

# Исследование возможностей нейронных сетей для прогнозирования показателей функционирования брокеров сообщений технологических платформ

И.А. Чирков, М.Е. Дунаев

**Аннотация.** Проблема программных сбоев при работе сложных программных систем является экономически ощутимой и, к сожалению, неизбежной. Можно попытаться минимизировать число сбоев путем их прогнозирования, используя информацию журналов событий (logs) отдельных приложений. В настоящей работе исследуются возможности применения нейронных сетей для прогнозирования показателей функционирования приложений технологических платформ. Для обучения сетей используется информация метрики качества MAE (средняя абсолютная ошибка). Сравнивается эффективность применения линейных, глубинных, свёрточных и рекуррентных нейронных сетей, а также модели Хольта-Винтерса и XGBoost. В итоге лучший результат показали рекуррентные нейронные сети (AR LSTM).

**Ключевые слова** – детектирование аномалий, машинное обучение, временной ряд, нейронные сети.

## I. ВВЕДЕНИЕ

Программные сбои являются сегодня ощутимой и неизбежной проблемой при работе корпоративных программных систем. При этом предпринять меры по предотвращению подобных сбоев или их последствий часто не удаётся из-за недостатка времени, отпущенного на эти действия. Поэтому, чтобы избежать сбоев, необходимо пытаться их прогнозировать, основываясь на единственном достоверном источнике информации - журналах событий (англ. logs) приложений [1], [2].

Лог - это файл с записями о различных измеряемых характеристиках событий, расположенных в хронологическом порядке. Сбор логов необходим для всех цифровых устройств и программных систем, чтобы оценивать последовательности их состояний и выявлять аномалии или сбои.

Статья получена 16 июня 2021.

Чирков И. А., магистр, Национальный исследовательский ядерный университет МИФИ (email: senior.chirkov@gmail.com)  
Дунаев М.Е., аспирант, Национальный исследовательский ядерный университет МИФИ (email: max.dunaev@mail.ru)  
Работа представляет собой результат магистерской диссертации

Анализируя логи разными методами, можно формировать последовательности действий администраторам платформ для управления процессами, идущими на платформе в реальном времени, по возможности избегая аварий и поломок. Обычно сбои обнаруживают, следя за превышением пороговых значений критических показателей системы (аномалиями). Детектирование аномалий в деятельности технологических платформ относится к классу задач поиска редких событий, т. е. шаблонов, несоответствующих их ожидаемому поведению.

Выявление аномалий при функционировании сложных технологических объектов в значительной степени основываются на традиционных методах интеллектуального анализа данных и машинного обучения [3]. Но в последнее время внимание исследователей все больше привлекают новые методы, использующие, в том числе и глубокие нейронные сети [4].

Система логирования процессов, идущих на платформе, становится основным источником информации для принятия решений по управлению различными приложениями, включая бизнес-процессы [5]. Обработывая логи, можно решать задачу детектирования процессов с целью снижения числа аварийных ситуаций, то есть предсказывать возможность их наступления и пытаться избежать.

Принимая во внимание значительный объём информации, хранимой в логах и большую вариабельность характеристик событий, задачи детектирования аномальных ситуаций можно пытаться решать с использованием разных методов машинного обучения. Целью данной работы является исследование возможностей нейронных сетей для прогнозирования отдельных характеристик логов при функционировании приложений.

## II. ПОДГОТОВКА К ОБУЧЕНИЮ

Чтобы использовать нейронные сети, работающие с временными рядами, желательно подготовить фрагменты временных рядов без разрывов и без искусственного сшивания фрагментов с поправкой на размер сдвига по времени в местах простоя исследуемого программного комплекса.

При выборе метода решения необходимо учитывать различные факторы. Так, для прогнозирования хорошо структурированных временных рядов, которые можно задать разностными уравнениями (системами разностных уравнений), существуют различные математические методы, которые не требуют специально обустроенной вычислительной системы.

При выборе метода решения необходимо учитывать различные факторы. Для прогнозирования хорошо структурированных временных рядов, которые можно задать системами разностных уравнений, целесообразно использовать известные математические методы, не требующие специальных компьютерных систем. Так же известно, что нейронная сеть может дать верный прогноз, если она хорошо обучена, и при этом не предполагает качественного анализа влияния изменений параметров модели на динамику процесса. [6]. Речь здесь идет не о весах, а о различных факторах процесса, изменение которых может существенно повлиять на его динамику. Сегодня нейронные сети можно эффективно использовать для предсказания событий, если эти события порождены эндогенными факторами [https://elibrary.ru/item.asp?id=44276383]. Если же в данных, используемых для обучения, нет изменений, обусловленных экзогенными факторами, то нейронная сеть будет бессильна в предсказании изменений в будущем.

Важно понимать, что цель использования искусственных нейронных сетей заключается не в том, чтобы вытеснить традиционные методы, а чтобы расширить круг решаемых задач. Дальнейшее изучение нейронных сетей и разработка критериев выбора обучающего метода приведет к более совершенным методам их обучения, и как следствие, к более широкому их использованию.

### III. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И СБОР МЕТРИК

Для экспериментов по исследованию эффективности работы алгоритмов прогнозирования временных рядов необходимо выбрать некоторую метрику журнала событий какого-нибудь известного программного инструмента, на которой будут тестироваться различные модели. Неплохое решение – Apache Kafka – одно из наиболее популярных программных решений для потоковой обработки данных в любом масштабе с открытым исходным кодом. Ее метрика «kafka consumer incoming byte rate» (рис.1), в которую искусственно были включены аномальные данные, полученные путём генерации большого количества сообщений, выбрана нами для тестирования моделей. События в данной метрике хранятся по темам (topics), наборы событий однотипны и не требуют значительной предобработки. Для работы с наборами событий воспользуемся распространенными инструментами мониторинга событий Prometheus и Grafana.



Рис 1 – Визуализация фрагмента временного ряда в Grafana

### IV. ОБУЧЕНИЕ МОДЕЛЕЙ

Сегодня не существует универсального алгоритма, одинаково хорошо пригодного для прогнозирования данных различной структуры. Поэтому для проведения экспериментов на языке программирования Python с использованием популярных библиотек были реализованы семь исследуемых алгоритмов: Хольт-Винтерс, XGBoost, линейная нейронная сеть, DNN, CNN, LSTM, AR LSTM.

Экспериментальный набор данных по классической схеме был разделен на 3 выборки - для обучения (70%), проверки (20%) и тестирования (10%). Для достижения большей реалистичности экспериментов данные набора случайно не перемешивались перед разделением.

Как известно, перед обучением нейронной сети важно провести масштабирование функций. Сделаем это с помощью процедуры нормализации с использованием метода скользящих средних.

Модели, которые будут далее использоваться, делают набор прогнозов на основе окна последовательных выборок данных. Основными характеристиками окон ввода являются ширина - число временных шагов или меток и величина временного сдвига между ними. Типы выборки и используемой модели могут потребовать генерации различных окон данных.

Существует два подхода для построения моделей прогнозирования последовательности будущих значений по текущим значениям всех функций:

1. Прогнозы одиночных снимков, при которых модель прогнозирует сразу весь временной ряд.

2. Авторегрессионные прогнозы, когда модель делает только одношаговые прогнозы, и ее выходные данные возвращаются в качестве входных для следующего шага.

Результаты прогнозирования будем представлять комбинированными графиками, левая часть каждого из которых показывает уже произошедшие события, а правая – предстоящие события. Зелёными маркерами обозначены реальные показатели временного ряда, а оранжевыми - предсказанные.

*Прогнозирование всего временного ряда*

Начнём эксперименты с модели, которая возвращает текущее значение метрики в качестве прогноза следующего временного промежутка (Last). Простой базовый план для этой модели - повторить последний временной шаг ввода для необходимого количества временных шагов вывода. Такая модель имеет право на существование, как исходная модель, если анализируемая метрика может изменяться медленно. Как видно на рис. 2, предсказания такой модели являются неточными (см. рис.2).

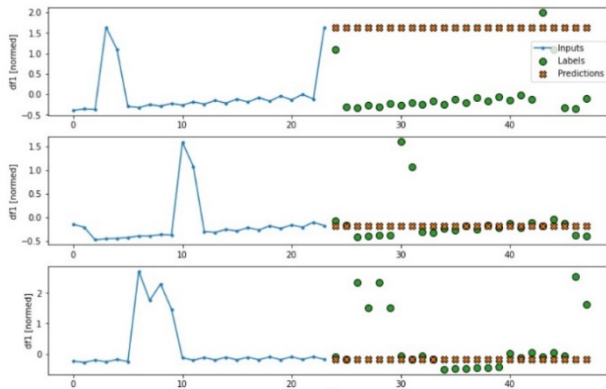


Рис. 2 – Результаты прогнозирования модели «Last».

Другим простым подходом является повторение предыдущих n временных шагов, предполагая, что следующие n шагов будут аналогичными (Repeat). Предсказания этой модели точнее (рис.3), но она, очевидно, не сможет предсказывать неожиданные выбросы.

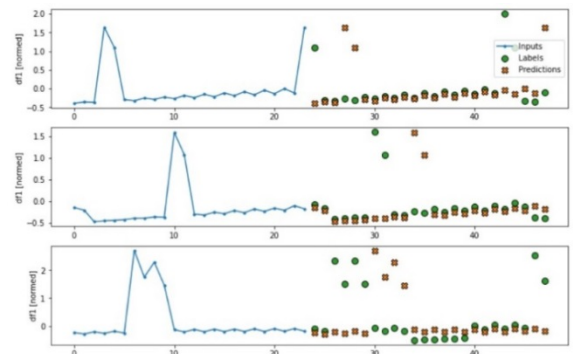


Рис. 3 – Результаты прогнозирования модели «Repeat».

Простая линейная модель, основанная на последнем временном шаге ввода, работает лучше, чем любая базовая, но имеет недостаточную мощность (рис. 4).

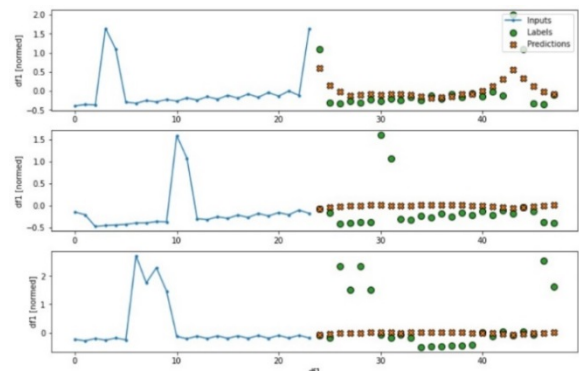


Рис. 4 – Результаты прогнозирования линейной модели.

Добавление дополнительного слоя (layers.Dense) нейронной сети между входом и выходом дает линейной модели больше возможностей, но по-прежнему основывается только на одном временном шаге входа (рис. 5).

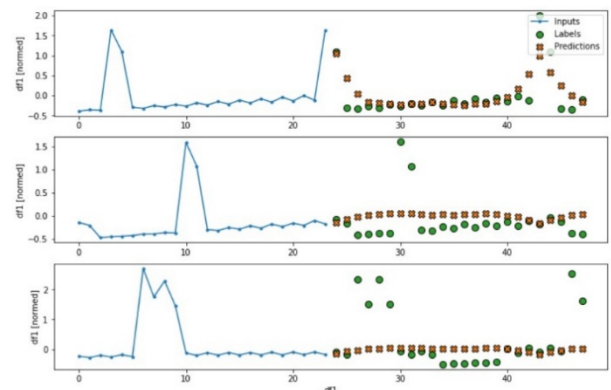


Рис. 5 – Результаты прогнозирования глубокой нейронной сети.

Свёрточная нейронная сеть делает прогнозы на основе истории фиксированной ширины, что может привести к лучшей производительности, чем у глубокой сети, поскольку она может

видеть, как вещи меняются с течением времени (рис.6).

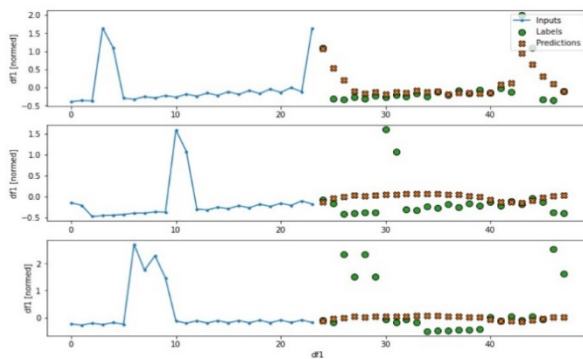


Рис. 6 – Результаты прогнозирования свёрточной нейронной сети.

Рекуррентная модель с длительной кратковременной памятью (LSTM) может научиться использовать длинную историю входных данных, если это имеет отношение к прогнозам, которые делает модель [7], [8]. Здесь модель будет накапливать внутреннее состояние в течение  $n$  временных шагов, прежде чем сделать единственный прогноз на следующие  $n$  шагов (рис.7). При текущем подходе, когда прогнозируется сразу весь временной ряд, LSTM должна выдавать результат только на последнем временном шаге, поэтому откажемся от установки возвращаемых последовательностей (`return_sequences=False`).

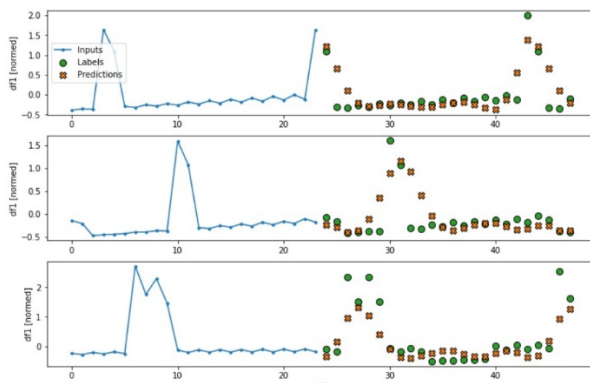


Рис. 7 – Результаты прогнозирования LSTM.

Все вышеперечисленные модели предсказывают всю выходную последовательность за один шаг.

В некоторых случаях модели может быть полезно разложить этот прогноз на отдельные временные шаги [9], [10]. Затем выходные данные возвращаются в качестве входных данных, и прогнозы могут быть сделаны с учетом предыдущего, как в классической генерации последовательностей с помощью рекуррентных нейронных сетей. Одним из явных преимуществ этого подхода является то, что сеть можно настроить для прогнозирования последовательности различной длины.

### Авторегрессивные модели

Построим авторегрессивную модель RNN [11], [12], [13]. Эта модель AR LSTM (рис.8) будет иметь ту же базовую форму, что и одношаговые модели LSTM, в которой есть `layers.Dense` и которые преобразует выходные данные LSTM в прогнозы модели. В этом случае модель должна вручную управлять входными данными для каждого шага, поэтому она использует параметр `layers.LSTMCell`.

Очевидно, что точность прогнозирования увеличивается в зависимости от сложности модели. Для оценки всех исследованных моделей используем среднюю абсолютную ошибку (рис.9) [14], [15].

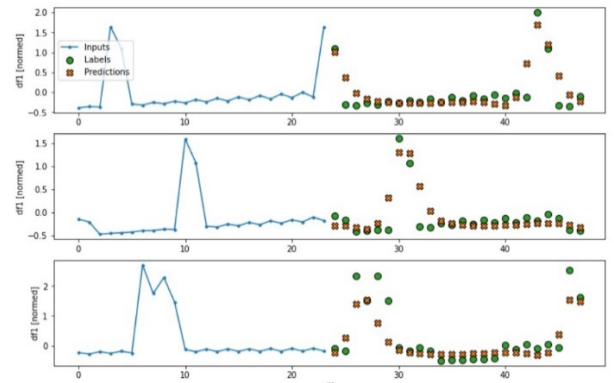


Рис. 8 – Результаты прогнозирования AR LSTM.

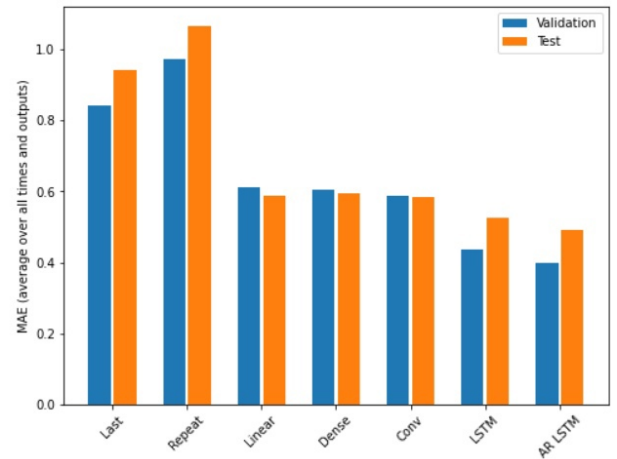


Рис. 9 – Сравнение точности моделей.

Заметна разница в точности между базовыми «Last», «Repeat» моделями и моделями, основанными на нейронных сетях. Прирост, достигнутый при переходе от глубокой нейронной сети к свёрточным, является небольшим.

Для сравнения с нейронными сетями были реализованы классические модели прогнозирования временных рядов: Хольт-Винтерс (рис.10) и XGBoost (рис.11).

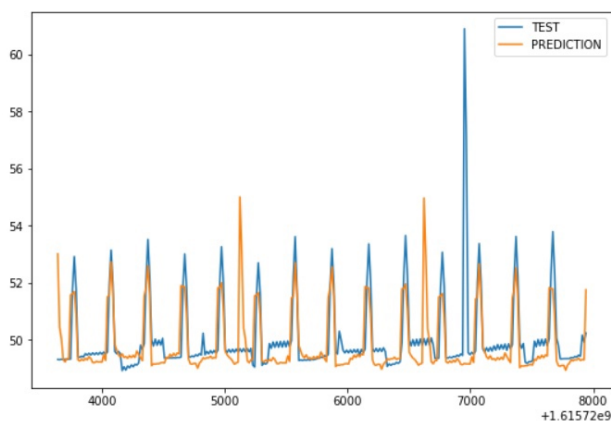


Рис. 10 – Результаты прогнозирования модели Хольта-Винтерса.

После применения тройного экспоненциального сглаживания результаты прогнозирования самые точные. На временном ряду прослеживается аддитивная сезонность.

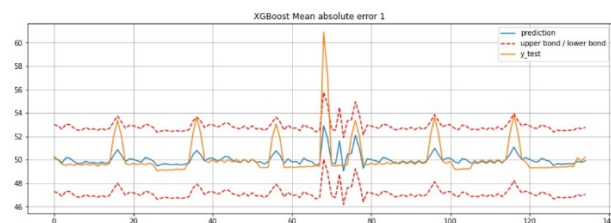


Рис. 11 – Результаты прогнозирования модели XGBoost.

Метод XGBoost имеет хорошую производительность. Из-за этого он является популярным в машинном обучении.

Заметно лучше показали себя рекуррентные нейронные сети. Лучший результат у авторегрессивной рекуррентной нейронной сети AR LSTM. Таким образом, в при решении данной задачи лучшие результаты по точности прогнозирования показали более сложные подходы, основанные на нейронных сетях, но мы бы этого не узнали, не проверив другие подходы.

## V. ЗАКЛЮЧЕНИЕ

Настоящая работа посвящена исследованию и разработке моделей предписывающей аналитики для диагностики деятельности технологических платформ крупных структур и ориентирована на интеллектуальный анализ больших данных.

В ходе работы были изучены особенности сбора данных, чтоб получить временной ряд без разрывов и не делать сшивание ряда в месте простоя. Также стало понятно, что большим удобством является мониторинг метрик брокера Kafka, т.к. набор данных для исследований получается однотипный и не требует глобальной предобработки.

При исследовании алгоритмов поиска частых последовательностей была сформирована выборка данных, на которой протестированы семь алгоритмов, написанных на Python 3. Лучшим оказался AR LSTM в рамках решения конкретных типов задач.

AR LSTM, показавший себя лучше остальных алгоритмов по точности решения, был также реализован на языке Scala для обеспечения большего быстродействия при работе с большими выборками.

После выполнения работы в Федеральной службе по интеллектуальной собственности были зарегистрированы программные комплексы «Анализ временных рядов с помощью Long short-term memoгу на Python», № свидетельства 2021617964 и «Анализ временных рядов с помощью Long short-term memoгу на Scala», № свидетельства 2021618076.

## БЛАГОДАРНОСТИ

Авторы выражают благодарность Высшей инженеринговой школе НИЯУ МИФИ за помощь в возможности опубликовать результаты выполненной работы и профессору Зайцеву К.С. за ценные замечания и полезные советы, сделанные в ходе подготовки данной статьи.

## СПИСОК ЛИТЕРАТУРЫ

- [1] Chandola V., Banerjee A., Kumar V. Anomaly Detection: A Survey // ACM Computing Surveys 41, Vol. 3, No. 15, 2009. pp. 1–58.
- [2] Рысин Н.А. Разработка высоконагруженного решения прогнозирования временных рядов: сб. науч. тр «Хроноэкономика» № 4(25). - Москва 2020 – С. 82-86.
- [3] Девяткин А. В., Извин А. В., Зыков Д. А. Применение метода группового учета аргументов в прогнозировании временных рядов // сб. науч. тр «E-Scio» № 4(25). - Москва 2020.
- [4] Indrajeet Y. Javeri, Mohammadhossein Toutiaee, Ismailcem B. Arpinar, and John A. Miller, Tom W. Miller Improving Neural Networks for Time-Series Forecasting using Data Augmentation and AutoML - University of Georgia, Athens, Georgia 2021.
- [5] Андрюхин Е. В., Ридли М. К., Правиков Д. И. Прогнозирование сбоев и отказов в распределенных системах управления на основе моделей прогнозирования временных рядов // Вопросы кибербезопасности. – 2019. – № 3 (31). – С. 24-32
- [6] Razvan-Gabriel Cirstea, Chenjuan Guo, Bin Yang Graph Attention Recurrent Neural Networks for Correlated Time Series Forecasting -- Full version - Aalborg 2021.

- [7] Selim Furkan Tekin, Oguzhan Karaahmetoglu, Fatih Ilhan, Ismail Balaban, Suleyman Serdar Kozat Spatio-temporal Weather Forecasting and Attention Mechanism on Convolutional LSTMs, Cornell University, 2021.
- [8] Алжеев А.В., Кочкаров Р.А. Сравнительный анализ прогнозных моделей ARIMA и LSTM на примере акций российских компаний. Финансы: теория и практика. 2020;24(1):14-23. DOI: 10.26794/2587-5671-2020-24-1-14-23
- [9] Бледнов Р. Г., Скворцов Н. А. Прогнозирование сбоев приложений по журналам состояния программных объектов // Современные информационные технологии и ИТ-образование. 2019. Т. 15, № 4. С. 945-953. DOI: 10.25559/SITITO.15.201904.945-953
- [10] Николенко С.И., Кадури А., Архангельская Е.В. Глубокое обучение. — СПб.: Питер, 2018. — 480 с.: ил. — (Серия «Библиотека программиста»).
- [11] Барский, А.Б. Логические нейронные сети / А.Б. Барский. - М.: Бином. Лаборатория знаний / Интернет-Университет Информационных Технологий (ИНТУИТ), 2017. - 336 с.
- [12] Галушкин, А.И. Нейронные сети: основы теории. / А.И. Галушкин. - М.: РиС, 2015. - 496 с.
- [13] Хайкин, С. Нейронные сети: полный курс / С. Хайкин. - М.: Диалектика, 2019. - 1104 с.
- [14] Maylawati, Dian S., Saptawati P. Set of Frequent Word Item sets as Feature Representation for Text with Indonesian Slang // Journal of Physics: Conference Series, Vol. 801, No. 1, 2017. P. 012066
- [15] Андреас, Мюллер Введение в машинное обучение с помощью Python. Руководство для специалистов по работе с данными // моногр. / Мюллер Андреас. - М.: Альфа-книга, 2017. - 697 с.

# Investigation of the capabilities of neural networks for predicting the functioning of message brokers of technological platforms

Igor Chirkov, Maxim Dunaev

*Annotation.* The problem of software failures in the operation of complex corporate software systems is economically significant and, unfortunately, inevitable. Therefore, to solve this problem, it is necessary to predict failures in a timely manner, based on information from the event logs (logs) of individual applications.

This paper investigates the capabilities of neural networks for predicting the performance of applications running on technological platforms and detecting anomalous situations. Network training is carried out on the basis of information from the log metrics of individual applications of technological platforms.

Were considered linear, dense, convolutional, recurrent neural networks, as well as the Holt-Winters and XGBoost models. The quality metric MAE (mean absolute error) is used to compare the results. As a result of comparisons, the best result was shown by recurrent neural networks (AR LSTM).

*Keywords* - anomaly detection, machine learning, time series, neural networks.

## REFERENCES

- [1] Chandola V., Banerjee A., Kumar V. Anomaly Detection: A Survey // ACM Computing Surveys 41, Vol. 3, No. 15, 2009. pp. 1–58.
- [2] Rysin N. Development of a highly loaded time series forecasting solution: collection of scientific papers «Chronoeconomics» № 4(25). - Moscow 2020 – pp. 82-86.
- [3] Devyatkin A., Izvin A., Zykov D. Application of the method of group accounting of arguments in time series forecasting // collection of scientific papers «E-Scio» № 4(25). - Moscow 2020.
- [4] Indrajeet Y. Javeri, Mohammadhossein Toutiaee, Ismailcem B. Arpinar, and John A. Miller, Tom W. Miller Improving Neural Networks for Time-Series Forecasting using Data Augmentation and AutoML - University of Georgia, Athens, Georgia 2021.
- [5] Andryukhin E., Ridley M., Pravikov D. Prediction of failures and failures in distributed control systems based on time series forecasting models // Cybersecurity issues. – 2019. – № 3 (31). – pp. 24-32
- [6] Razvan-Gabriel Cirstea, Chenjuan Guo, Bin Yang Graph Attention Recurrent Neural Networks for Correlated Time Series Forecasting -- Full version - Aalborg 2021.
- [7] Selim Furkan Tekin, Oguzhan Karaahmetoglu, Fatih Ilhan, Ismail Balaban, Suleyman Serdar Kozat Spatio-temporal Weather Forecasting and Attention Mechanism on Convolutional LSTMs, Cornell University, 2021.
- [8] Alzheev A., Kochkarov R. Comparative analysis of predictive models ARIMA and LSTM on the example of Russian stocks. Finance: theory and practice. 2020;24(1):14-23. DOI: 10.26794/2587-5671-2020-24-1-14-23
- [9] Blednov R., Skvortsov N. Predicting application crashes based on the state logs of software objects // Modern information technology and IT education. 2019. T. 15, № 4. C. 945-953. DOI: 10.25559/SITITO.15.201904.945-953
- [10] Nikolenko S., Kadurin A., Arkhangelskaya E. Deep learning. — St. Petersburg.: Peter, 2018. — 480 p.: illustrated. — (Series « Programmer's library »).
- [11] Barsky A. Logical neural networks / A. Barsky. - M.: Binomial. Knowledge laboratory / Internet University of Information Technologies (INTUIT), 2017. - 336 p.
- [12] Galushkin A. Neural networks: basic theory / A. Galushkin. - M.: editorial and publishing system, 2015. - 496 p.
- [13] Khaikin S. Neural Networks: Complete Course / S. Khaikin. - M.: Dialectics, 2019. - 1104 p.
- [14] Maylawati, Dian S., Saptawati P. Set of Frequent Word Item sets as Feature Representation for Text with Indonesian Slang // Journal of Physics: Conference Series, Vol. 801, No. 1, 2017. P. 012066.
- [15] Andreas, Muller An introduction to machine learning with Python. Data Scientist's Guide // monograph. / Muller Andreas. - M.: Alpha-book, 2017
- First Chirkov Igor Aleksandrovich. Date of birth: January 13, 1997. Place of birth: Russia, reg. Smolensk, Roslavl. Education: IATE MEFH, «Informatics and Computer Engineering», Bachelor's Degree (2015-2019); NRNU MEFH, «Software Engineering», Master's Degree (2019-2021).
- He works at The Rosatom State Corporation Engineering Division ASE, Engineer from 2019 to the present.
- Second Dunaev Maxim Evgenievich. Date of birth: March 15, 1996. Place of birth: Russia, Morshansk. Education: NRNU MEFH, «applied informatics», Bachelor's Degree (2013-2017); NRNU MEFH, «business Informatics», Master's Degree (2017-2019); NRNU MEFH «Informatics and Computer Engineering», Postgraduate Degree(2019-present).
- M. Dunaev, K. Zaytsev Logs analysis to search for anomalies in the functioning of large technology platforms // Journal of Theoretical and Applied Information Technology, 2019 Vol. 97, No. 11, Q3 pp. 3111-3123;
- M. Dunaev, K. Zaytsev, M. Titov A study of sequential pattern mining algorithms for use in detection of user activity patterns // Journal of Theoretical and Applied Information Technology, 2018 Vol. 96, No. 13, Q3 pp. 4306-4315