УДК 004.9

СРАВНЕНИЕ ТОЧНОСТИ ПРОГНОЗИРОВАНИЯ СТОИМОСТИ АКЦИЙ КОМПАНИИ С ИСПОЛЬЗОВАНИЕМ СВЁРТОЧНОЙ НЕЙРОННОЙ СЕТИ (CNN), ПРОСТОЙ РЕКУРРЕНТНОЙ НЕЙРОННОЙ СЕТИ (SIMPLE RNN) И РЕКУРРЕНТНОЙ НЕЙРОННОЙ СЕТИ НА БАЗЕ ДОЛГОЙ КРАТКОСРОЧНОЙ ПАМЯТИ (LSTM)  
  
Качалов О.И., Миронов А.Н., Володина А.М.

МИРЭА - Российский технологический университет, 119454, Россия, г. Москва, проспект Вернадского, 78, e-mail: [smoltan@bk.ru](mailto:smoltan@bk.ru), [ichcp@mail.ru](mailto:ichcp@mail.ru)

В данной статье приведены результаты сравнения точности прогнозирования стоимости акций компании с использованием свёрточной нейронной сети (CNN), простой рекуррентной нейронной сети (SIMPLE RNN) и рекуррентной нейронной сети на базе долгой краткосрочной памяти (LSTM).

Ключевые слова: прогнозирование, свёрточная нейронная сеть (CNN), простая рекуррентная нейронная сеть (SIMPLE RNN), рекуррентная нейронная сеть на базе долгой краткосрочной памяти (LSTM).

COMPARISON OF THE ACCURACY OF FORECASTING THE VALUE OF THE COMPANY'S SHARES USING A CONVOLUTION NEURAL NETWORK (CNN), A SIMPLE RECURSIVE NEURAL NETWORK (SIMPLE RNN) AND A RECURSIVE NEURAL NETWORK BASED ON LONG-TERM SHORT-TERM MEMORY (LSTM)  
  
Kachalov O.I., Mironov A.N., Volodina A.M.

MIREA - Russian Technological University, 119454, Russia, Moscow, 78 Vernadsky Avenue, e-mail: [smoltan@bk.ru](mailto:smoltan@bk.ru), [ichcp@mail.ru](mailto:ichcp@mail.ru)

This article compares the accuracy of forecasting the value of a company's shares using a convolution neural network (CNN), a simple recursive neural network (SIMPLE RNN) and a long term memory based recursive neural network (LSTM).

Keywords: forecast, convolution neural network (CNN), simple recursive neural network (SIMPLE RNN), long term memory based recursive neural network (LSTM).

**Введение**

Анализ - метод исследования путём рассмотрения отдельных сторон, свойств, составных частей чего-либо. Из определения следует, что основной целью любого анализа является извлечение полезной информации, знаний. Существует большое количество видов анализа. Для анализа стоимости акций лучше всего подходить технический вид анализа. Технический анализ не рассматривает причины, по которым акции изменяют свою стоимость; учитывается только тот факт, что стоимость движется тем или иным образом или колеблется в определенном интервале. Одним из методов технического анализа является применение нейронных сетей. Нейронные сети способны эффективно решить поставленную перед ними задачу, а различные архитектуры нейронных сетей, такие как свёрточные нейронные сети или рекуррентные нейронные сети, способны предоставить гибкий подход к решению поставленных задач.

Предметом исследования данной статьи является сравнение различных архитектур нейронных сетей для технического анализа графика стоимости компании.

**Описание свёрточной нейронной сети (CNN)**

Свёрточная нейронная сеть – архитектура нейронной сети, которая разработана специально для эффективного извлечения абстрактных признаков. Свёрточная нейронная сеть представляет собой множество слоев нейронов, которые соединены между собой операцией свертки, т.е. каждый фрагмент предыдущего слоя умножается на матрицу весов (ядро свертки) поэлементно, результат суммируется и записывается в аналогичную позицию следующего слоя.

Основные характеристики свёрточного слоя:

* Количество признаков (filters)
* Размер ядра свертки (kernel size)
* Смещение (strides) – отвечает за смещение ядра свертки
* Активационная функция (activation function)
* Padding - дополнение к слою

Ядро свертки – матрица весов, с помощью который извлекаются признаки. Именно эти веса изменяются во время обучения нейронной сети. Также ядра свертки называют признаками (features), а набор признаков часто называют картой признаков (feature map).

Операция свертки работает достаточно просто: ядро, размером NxN (размер ядра свертки; kernel size), «скользит» над предыдущем слоем нейронной сети, выполняя поэлементно умножение над частью данных, над которыми ядро находится. Результаты умножений суммируются. Просуммированный результат попадает в активационную функцию, к примеру ReLU, если таковая требуется, и далее результат записывается в следующий слой нейронной сети. Затем ядро смещается на T-элементов (сдвиг; stride) и операция повторяется снова.

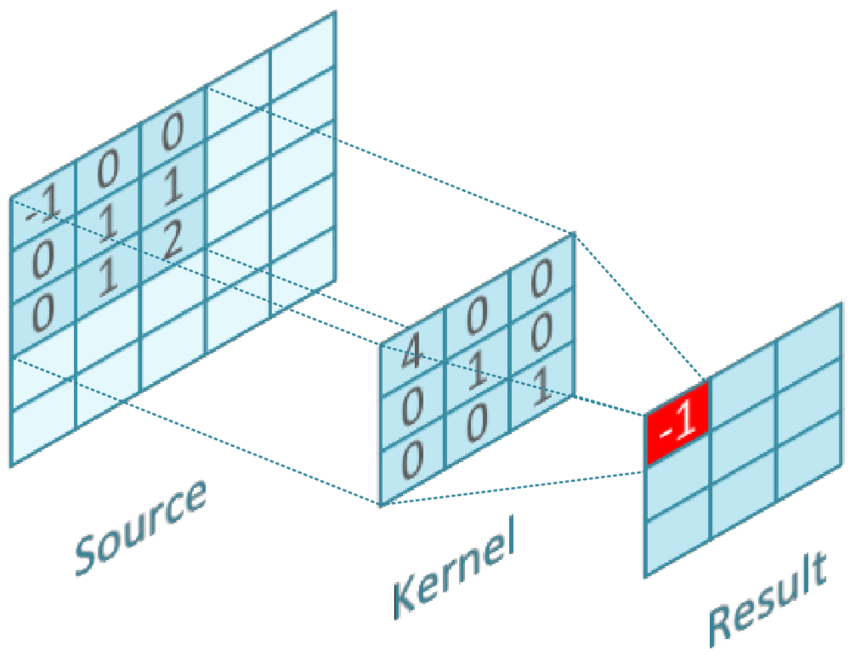


Рис. 1. Пример работы операции свёртки (слой Conv2D)

Обучение свёрточных нейронных сетей происходит с помощью алгоритма обратного распространения ошибки.

В эксперименте будет использоваться архитектура нейронной сети WaveNet. Данная архитектура использует свёрточные слои. Каждый слой нейронной сети представляет собой слой Conv1D с размером ядра 2, смещением 1, padding установлен в такой режим, чтобы размер выходного массива равнялся размеру входного и активационной функцией ReLU. Однако, от слоя к слою изменяется dilation=N, который показывает, что для вычисления будет будет браться каждый N-й нейрон.

A close up of a device

Description automatically generated

Рис. 2. Пример архитектуры нейронной сети WaveNet

Рассмотрим пример работы нейронной сети WaveNet на примере Рис. 2. Первый слой (синий) является рецептором, или входным слоем. Далее производится операция свёртки и происходит переход к первому скрытому (белый) слою. Над ним производится та же операция, что и над входным, однако dilation равен 2. Это значит, что из всех нейронов во скрытом слое будет использоваться каждый второй нейрон. Затем происходит операция свертки и переход ко второму скрытому слою нейронной сети. Во втором скрытом слое dilation равен 4, что значит, что в операции свертки будет использоваться каждый четвертый нейрон. Далее, описанный цикл производится, пока в результате на выходе не останется требуемое количество нейронов.

В эксперименте будет использоваться нейронная сеть с архитектурой WaveNet и будет насчитывать 7 слоев. На вход будет поступает вектор, размерностью (12, 1), что соответствует значению цен акции сегодня и 11 дней в прошлом. На выходе - вектор, размерностью (1, 1), что соответствует одному значению цены акции в будущем.

**Описание простой рекуррентной нейронной сети (Simple RNN)**

Рекуррентная нейронной сети – архитектура нейронной сети, в которой существуют обратные связи между элементами нейронной сети. Рекуррентные нейронные сети имеют свою собственную внутреннюю память (memory sell), поэтому данный тип нейронной сети способен обрабатывать последовательность во времени произвольной длины. Внутренняя память представляет собой маленькую нейронную сеть, которая может быть реализована несколькими способами: простой персептрон (Simple RNN (Рис. 4)), или более сложные структуры сети, такие как GRU или LSTM.

A screenshot of a cell phone

Description automatically generated

Рис. 3. Архитектура свёрточной нейронной сети, используемая в эксперименте

Обучение рекуррентных нейронных сетей происходит с помощью алгоритма обратного распространения ошибки во времени.

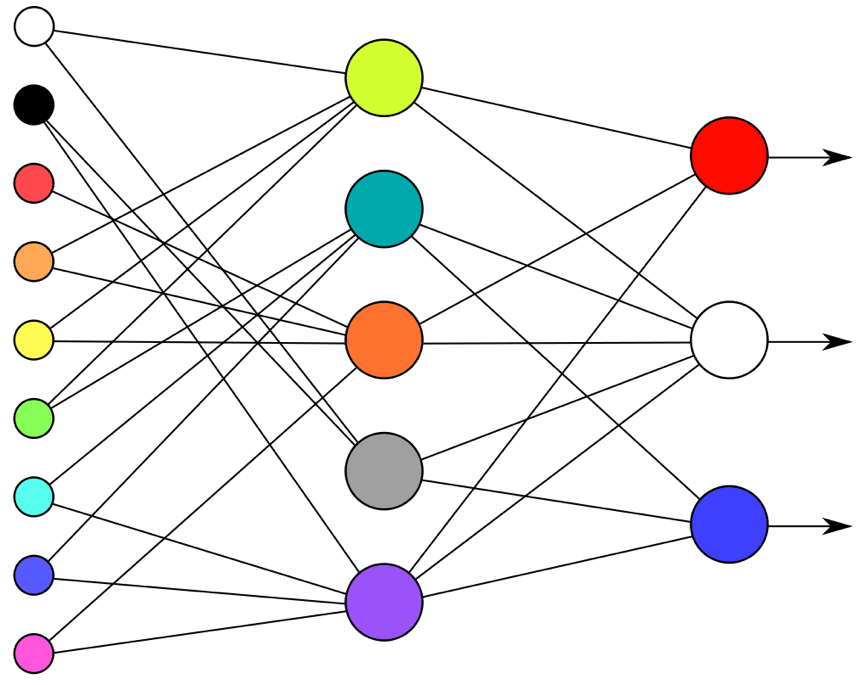


Рис. 4. Логическая схема персептрона

Рекуррентную нейронную сеть следует рассматривать как несколько копий одной и той же нейронной сети, которая передает вектор преемнику, т.е. следующая ячейка памяти получает на вход и новые вектор (X0 … X29 (Рис. 5)), и вектор из предыдущей ячейки (Y0 = H0 … Y29 = H29 (Рис. 5)).

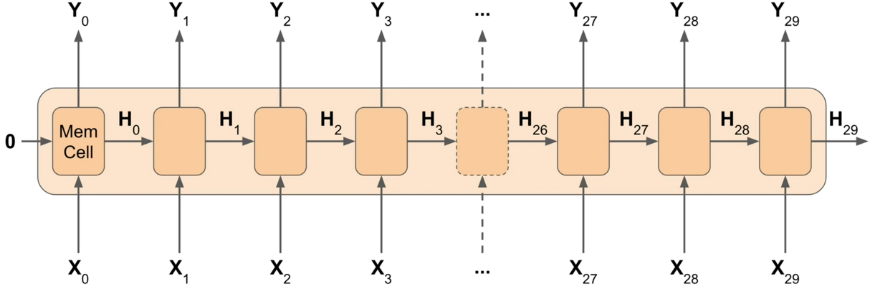


Рис. 5. Пример архитектуры рекуррентной нейронной сети

Как уже было сказано ранее, внутренняя память может быть реализована несколькими способами. Однако результат, который выдает рекуррентная нейронная сеть может отличаться в зависимости от модели:

* Sequence-to-Sequence
* Sequence-to-Vector

Sequence-to-sequence рекуррентная нейронная сеть выдает вектор на каждом моменте времени Y0…Y29 (Рис. 5), в свою очередь sequence-to-vector рекуррентная нейронная сеть выдает только финальный результат Y29 (Рис. 5).

Основные характеристики:

* Количество узлов (units)
* Активационная функция (activation function)
* Передача промежуточного результата в следующий слой (return sequences)

В эксперименте будет использоваться нейронная сеть Sequence-to-sequence рекуррентная нейронная сеть.

A screenshot of a cell phone

Description automatically generated

Рис. 6. Архитектура простой рекуррентной нейронной сети, используемая в эксперименте

**Описание рекуррентной нейронной сети на базе долгой краткосрочной памяти (LSTM)**

LSTM нейронная сеть – разновидность рекуррентной нейронной сети, которая использует более сложную структуру ячеек памяти (Рис. 4).

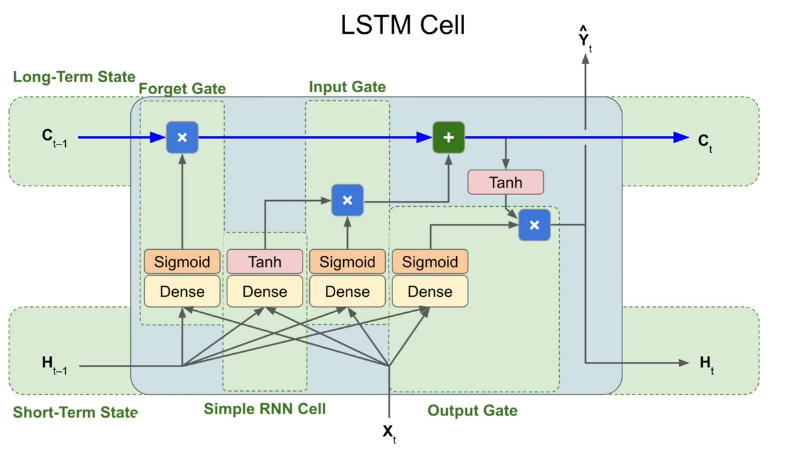


Рис. 6 Логическая схема ячейки LSTM

Также, как и у простой рекуррентной нейронной сети, ячейка LSTM получает на вход (Hi (Рис. 6)) вектор из предыдущей ячейки (Short-Term State). Однако существует второй вектор, который передается между ячейками – Long-Term State. Данный вектор помогает нейронной сети запоминать значения на длинные промежутки времени. Над данным вектором производятся только простые операции, в результате чего хранимое значение не размывается во времени, и градиент не исчезает при использовании метода обратного распространения ошибки во времени при обучении нейронной сети.

**Описание эксперимента**

Цель эксперимента - сравнение качества обучения трёх нейронных сетей на одинаковой обучающей выборке.

Задачи:

* Скачать график стоимости акций компании
* Подготовить обучающую, валидационную и тестовую выборку
* Разработать 3 нейронные сети
* Провести анализ полученных результатов, сделать выводы

A screenshot of a cell phone

Description automatically generated

Рис. 6. Архитектура рекуррентной нейронной сети с элементами LSTM, используемая в эксперименте

**Используемые языки программирования и программные библиотеки**

В качестве основного языка программирования будет использоваться язык высокого уровня Python, версии 3.7. Выбранный язык программирования отлично подходит для разработки искусственных нейронных сетей, поскольку для данного языка существует большое количество мощных библиотек для создания искусственных нейронных сетей.

Для разработки нейронных сетей использовалась библиотека Keras. Keras представляет собой надстройку (высокоуровневый API) для библиотеки Tensorflow. Keras обладает всеми необходимыми слоями для создания представленных нейронных сетей.

Для разработки, обучения и тестирования нейронных сетей использовался сервис Google Colab. Представленных сервис оснащен всеми необходимые средствами разработки и доступны все перечисленные библиотеки.

**Описание способа получения и деления графика**

В качестве графика, на котором будем обучай нейронные сети, возьмем график стоимость акций компании Apple с 2011 года. Всего график насчитывает 2012 значений.

A close up of a map

Description automatically generated

Рис. 7. График стоимости акции компании Apple

Для успешного обучения нейронных сетей требуется разделить график на 3 части: обучающая выборка, валидационная выборка и тестовая выборка. На обучающей выборке будет происходить обучение нейронных сетей. Валидационная выборка нужна для проверки качества обучения во время самого обучения нейронной сети.

Разделим график на 3 части, в произвольном соотношении – 1400 элементов в обучающей выборке, 388 элементов в валидационной выборке и 188 в тестовой выборке. Поскольку при создании обучающей выборке строго ставится в соответствие 12 «прошлых» значений к одному значению «будушего», то суммарное количество элементов в трех обучающих меньше на 36 элементов.

Выборка формируется при помощи «скользящего окна», индивидуально для каждой подвыборки. На вход нейронной сети подается 12 значений на основании которой нейронная сеть будет предсказывать 1 значение. Окно, размером в 12 значений проходит весь график функций, устанавливая соответствие 13-ое значение как ответ. Затем происходит смещение на одну единицу и процесс повторяется снова.

A close up of a map

Description automatically generated

Рис. 8. График обучающей выборки

A close up of a map

Description automatically generated

Рис. 9. График валидационной выборки

A close up of a map

Description automatically generated

Рис. 10. График тестовой выборки

Обучающая выборка – выборка, на которой будет происходить обучение. Валидационная выборка – выборка, при помощи которой будут проверяться качество обучения в процессе обучения. Тестовая выборка – выборка, при помощи которой будет проводиться оценка качества обучения после завершения обучения нейронной сети.

Во время обучения с помощью специальных механизмов (callback) будет производиться контроль за обучением нейронной сети:

* Сохранение самой лучшей версии нейронной сети – в процессе обучения, с помощью валидационной выборки, будет проводиться анализ качества нейронной сети и будет сохраняться самая обученная нейронная сеть.
* Контроль за переобчучением – в процессе обучения будет проводиться анализ ошибки нейронной сети с целью предотвращения переобучения нейронной сети.

**Описание процесса обучения**

В процессе обучения будут сохраняться история значений:

* ошибка сети (loss)
* среднеквадратичная ошибка (mse)
* средняя абсолютная ошибка (mae)
* ошибка значения (value loss)

Обучение нейронной сети будет производиться с помощью метода обучения с учителем, т.е. значению во входной выборке ставится в соответствие значение выходной выборке.

Поскольку начальные веса в нейронных сетях инициализируются с помощью генератора псевдослучайных чисел, каждая архитектура нейронной сети будет обучаться 10 раз и из полученных нейронных сетей будет выбрана сеть с наилучшими показателями.

В качестве функции потерь (loss) будет использоваться функция потерь Хьюбера, в качестве оптимайзера (optimizer) – Adam. Для ускорения обучения batch\_size (количество одновременно поступаемых значений в нейронную сеть) равен 700.

**Результаты обучения**

Для сравнения качества обучения простоим графики ошибки сети, среднеквадратичной ошибки, средней абсолютной ошибки и ошибки значения для каждой нейронной сети и объеденим графики в один график.

Анализируя результаты четырех графиков, можно сделать следующие заключения:

* самая точная нейронная сеть для предсказаний (*ошибка значения* наименьшая) – свёрточная нейронная сеть, на втором месте – простая рекуррентная нейронная сеть и на последнем LSTM нейронная сеть
* самая быстро обучаемая архитектура нейронной сети – LSTM RNN (при условии, что в процессе обучение callback не видит признаков улучшения точности предсказаний сети на протяжении 100 эпох)
* анализируя график *ошибки сети*, *среднеквадратичной и средней абсолютной ошибки* можно сделать вывод, что свёрточная нейронная сеть ведет себя схоже с LSTM RNN, однако точность LSTM RNN сети гораздо хуже
* дольше всех обучалась простая рекуррентная нейронная сеть. Это свидетельствует о том, что на протяжении всего процесса обучения точность предсказаний данной сети постепенно улучшалось, вследствие чего callback не прекращал обучение

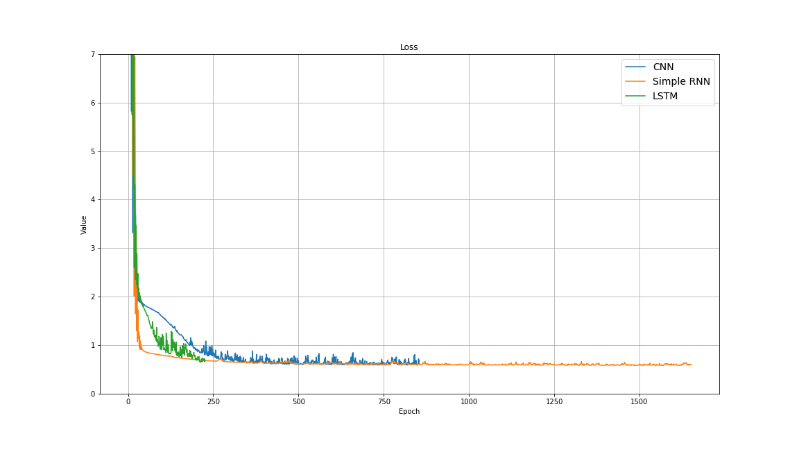


Рис. 11. График ошибки сети

A close up of text on a white background

Description automatically generated

Рис. 12. График среднеквадратичной ошибки



Рис. 13. График средней абсолютной ошибки

A screenshot of a cell phone

Description automatically generated

Рис. 14. График ошибки значения

A screenshot of a cell phone

Description automatically generated

Рис. 15. Сводная таблица результатов

Одним из важных параметров при оценке качества обучения является график *ошибки значения* (Рис. 14). Как видно из графика самая большая ошибка у LSTM сети. Сводная таблица результатов (Рис. 15) в количественном виде показывает, что значение ошибки у LSTM сети существенно больше, нежели у двух других сетей. Это связано с тем, что LSTM рекуррентные нейронные сети крайне эффективнее при использовании больших временных отрезках. В эксперименте используется временной отрезок с 12 значениями и при использовании такого временного отрезка LSTM сеть, по результатам эксперимента, не способна точно работать. В свою очередь свёрточная нейронная сеть и простая рекуррентная нейронная сеть показывает схожие результаты, которые лучше, по сравнению с LSTM сетью.

A picture containing text, map

Description automatically generated

Рис. 16. График предсказаний свёрточной нейронной сети на тестовой выборке

A close up of a map

Description automatically generated

Рис. 17. График предсказаний простой нейронной сети на тестовой выборке

A close up of a map

Description automatically generated

Рис. 18. График предсказаний LSTM нейронной сети на тестовой выборке

Графики предсказаний для тестовой выборки подтверждает, что LSTM нейронная сеть справляется хуже в сравнении с двумя другими сетями, однако, в целом, неплохо повторяет очертание графика.

Проведем еще одно тестирование. Загрузим график стоимости акции компании Google и попробуем предсказать будущие значения. Так же, как и на тестовой выборке, временной отрезок насчитывает 12 значений; результат – одно значение в будущем.

A close up of a map

Description automatically generated

Рис. 19. График предсказаний свёрточной нейронной сети

A close up of a map

Description automatically generated

Рис. 20. График предсказаний простой рекуррентной нейронной сети

A close up of a map

Description automatically generated

Рис. 21. График предсказаний LSTM нейронной сети

В данном тесте просматривается аналогичный результат – LSTM нейронная сеть справляется хуже, нежели другие сети. Однако разброс значений уже гораздо выше. Свёрточная нейронная сеть и простая рекуррентная нейронная сеть справляются одинаково хорошо.

**Выводы**

В результате эксперимента было определено, что свёрточные нейронные сети способны давать аналогичной по качеству прогноз, что и рекуррентные нейронные сети, одновременно с этим демонстрируя высокую скорость обучения за счет более простых операций. Анализируя результаты обучения, было установлено, что свёрточная нейронная сеть, на тестовых выборках, показала практические одинаковый результат, что и простая рекуррентная нейронная сеть. LSTM нейронная сеть справилась хуже, однако это связано с небольшой входной временной последовательностью. В результате эксперимента с уверенностью можно сказать, что технический анализ с помощью машинного обучения эффективен и скорее всего в будущем все больше и больше компаний будут использовать данный подход для анализа.

Список литературы

1. Актуальные киберугрозы — 2018. Тренды и прогнозы. [Электронный ресурс] URL: <https://www.ptsecurity.com/ru-ru/research/analytics/cybersecurity-threatscape-2018/> (16.02.2020)
2. Бодягин Е. Фундаментальный анализ стоимости акций. [Эл. ресурс] // URL: [https://www.cfin.ru/management/ practice/value\_ stock.shtml](https://www.cfin.ru/management/%20practice/value_%20stock.shtml) (Дата обращения: 02.11.19)
3. Keras: The Python Deep Learning library. [Эл. ресурс] // URL: <https://keras.io> (Дата обращения: 28.11.19)
4. Adam: A Method for Stochastic Optimization. [Эл. ресурс] // URL: <https://arxiv.org/abs/1412.6980v8> (Дата обращения: 05.11.19)
5. Generalized Huber Regression. [Эл. ресурс] // URL: <https://towardsdatascience.com/generalized-huber-regression-505afaff24c> (Дата обращения: 29.11.19)

References

1. Actual cyberthreats - 2018. Trends and forecasts. [Electronic resource] URL: https://www.ptsecurity.com/ru-ru/research/analytics/cybersecurity-threatscape-2018/ (16.02.2020).
2. Bodiagin E. Fundamental analysis of share value. [Electronic resource] // URL: https://www.cfin.ru/management/ practice/value\_ stock.shtml (Date of circulation: 02.11.19)
3. Keras: The Python Deep Learning library. [Electronic resource] // URL: https://keras.io (Date of address: 28.11.19)
4. Adam: A Method for Stochastic Optimisation. [Electronic resource] // URL: https://arxiv.org/abs/1412.6980v8 (Date of address: 05.11.19).
5. Generalized Huber Regression. [Electronic resource] // URL: https://towardsdatascience.com/generalized-huber-regression-505afaff24c (Date of address: 29.11.19)