

**Трибуна молодого ученого**

УДК 336.76, 004.032.26

JEL G17

**СТАРКОВ Илья Алексеевич**

Финансовый университет при Правительстве Российской Федерации, ул. Малый Златоустинский переулок, 7, Москва, 129164, Россия.

<https://orcid.org/0000-0003-0212-7737>

Старков Илья Алексеевич, студент Финансового университета, Финансовый факультет, Финансовые рынки и финтех, 3 курс, Москва, Россия. E-mail: il0106@yandex.ru

*Научный руководитель: Захарова Ольга Владимировна, старший преподаватель Департамента банковского дела и финансовых рынков, Москва. Финансовый университет при Правительстве Российской Федерации, ул. Малый Златоустинский переулок, 7, Москва, 129164, Россия. E-mail: olvzakharova@mail.ru*

**ОЦЕНКА ПРОГНОЗИРОВАНИЯ БИРЖЕВОГО ИНДЕКСА  
С ПОМОЩЬЮ LSTM-НЕЙРОСЕТИ**

**Аннотация**

**Предмет/тема.** В исследовании рассмотрена работа слоя LSTM в составе нейросети для прогнозирования значений биржевого индекса РТС на следующий рабочий день.

**Цели/задачи.** Измерение эффективности реализации скалярной регрессии нейросетевой архитектуры с применением LSTM-слоя на примере прогнозирования индекса РТС.

**Методология.** Моделируется одно из возможных базовых (наивных) решений, таким образом определяется целевая эффективность для создаваемой нейросети. Создаётся простейшее нейросетевое решение – неглубокая полносвязная сеть, на примере которой определяется обоснованность использования перекрёстной проверки по K блокам. Сравнивается эффективность LSTM-сети с неглубокой полносвязной нейросетью по метрике MAE (mean absolute error). Проводится постепенная оптимизация избранной архитектуры, затем анализируется работа финального варианта нейросети на тестовой выборке, которая сопоставляется с эффективностью наивного решения.

**Результаты/выводы.** LSTM-сеть не показала хороших результатов - эффективность наивного (базового) метода прогнозирования оказалась выше. Делается вывод о том, что LSTM лучше делать в стеке с другими технологиями, алгоритмами и входными данными. Выдвигается предположение, что LSTM может больше подходить для бинарной или многоклассовой классификации, нежели для скалярной регрессии.

**Ключевые слова:** LSTM, нейросеть, прогнозирование биржевого индекса, полносвязная нейросеть, рекуррентная нейросеть, финансовый рынок, машинное обучение, оптимизация.

### Young scientist tribune

**Ilya A. Starkov**

Student of the Faculty of Finance at Financial University, Financial Markets and Fintech, st. Maly Zlatoustinsky Lane, 7, Moscow, 129164, Russia.

E-mail: il0106@yandex.ru

*Scientific advisor: Olga V. Zakharova, Senior Lecturer, Banking and Financial Markets Department at Financial University under the Government of the Russian Federation.*

*E-mail: olvzakharova@mail.ru*

## ESTIMATION OF STOCK INDEX FORECASTING USING LSTM NEURAL NETWORK

### Abstract

**Subject/topic** The study examines the work of the LSTM layer as part of a neural network for predicting the values of the RTS stock index on the next working day.

**Goals/tasks** Measuring the effectiveness of implementing scalar regression of neural network architecture using the LSTM layer on the example of forecasting the RTS index.

**Methodology** One of the possible basic (naive) solutions is modeled, thus the target efficiency for the created neural network is determined. The simplest neural network solution is being created – a shallow fully connected network, on the example of which the validity of using cross-validation for K blocks is determined. The efficiency of an LSTM network with a shallow fully connected neural network is compared according to the MAE (mean absolute error) metric. A gradual optimization of the chosen architecture is carried out, then the work of the final version of the neural network is analyzed on a test sample, which is compared with the effectiveness of the naive solution.

**Results/conclusions** The LSTM network did not show good results - the effectiveness of the naive (basic) forecasting method turned out to be higher. It is concluded that it is better to include LSTM in a stack with other technologies, algorithms and input data. It is suggested that LSTM may be more suitable for binary or multiclass classification than for scalar regression.

**Keywords:** *LSTM, neural network, stock index forecasting, fully connected neural network, recurrent neural network, financial market, machine learning, optimization.*

Мечта каждого аналитика, трейдера-спекулянта, инвестора, информационного агентства, политика, да и любого лица, интересы которого переплетаются с финансовыми рынками, - знание будущего положения дел, а именно, будущих цен каких-либо активов, начиная от товаров и заканчивая сложными высоко виртуализированными инструментами. Отдача от финансовых вложений не является предсказуемой, наоборот, каждое ценовое колебание актива - невероятно многофакторный процесс, включающий в себя многие явления, которые рассчитать в точности невозможно [1]. В такой ситуации на первое место выступает знание о том, что будет в будущем (или хотя бы попытка узнать).

Способов прогнозирования ценовых колебаний огромное множество и в последнее время набирает особую популярность применение нейросетей в качестве агента, позволяющего с определённой долей уверенности сказать что-либо о завтрашнем дне (это может быть волатильность или кривизна её «улыбки», цены закрытия, открытия, максимумов, минимумов, открытый интерес, объём торгов за сессию и т.д.).

В данной статье будет проанализирована работа популярного сегодня рекуррентного слоя нейросетевых архитектур – несвёрточного LSTM (long short-term memory). Исследование будет проходить следующим образом: во-первых, будет смоделировано базовое решение – наш бенчмарк эффективности, который нам предстоит преодолеть созданной нейросетью, во-вторых, необходимо создать простейшую нейросеть для решения поставленной задачи прогнозирования, как некий аналог бенчмарка, только в пространстве нейросетей, в-третьих, должна быть скомпилирована сеть хотя бы с одним слоем LSTM, которая по эффективности будет лучше как простейшей нейросети, так и базового решения на валидационной выборке, в-четвёртых, будет воспроизведён точечный прогноз на тестовой выборке, а также подсчитаны основные метрики, в конце же последует вывод по проделанной работе.

Исследование будет проводиться в следующих рамках:

- 1) Эффективность LSTM будет оценена только со стороны скалярной регрессии;
- 2) На вход нейросети будут поданы временные ряды дневных цен закрытия индекса РТС, индекса ММББ, индекса S&P500, склеенного фьючерса на нефть Brent и валютная пара USD/RUB (FOREX) с 11 декабря 2011 года (все ряды нормализованы, среднее и стандартное отклонение для этого, естественно, взяты от тренировочной выборки);
- 3) Количество последних данных, учитываемых для реализации прогноза – 15 дней, то есть три недели рабочих дней;
- 4) Горизонт прогнозирования - цена закрытия РТС на завтрашний день;
- 5) Некоторые решения будут содержать определённые константы (например, число блоков при разбивке по К блокам и т.д.).

Нужно учитывать, что все дальнейшие выводы, как промежуточные, так и итоговые, будут сделаны именно в этих рамках. Это говорит о том, что если другой исследователь возьмёт другие входные данные или воссоздаст альтернативные инженерные решения, то, скорее всего, результаты будут розниться.

Базовое (наивное) решение. Нашим базовым (наивным) решением будет логика, прогнозирующая на следующий день такое значение цены закрытия индекса РТС, которое было сегодня. Логично, что при таком прогнозировании будет наименьший спред между наивным прогнозом и реальной ценой на завтра. Но здесь есть несколько деталей, которые мы обязаны учесть для полноты картины, и для этого нужно прояснить

особенности нейросетевых структур, создаваемых нами в следующих пунктах. Дело всё в том, что мы не можем сразу определить варианты обучения моделей «на глаз», потому что количество собранных данных относительно не велико и равняется 2661 дневных закрытий, единственное, мы можем с уверенностью сказать, что временной ряд рыночных цен лучше не перемешивать. Однако остаётся два варианта препроцессинга и обучения: перекрёстная проверка по K блокам (положение валидационных данных в таком случае не будет являться константой) и обучение на обычном срезе главной выборки (здесь уже каждая подвыборка закрепит за собой определённое неизменное положение) [2].

Выбор одного из методов будет описан в следующих пунктах, однако наивное решение должно быть бенчмарком для любых путей, которые мы выберём далее, следовательно, мы должны реализовать наивное решение как через распределение по K блокам, так и через обычное разбиение на подвыборки.

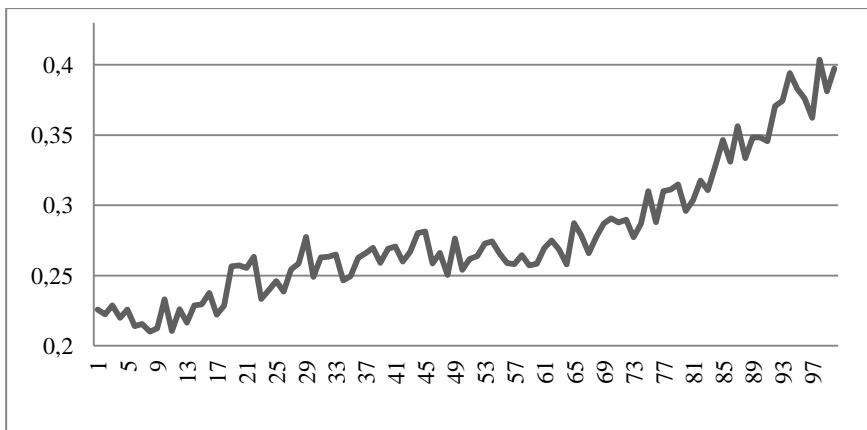
Нашей основной метрикой эффективности, которую мы будем отслеживать до конца анализа, будет MAE (mean absolute error, средняя абсолютная ошибка). Так как данные для всех моделей и оценок будут нормализованные, то для репрезентативности MAE лучше «восстанавливать» до исходных единиц измерения индекса РТС, умножая полученную ошибку на среднее квадратичное отклонение тренировочных данных. В статье MAE, не восстановленное до оригинальных единиц, будет только на графиках, в остальных случаях показатель будет представлен уже с поправкой на нормализацию (в долларах).

В изначальном наборе рядов на финальный тест будет приходиться 25%, на валидацию (или контроль/проверку) 25% от оставшихся входных данных (в конкретных индексах, считая, что первое измерение было нулевым: тренинг оканчивается на 1496 дне, а валидация / контроль – на 1995). Перекрёстная проверка по K блокам будет проходить на совмещённой подвыборке «тренинг + валидация», с той целью, чтобы предусмотреть изменение положения контрольных данных. Блоков в проверке будет 4 штуки. Таким образом, MAE базового решения при таком методе проверки составляет 15,22 доллара США. Теперь вычислим MAE на всём промежутке данных, а также на валидационной выборке (с 1497 дня до 1995), они соответственно равны 15,18 и 11,15 долларов. Мы получили диапазон нашего бенчмарка – от 11,15 до 15,22 долларов, значит все последующие структуры должны как минимум на валидационной выборке в него попасть.

Простейшее нейросетевое решение. Простейшим решением будет неглубокая сеть (обычный перцептрон) всего с двумя полностью связанными слоями (Dense): входной слой (16 нейронов, функция активации – линейный выпрямитель, или полулинейный элемент) и выходной слой (так как мы хотим воспроизвести скалярную регрессию, у нас в этом слое будет

1 нейрон без функции активации); функцией потерь будет выступать MAE, а её оптимизатором – градиентный спуск с импульсом. Мы проведём проверку на 100 эпохах. Эта и дальнейшие реализации будут производиться в Keras<sup>1</sup>.

Перекрёстная проверка такой базовой архитектуры по 4 блокам показала, что минимальное значение MAE (считались средние MAE на каждую эпоху по 4 блокам обучения) составило 63,26 доллара (см. рис. 1). Видно, что переобучение наступило практически мгновенно, и среднее потерь на валидационных выборках начало постепенно увеличиваться сразу с запуска процесса.



**Рисунок 1 – Среднее MAE (нормализованное) на валидационных выборках перекрёстной проверки простейшего нейросетевого решения / Mean MAE (normalized) on validation samples of cross-validation of the simplest neural network solution**

Как уже упоминалось ранее, обучение можно выполнить не перекрёстной проверкой, а на обычном дроблении начальной выборки на тренировочный, валидационный и тестовый срезы. Параметры срезов такие же, как и в предыдущем пункте. Обучение проводилось так же на протяжении 100 эпох, размер пакетов равен 128 единицам. Таким образом, минимальное значение показателя MAE (см. рис. 2) на валидационной выборке равно 41,62 доллара, а переобучение (экстремум-впадина графика потерь) наступило на 91-ой эпохе. Получилось так, что ошибка при обучении на обычных срезах данных меньше, чем при обучении на разбивке по K блокам, поэтому перекрёстную проверку мы использовать в дальнейшем не будем.

LSTM-сеть и достижение бенчмарка. Нашей первой целью являются результаты неглубоких сетей, описанных в предыдущих пунктах, чтобы

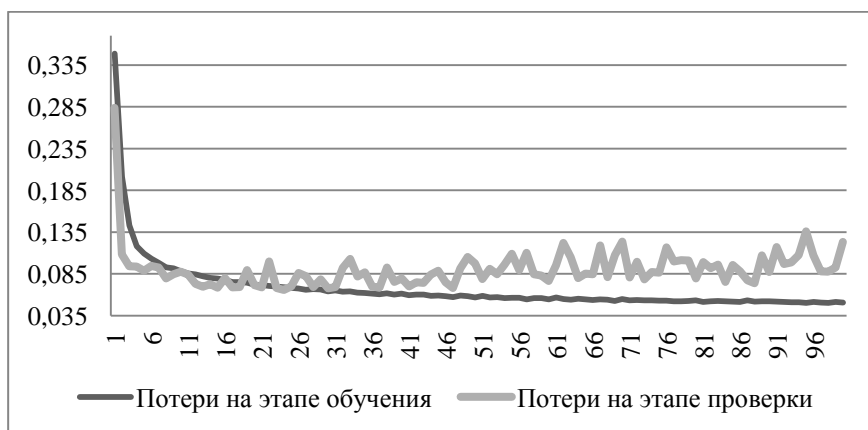
<sup>1</sup> API «Keras». [Электронный ресурс] // URL: <https://ru-keras.com/home/> (Дата обращения: 06.12.2021)

доказать, что рекуррентный слой, как минимум, лучше полносвязного, второй целью будет достижение диапазона базового решения.



**Рисунок 2 – MAE (нормализованное) на валидационной выборке без перекрёстной проверки простейшего нейросетового решения / MAE (normalized) on a validation sample without cross-validation of the simplest neural network solution**

Для первого случая мы просто заменим полносвязный слой на рекуррентный LSTM с гиперболическим тангенсом в качестве функции активации, остальные гиперпараметры останутся те же. По результатам обучения минимальное значение MAE на валидационной выборке (см. рис. 3) получилось равным 22,23 доллара на 24 эпоху, что доказывает большую эффективность скалярной регрессии с помощью LSTM по сравнению с полносвязными слоями.



**Рисунок 3 – MAE (нормализованное) на валидационной выборке первой LSTM-архитектуры, превосходящей простейший перцептрон / MAE (normalized) on a validation sample of the first LSTM architecture superior to the simplest perceptron**

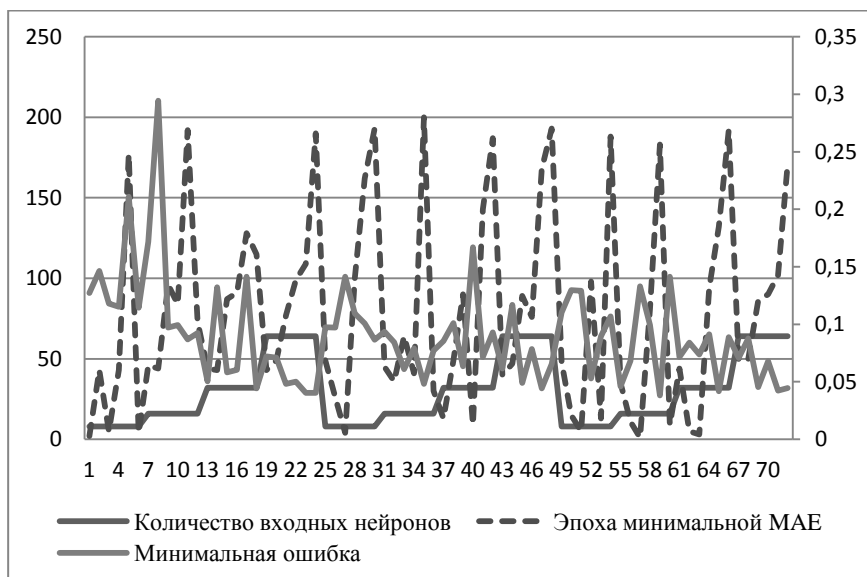
Однако для того, чтобы подтвердить обоснованность использования LSTM как приемлемое решение этой задачи в целом, необходимо достижение (пусть даже и теоретическое) результатов наивного решения, то есть диапазона от 11,15 до 15,22 долларов. Поэтому теперь нам надо реализовать подобающую оптимизацию, затрагивающую все параметры нейросети.

Вначале нам нужно выяснить в каком направлении оптимизации вообще двигаться. Представляется логичным «отместить» сразу самые серьёзные инженерные решения: количество слоёв LSTM и использование двунаправленной LSTM-сети. Тестируются такие имплементации будут на той архитектуре, которая превзошла простейшее нейросетевое решение. При использовании двунаправленной сети мы получили минимум по MAE равный 28,17 долларов, что хуже, чем первоначальная ошибка LSTM в 22,23 доллара. Удвоение LSTM (во втором слое тоже 16 нейронов, как и в первом) дало MAE, равное 36,84 долларов, удвоение входных нейронов во втором LSTM дало ошибку в 28,59 долларов, уменьшение нейронов вдвое – 33,59. Смена во втором слое гиперболического тангенса на популярный линейный выпрямитель ухудшило все три результата, увеличив минимальную потерю. Таким образом, мы выяснили, что слой LSTM будет один, а сеть не двунаправленная.

Теперь наша оптимизация перемещается в плоскость других гиперпараметров: эпох, входных нейронов, величины пакетов. Достаточно эффективное и быстрое решение - деревья оценок Парзена-Розенблатта<sup>1</sup>, однако, в контексте данной работы более репрезентативной и не менее эффективной будет простой перебор с постоянным повторным воссозданием и компиляцией архитектуры на каждой итерации цикла, дабы избежать «утечек» из контрольной выборки. Перебор будет произведён по следующим элементам: эпохи – 50, 100, 200, входные нейроны – 8, 16, 32, 64, размера пакета – 64, 128. Всего было осуществлено 10 циклов перебора, это значит, что каждому конкретному образцу архитектуры соответствует 10 его одинаковых версий, это нужно, чтобы учесть возможные «всплески» минимальных ошибок из-за определённой расстановки весов внутри слоя. Для наглядного примера были взяты три первых цикла перебора (см. рис. 4), маркерами на линии минимальной ошибки отмечены экстремумы минимальной ошибки (MAE) за каждый цикл. Левая ось – ось количества нейронов и номера эпохи, на которой возник минимум ошибки, правая ось – MAE, горизонтальная ось – номер итерации цикла по перебору. Сразу видно, что каждому минимуму соответствуют 64 нейрона, пакет размером в 64 единицы (ряд пакетов не нанесён на график, чтобы не мешать наглядности), также можно заметить, что 2 из 3 экстремумов возникли в районе 100-ой эпохи.

---

<sup>1</sup> API «Hyperas». [Электронный ресурс] // URL: <https://github.com/maxpumperla/hyperas> (Дата обращения: 06.12.2021)



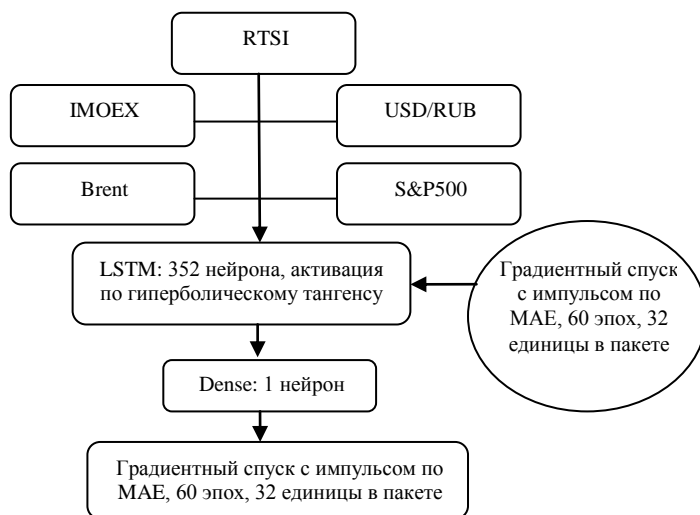
**Рисунок 4 – Первые три цикла перебора гиперпараметров / The first three cycles of iterating over hyperparameters**

Теперь понятно, что наше основное направление оптимизации будет идти в сторону увеличения нейронов слоя (последовательного удвоения) и уменьшения пакета (последовательного деления на два), детализация этих двух параметров будет проходить после нахождения точки, в которой эффективность уже практически не прибавляется, эпохи же будут подгоняться в районы найденных минимумов MAE. Руководствуясь такими принципами, получаем сеть с оптимальным размером пакета в 32 единицы, 352 нейронами и временем на обучение, равное 60 эпохам. MAE такого решения равняется 12,66 долларов. Но эти параметры ещё не дают нам финальную структуру, хоть и вошли в диапазон бенчмарка, что уже доказывает теоретическую пригодность нашей LSTM-сети. Окончательная система будет получена после регуляризации.

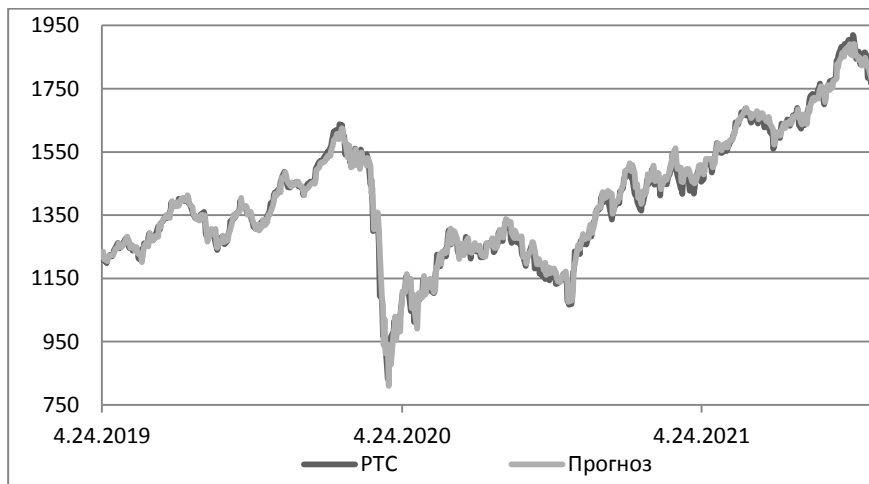
Регуляризация данной модели представляет из себя подгон таких элементов, как коэффициентов рекуррентного прореживания [3] и регуляризации L1, L2. Однако, что не очевидно, ни одно средство из перечисленных не улучшило результатов. Таким образом, получено финальное рекуррентное LSTM-решение для скалярной регрессии (см. рис. 5).

Реализация прогноза. Для оценки работы LSTM-сети нужно выполнить точечные прогнозы на завтра на тестовой выборке (см. рис. 6): на график нанесены значения индекса РТС (восстановленные из нормализованных), сдвинутые на день вперёд, то есть это таргет, который должна достичь нейросеть, вместе с ним наложен ещё один график – прогнозы сети на этот день (также восстановленные из нормализованных значений).





**Рисунок 5 – Финальная архитектура / Final architecture**

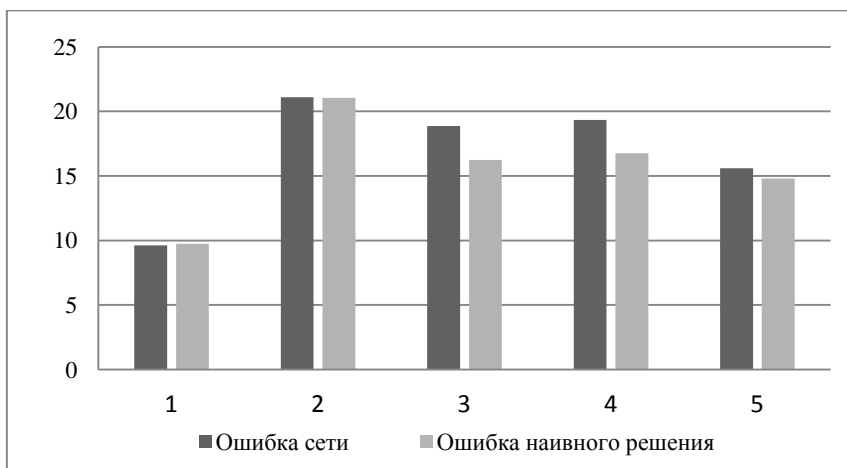


**Рисунок 6 – Работа финальной архитектуры на тестовой выборке / The work of the final architecture on a test sample**

Видно, что в определённые моменты, спред прогноза и реального значения расходится – связи, обеспеченные преобразованием входных данных через гиперпространства LSTM, затухают. Это прямо доказывает изменчивости рынка, его конъюнктуры и настроений.

Для более точной оценки, нам нужно раздробить рассмотренный промежуток на несколько более малых и посчитать на каждом из них среднюю абсолютную ошибку сети и нашего наивного решения; для примера, будет 5 отрезков по 130 дней, почти полгода рабочих дней, когда считался индекс (см. рис. 7). В первый отрезок времени нейросеть выдала

ошибку в 9,62 доллара, а наивное решение показало ошибку 9,74 доллара - немного, однако это значит, что, по крайней мере в этот промежуток, использование нейросети было выгоднее базового прогноза. Связь, найденная нейросетью «продержалась» почти полгода. Настройки спроектированной архитектуры не смогли «перегнать» наивное решение и, хоть сеть и выдавала первое время ошибки близкие к бенчмарку, её использование с такими параметрами представляется нецелесообразным.



**Рисунок 7 – Ошибки сети и наивного решения на 5 отрезках тестовой выборки, \$ / Network errors and naive solutions errors on 5 segments of the test sample, \$**

Внимания заслуживает тот факт, что сеть спрогнозировала само направление движения (выше или ниже будет завтрашнее закрытие) в 371 случаях из 650, это 57,08% точности. Может быть, при направленности сети на классификацию (отслеживание тренда), а не скалярную регрессию (прогноз конкретного значения) LSTM-структура была бы эффективнее, ведь сама постановка задачи в данном случае играет очень важную роль для результатов [4].

Итак, LSTM-сеть для скалярной регрессии показала свою неэффективность – построенная архитектура не превзошла наивного решения на тестовой выборке. Было видно, что сначала теста сеть, как минимум, недалеко отставала от бенчмарка эффективности, но в конце выборки сеть полностью утратила возможность прогнозировать ближайшие значения индекса на следующий день. Так произошло из-за самой природы рынка – его изменчивости: связи, когда-то найденные, после небольшого промежутка времени теряют свою значимость [5].

Ясно, что в таком виде (когда на вход подаются цены закрытий, и нет связки с другими технологиями) LSTM-сеть бесполезна независимо от мощности и направленности оптимизации. Однако очень важно изучить LSTM с других сторон, например, в классификации (бинарной и

многоклассовой), так как в данном исследовании нейросеть показала, что может с положительным математическим ожиданием работать по этому направлению, касающегося финансового рынка.

LSTM-структуру в целях скалярной регрессии представляется логичнее использовать в стеке других программ и алгоритмов с другими входными данными [6]. Например, использовать на вход в нейросеть временной ряд рыночного сентимента, либо применять модифицированные слои (свёрточный LSTM), нестандартные архитектуры ориентированных ациклических графов [7] или модели остаточных связей [8], также любопытно использовать при прогнозах текстовые предикторы в виде постов из социальных сетей, ленты новостей, либо частоту запроса какого-либо явления в браузере [9].

#### **Список источников:**

1. Загалаева (2019) – Загалаева Ж.А. Место и роль фондового рынка в финансовой системе страны в современных условиях // Гуманитарные, социально-экономические и общественные науки. 2019. №11.

2. Шолле (2018) – Шолле Ф. Глубокое обучение на Python. – СПб.: Питер, 2018. – 400 с.

3. Gal (2016) – Gal Y., Ghahramani Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning // International conference on machine learning. – PMLR, 2016. – С. 1050-1059.

4. Сергеев (2018) – Сергеев В.А. Использование нейросетей в прогнозировании фондового рынка / В. А. Сергеев // Азимут научных исследований: экономика и управление. – 2018. – Т. 7. – № 4(25). – С. 280-282.

5. Видмант (2018) – Видмант О.С. Прогнозирование финансовых временных рядов с использованием рекуррентных нейронных сетей LSTM // Общество: политика, экономика, право, 2018 № 5.-С.63-66. — Только в электронном виде.

6. Лабусов (2020) – Лабусов М.В. Нейронные сети долгой краткосрочной памяти и их использование для моделирования финансовых временных рядов // Инновации и инвестиции, 2020 № 3.-С.167-171. — Только в электронном виде.

7. Szegedy (2015) - Szegedy C. et al. Going deeper with convolutions // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2015. – С. 1-9.

8. He K. et al. Deep residual learning for image recognition // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2016. – С. 770-778.

9. Curme (2014) – Curme C. et al. Quantifying the semantics of search behavior before stock market moves // Proceedings of the National Academy of Sciences. – 2014. – Т. 111. – №. 32. – С. 11600-11605.

#### **References:**

Zagalaeva (2019) – Zagalaeva Zh.A. The place and role of the stock market in the financial system of the country in modern conditions [Zagalaeva Zh. A. Mesto i rol fondovogo rynka v finansovoj sisteme strany v sovremennyh

usloviyah] // Humanities, socio-economic and social sciences [Gumanitarnye, socialno-ekonomicheskie i obshchestvennye nauki]. 2019. # 11.

Chollet (2018) – Chollet Fr. Deep learning in Python [Glubokoe obuchenie na Python]. – St. Petersburg: Peter, 2018. - 400 p.

Gal, Ghahramani (2016) – Gal Y., Ghahramani Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning // International conference on machine learning. – PMLR, 2016. – P. 1050-1059.

Sergeev (2018) – Sergeev V.A. The use of neural networks in stock market forecasting [Ispolzovanie nejrosetej v prognozirovanii fondovogo rynka] / V. A. Sergeev // Azimut of scientific research: economics and management [Azimut nauchnyh issledovanij: ekonomika i upravlenie]. - 2018. - T. 7. – # 4(25). – Pp. 280-282.

Vidmant (2018) – Vidmant O.S. Forecasting financial time series using recurrent neural networks LSTM [Prognozirovanie finansovyh vremennyh ryadov s ispolzovaniem rekurrentnyh nejronnyh setej LSTM] // Society: politics, economics, law [Obshchestvo: politika, ekonomika, pravo], 2018 No. 5.- Pp.63-66. - Only in electronic form.

Labuzov (2020) – Labuzov M.V. Neural networks of long-term short-term memory and their use for modeling financial time series [Nejronnye seti dolgoj kratkosrochnoj pamyati i ih ispolzovanie dlya modelirovaniya finansovyh vremennyh ryadov] // Innovations and investments [Innovacii i investicii], 2020 No. 3. - Pp.167-171. - Only in electronic form.

Szegedy (2015) – Szegedy C. et al. Going deeper with convolutions // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2015. – Pp. 1-9.

He K. et al. Deep residual learning for image recognition // Proceedings of the IEEE conference on computer vision and pattern recognition. – 2016. – Pp. 770-778.

Curme (2014) – Curme C. et al. Quantifying the semantics of search behavior before stock market moves // Proceedings of the National Academy of Sciences. – 2014. – T. 111. – No. 32. – Pp. 11600-11605.