

Курсовая работа защищена с оценкой _

Ученый секретарь кафедры доцент Валединский В. Д.

Московский государственный университет имени М.В. Ломоносова

Механико-математический факультет

Кафедра Вычислительной математики



КУРСОВАЯ РАБОТА

**Прогнозирование финансовых временных
рядов с помощью нейронных сетей
с памятью**

Выполнена студенткой 309 группы

Маховой Анастасией Геннадьевной

Научный руководитель:

д.ф.-м.н М.И. Кумсков

Москва, 2023

Аннотация

Содержание

1	Введение	3
2	Задачи прогнозирования	
	финансовых временных рядов	3
2.1	Постановка задачи	5
3	Временные ряды	5
3.1	Определения	5
3.2	ETS models	7
4	LSTM	8
4.1	Архитектура	10
4.1.1	слой фильтра забывания	12
4.1.2	слой входного фильтра	12
4.1.3	обновляем состояние ячейки	13
4.1.4	слой выходного фильтра	13
4.1.5	Общая формула	14
5	Модель	15
5.1	Функция потерь	15
5.2	Алгоритм оптимизации	15
5.3	Метрика оценки качества	15
6	Python	15
7	Исследование	15
	Список литературы	15

1 Введение

2 Задачи прогнозирования финансовых временных рядов

Задачи о прогнозировании финансовых временных рядов можно разделить на 2 группы в зависимости от ожидаемых выходных данных[3]:

1. прогноз цены - необходимо предсказать стоимость на некоторое время вперед как можно точнее.
2. прогноз тренда - направления движения графика стоимости:
 - (a) 2-class problem – предсказать восходящий и нисходящий тренд
 - (b) 3-class problem - предсказать восходящий, нисходящий и боковой тренды

Входные данные для прогнозирования можно брать из:

- Цен за предыдущие периоды

«цены устанавливаются отнюдь не на основе объективных показателей, таких как рентабельность продаж или прибыль. Курс акций может вдруг резко подскочить, но связано это будет лишь с улучшением ожиданий инвесторов, а вовсе не с ростом продаж, рентабельности или прибыли компании»

(с)Роджер Мартин

То есть на цены влияет и эмоциональный аспект, который можно отследить с помощью:

- **Фундаментального анализа**

Благодаря методам фундаментального анализа можно проанализировать справедливую стоимость на данный момент и предсказать повышение/понижение спроса

- **Технического анализа**

Анализ графиков на характерные предпосылки к изменению направления движения стоимости. На рисунке 1 изображены основные виды графиков, которые используются для отображения цен.

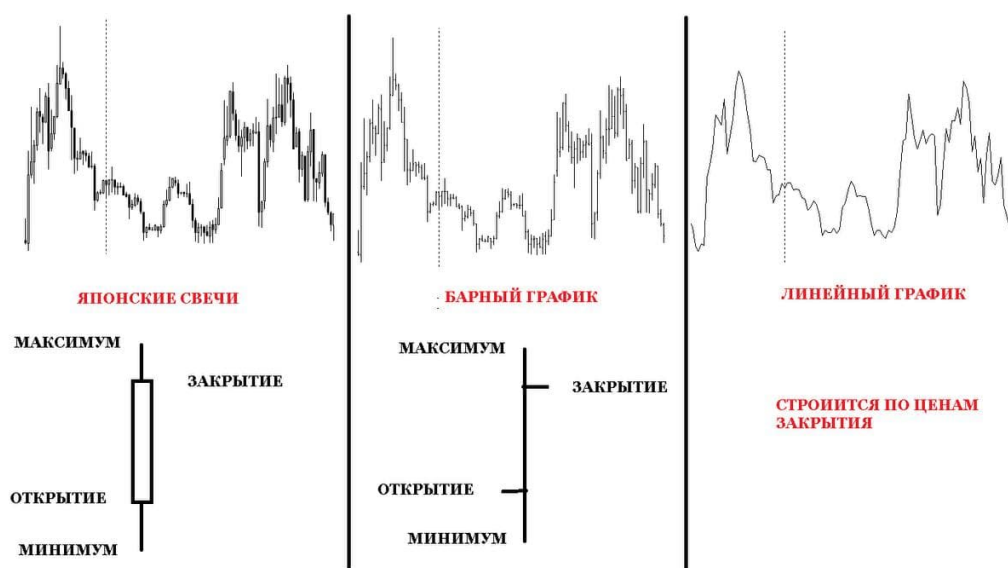


Рис. 1: виды графиков

- **Текстовой информации**

Определение эмоционального настроения и мнений на основе постов и комментариев в соцсетях также помогает спрогнозировать поведение инвесторов.

2.1 Постановка задачи

Дан финансовый временной ряд $X = \{X_1, X_2, \dots, X_N\}$

Составим прогноз тренда основываясь только на ценах, то есть $X_i \in \mathbb{R}_+$
Решим задачу классификации: сопоставим временному интервалу $[t_M, t_L]$,
 $N < M \leq L$ вектор со значениями из \mathbb{Z}_3 , так как тренд бывает трех типов: восходящим, нисходящим и боковым.

3 Временные ряды

3.1 Определения

Дадим несколько определений, связанных с временными рядами:

Временной ряд - это последовательность значений, описывающих протекающий во времени процесс, измеренных в последовательные моменты времени, обычно через равные промежутки.

t - настоящее время,

$t - 1, t - 2, t - 3 \dots$ - прошлое,

$t + 1, t + 2, t + 3 \dots$ - будущее

Лег – временной период из прошлого (задержка), например, лег h соответствует $t - h$, где h , характеризующая разницу во времени между элементами временного ряда, называется лаговой переменной или запаздыванием.

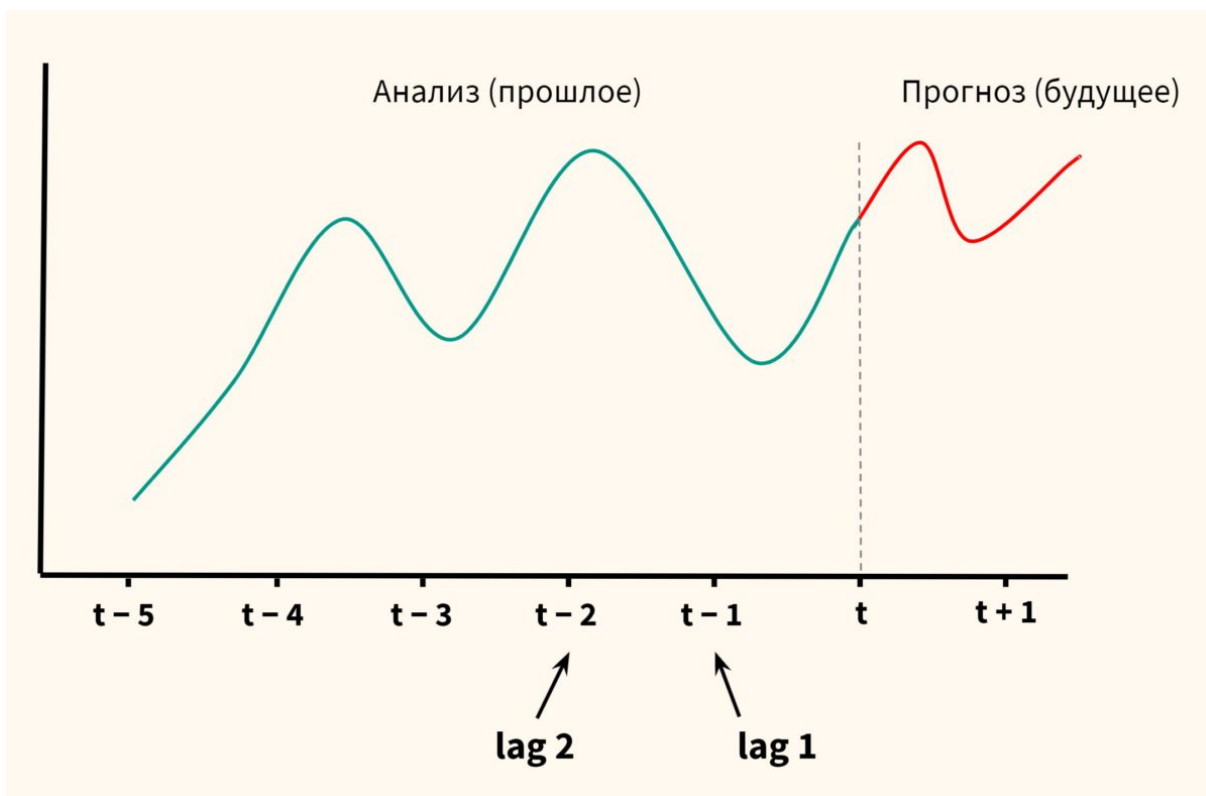


Рис. 2: временной ряд и лаг

Тик - шаг по оси времени $\Delta t = t_i - t_{i+1} = const \forall i$

Тренд – долгосрочное изменение уровня временного ряда

1. Нисходящий(медвежий) – каждый локальный максимум цены ниже предыдущего, как и локальный минимум
2. восходящий (бычий) – каждый локальный минимум цены выше предыдущего, как и локальный максимум
3. боковой (флэт) – локальные максимумы и минимумы примерно на одном значении

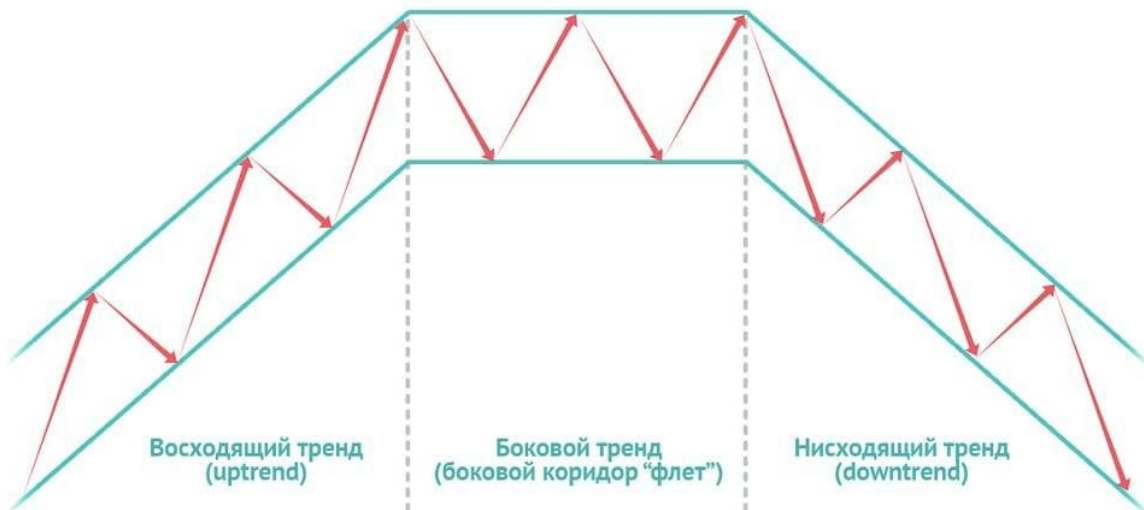


Рис. 3: виды тренда

Сезонность – циклические изменения уровня ряда с постоянным периодом

Шум - случайное изменение в ряду, не коррелирующее с другими данными

Финансовый временной ряд -

3.2 ETS models

ETS (Error-Trend-Seasonality) модель представляет временной ряд как композицию тренда, сезонности и ошибки(шума). Существует аддитивная модель, в общем случае представляющая уровень как сумму компонент:

$$y(t) = Error + Seasonality + Trend$$

Ее стоит использовать, когда величина изменяется линейно (например, у авиакомпании +1000 пассажиров в год), а мультипликатив-

ную

$$y(t) = Error * Seasonality * Trend$$

когда величина изменяется нелинейно (например, количество пассажиров увеличивается каждый год в 2 раза). На рисунке 4 изображено разложение временного ряда на соответствующие компоненты:

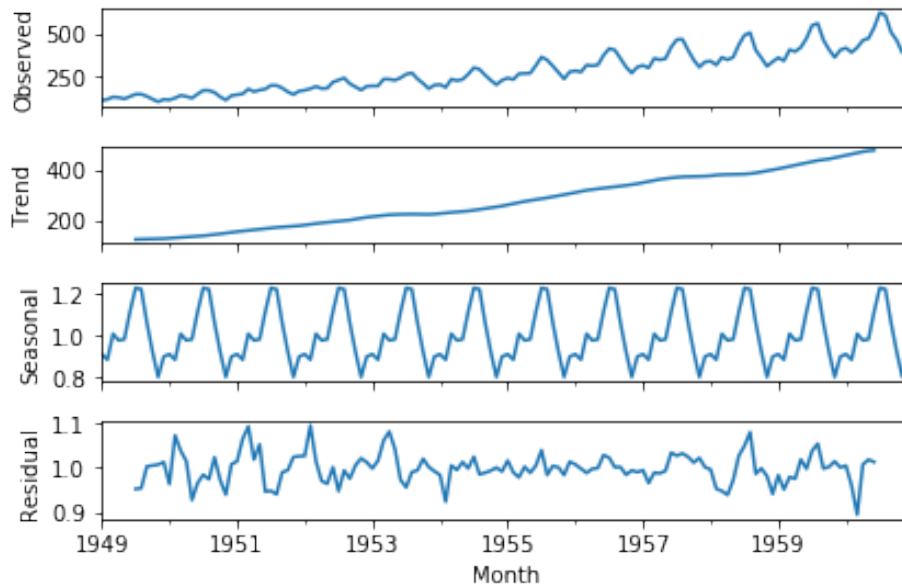


Рис. 4: декомпозиция временного ряда (мультипликативная модель)

4 LSTM

Исследования в области прогнозирования временных рядов ведутся в течение многих лет, но с открытием применения глубокого обучения для решения данной задачи интерес к теме невероятно возрос. За последние десятилетия вышло огромное количество работ о прогнозировании финансовых временных рядов с использованием глубокого обучения (Deep Learning (DL)), рекуррентных

нейронных сетей (Recurrent Neural Networks (RNN)), в частности, LSTM (Long Short-Term Memory) нейронных сетей с долгой краткосрочной памятью.

Минус RNN в невозможности смотреть далеко в прошлое, поэтому, работая в 1991 году над решением проблемы затухающего(исчезающего) градиента (the vanishing gradient problem) уже в 1997 году Зеппом Хохрайтером и Юргеном Шмидхубером была представлена новая архитектура рекуррентной нейронной сети - LSTM - которая смогла эффективно решать следующие задачи[2]:

1. Распознавание долгосрочных закономерностей в зашумленных входных последовательностях
2. Определение в зашумленных входных потоках порядка событий, находящихся во времени далеко друг от друга.
3. Извлечение информации, передаваемой расстоянием между событиями
4. Точная генерация периодических событий, закономерностей.
5. Надежное и длительное хранение действительных чисел

LSTM является самой цитируемой нейронной сетью 20 века [1], а современные алгоритмы LSTM разрабатываются и по сей день и используются для решения широкого спектра задач[2]: распознавание речи, машинный перевод, распознавание видео, распознавание рукописного ввода, прогнозирование временных рядов. LSTM-сети используются в робототехнике, видеоиграх, чат-ботах, в сфере здравоохранения и тд.

4.1 Архитектура

Любая рекуррентная нейронная сеть имеет форму цепочки повторяющихся модулей нейронной сети:

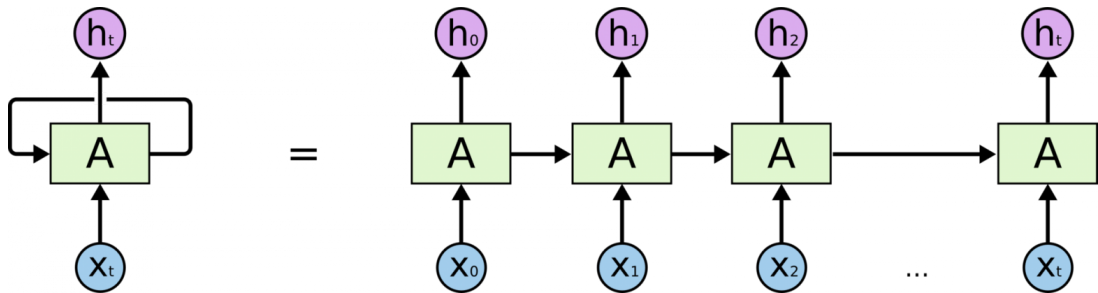


Рис. 5: общий вид рекуррентной нейронной сети (RNN)

Структура LSTM также напоминает цепочку, но модули выглядят иначе. Рассмотрим один из них:

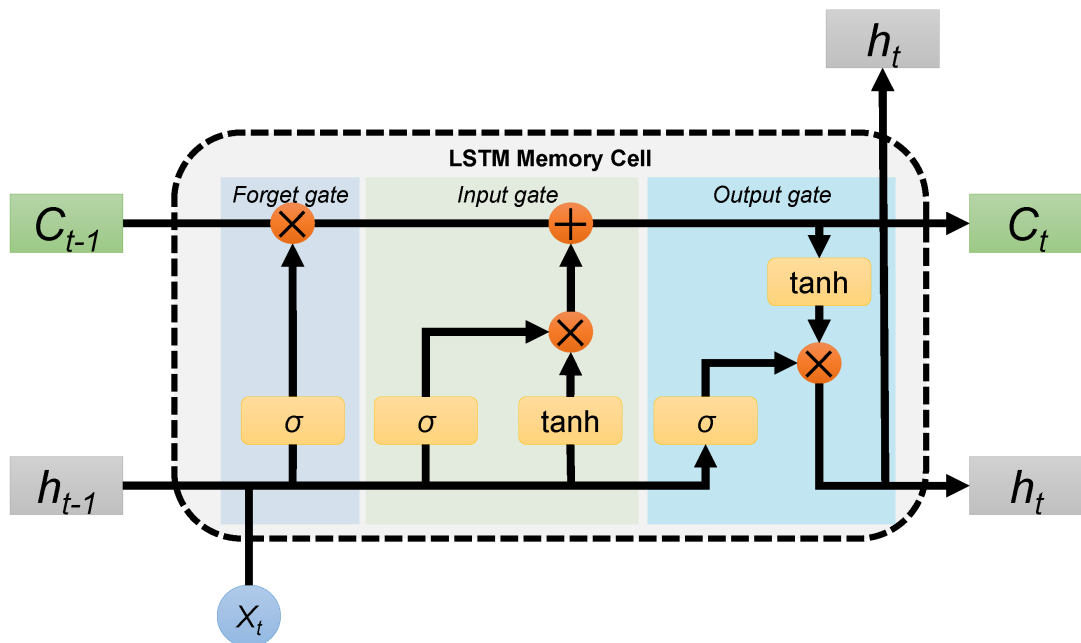


Рис. 6: модуль LSTM

На схеме выше каждая линия переносит целый вектор от выхода одного узла ко входу другого. Сливающиеся линии означают объединение, а разветвляющиеся стрелки говорят о том, что данные копируются и копии уходят в разные компоненты сети.

Красными кружочками обозначены поточечные операции:

$+$ - сложение векторов,

\times - произведение Адамара (поточечное умножение)

Желтые прямоугольники – это обученные слои нейронной сети.

Для более строгого описания схемы напомним вид функций активации:

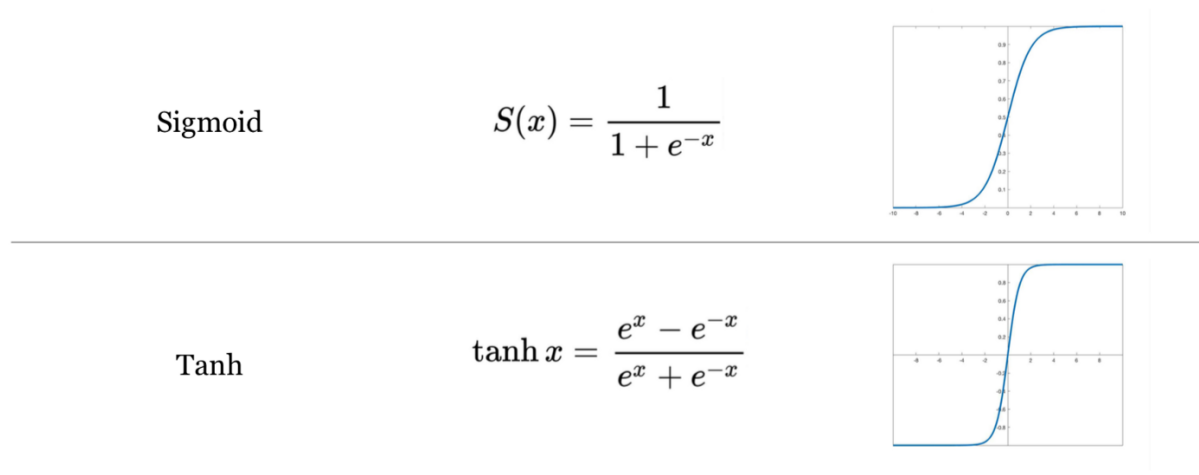


Рис. 7: функции активации

Ключевой компонент LSTM – это состояние ячейки (cell state) – горизонтальная линия, проходящая по верхней части схемы. LSTM может удалять информацию из состояния ячейки; этот процесс регулируется структурами, называемыми фильтрами (gates). Они состоят из слоя сигмоидальной нейронной сети и операции поточечного умножения. Сигмоидальный слой возвращает числа от нуля до единицы, которые обозначают, какую долю каждого блока ин-

формации следует пропустить дальше по сети. Ноль в данном случае означает “не пропускать ничего”, единица – “пропустить все”. В LSTM три таких фильтра, позволяющих защищать и контролировать состояние ячейки.

4.1.1 слой фильтра забывания

Первый шаг – определить, какую информацию можно выбросить из состояния ячейки. Это решение принимает сигмоидальный слой, называемый “слоем фильтра забывания” (forget gate layer). Он смотрит на h_{t-1} и x_t и возвращает число от 0 до 1 для каждого числа из состояния ячейки C_{t-1} . 1 означает “полностью сохранить”, а 0 – “полностью выбросить”.

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f)$$

где W, b - матрица и вектор коэффициентов

4.1.2 слой входного фильтра

Следующий шаг – решить, какая новая информация будет храниться в состоянии ячейки. Этот этап состоит из двух частей. Сначала сигмоидальный слой под названием “слой входного фильтра” (input layer gate) определяет, какие значения следует обновить. Затем tanh-слой строит вектор новых значений-кандидатов \tilde{C}_t , которые можно добавить в состояние ячейки.

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_C * [h_{t-1}, x_t] + b_C)$$

4.1.3 обновляем состояние ячейки

Настало время заменить старое состояние ячейки C_{t-1} на новое состояние C_t . Что нам нужно делать — мы уже решили на предыдущих шагах, остается только выполнить это.

Мы умножаем старое состояние на f_t , забывая то, что мы решили забыть. Затем прибавляем $i_t * \tilde{C}_t$. Это новые значения-кандидаты, умноженные на t — на сколько мы хотим обновить каждое из значений состояния.

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

4.1.4 слой выходного фильтра

Наконец, нужно решить, какую информацию мы хотим получать на выходе. Выходные данные будут основаны на нашем состоянии ячейки, к ним будут применены некоторые фильтры. Сначала мы применяем сигмоидальный слой, который решает, какую информацию из состояния ячейки мы будем выводить. Затем значения состояния ячейки проходят через \tanh -слой, чтобы получить на выходе значения из диапазона от -1 до 1, и перемножаются с выходными значениями сигмоидального слоя, что позволяет выводить только требуемую информацию.

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

4.1.5 Общая формула

Итого, простейший LSTM-модуль можно представить в виде системы уравнений:

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f)$$

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_C * [h_{t-1}, x_t] + b_C)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

5 Модель

5.1 Функция потерь

5.2 Алгоритм оптимизации

5.3 Метрика оценки качества

6 Python

7 Исследование

Список литературы

- [1] The most cited neural networks all build on work done in my labs.Jürgen Schmidhuber (2021, slightly updated 2022) <https://people.idsia.ch/~juergen/most-cited-neural-nets.html> (27.11.2022)
- [2] Jürgen Schmidhuber's page on Recurrent Neural Networks (updated 2017)<https://people.idsia.ch/~juergen/rnn.html> (27.11.2022)
- [3] Financial Time Series Forecasting with Deep Learning : A Systematic Literature Review: 2005-2019<https://arxiv.org/pdf/1911.13288.pdf> (27.11.2022)