли алгоритм Elastic Weight Consolidation, адаптированный под трансформер ([\[PDF\] An enhanced Transformer framework with incremental learning for ...](#))). Важным вопросом является и **интерпретируемость**: трансформеры – сложные «чёрные ящики», но механизмы внимания можно использовать для оценки важности признаков или влияния одних акций на другие. В работах отмечается, что визуализация матриц внимания трансформера способна выявить осмысленные зависимости (например, MASTER сумел выделить кластеры акций по отраслям через веса межакционного внимания ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#))). Temporal Fusion Transformer предоставляет встроенную интерпретацию – важность каждого входного признака и каждого временного шага через вычисленные коэффициенты внимания, что ценно для аналитиков ([Temporal Fusion Transformers for Enhanced Multivariate Time Series Forecasting of Indonesian Stock Prices](#)). Таким образом, методология применения трансформеров к акциям включает в себя не только саму модель внимания, но и целый ряд архитектурных модификаций и приёмов обучения, призванных учесть специфику финансовых данных.

Анализ и результаты

Вычислительные требования и данные

Модели трансформеров обычно **требовательны к вычислительным ресурсам**. Алгоритм самовнимания имеет вычислительную сложность $O(n^2)$ от

длины последовательности, поэтому большие окна исторических данных резко увеличивают время обучения и память. В экспериментах отмечалось, что прямое применение трансформера к большим биржевым датасетам затруднительно из-за **больших потребностей в памяти**, и это ограничивает практичность подхода без оптимизации ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Например, для прогноза на основе минутных данных за год ($\approx 390 \cdot 252 \approx 100k$ точек на акцию) стандартный Transformer потребует распределения в десятки гигабайт памяти GPU, что мало доступно вне исследовательской среды. Для преодоления этого применяют методы снижения сложности: **Sparse Attention** (разреженное внимание, как в модели Informer), усечение длины контекста, агрегирование данных по интервалам. Тем не менее, для обучения трансформерных моделей на истории цен предпочтительно иметь доступ к GPU/TPU и эффективно батчить данные.

Кроме вычислительной сложности, трансформеры **требовательны к объёму данных** для обучения. Поскольку число параметров велико, нужна длительная история или расширенный набор данных (например, данные сразу по многим акциям) чтобы модель не переобучилась. Практика показывает, что на небольших выборках простые модели могут превосходить глубокие: «*LSTM лучше работает при большом количестве данных, а ARIMA – на малых выборках*» ([Time series prediction using ARIMA vs LSTM](#)). Аналогично, трансформер при недостатке данных рискует подогнаться под шум. Поэтому часто делают *преобучение* (pre-training) на больших наборах – например, на целом индексе акций – а затем *дообучение* (fine-tuning) на конкретной акции или секторе. Это позволяет перенести знания о общих рыночных закономерностях. Преимущество трансформеров в том, что они поддерживают **Transfer Learning** – можно обучить модель на одних последовательностях и использовать на других, чего сложно добиться с традиционными статистическими моделями или даже с LSTM ([Temporal Fusion Transformers for Enhanced Multivariate Time Series Forecasting of Indonesian Stock Prices](#)). В итоге, для достижения высокой точности трансформерам обычно нужен либо большой датасет, либо комплексный подход к генерации признаков (например, добавление теханализа, других акций, новостей), чтобы модель обучалась на богатом представлении рынка.

Ограничения и проблемы применения

Несмотря на многообещающие результаты, исследования отмечают ряд проблем при применении трансформеров в финансах.

- **Переобучение и обобщающая способность.** Финансовые ряды содержат много шума и случайных колебаний, на которых мощная модель может легко переобучиться. Простые модели (ARIMA, регрессия) за счёт своей жёсткой структуры менее склонны подгоняться под шум, тогда как трансформер может “выучить” случайные паттерны, не повторяющиеся в будущем. Чтобы смягчить это, необходимы регуляризация и контроль сложность модели. Например, Xie et al. использовали dropout=0.2 в каждом слое и пониженную размерность слоёв, отмечая, что это помогло снизить риск переобучения при обучении DCT ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)) ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Тем не менее, остаётся вызов: как отличить рыночный шум от информативных сигналов – трансформер сам по себе этому “не обучен”. Требуются методы фильтрации признаков: MASTER решает через **маркет-гид (market guide)** – подмешивая агрегированные рыночные показатели, он **направляет модель фокусироваться на значимых признаках** и отсеивать нерелевантные ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)). В других работах добавляют осмысленные признаки (волатильность, корреляции), что снижает нагрузку на модель по изобретению сигналов самостоятельно.

- **Интерпретируемость решений.** Для трейдеров и аналитиков важно понимать, почему модель делает тот или иной прогноз. Классический ARIMA предоставляет понятные коэффициенты (автокорреляции), решающие деревья – легко интерпретируемые правила, а трансформеры – нет. Хотя механизм внимания можно визуализировать, **интерпретация attention-скоров неоднозначна**. Тем не менее, прогресс есть: например, TFT предоставляет оценки важности статических и динамических признаков, а MASTER визуализировал матрицы внимания между акциями, выявив кластеры отраслевых корреляций ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)). Это ценно, так как даёт доверие пользователям модели. В целом же, интерпретируемость трансформеров пока уступает простым моделям, поэтому их используют чаще как часть

автоматических торговых стратегий, нежели для поддержки решений людей (где объяснимость критична).

- **Стабильность и устойчивость.** Под стабильностью понимается, насколько надёжно модель работает при изменениях данных. Рынки подвержены **структурным разрывам** (смена режима, кризисы). Модель, обученная на одном режиме, может резко потерять точность при смене условий. Трансформеры, будучи неглубоко привязанными к последним данным (в отличие от ARIMA с обновляемыми коэффициентами), могут выдавать непредсказуемые ошибки при выходе за рамки обучающей выборки. Исследователи пытаются внедрять механизмы адаптации: инкрементальное обучение, fine-tuning по мере поступления новых данных, либо **robust training** (обучение на разных периодах, перемешивая их, чтобы модель не переобучилась на один режим). Однако эта проблема далека от решения – в реальном рынке может случиться событие, не похожее ни на что в обучающих данных. Здесь преимущество простых моделей в том, что они, как правило, опираются на меньшее число предположений, и их можно быстрее перенастроить. В работах подчёркивается, что **уникальность трендов каждой акции** затрудняет построение единой универсальной модели ([Temporal Fusion Transformers for Enhanced Multivariate Time Series Forecasting of Indonesian Stock Prices](#)). То, что работает для одной акции, может не работать для другой – трансформеру приходится “вспоминать” контекст конкретного инструмента. Поэтому иногда выбирают компромисс: обучать отдельные небольшие модели на конкретные акции или сектора (жертвуя универсальностью ради стабильности на узкой области).

- **Высокая вариативность результатов.** В сложных моделях результаты могут меняться от запуска к запуску (из-за случайной инициализации, стохастичности оптимизации). Наблюдение из практики – нужно тщательно подбирать гиперпараметры (размер окна, число слоёв, learning rate) для каждого набора данных. Нет гарантии, что параметры, успешные на одном рынке, будут оптимальны на другом. Это затрудняет промышленное внедрение: требуется больше экспериментов для калибровки трансформера, тогда как методы вроде Prophet/ARIMA более стандартизированы.

Суммируя: трансформеры предоставляют мощный аппарат для моделирования финансовых временных рядов, но их применение сопряжено с рядом технических трудностей. Нужны большие вычислительные мощности и данные, осторожность во избежание переобучения, и дополнительные усилия для обеспечения интерпретируемости и стабильности модели.

Сравнение с традиционными методами

Сравним, насколько трансформеры эффективнее классических подходов в задаче прогнозирования цен акций:

- **ARIMA/Prophet.** Модели типа ARIMA строят линейные регрессии на лаги ряда и хорошо работают на **стационарных** временных рядах с понятными сезонностями. Однако фондовые рынки, как правило, нестационарны и содержат нелинейные эффекты, поэтому ARIMA часто недооценивает волатильность и не улавливает резкие развороты. В долгосрочных прогнозах ARIMA быстро теряет точность ([Predicting Stock Market Trends with Data Science: Part 4 — ARIMA ...](#)). Гибридные модели (ARIMA+GARCH) учитывают волатильность, но всё же ограничены. Пространство состояний ARIMA невелико, и при наличии достаточных данных **глубокие нейросети обычно превосходят ARIMA**. Например, в одном эксперименте на акциях Ibovespa трансформер показал более высокую точность прогноза, превзойдя ARIMA примерно в 60% случаев при различных настройках горизонта (). Тем не менее, на **коротких рядах или при очень небольшом обучающем наборе** ARIMA может дать конкурентный результат, тогда как трансформер без данных работать не будет ([Time series prediction using ARIMA vs LSTM](#)).

- **Деревья решений и XGBoost.** Методы ансамблей деревьев (Random Forest, XGBoost) широко применяются в алгоритмической торговле для прогнозирования на основании большого набора индикаторов. Их плюс – умение работать с разнообразными нелинейными признаками и относительная устойчивость к шуму, а также высокая скорость обучения. При грамотном отборе признаков XGBoost способен обнаруживать паттерны, подобные тем, что ловит нейросеть. Однако деревья **не моделируют временную зависимость явно** – им нужно скормить лаги ряда или агрегаты в качестве отдельных признаков. В результате для захвата долгосрочной

динамики приходится вручную задавать значимые интервалы. Трансформер же учится **автоматически выявлять важные лаги и сочетания** через механизм внимания. В исследованиях сравнения часто включают XGBoost или светлые GBM как бенчмарк: нейросетевые модели начинают выигрывать, когда паттерны сложные или данные мультимодальные (например, включены тексты). Если же зависимость почти линейна или размер выборки мал, ансамбли могут быть лучше. К сожалению, прямых численных сравнений “Transformer vs XGBoost” в литературе меньше, но косвенно известно, что гибриды типа “признаки через нейросеть + XGBoost” показывали себя хуже, чем сквозное обучение end-to-end на трансформере ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Поэтому в новейших работах XGBoost служит скорее базовым ориентиром: например, MASTER сравнивался с градиентным бустингом и значительно его превзошёл по метрикам RMSE и MAPE (точные цифры в тексте не приведены, но говорится о превосходстве над “предыдущими работами”) ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)).

- **LSTM/GRU (рекуррентные нейросети).** До эры трансформеров наиболее популярными моделями для цен акций были LSTM и GRU – рекуррентные сети с памятью. Они умеют моделировать нелинейности и некоторую долговременную память, что уже даёт преимущество над ARIMA. Сравнение Transformer vs LSTM – один из ключевых вопросов последних лет. Ряд работ отмечал, что *«трансформеры не дают существенного прироста по сравнению с RNN аналогичного размера»* ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)), особенно если не настроены должным образом. Это связано с тем, что LSTM тоже может выучить многие паттерны, а на ограниченных данных даже обгоняет трансформер (у последнего больше параметров). Однако по мере усложнения задачи трансформеры начинают выигрывать. В репозитории ctxj (2023) зафиксировано, что при прогнозе минутных цен акций (по 10 предыдущих шагов) **трансформер обучается быстрее и достигает меньшей ошибки, чем LSTM** ([GitHub - ctxj/Time-Series-Transformer-Pytorch](#)). Академические исследования также сообщают о превосходстве: гибрид CNN-Trans-SPP уменьшил ошибку MAPE на 36.8% по сравнению с обычным трансформером и на ~45% по сравнению с LSTM ([CNN-Trans-SPP: A small Transformer with](#)

[CNN for stock price prediction](#)). В модели DGT с вниманием на графах улучшение по RMSE ~13.5% относительно GRU-бенчмарка ([Stock Market Forecasting with Differential Graph Transformer: A Novel Approach to Temporal and Spatial Stock Data Analysis | Journal of Information Systems Engineering and Management](#)). Эти результаты свидетельствуют, что трансформеры (особенно усовершенствованные) **могут существенно превзойти LSTM/GRU** в точности. Кроме того, трансформеры имеют архитектурные преимущества: параллельная обработка последовательности (обучение быстрее, т.к. нет поэлементной рекурсии) и возможность масштабирования на большие окна контекста. LSTM же приходится увеличивать число временных шагов итеративно, что и медленнее, и может привести к ухудшению обучения (градиенты затухают). С другой стороны, **LSTM менее требовательны к памяти** и зачастую стабильнее на небольших данных. Таким образом, в рамках одной задачи при достаточном объёме данных правильно настроенный трансформер обычно не уступает или превосходит LSTM по качеству прогноза, а иногда весьма заметно. Например, в работе Wang et al. (2022) глубокий трансформер превзошёл LSTM и CNN, дав прирост точности порядка нескольких процентов ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)). Но если данные ограничены, LSTM может быть надёжнее.

В целом, трансформеры представляют следующий шаг по сравнению с традиционными методами: они объединяют способность **учиться у данных** (как нейросети) с **гибкостью по временным зависимостям** (благодаря вниманию). Практические сравнения показывают преимущество трансформеров во многих случаях, особенно на **длинных последовательностях** и **многомерных данных**. Однако традиционные методы не списаны со счетов: они проще, понятнее и могут выступать сильной базой для небольших задач. Часто в работах лучший результат достигается не чисто трансформером, а **ансамблем методов**. Например, можно усреднить прогнозы ARIMA, LSTM и Transformer – это иногда повышает стабильность и уменьшает риск ошибки в одну сторону. Но с ростом количества данных и сложности зависимостей трансформеры всё явнее выходят вперёд.

Заключение и рекомендации

Итоги текущих исследований. На сегодняшний день трансформеры зарекомендовали себя как перспективный инструмент для прогнозирования цен акций. За последние два года опубликовано множество работ, продемонстрировавших успешность трансформеров при правильной адаптации: от специализированных архитектур вроде MASTER ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)) до гибридных моделей с CNN и графами ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)) ([Stock Market Forecasting with Differential Graph Transformer: A Novel Approach to Temporal and Spatial Stock Data Analysis | Journal of Information Systems Engineering and Management](#)). Текущие исследования показывают, что трансформеры способны превосходить классические модели и предыдущие нейросетевые подходы по точности прогноза ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)) ([CNN-Trans-SPP: A small Transformer with CNN for stock price prediction](#)). Особенно эффективно трансформеры работают, когда нужно учитывать множество факторов и зависимостей (различные акции, индикаторы, текстовые данные и т.д.) – их многоголовый механизм внимания естественным образом это реализует. Тем не менее, универсального решения пока не достигнуто: каждый рынок и задача требуют тонкой настройки модели. Некоторые исследования отмечают, что базовый Transformer без улучшений даёт скромный выигрыш или вообще не превосходит LSTM ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)), но модификации архитектуры и добавление данных существенно повышают результат.

Преимущества трансформеров:

- Умеют *захватывать долгосрочные зависимости* в данных лучше рекуррентных сетей, что важно для распознавания трендов и циклов ([Deep Convolutional Transformer Network for Stock Movement Prediction](#)).
- *Параллельная обработка последовательности*: обучение и предсказание могут быть быстрее, чем у последовательных моделей, особенно на современных GPU ([Temporal Fusion Transformers for Enhanced Multivariate Time Series Forecasting of Indonesian Stock Prices](#)).
- *Гибкость архитектуры*: легко расширяются под мультимодальные данные (цены, новости, граф связей) – достаточно добавить соответствующие блоки внимания или эмбединги.

- *Возможность трансферного обучения:* можно преобучить на одних данных и адаптировать к другим, что перспективно для финансов (например, обучить модель на истории индексов, затем перенастроить под отдельную акцию).

- *Состояние искусства на многих задачах:* трансформеры побили рекорды точности не только в NLP, но и в общем прогнозировании временных рядов (метео, электроэнергия), и финансовая область начинает получать выгоды от этих достижений.

Недостатки и ограничения:

- *Высокий риск переобучения* на ограниченных данных и шумных рядах – требуется большой объём данных или регуляризация, иначе модель может ловить случайные паттерны.

- *Требовательность к ресурсам:* для обучения нужен мощный GPU/TPU, много памяти (особенно для длинных последовательностей). В реальном времени громоздкие модели могут быть трудны для развёртывания.

- *Низкая интерпретируемость:* трудно объяснить, почему сделан тот или иной прогноз. Это препятствие для использования в сфере финансовых услуг, где нужна объяснимость. Требуются специальные техники интерпретации внимания или пост-хок анализ.

- *Сложность настройки:* много гиперпараметров (длина окна, архитектура, количество слоёв, головы внимания, скорость обучения и др.), чувствительность к их выбору. Оптимизация требует экспертного опыта и экспериментов.

- *Не гарантируют превосходство во всех ситуациях:* существуют режимы рынка или отдельные акции, где простые модели выступают не хуже. Трансформер – не панацея, и на него не следует полагаться без сравнения с бенчмарками.

Рекомендации и направления дальнейших исследований:

1. **Улучшение устойчивости к смене режимов.** Стоит развивать методы адаптации трансформеров к меняющимся рыночным условиям.

Перспективны *онлайн-обучение* и *мета-обучение*, при которых модель могла бы быстро перестраиваться при появлении новых данных, сохраняя знание базовых паттернов. Например, внедрение механизмов вроде Non-stationary Transformer ([Non-stationary Transformers: Exploring the Stationarity in Time ...](#)) ([Non-stationary Transformers: Exploring the Stationarity in Time ...](#)) или обучающих регуляризаций, не дающих забыть старое, поможет бороться с нестационарностью.

2. Интерпретируемые финансовые трансформеры.

Необходимо уделять внимание интерпретации моделей. Одно направление – разрабатывать **attention-метрики**, показывающие влияние входных факторов на прогноз, в удобной для аналитиков форме. Это может быть интеграция идей из Explainable AI (SHAP-values, локальные объяснения) с трансформерами. Другой подход – создавать упрощённые суррогатные модели, приближенно объясняющие решения трансформера. Повышение прозрачности повысит доверие к таким моделям в индустрии.

3. **Интеграция экономических знаний.** Трансформеры как универсальные аппроксиматоры могут выиграть, если направлять их с помощью знаний о предметной области. MASTER показал пользу от использования агрегированной информации рынка для выбора признаков ([\[2312.15235\] MASTER: Market-Guided Stock Transformer for Stock Price Forecasting](#)). В будущем можно интегрировать в архитектуру, например, знание о торговых сессиях, календарных эффектах, граф связей между компаниями (вплоть до знаний о поставщиках/клиентах компании). Архитектуры типа **Graph Transformer** и **Temporal Fusion** – шаги в эту сторону. Возможно появление *доменно-специфических трансформеров* для финансов, которые встраивают в себя основы финансовой теории (например, принципы арбитража или «эффективного рынка») как ограничения при обучении.

4. **Повышение эффективности и уменьшение размеров модели.** Для практического применения желательно облегчить трансформеры. Будущие исследования могут сфокусироваться на **сжатии моделей** (применение знаний из TinyML к трансформерам), разработке более эффективных механизмов внимания (линейное или логарифмическое

время). Уже предложены варианты (Informer, Performer и др.), их адаптация к финансовым данным – перспективная задача. Это позволит использовать трансформеры для **высокочастотного трейдинга** и на встраиваемых устройствах, где ресурсы ограничены.

5. **Мультимодальные и многоцелевые модели.** Реальные инвестиционные решения опираются на сочетание прогнозов по разным активам и факторам. Интересным направлением является создание **единой трансформер-модели, способной предсказывать сразу несколько связанных величин** – цены множества акций, совместно с макропоказателями. Такая модель могла бы учиться на более широком контексте (например, одновременно прогнозировать индекс и входящие в него акции, что улучшит согласованность предсказаний). Также мультимодальные трансформеры, обрабатывающие текст новостей, числовые ряды и может быть даже изображения (например, графики или спутниковые снимки для агро-сектора), могут предоставить более полную картину рынка. Первые шаги в этом направлении уже есть (учёт новостей, соцмедиа), и их следует развивать.

В заключение отметим, что применение трансформеров в прогнозировании цен акций – быстро эволюционирующая область. Трансформеры уже показали **высокий потенциал**, но чтобы полностью раскрыть его в условиях реального рынка, требуются дальнейшие исследования. Комбинируя мощь трансформер-архитектур с экономической экспертизой и новыми идеями (графы, мультимodal, интерпретируемость), исследователи и практики смогут создать модели, которые не только точны, но и надёжны и понятны. Такая синергия поспособствует появлению более совершенных систем поддержки решений для инвесторов и трейдеров.

Источники и полезные ресурсы: Все утверждения и результаты, упомянутые в отчёте, подкреплены ссылками на литературу. Для практического ознакомления с моделями рекомендуем репозиторий **MASTER** ([GitHub - SJTU-DMTai/MASTER: This is the official code and supplementary materials for our AAAI-2024 paper: MASTER: Market-Guided Stock Transformer for Stock Price Forecasting. MASTER is a stock transformer for stock price forecasting, which models the momentary and cross-time stock correlation and guide feature selection with market information.](#)) – код продвинутого трансформера для акций, библиотеку **TrendMaster** ([GitHub -](#)

[hemangioshi37a/TrendMaster: Using Transformer deep learning architecture to predict stock prices.](#)) – прикладной инструмент на основе трансформера, а также открытые реализации простых трансформеров для временных рядов (например, проект ctxj ([GitHub - ctxj/Time-Series-Transformer-Pytorch](#)), демонстрирующий преимущество трансформера над LSTM на минутных данных). Эти ресурсы помогут заинтересованному читателю углубиться в тему и попробовать реализовать рассмотренные модели самостоятельно.