

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И ПРИНЯТИЕ РЕШЕНИЙ
ARTIFICIAL INTELLIGENCE AND DECISION MAKING

УДК 004.032.26

DOI: 10.18413/2518-1092-2024-9-4-0-3

Шапалин В.Г.
Николаенко Д.В.

СРАВНЕНИЕ СТРУКТУРЫ, ЭФФЕКТИВНОСТИ
И СКОРОСТИ РАБОТЫ ПОЛНОСВЯЗНЫХ,
СВЕРТОЧНЫХ И РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Донецкий национальный технический университет», ул. Артёма, 58, г. Донецк, 283001

e-mail: shapalinv@gmail.com, dv.nikolaenko@yandex.ru

Аннотация

Рассмотрение оптимальности работы различных архитектур нейронных сетей для различных задач является на данный момент актуальной темой для исследования. В данной статье была рассмотрена эффективность работы полносвязных, рекуррентных и сверточных нейронных сетей в контексте разработки простой модели для прогноза погоды. Были рассмотрены архитектуры и принципы работы полносвязных нейронных сетей, устройство одномерных и двумерных сверточных нейронных сетей, а также архитектуру, особенности, преимущества и недостатки рекуррентных нейронных сетей: простых рекуррентных нейронных сетей, сетей LSTM и GRU, а также их двунаправленные подвиды для каждого из трех вышеперечисленных видов. На основе имеющихся теоретических материалов были разработаны простые нейронные сети для сравнения эффективности той или иной архитектуры, где в качестве критерия выступают время обучения и величина ошибки, а в качестве данных для обучения: температура, скорость ветра и атмосферное давление. Были рассмотрены скорость обучения, минимальное и среднее значение ошибки для полносвязной нейронной сети, сверточной нейронной сети, простой рекуррентной сети, LSTM и GRU, а также для двунаправленных рекуррентных нейронных сетей. На основе полученных результатов был проведен анализ возможных причин эффективности той или иной архитектур. На основе полученных данных были построены графики зависимости скорости работы от величины ошибки для трех исследуемых наборов данных: температуры, скорости ветра и атмосферного давления, а также сделаны выводы об эффективности определенной модели в контексте прогноза временных рядов метеорологических данных. Полученные результаты планируется использовать в дальнейших работах со смежными темами, либо развивать данную тему.

Ключевые слова: LSTM; GRU; двунаправленные рекуррентные нейронные сети; сверточные нейронные сети; keras

Для цитирования: Шапалин В.Г., Николаенко Д.В. Сравнение устройства, эффективности и скорости работы прямых, сверточных и рекуррентных нейронных сетей // Научный результат. Информационные технологии. – Т.9, №4, 2024. С. 21-35. DOI: 10.18413/2518-1092-2024-9-4-0-3

Shapalin V.G.
Nikolayenko D.V.

COMPARISON OF THE STRUCTURE, EFFICIENCY,
AND SPEED OF OPERATION OF FEEDFORWARD,
CONVOLUTIONAL, AND RECURRENT NEURAL NETWORKS

Federal state budgetary educational institution of higher education «Donetsk national technical university»,
58 Artema St., Donetsk, 283001

e-mail: shapalinv@gmail.com, dv.nikolaenko@yandex.ru

Abstract

This article examines the efficiency of fully connected, recurrent, and convolutional neural networks in the context of developing a simple model for weather forecasting. The architectures

and working principles of fully connected neural networks, the structure of one-dimensional and two-dimensional convolutional neural networks, as well as the architecture, features, advantages, and disadvantages of recurrent neural networks—specifically, simple recurrent neural networks, LSTM, and GRU, along with their bidirectional variants for each of the three aforementioned types—are discussed. Based on the available theoretical materials, simple neural networks were developed to compare the efficiency of each architecture, with training time and error magnitude serving as criteria, and temperature, wind speed, and atmospheric pressure as training data. The training speed, minimum and average error values for the fully connected neural network, convolutional neural network, simple recurrent network, LSTM, and GRU, as well as for bidirectional recurrent neural networks, were examined. Based on the results obtained, an analysis was conducted to explore the possible reasons for the effectiveness of each architecture. Graphs were plotted to show the relationship between processing speed and error magnitude for the three datasets examined: temperature, wind speed, and atmospheric pressure. Conclusions were drawn about the efficiency of specific models in the context of forecasting time series of meteorological data.

Keywords: LSTM; GRU; bidirectional recurrent neural networks; convolutional neural networks keras; tensorflow

For citation: Shapalin V.G., Nikolayenko D.V. Comparison of the structure, efficiency, and speed of operation of feedforward, convolutional, and recurrent neural networks // Research result. Information technologies. – Т.9, №4, 2024. – P. 21-35. DOI: 10.18413/2518-1092-2024-9-4-0-3

ВВЕДЕНИЕ

Современные методы машинного обучения, такие как нейронные сети, предоставляют новые возможности для решения задач прогноза временных рядов. Существует множество подходящих для этого архитектур машинного обучения, как полносвязные, так и рекуррентные и сверточные нейронные сети. В существующих статьях рассмотрена теоретическая часть вопроса ускорения обучения нейронной сети [1], а также рассмотрены возможные способы создания нейросетей для наилучшего решения задачи прогноза погоды [2]. Тем не менее, в данных статьях не было уделено внимания сравнению представленных нейронных сетей. Целью данной статьи является проведение сравнительного анализа работы различных типов нейронных сетей в плане обучения и прогноза нескольких метеорологических величин. К рассматриваемым величинам относятся температура, скорость ветра и давление. Основными критериями оценки выступают точность и скорость обучения моделей. Ввиду ограничения на размер статьи, а также по причине ограничений во времени, в данной статье не рассматривались более сложные архитектуры, а именно трансформеры и остаточные сети. В рамках исследования будут рассмотрены следующие аспекты: теоретические основы сверточных нейронных сетей, RNN, а также их модификаций, таких как Long-short term memory (LSTM) и Gated recurrent unit (GRU).

Данное исследование не только демонстрирует применение различных архитектур нейронных сетей для прогноза погоды, но и позволяет оценить первичную оценку эффективности тех или иных моделей, а полученные результаты могут быть использованы для дальнейших исследований в смежных областях. Следует заметить, что выявление баланса между точностью и скоростью обучения позволяет оптимизировать использование вычислительных ресурсов, что особенно важно для приложений с ограниченными возможностями, или приложениями, работающими в реальном времени.

1. Рассмотрение эффективности прогноза с помощью полносвязной нейронной сети

Общая схема полносвязной нейронной сети представлена на рисунке 1.

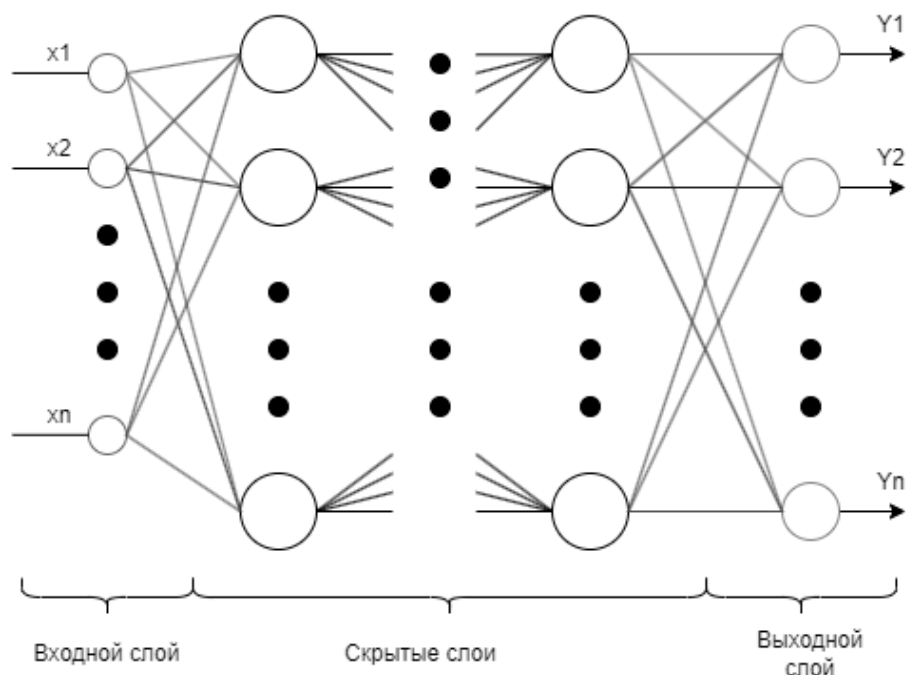


Рис. 1. Общая схема полносвязной нейронной сети
Fig. 1. General scheme of feedforward neural network

где x_1 - x_n – входные параметры, Y_1 - Y_n – выходные параметры. В зависимости от требований задачи входных и выходных параметров может быть любое количество. Совокупность входных параметров образует входной слой, на который подаются данные. Совокупность выходных параметров образует выходной слой, откуда извлекаются полученные результаты. Между ними находится неограниченное количество скрытых слоев, представляющих группу нейронов, где выходы предыдущего слоя поступают на входы следующего слоя. [3-4] Однако, полносвязная нейронная сеть имеет ряд недостатков. В контексте задачи, данная сеть плохо обрабатывает набор последовательных данных, где существуют временные зависимости. Это можно объяснить несвязностью отдельных выходных данных – при выводе более поздних данных, нейросеть не учитывает предыдущие выходные данные [5-6].

Для рассмотрения эффективности данной модели была разработана полносвязная нейронная сеть. В качестве обучающих данных был выбран датасет с kaggle, где указан почасовой отчет о погодных условиях с 2006 по 2016 год [7]. Для простоты работы, в нашем случае из всех данных будет использованы только данные о температуре, скорости ветра и давлении, а также сокращен период до 4 лет. Перед разработкой архитектуры, необходимо разметить и упорядочить данные. Обработка исходного файла, а также обучение модели будет происходить на языке python. Исходный код для разметки данных, а также для обучения расположен в публичном github репозитории [8]. Для всех случаев количество параметров будет равно 10000 с погрешностью до 400 параметров.

Нейросеть будет построена с помощью фреймворка keras [9]. Она состоит из 2х скрытых слоев: первый полносвязный слой, и второй из одного нейрона для вывода результата в виде одного числа. Размер одного батча (отдельного пакета данных для обучения) – 5 элементов, количество эпох – 30. Результат обучения представлен в таблице 1.

Таблица 1

Результат обучения полносвязной нейронной сети

Table 1

Result of the training feedforward neural network	
Время обучения	9:19
Минимальная ошибка обучающей выборки	1.46
Минимальная ошибка тестовой выборки	1.47

2. Рассмотрение эффективности прогноза с помощью одномерной сверточной нейронной сети

Альтернативой для решения подобных задач может являться сверточная нейронная сеть. Основная идея данной нейронной сети заключается в операции свёртки. Сверткой является процесс поэлементного умножения и суммирования элементов исходного n-мерного массива чисел с элементами ядра свёртки с дальнейшим смещением ядра вдоль элементов массива и дальнейшим повторением поэлементного умножения и сложения [10-11]. Свертка может производиться также над элементами одномерного массива. Ядро свертки в таком случае тоже будет одномерным [12]. На рисунке 2 показан пример операции свертки с двумерным массивом.

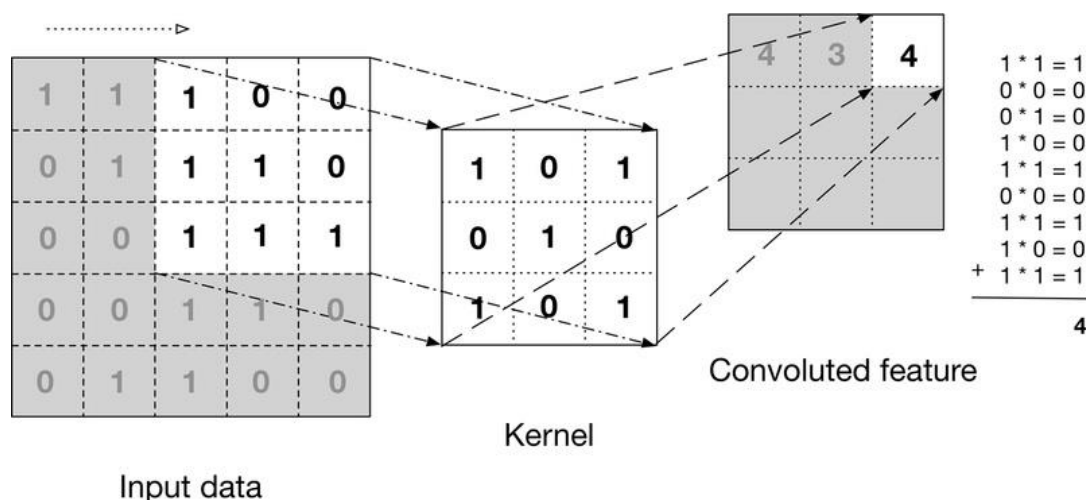


Рис. 2. Пример операции свертки над двумерным массивом данных
Fig. 2. Example of convolution operation of two-dimensional data array

В контексте машинного обучения, элементы ядра матрицы могут быть представлены в виде изменяемых весов. Основное применение подобных нейронных сетей заключается в анализе или преобразовании изображений из-за способности анализировать данные в двумерном представлении. Однако, одномерные сверточные нейронные сети также могут использоваться для анализа временных рядов. Наиболее эффективны данные сети для анализа некоторых повторяющихся локальных закономерностей.

Простая архитектура сверточной нейронной сети состоит из двух слоев: непосредственно слой свёртки и слой подвыборки. Слой подвыборки необходим для уменьшения количества признаков. Основными методами уменьшения признаков является нахождение максимального элемента из ряда имеющихся, либо нахождение среднего значения. Тем не менее, функция подвыборки может быть любой, а использование определенной функции зависит от поставленной задачи [13-14].

Учитывая представленные выше данные, создадим простую одномерную сверточную нейронную сеть для обработки временных последовательностей данных путем замены

полносвязного слоя сверточным. Схематическое изображение обработки данных показано на рисунке 3, результат обучения показан в таблице 2:

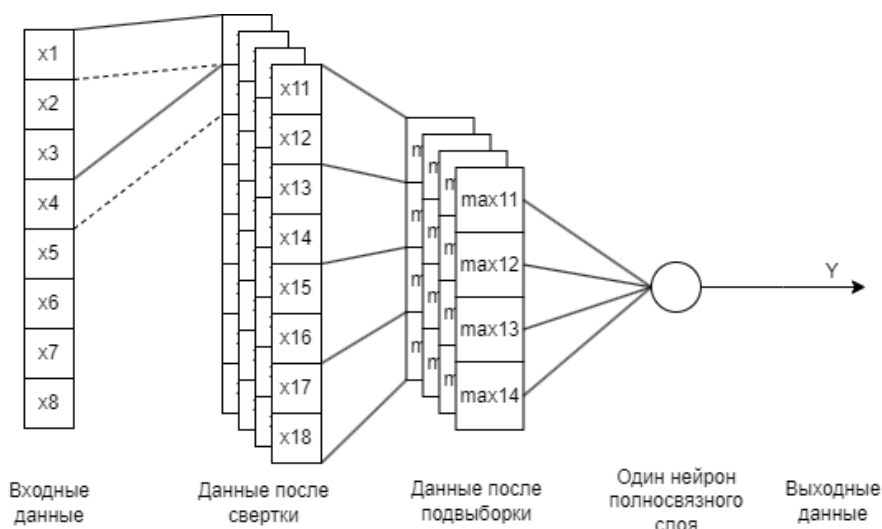


Рис. 3. Преобразование данных в сверточной нейронной сети с одним слоем свертки и подвыборки

Fig. 3. Data transformation in convolutional neural network with single convolutional and pooling layers

Таблица 2

Результат обучения сверточной нейронной сети

Table 2

Result of the training convolutional neural network

Время обучения	13:45
Минимальная ошибка обучающей выборки	1.61
Минимальная ошибка тестовой выборки	1.46

Как можно заметить, при одинаковом количестве параметров, общая ошибка стала больше, несмотря на большее время обучения, что говорит о неэффективности данной архитектуры для нашей задачи. Дополнительным признаком неэффективности модели является большая разница между тренировочной и тестовой ошибками, что говорит о том, что данная нейросеть не может извлечь признаки, необходимые для верного прогноза следующего значения.

3. Рассмотрение эффективности работы рекуррентных нейронных сетей

Необходимо также рассмотреть архитектуру рекуррентных нейронных сетей, которые хорошо показывают себя в обработке временных последовательностей. Первой рассматриваемой архитектурой является архитектура простой рекуррентной нейронной сети [15]. Основная идея заключается в том, что выходные данные подаются обратно на вход по формуле 1:

$$Y = W_1 * X + W_2 * Y_{-1} + b; \quad (1)$$

где X – входное значение,

Y – выходное значение,

Y_{-1} – выходное значение на предыдущем шаге,

$W_{1,2}$ – соответствующие веса,

b – смещение.

Схема одной рекуррентной ячейки представлена на рисунке 4.

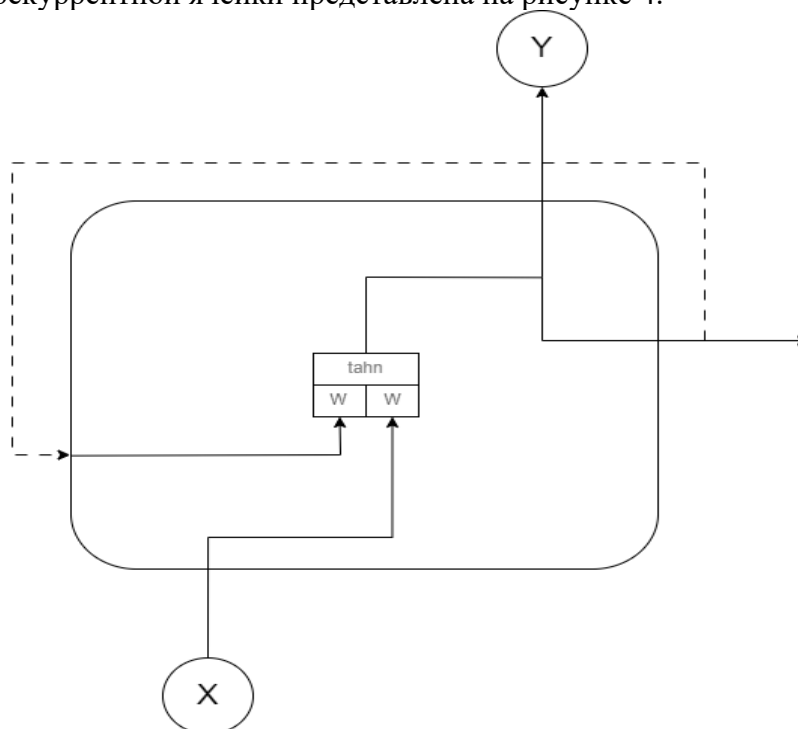


Рис. 4. Схема одиночной ячейки рекуррентной нейронной сети
Fig. 4. Scheme of single recurrent neural network cell

После обработки входного значения x_1 , выходное значение H_1 (оно же скрытое состояние) отправляется обратно на вход, и после умножения на весовой коэффициент суммируется с новым входным значением x_2 и коэффициентом смещения b , который не отображен на рисунке 4. Полученная сумма затем передается на функцию активации и является новым скрытым состоянием. Данный процесс повторяется n раз, в зависимости от параметров, заданных разработчиком. При этом нужно помнить, что веса остаются теми же. Эта особенность рекуррентных нейронных сетей обеспечивает модель способностью обобщать на последовательных данных и захватывать временные зависимости, что является основной идеей RNN. При увеличении количества нейронов, увеличится количество весовых коэффициентов для входного и скрытого состояния, и соответственно, увеличится количество скрытых состояний. Они будут представлены в виде вектора, а весовые коэффициенты будут представлены в виде матрицы весов и смещений [16-17].

Результат работы простой рекуррентной нейронной сети показан в таблице 3.

Таблица 3

Результат обучения простой рекуррентной нейронной сети

Table 3

Result of the training simple recurrent neural network	
Время обучения	40:55
Минимальная ошибка обучающей выборки	1.81
Минимальная ошибка тестовой выборки	1.89

Как видно из результатов, ошибка стала немного больше по сравнению с полносвязной нейронной сетью, но время обучения кардинально выросло. Исходя из полученных данных можно сказать, что простая рекуррентная нейронная сеть работает примерно так же, если не хуже в сравнении с полносвязной сетью. Причину можно объяснить проблемой "затухающего" или "взрывающегося" градиента. Как можно заметить на рисунке 5, рекуррентную нейронную сеть

можно представить, как последовательность нейронов, где выход из одной ячейки идет на вход следующего.

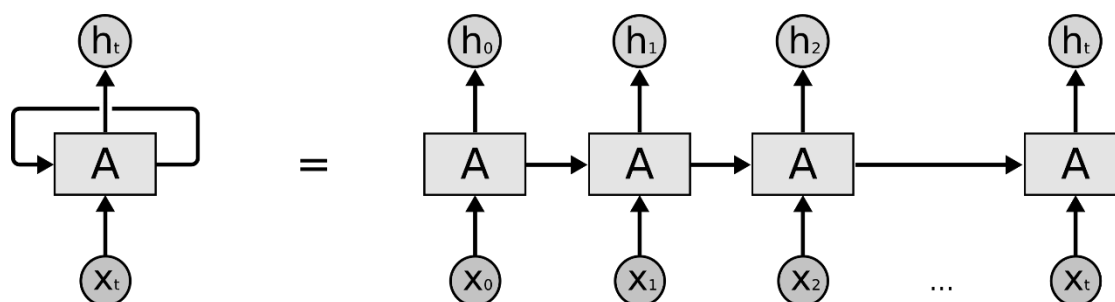


Рис. 5. Развернутая схема рекуррентной нейронной сети
Fig. 5. Unfolded scheme of recurrent neural network

Обучение подобной сети проходит с помощью стандартного метода обратного распространения ошибки, который используется и в полносвязных сетях. Отличие заключается в том, что в рекуррентных нейронных сетях градиент проходит через все временные шаги в обратном направлении, поэтому данный метод называется обратное распространение ошибки через время (backpropagation through time) [18-19]. Временные шаги в данном случае можно представить как слои нейронной сети. При прохождении градиента через большое количество слоев, значение градиента может уменьшиться, приближаясь к нулю, из-за чего значения весов практически не будут обновляться. Это является проблемой затухающих градиентов. В рекуррентных нейронных сетях может быть и обратная ситуация – когда градиент начинает экспоненциально расти, что приводит к очень нестабильному изменению весов, и соответственно, обучения. Это является проблемой взрывающихся градиентов.

4. LSTM

Для решения данной проблемы в рекуррентных нейронных сетях была разработана сеть Long-short term memory (LSTM). Основная идея заключалась в добавлении новых необходимых данных в память, и управлении добавлением с помощью «врат» (gates). Схема LSTM представлена на рисунке 6:

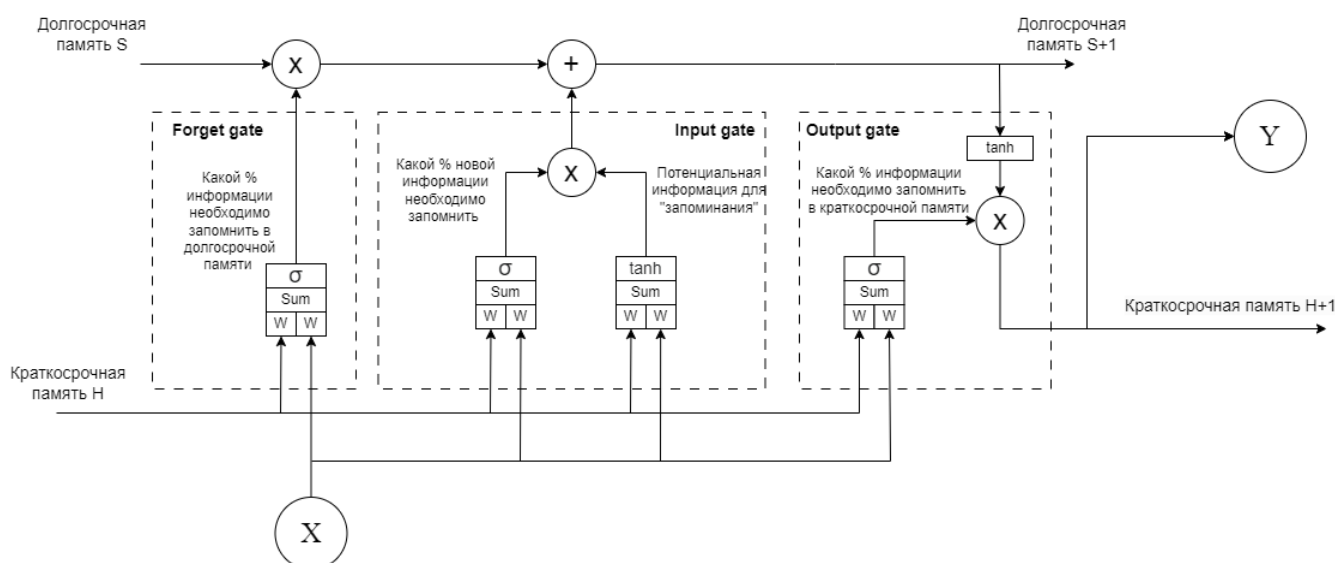


Рис. 6. Схема работы LSTM
Fig. 6. Scheme of LSTM work

В keras LSTM использует 3 вида врат: forget gate, input gate и output gate:

- Forget gate используется для «забывания» лишней информации, а точнее для запоминания нужной. На его входы подается входные данные и информация с краткосрочной памяти. На выходе используется функция активации σ , принимающая значения от 0 до 1, и в своей сути показывающая, какой процент информации долгосрочной памяти необходимо забыть.

- Input gate используется для запоминания новой информации. Входной сигнал и краткосрочная память проходят через два элемента со своими весами, но разными функциями. Как видно из рисунка 6, правая имеет функцию активации \tanh , принимающая на выходе значения от -1 до 1, и представляющая действительный сигнал, тогда как левый элемент определяет процент полезной информации, который будет просуммирован с долгосрочной памятью.

- Output gate используется для определения информации, которая идет на выход. После обработки долгосрочной памяти с помощью функции \tanh , также определяется процент запоминаемой информации в краткосрочной памяти с помощью той же функции σ . Данное значение также является выходным сигналом Y [20].

Заменяем слой простой рекуррентной нейронной сети на слой LSTM. Результат указан в таблице 4:

Таблица 4

Результат обучения нейронной сети LSTM

Table 4

Result of the training LSTM neural network

Время обучения	80:29
Минимальная ошибка обучающей выборки	1
Минимальная ошибка тестовой выборки	1.17

Как можно заметить, результат работы стал заметно лучше. Однако, существенно увеличилось время обучения (примерно в 2 раза по сравнению с простой RNN).

5. GRU

Для ускорения обучения, была создана сеть Gated recurrented units (GRU), схема которой показана на рисунке 7.

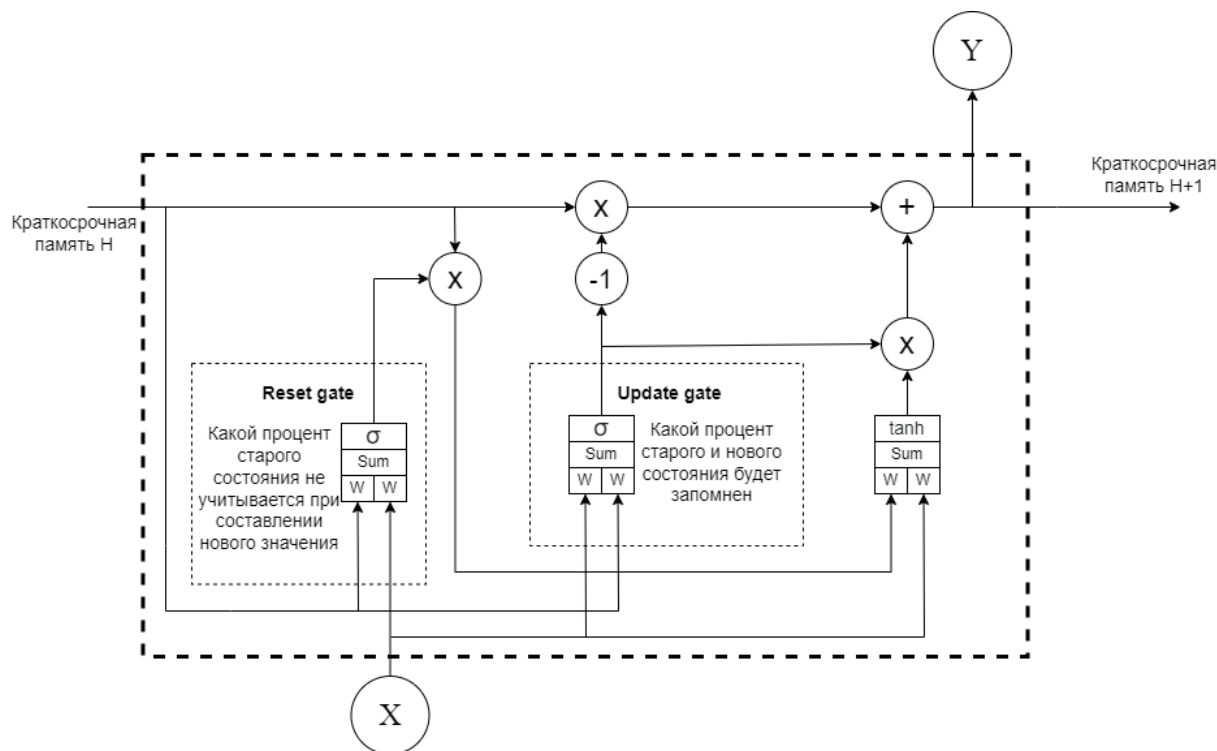


Рис. 7. Схема работы GRU
Fig. 7. Scheme of GRU work

Основной смысл заключается в замене входных и выходных ворот одними воротами обновления (Update gate), что уменьшает общее количество параметров, улучшая точность работы, практически не меняя при этом в скорости обучения. Также убирается долгосрочная память, и используется только краткосрочная. Преимуществом данного подхода также является более простая реализация по сравнению с LSTM. Ворота обновления одновременно определяют, какой процент данных будет добавлен к скрытому состоянию, и какой процент данных в краткосрочной памяти будет отброшен. Символ «-1» обозначает операцию 1-х, следовательно, количество запоминаемой памяти будет обратно пропорциональным количеству добавляемой новой памяти [21]. Заменяем слой LSTM на слой GRU, полученный результат показан в таблице 5.

Таблица 5

Результат обучения нейронной сети GRU

Table 5

Result of the training GRU neural network

Время обучения	91:19
Минимальная ошибка обучающей выборки	0.99
Минимальная ошибка тестовой выборки	1.18

Можно заметить, что при примерно равном количестве параметров точность увеличилась незначительно. Однако, также незначительно увеличилось время обучения, из чего следует, что в общем случае они равноценны, и предпочтительность того или иного метода зависит от конкретных условий задачи.

6. Рассмотрение эффективности двунаправленных рекуррентных нейронных сетей

Последним рассматриваемым подвидом рекуррентных нейронных сетей являются двунаправленные нейронные сети. В общем случае они представляют из себя два набора

рекуррентных нейронных сетей, принимающих и анализирующих данные в прямом и обратном порядке. После обработки выходные данные из двух наборов объединяются согласно определенному алгоритму, выбираемому при разработке – суммирование, конкатенация и др. [22-23]. Основная идея данных нейронных сетей является в поиске определенных паттернов, которые можно обнаружить при рассмотрении временной последовательности в обратном направлении. Общая схема двунаправленной нейронной сети показана на рисунке 8^[24].

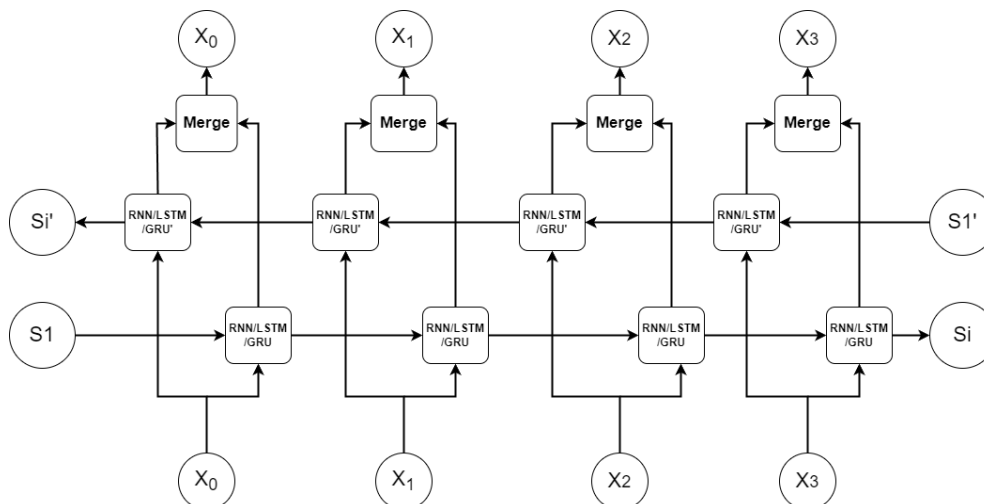


Рис. 8. Общая схема двунаправленной рекуррентной нейронной сети
Fig. 8. General scheme of bidirectional recurrent neural network

Создадим 3 различные двунаправленные рекуррентные нейронные сети, отличающиеся типом ячеек. Результат работы различных двунаправленных нейронных сетей показаны в таблицах 6, 7 и 8.

Результат обучения двунаправленной простой рекуррентной нейронной сети

Таблица 6

Table 6

Result of the training bidirectional simple recurrent neural network

Время обучения	80:27
Минимальная ошибка обучающей выборки	1.7
Минимальная ошибка тестовой выборки	1.8

Результат обучения двунаправленной LSTM

Таблица 7

Table 7

Result of the training bidirectional LSTM

Время обучения	97:12
Минимальная ошибка обучающей выборки	1.05
Минимальная ошибка тестовой выборки	1.22

Таблица 8

Результат обучения двунаправленной GRU

Table 8

Result of the training bidirectional GRU

Время обучения	108:16
Минимальная ошибка обучающей выборки	1.07
Минимальная ошибка тестовой выборки	1.21

Во всех трёх случаях при увеличении времени обучения значение ошибки не уменьшилось, а даже немного увеличилось, что уже говорит о слабой эффективности данной архитектуры.

Аналогично предыдущим пунктам, выведем результаты обучения для значений скорости ветра и давления. Итоговые результаты по всем трем наборам данных сведены к точечным диаграммам на рисунках 9, 10 и 11, где показана зависимость ошибки от времени обучения (Dense – полносвязный слой, CNN – сверточная нейронная сеть, SimpleRNN – простая рекуррентная нейронная сеть, Bidir – двунаправленная НС).

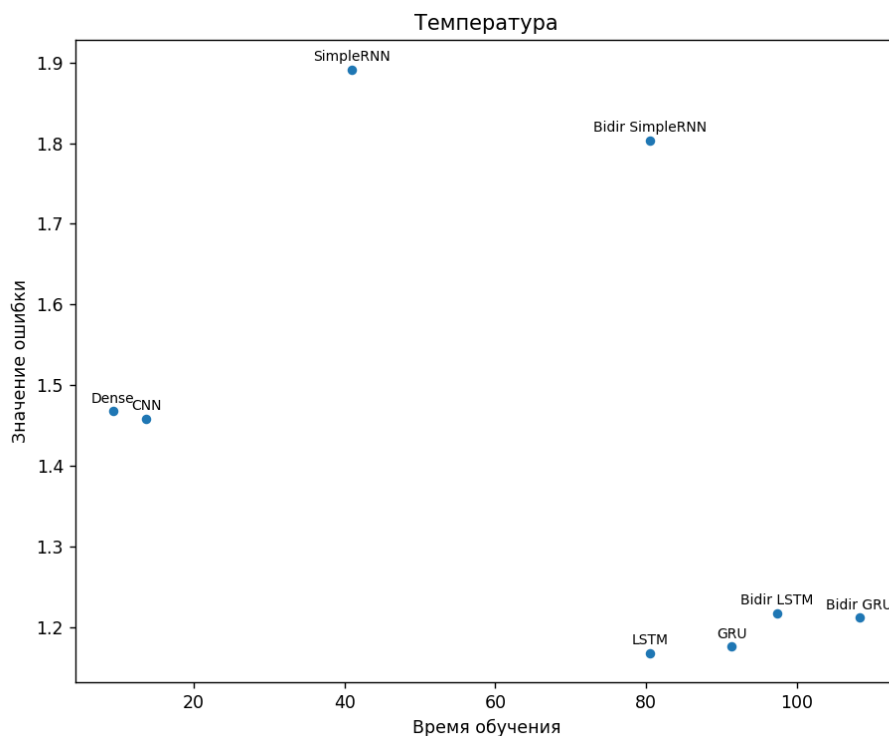


Рис. 9. Точечная диаграмма зависимости ошибки от времени обучения. Температура

Fig. 9. Error-training time scatter plot. Temperature

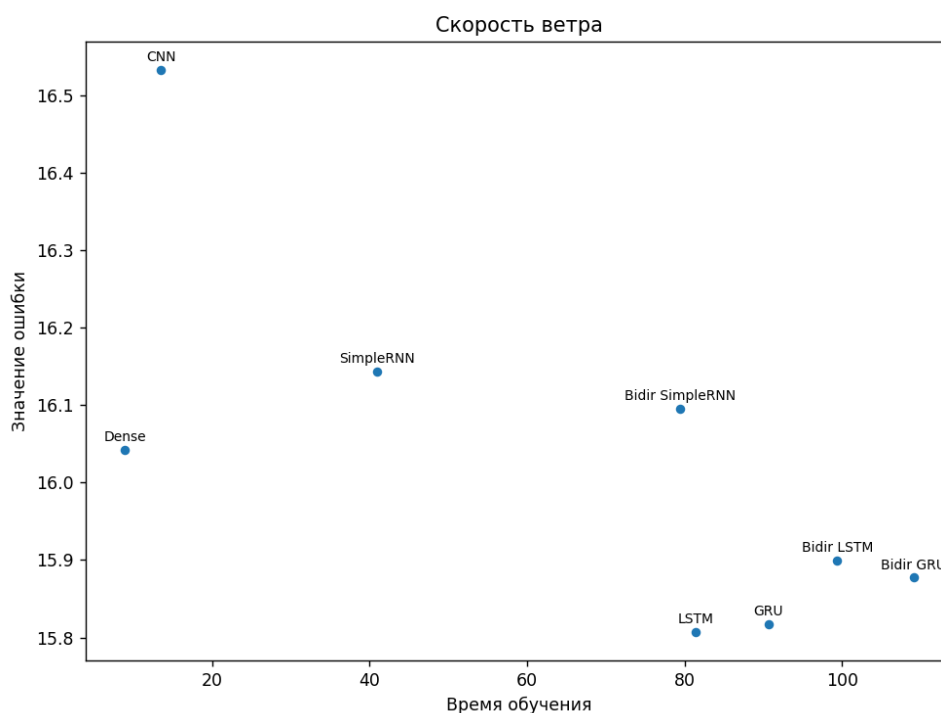


Рис. 10. Точечная диаграмма зависимости ошибки от времени обучения. Скорость ветра
Fig. 10. Error-training time scatter plot. Wind speed

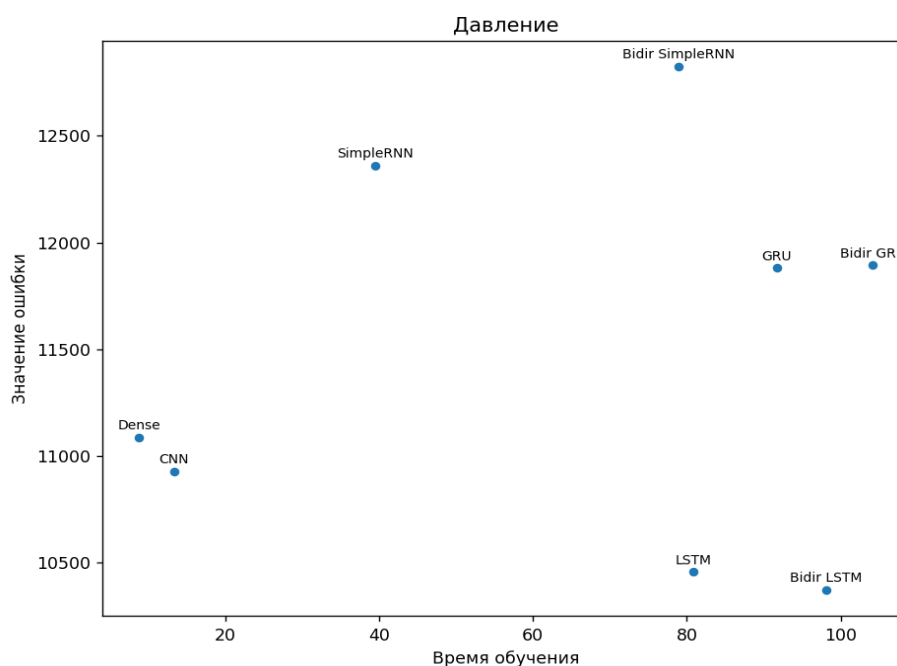


Рис. 11. Точечная диаграмма зависимости ошибки от времени обучения. Давление.
Fig. 11. Error-training time scatter plot. Pressure

ЗАКЛЮЧЕНИЕ

Из полученных результатов можно сделать вывод, что наименьшую ошибку в контексте обработки временных последовательностей метеорологических данных дают сети LSTM и GRU. Однако, при соотношении время/точность наиболее оптимальной оказывается полносвязная нейронная сеть. В случае критичности скорости работы (например, на слабых машинах, или для

обработки данных в реальном времени) наиболее приемлемой из всех рассмотренных будет являться именно полносвязная нейронная сеть. Однако, если приоритетом является точность работы, предпочтение следует отдавать сетям LSTM или GRU. Эффективность работы обеих моделей различается незначительно, и выбор первой или второй зависит от имеющихся входных данных. Тем не менее, при всех трех случаях LSTM работает примерно одинаково, тогда как ошибка в GRU сильно выросла при обучении на данных атмосферного давления, что видно на рисунке 11. Также при одинаковом значении ошибки, LSTM работает немного быстрее, и, в данном случае, более предпочтительнее. Двухнаправленные рекуррентные нейронные сети по эффективности немного уступают обычным рекуррентным нейронным сетям, за исключением прогноза давления. Тем не менее, они требуют больше времени на обработку, что делает их менее предпочтительными для работы с похожими данными. Наихудший результат показывают сверточные нейронные сети: они недостаточно быстро работают, и при этом недостаточно точны. Подводя общий итог, можно сказать, что полученные данные могут использоваться для построения более сложных нейронных сетей с учетом имеющихся особенностей данных временных рядов атмосферных явлений, а также быть основой для дальнейших исследований в смежных областях.

Список литературы

1. Рындин А.А., Ульев В.П., Исследование скорости обучения нейронных сетей [Электронный ресурс]. URL: <https://cyberleninka.ru/article/n/issledovanie-skorosti-obucheniya-neyronnyh-setey> (Дата обращения: 12.10.2024)
2. Быков Ф.Л., Царалов Н.Д., Современные практики применения машинного обучения в задаче прогноза погоды [Электронный ресурс]. URL: <https://cyberleninka.ru/article/n/sovremennye-praktiki-primeneniya-mashinnogo-obucheniya-v-zadache-prognoza-pogody> (Дата обращения: 13.10.2024)
3. Логином вики: многослойный перцептрон [Электронный ресурс]. URL: <https://wiki.loginom.ru/articles/multilayered-perceptron.html> (Дата обращения: 03.07.2024)
4. Murat H. Sazli, A brief review of feed-forward neural networks [Электронный ресурс]. URL: https://www.researchgate.net/publication/228394623_A_brief_review_of_feed-forward_neural_networks (Дата обращения: 03.07.2024)
5. Полносвязные слои нейронных сетей в машинном обучении [Электронный ресурс]. URL: <https://habr.com/ru/articles/718044/> (Дата обращения: 03.07.2024)
6. Горбачевская Е.Н., Классификация нейронных сетей [Электронный ресурс]. URL: <https://cyberleninka.ru/article/n/klassifikatsiya-neyronnyh-setey/viewer> (Дата обращения: 03.07.2024)
7. Kaggle weather dataset [Электронный ресурс]. URL: <https://www.kaggle.com/datasets/muthuj7/weather-dataset/data> (Дата обращения: 08.07.2024)
8. Shapalin Vitaliy: «Forecast» github repository [Электронный ресурс]. URL: <https://github.com/ShapalinVitaliy/Forecast> (Дата обращения: 09.07.2024)
9. Keras official website [Электронный ресурс]. URL: <https://keras.io/> (Дата обращения: 10.07.2024)
10. Simon J. D. Prince Understanding deep learning. MIT press, udlbook.com, 2023. 161 с.
11. Хабр, «Свертка» [Электронный ресурс]. URL: <https://habr.com/ru/articles/795223/> (Дата обращения: 05.08.2024)
12. Д.А. Маршалко, О.В. Кубанский, архитектура сверточных нейронных сетей [Электронный ресурс]. URL: <https://cyberleninka.ru/article/n/arhitektura-svyortochnyh-neyronnyh-setey/viewer> (Дата обращения: 05.08.2024)
13. 1D convolutional neural networks and applications: A survey [Электронный ресурс]. URL: <https://www.sciencedirect.com/science/article/pii/S0888327020307846> (Дата обращения: 06.08.2024)
14. Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach [Электронный ресурс]. URL: <https://www.sciencedirect.com/science/article/pii/S1877050918308019> (06.08.2024)
15. I. Goodfellow, I. Bengio, A. Courville Deep learning. MIT press, Deeplearningbook.org, 2023. 373 с.
16. RNN, LSTM, GRU и другие рекуррентные нейронные сети [Электронный ресурс]. URL: http://vbystricky.ru/2021/05/rnn_lstm_gru_etc.html (Дата обращения: 12.07.2024)

17. Robin M. Schmidt, Recurrent Neural Networks (RNNs): A gentle Introduction and Overview [Электронный ресурс]. URL: <https://arxiv.org/abs/1912.05911> (Дата обращения: 13.07.2024)
18. Recurrent Neural Networks (RNN) - The Vanishing Gradient Problem [Электронный ресурс]. URL: <https://www.superdatascience.com/blogs/recurrent-neural-networks-rnn-the-vanishing-gradient-problem> (Дата обращения: 13.07.2024)
19. Paul Werbos, Backpropagation through time: what it does and how to do it [Электронный ресурс]. URL: https://www.researchgate.net/publication/2984354_Backpropagation_through_time_what_it_does_and_how_to_do_it (Дата обращения: 12.07.2024)
20. LSTM — нейронная сеть с долгой краткосрочной памятью [Электронный ресурс]. URL: <https://neurohive.io/ru/osnovy-data-science/lstm-nejronnaja-set/> (Дата обращения: 29.07.2024)
21. A survey on long short-term memory networks for time series prediction [Электронный ресурс]. URL: <https://www.sciencedirect.com/science/article/pii/S2212827121003796> (Дата обращения: 30.07.2024)
22. Рекуррентные блоки GRU. Пример их реализации в задаче сентимент-анализа [Электронный ресурс]. URL: https://proproprogs.ru/neural_network/rekurrentnye-bloki-gru-primer-realizacii-v-zadache-sentiment-analiza (Дата обращения: 30.07.2024)
23. Двухнаправленные (bidirectional) рекуррентные нейронные сети [Электронный ресурс]. URL: https://proproprogs.ru/neural_network/bidirectional-rekurrentnye-neyronnye-seti (Дата обращения: 30.07.2024)
24. Mike Schuste, Kuldeep K. Paliwal Bidirectional Recurrent Neural Networks [Электронный ресурс]. URL: <https://deeplearning.cs.cmu.edu/S24/document/readings/Bidirectional%20Recurrent%20Neural%20Networks.pdf> (Дата обращения: 30.07.2024)

References

1. Ryndin A.A., Ulev V.P., Neural networks learning speed research. <https://cyberleninka.ru/article/n/issledovanie-skorosti-obucheniya-neyronnyh-setey> (accessed October 12, 2024) (in Russian).
2. Bykov F.L., Tsaralov N.D., Modern practices of machine learning applying in weather forecast task. <https://cyberleninka.ru/article/n/sovremennye-praktiki-primeneniya-mashinnogo-obucheniya-v-zadache-prognoza-pogody> (accessed October 13, 2024)
3. Loginom wiki: multilayer perceptron. <https://wiki.loginom.ru/articles/multilayered-perceptron.html> (accessed July 3, 2024) (in Russian).
4. Murat H. Sazli, A brief review of feed-forward neural networks. https://www.researchgate.net/publication/228394623_A_brief_review_of_feed-forward_neural_networks (accessed July 3, 2024).
5. Fully connected layers of neural networks in machine learning. <https://habr.com/ru/articles/718044/> (accessed July 3, 2024) (in Russian).
6. Gorbachevskaya E.N., Neural network classification. <https://cyberleninka.ru/article/n/klassifikatsiya-neyronnyh-setey/viewer> (accessed July 3, 2024) (in Russian).
7. Kaggle weather dataset. <https://www.kaggle.com/datasets/muthuj7/weather-dataset/data> (accessed July 8, 2024) (in Russian).
8. Shapalin Vitaliy: "Forecast" github repository. <https://github.com/ShapalinVitaliy/Forecast> (accessed July 9, 2024) (in Russian).
9. Keras official website. <https://keras.io/> (accessed July 10, 2024)
10. Simon J. D. Prince, Understanding deep learning. MIT press, udlbook.com, 2023, 161.
11. Habr, "Convolution". <https://habr.com/ru/articles/795223/> (accessed August 5, 2024) (in Russian).
12. D.A. Marshalko, O.V. Kubansky, Architecture of convolutional neural networks. Available at: <https://cyberleninka.ru/article/n/arhitektura-svyortochnyh-neyronnyh-setey/viewer> (accessed August 5, 2024) (in Russian).
13. 1D convolutional neural networks and applications: A survey. <https://www.sciencedirect.com/science/article/pii/S0888327020307846> (accessed August 8, 2024)
14. Conceptual Understanding of Convolutional Neural Network – A Deep Learning Approach. URL: <https://www.sciencedirect.com/science/article/pii/S1877050918308019> (accessed August 6, 2024)
15. I. Goodfellow, Y. Bengio, A. Courville, Deep learning. MIT press, deeplearningbook.com, 2016, 373.

16. RNN, LSTM, GRU and other recurrent neural networks. http://vbystricky.ru/2021/05/rnn_lstm_gru_etc.html (accessed July 12, 2024) (in Russian).

17. Robin M. Schmidt, Recurrent Neural Networks (RNNs): A gentle Introduction and Overview. <https://arxiv.org/abs/1912.05911> (accessed July 13, 2024)

18. Recurrent Neural Networks (RNN) - The Vanishing Gradient Problem. <https://www.superdatascience.com/blogs/recurrent-neural-networks-rnn-the-vanishing-gradient-problem> (accessed July 13, 2024)

19. Paul Werbos, Backpropagation through time: what it does and how to do it. https://www.researchgate.net/publication/2984354_Backpropagation_through_time_what_it_does_and_how_to_do_it (accessed July 12, 2024).

20. LSTM — neural network with Long Short-Term Memory. <https://neurohive.io/ru/osnovy-data-science/lstm-nejronnaja-set/> (accessed July 29, 2024) (in Russian).

21. A survey on long short-term memory networks for time series prediction. <https://www.sciencedirect.com/science/article/pii/S2212827121003796> (accessed July 30, 2024)

22. GRU recurrent blocks. An example of their implementation in a sentiment analysis task. https://proproprogs.ru/neural_network/rekurrentnye-bloki-gru-primer-realizacii-v-zadache-sentiment-analiza (accessed July 30, 2024) (in Russian).

23. Bidirectional recurrent neural networks. https://proproprogs.ru/neural_network/bidirectional-rekurrentnye-neyronnye-seti (accessed July 30, 2024) (in Russian).

24. Mike Schuste, Kuldip K. Paliwal Bidirectional Recurrent Neural Networks. <https://deeplearning.cs.cmu.edu/S24/document/readings/Bidirectional%20Recurrent%20Neural%20Networks.pdf> (accessed July 30, 2024)

Шапалин Виталий Геннадиевич, Аспирант Донецкого национального технического университета

Николаенко Денис Владимирович, Кандидат технических наук, доцент, доцент кафедры "Компьютерная инженерия"

Shapalin Vitaliy Gennadiyevich, Postgraduate of Donetsk national technical university

Nikolayenko Denis Vladimirovich, Candidate of Technical Sciences, Associate Professor, Associate Professor at the Department of Computer Engineering