Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра автоматизации обработки информации (АОИ)

Отчёт к Лабораторной работе №5

по дисциплине «Нейронные сети и их применение»

Выполнил:

Студент группы 422–M1

\_\_\_\_Белоус Г.В.

Принял:

К.т.н., Доцент кафедры АОИ

\_\_\_\_Аксёнов С.В.

Оглавление

[1 Введение 3](#_Toc135653711)

[1.1 Цель лабораторной работы 3](#_Toc135653712)

[1.2 Задание на лабораторную работу 3](#_Toc135653713)

[2 Ход выполнения работы 5](#_Toc135653714)

[2.1 Регрессор встречного распространения 6](#_Toc135653715)

[2.2 Регрессор, использующий слой RNN в качестве скрытого слоя 7](#_Toc135653716)

[2.3 Регрессор, ячейку LSTM (или GRU) в скрытом слое 9](#_Toc135653717)

[2.4 Итоговые метрика каждой модели 10](#_Toc135653718)

[3 Вывод 13](#_Toc135653719)

# Введение

## 1.1 Цель лабораторной работы

Получить навыки создания нейронных сетей для анализа сигналов с помощью моделей рекуррентных сетей и ячеек LSTM (GRU).

## **1.2 Задание на лабораторную работу**

Построить три нейросетевые модели:

a) Регрессор встречного распространения

b) Регрессор, использующий слой RNN в качестве скрытого слоя

c) Регрессор, ячейку LSTM (или GRU) в скрытом слое

По выборке – погода в Сочи.

Выполнить загрузку и предварительную обработку данных из наборов. Разделить каждую выборку на обучающую, тестовую и валидационную. Произвести обучение набора нейросетевых архитектур, отличающихся разным набором параметров: количество нейронов в слоях (ячеек), процедур оптимизации:

- Подобрать архитектуры нейронных сетей, которые с одной стороны позволяют получить модели с лучшими метриками качества работы, с другой стороны не являются избыточными и не переобученными для следующих задач: a) прогноз температуры на следующий день по температуре предыдущих дней, b) прогноз средней температуры на следующей неделе по данным предыдущих дней (или недель). Сделать подбор количества предыдущих значений, используемых для прогнозной модели, для более точной оценки целевой переменной. Сделать эксперименты с прореживанием (Dropout).

- Вычислить следующие метрики работы: R2, MSE, MAE для всех исследованных моделей.

Сделать выводы по результатам построения моделей.

Содержание отчета.

1. Описание наборов данных.

2. Признаки, которые были использованы для анализа.

3. Параметры архитектур и обучения нейронных сетей, использованные для обучения.

4. Графики обучения для архитектур нейронных сетей с лучшими характеристиками эффективности

5. Оценки моделей на тестовых выборках в виде таблиц/ диаграмм, отображающих метрики качества.

6. Программный код.

# Ход выполнения работы

Датасет, данные из которого обрабатывались в данной работе, представляет из себя набор данных о погоде в Сочи между 2011-м и 2020-м годом с указанием времени замеров, температуры по шкале Цельсия, атмосферных давлений и ультрафиолетовым индексом.

Перед созданием моделей наборы данных были проверен на отсутствие пропущенных записей, в наборе данных для удаления строк с отсутствующими значениями необходимо было удалить полностью столбец DD в котором все значения были нулевые, в результате отчистки набора данных от строк с пропущенными значениями было удалено 3921 запись.

График температуры для всех записей показан на рисунке 2.1:

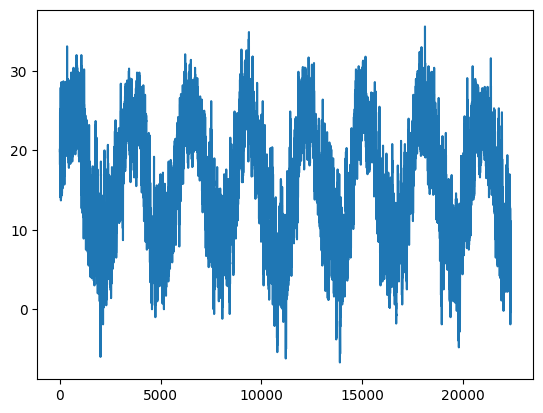


Рисунок 2.1 – График температуры

Исходя из графика можно увидеть уменьшение и увеличение температуры в связи со сменой времени года и определить абсолютный минимум и максимум за девять лет измерений.

Обработанные данные были разделены на выборки для обучения, валидации и проверки (соответственно 70, 15 и 15 %), при этом использовались значения только температуры, занесенные в один массив. А после этого полученный одномерный массив был преобразован в многомерный в каждой выборке (многомерность определялась переменной window = 20 для всех моделей).

## Регрессор встречного распространения

Так как данные были взяты из одного столбца этот же атрибут и является целевым.

Регрессор представляет из себя нейронную сеть из трех слоев:

* Входной слой с нейронами в количестве window;
* Скрытый слой, состоящий из двенадцати нейронов и функцией активации ReLU (rectified linear unit);
* Выходной слой состоит из одного нейрона, так как в результате предсказания модели необходимо получить величину температуры.

Оптимизатор был выбран adam, так как данный оптимизатор показывает более гладкий график, в отличии от SGD и RMSprop, как функция потери была взята средне квадратическая ошибка, а в качестве метрики обучения средне абсолютная.

Для обучения модели были определены значения эпох (60) и размер выборки (25).

В итоге модель была обучена за тридцать четыре эпохи, график функции ошибки (MSE) от количества эпох обучения показан на рисунке 2.2:

Изображение выглядит как текст, снимок экрана, линия, диаграмма

Автоматически созданное описание

Рисунок 2.2 – График величины функции ошибки от количества эпох при обучении модели

## Регрессор, использующий слой RNN в качестве скрытого слоя

Регрессор представляет из себя нейронную сеть из трех слоев:

* Входной слой с нейронами в количестве window;
* Скрытый слой, в котором используется слой SimpleRNN состоящий из двенадцати нейронов и функцией активации ReLU (rectified linear unit);
* Выходной слой состоит из одного нейрона, так как в результате предсказания модели необходимо получить величину температуры.

SimpleRNN — это один из типов слоев рекуррентных нейронных сетей, который применяет простую рекуррентную архитектуру. Он принимает входные данные с заданным временным окном и передает информацию из предыдущего временного шага на следующий временной шаг.

Оптимизатор был выбран adam, так как данный оптимизатор показывает более гладкий график, в отличии от SGD и RMSprop, как функция потери была взята средне квадратическая ошибка, а в качестве метрики обучения средне абсолютная.

Для обучения модели были определены значения эпох (60) и размер выборки (25).

В итоге модель была обучена за тридцать четыре эпохи, график функции ошибки (MSE) от количества эпох обучения показан на рисунке 2.3:

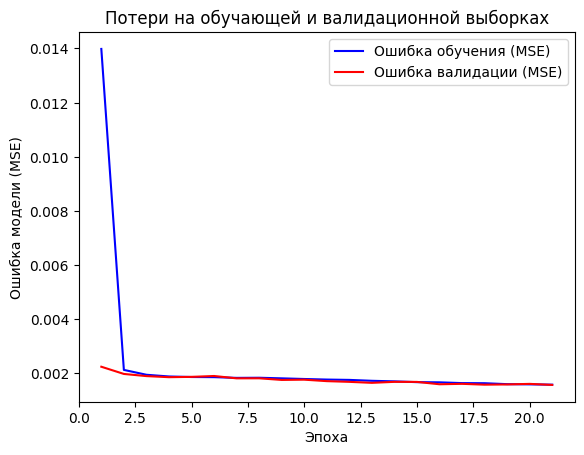


Рисунок 2.3 – График величины функции ошибки от количества эпох при обучении модели с рекуррентным слоем

## Регрессор, ячейку LSTM (или GRU) в скрытом слое

Регрессор представляет из себя нейронную сеть из трех слоев:

* Входной слой с нейронами в количестве window;
* Скрытый слой, в котором используется слой LSTM состоящий из шестнадцати нейронов и функцией активации ReLU (rectified linear unit);
* Выходной слой состоит из одного нейрона, так как в результате предсказания модели необходимо получить величину температуры.

LSTM (Long Short-Term Memory) — это тип рекуррентного слоя, используемого в нейронных сетях для моделирования временных последовательностей и учета долгосрочных зависимостей в данных. Он отличается от простого RNN (рекуррентная нейронная сеть) тем, что обладает дополнительной внутренней структурой, позволяющей сохранять и использовать информацию в течение длительных временных интервалов.

Оптимизатор был выбран adam, так как данный оптимизатор показывает более гладкий график, в отличии от SGD и RMSprop, как функция потери была взята средне квадратическая ошибка, а в качестве метрики обучения средне абсолютная.

Для обучения модели были определены значения эпох (60) и размер выборки (25).

В итоге модель была обучена за тридцать четыре эпохи, график функции ошибки (MSE) от количества эпох обучения показан на рисунке 2.3:



Рисунок 2.3 – График величины функции ошибки от количества эпох при обучении модели с рекуррентным слоем

## Итоговые метрика каждой модели

Графики с оригинальными и предсказанными значениями температуры показаны на рисунках 2.4, 2.5 и 2.6:

Изображение выглядит как снимок экрана, График, текст, диаграмма

Автоматически созданное описание

Рисунок 2.4 – Оригинальные (синие) и предзаказанные (красные) значения модели встречного распространения

Изображение выглядит как снимок экрана, График, текст, диаграмма

Автоматически созданное описание

Рисунок 2.5 – Оригинальные (синие) и предзаказанные (красные) значения модели встречного распространения

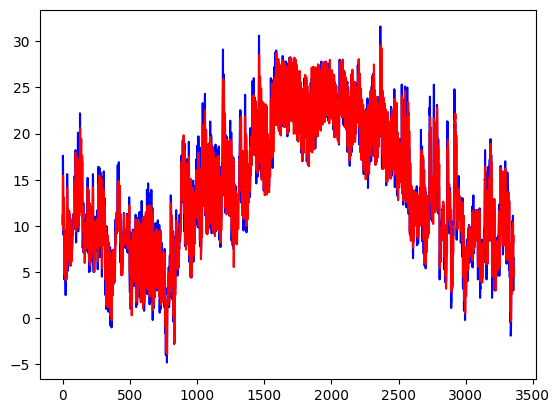


Рисунок 2.6 – Оригинальные (синие) и предзаказанные (красные) значения модели встречного распространения

В таблице 2.1 собраны метрики всех моделей полученные с помощью библиотеки sklearn.metrics.

Таблица 2.1 – Метрики качества моделей

|  |  |  |  |
| --- | --- | --- | --- |
| Метрика\Модель | Регрессор встречного распространения | SimpleRNN | LSTM |
| MSE | 3.15 | 3.371 | 3.31 |
| MAE | 1.243 | 1.293 | 1.293 |
| R2 | 0.939 | 0.935 | 0.936 |

# Вывод

В ходе выполнения лабораторной работы были построены модели регрессоров встречного распространения, использующего слой RNN в качестве скрытого слоя и с ячейкой LSTM в скрытом слое с помощью библиотеки построения нейронных сетей.

Стоит отметить, что для совмещения минимальной ошибки и высокой скорости расчётов моделей для всех трех моделей было использовано одинаковое число эпох (60) и размер выборки (25). Количество нейронов для первых двух слоев одинаково (12), а в третьей модели было выбрано другое количество (16). При сравнении итоговых метрик было выявлен, что модель регрессора встречного распространения показало лучше результаты, а худшим оказался регрессор использующий слой RNN.

Листинг программы находится в соответствующей папке Lab5 по ссылке на репозиторий: https://github.com/peremichka256/NSiIP