

## ПРИМЕНЕНИЯ РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ LSTM ДЛЯ ПРОГНОЗИРОВАНИЯ ВРЕМЕННЫХ РЯДОВ СРЕДНЕМЕСЯЧНОЙ ТЕМПЕРАТУРЫ ВОЗДУХА

Абдрасакова А.Б.<sup>(1)</sup>, Сулайманова С.М.<sup>(2)</sup>

<sup>(1)</sup>КГТУ им. И. Раззакова Бишкек, КР,

<sup>(2)</sup>КНАУ им. К.И. Скрябина Бишкек, КР

[aizada\\_abdrasakova@yahoo.com](mailto:aizada_abdrasakova@yahoo.com), [sulai@mail.ru](mailto:sulai@mail.ru)

**Аннотация:** Рассматриваются тенденции изменений временных рядов среднемесячного температурного воздуха Ошской области за 100-летний период 1921-2021гг с помощью LSTM модели нейронных сетей для решения задачи прогнозирования.

**Ключевые слова:** температура воздуха, задача прогнозирования, нейронная сеть, временной ряд, изменение климата.

## АБАНЫН ОРТОЧО АЙЛЫК ТЕМПЕРАТУРАСЫНЫН УБАКЫТ КАТАРЫН БОЛЖОЛДОО ҮЧҮН LSTM РЕКУРРЕНТТИК НЕЙРОН ТАРМАКТАРЫН КОЛДОНУУ

Абдрасакова А.Б.<sup>(1)</sup>, Сулайманова С.М.<sup>(2)</sup>

<sup>(1)</sup>И.Раззаков атындагы КМТУ, Бишкек, КР,

<sup>(2)</sup>К.И.Скрябин атындагы КУАУ, Бишкек, КР

[aizada\\_abdrasakova@yahoo.com](mailto:aizada_abdrasakova@yahoo.com), [sulai@mail.ru](mailto:sulai@mail.ru)

**Аннотация:** 1921-2021-жылдар аралыгындагы 100 жылдык мезгилге Ош облусунун аймагындагы абанын орточо айлык температурасынын убакыт катарынын өзгөрүү тенденциялары болжолдоо маселесин чыгаруучу нейрондук тармактардын LSTM моделин колдонуу менен каралат.

**Ачкыч сөздөр:** абанын температурасы, болжолдоо маселеси, нейрондук тармак, убакыт катары, климаттын өзгөрүшү.

## APPLICATION OF RECURRENT NEURAL NETWORKS LSTM FOR FORECASTING TIME SERIES OF AVERAGE MONTHLY AIR TEMPERATURE

Abdrasakova A.B.<sup>(1)</sup>, Sulaimanova S.M.<sup>(2)</sup>

<sup>(1)</sup>KSTU n. a. I. Razzakov, Bishkek, Kyrgyz Republic,

<sup>(2)</sup>KNAU n.a. K.I. Skryabin, Bishkek, Kyrgyz Republic

[aizada\\_abdrasakova@yahoo.com](mailto:aizada_abdrasakova@yahoo.com), [sulai@mail.ru](mailto:sulai@mail.ru)

**Annotation:** The trends of changes in the time series of the average monthly temperature air in the Osh region for the 100-year period 1921-2021 are considered using the LSTM model of neural networks to solve the forecasting problem.

**Key words:** air temperature, forecasting problem, neural network, time series, climate change.

**Введение.** Региональные климатические изменения во многом определяются не только глобальными процессами, но известными физико-географическими условиями конкретного региона, что в полной мере можно отнести к некоторым областям Кыргызстана. Оценка трендов только среднемесячной температуры воздуха данного региона не позволяют выявить структуру температурных изменений. В данной работе

применяется прогнозирование временных рядов для данных среднемесячных температур воздуха, что дает возможность исследовать изменения регионального климата. Задача прогнозирования временных рядов заключается в том, чтобы продолжить значения данных среднемесячной температуры воздуха прошлых периодов в будущее, где эти значения еще не известны.

Рекуррентная сеть с долгосрочной и краткосрочной памятью LSTM (Long Short Term Memory) глубокого обучения более точно определяет закономерности в данных временных рядов, и эффективно может применяться для прогнозирования будущего тренда данных [1].

Данные представляются как среднемесячная температура воздуха Ошской области в промежутке с 1921-2021гг. Они разбиваются на обучающий и тестовый набор, где данные до 2001г. используются в качестве обучающих данных для модели, а затем пытаемся предсказать среднемесячную температуру в течение последней части набора данных[7].

Чтобы спрогнозировать значения будущих временных шагов последовательности, можно обучить сеть LSTM регрессии от последовательности к последовательности, где ответами являются обучающие последовательности со значениями, сдвинутыми на один временной шаг. То есть на каждом временном шаге входной последовательности сеть LSTM учится предсказывать значение следующего временного шага. При прогнозировании значения нескольких временных шагов в будущем, используем функцию `predictAndUpdateState` для прогнозирования временных шагов по одному и обновления состояния сети при каждом прогнозе.

В представленном случае используется набор данных среднемесячная температура воздуха `avgtemp_dataset`. В этом примере сеть LSTM обучается прогнозировать количество случаев среднемесячной температуры воздуха с учетом количества случаев в предыдущие месяцы.

**Методы решение задачи.** Данные загружаются в `avgtemp_dataset`, где один временной ряд с временными шагами, соответствующими месяцам, и значениями, соответствующими количеству температуры. Выход представляет собой массив ячеек, где каждый элемент представляет собой один временной шаг. Изменяются данные, чтобы они были вектором-строкой.

```
data = avgtemp_dataset;  
data = [data{:}];  
figure  
plot(data)  
xlabel("years")  
ylabel("Temperature")  
title("Ежемесячные средние температуры в г.Ош с 1921-2021")
```

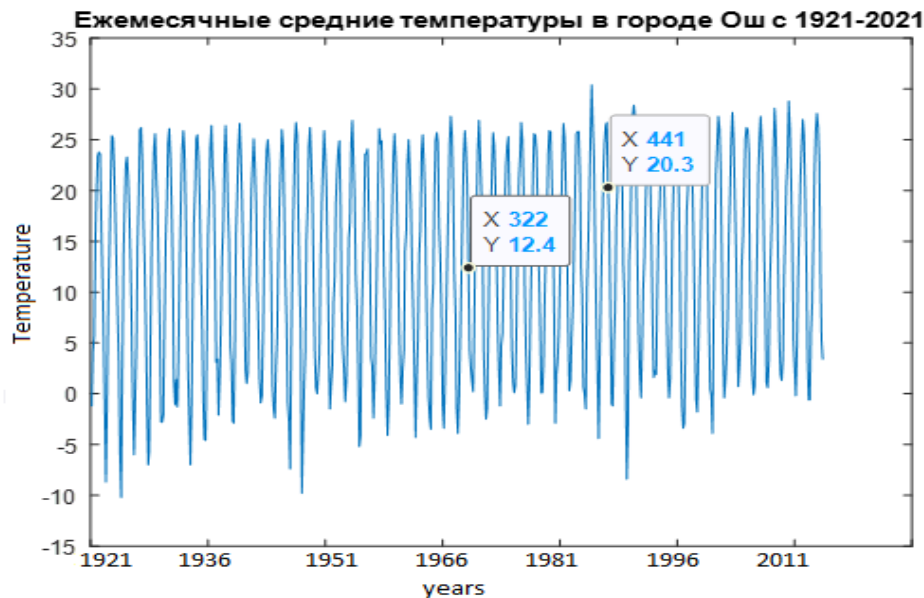


Рис.1. Временные ряды среднемесячных температур воздуха (С°) по Ошскому региону Кыргызской Республики за 1921-2021 гг.

Разделяются на обучающие и тестовые данные. Вводятся данные на первых 90% последовательности и тестируются на последних 10%.

```
numTimeStepsTrain = floor(0.9*numel(data));
dataTrain = data(1:numTimeStepsTrain+1);
dataTest = data(numTimeStepsTrain+1:end);
```

Для лучшего соответствия и предотвращения отклонения обучения стандартизируются данные обучения, чтобы иметь нулевое среднее значение и единичную дисперсию. Во время прогнозирования должны стандартизоваться тестовые данные, используя те же параметры, что и обучающие данные.

```
mu = mean(dataTrain);
sig = std(dataTrain);
dataTrainStandardized = (dataTrain - mu) / sig;
```

Чтобы спрогнозировать значения будущих временных шагов последовательности, нужно задать ответы как обучающие последовательности со значениями, сдвинутыми на один временной шаг. То есть на каждом временном шаге входной последовательности сеть LSTM учится предсказывать значение следующего временного шага. Предикторы — это обучающие последовательности без конечного временного шага.

```
XTrain = dataTrainStandardized(1:end-1);
YTrain = dataTrainStandardized(2:end);
```

Определение сетевой архитектуры LSTM. Создается регрессионная сеть LSTM. Указывается слой LSTM, чтобы иметь 200 скрытых единиц.

```
numFeatures = 1;
numResponses = 1;
numHiddenUnits = 200;
```

```
layers = [ ...
    sequenceInputLayer(numFeatures)
    lstmLayer(numHiddenUnits)
    fullyConnectedLayer(numResponses)
    regressionLayer];
```

Указание вариантов обучения. Устанавливается решатель на «Адам» и тренируется в течение 250 эпох. Чтобы предотвратить взрыв градиентов, устанавливается порог градиента на 1. Указывается начальная скорость обучения 0,005 и уменьшается скорость обучения после 125 эпох, умножив на коэффициент 0,2.

```
options = trainingOptions('Adam', ...
    'MaxEpochs',250, ...
    'GradientThreshold',1, ...
    'InitialLearnRate',0.005, ...
    'LearnRateSchedule','piecewise', ...
    'LearnRateDropPeriod',125, ...
    'LearnRateDropFactor',0.2, ...
    'Verbose',0, ...
    'Plots','training-progress');
```

Обучение сети LSTM. Для обучения нейронной сети была построена модель рекуррентной нейронной сети LSTM со следующими параметрами обучения: длина последовательности, равная 500; количество единиц скрытого слоя, равное 200; количество элементов вывода, равное 1. указанными параметрами обучения с помощью trainNetwork показан на рис.2.

```
net = trainNetwork (XTrain, YTrain, слоу, параметры);
```

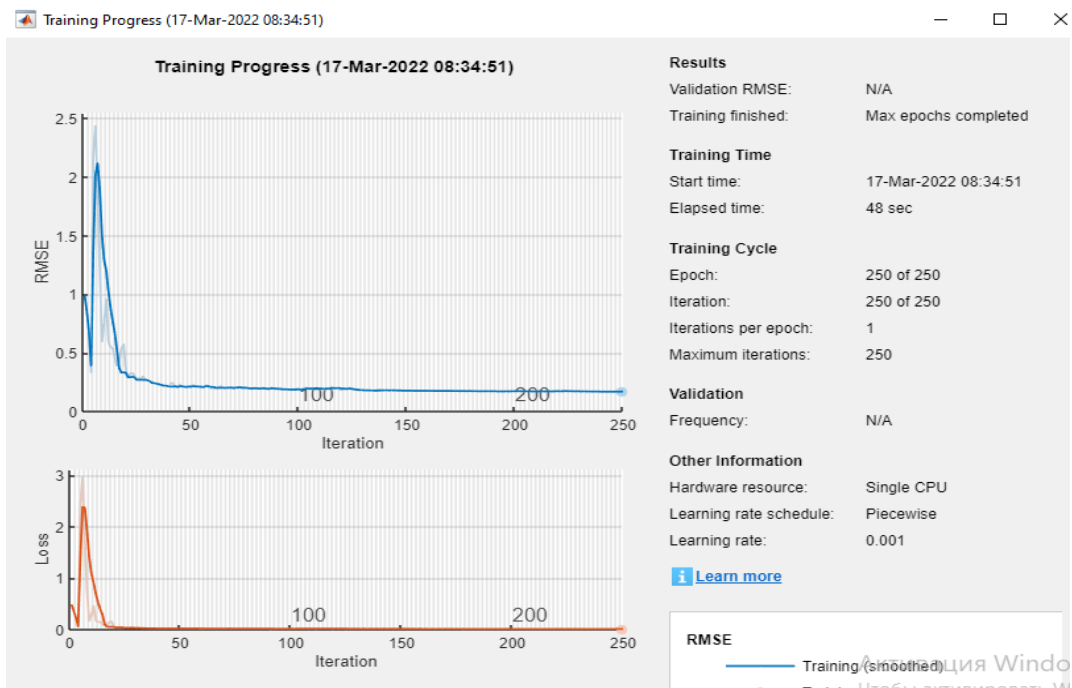


Рис.2. Обучение LSTM с указанными параметрами обучения с помощью trainNetwork

**Получение результатов.** Чтобы спрогнозировать значения нескольких временных шагов в будущем, используются функция `predictAndUpdateState` для прогнозирования временных шагов по одному и обновления состояния сети при каждом прогнозе. Для каждого прогноза используется предыдущий прогноз в качестве входных данных для функции.

Стандартизировать тестовые данные, используя те же параметры, что и обучающие данные.

```
dataTestStandardized = (dataTest - mu) / sig;
XTest = dataTestStandardized(1:end-1);
```

Чтобы инициализировать состояние сети, сначала предсказывается обучающие данные `XTrain`. Затем выводится первый прогноз, используя последний временной шаг обучающего ответа `YTrain(end)`. Переберем оставшиеся прогнозы и вводим предыдущий прогноз в `predictAndUpdateState`.

Для больших наборов данных, длинных последовательностей или больших сетей прогнозы на GPU обычно выполняются быстрее, чем прогнозы на CPU. В противном случае прогнозы на ЦП обычно вычисляются быстрее. Для прогнозов с одним временным шагом используйте ЦП. Чтобы использовать ЦП для прогнозирования, устанавливаем для параметра «`ExecutionEnvironment`» функции «`PredictAndUpdateState`» значение «`cpu`».

```
net = predictAndUpdateState(net,XTrain);
[net,YPred] = predictAndUpdateState(net,YTrain(end));
numTimeStepsTest = numel(XTest);
for i = 2:numTimeStepsTest
[net,YPred(:,i)] = predictAndUpdateState(net,YPred(:,i-1),'ExecutionEnvironment','cpu');
end
```

Нестандартизируем прогнозы, используя параметры, рассчитанные ранее.

```
YPred = sig*YPred + mu;
```

График прогресса обучения показывает среднеквадратичную ошибку (RMSE), рассчитанную на основе стандартизированных данных. Вычислим RMSE из нестандартизированных прогнозов.

```
YTest = dataTest(2:end);
rmse = sqrt(mean((YPred-YTest).^2))
rmse = single 248.5531
```

*Построим график временного ряда обучения с прогнозируемыми значениями.*

```
figure
plot(dataTrain(1:end-1))
hold on
idx = numTimeStepsTrain:(numTimeStepsTrain+numTimeStepsTest);
plot(idx,[data(numTimeStepsTrain) YPred],'-')
hold off
xlabel("Years")
```

```
ylabel("Temperature")
title("Прогнозирование")
legend(["Observed" "Forecast"])
```

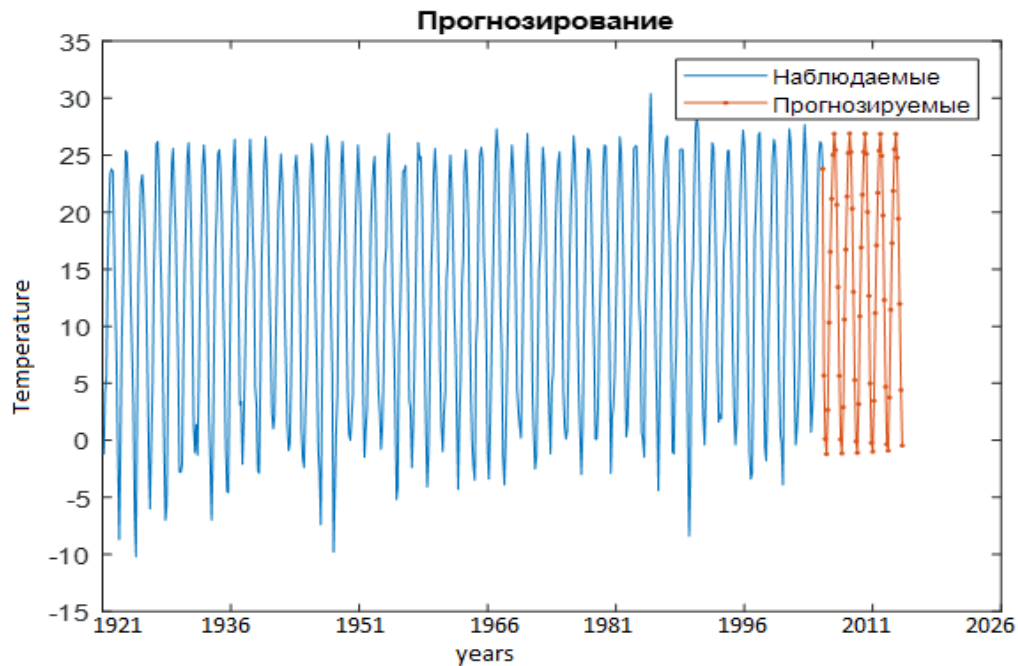


Рис.3.

Временной ряд обучения с прогнозируемыми значениями

Обновление состояния сети с помощью наблюдаемых значений. Если есть доступ к фактическим значениям временных шагов между прогнозами, можно обновить состояние сети с помощью наблюдаемых значений вместо прогнозируемых значений.

Во-первых, инициализируем состояние сети. Чтобы делать прогнозы для новой последовательности, сбрасываем состояние сети с помощью `resetState`. Сброс состояния сети предотвращает влияние предыдущих прогнозов на прогнозы новых данных. Сбрасываем состояние сети, а затем инициализируем состояние сети, предсказав обучающие данные.

```
net = resetState(net);
net = predictAndUpdateState(net,XTrain);
```

Делаем прогнозы на каждом временном шаге. Для каждого прогноза предсказываем следующий временной шаг, используя наблюдаемое значение предыдущего временного шага. Устанавливается для параметра «Среда выполнения» в `PredictAndUpdateState` значение «ЦП».

```
YPred = [];
numTimeStepsTest = numel(XTest);
for i = 1:numTimeStepsTest
    [net,YPred(:,i)] = predictAndUpdateState(net,XTest(:,i),'ExecutionEnvironment','cpu');
end
```

Нестандартизируем прогнозы, используя параметры, рассчитанные ранее.

$$YPred = sig*YPred + mu;$$

Calculate the root-mean-square error (RMSE).

$$rmse = \sqrt{\text{mean}((YPred - YTest).^2)}$$

$$rmse = 158.0959$$

Сравниваем прогнозируемые значения с тестовыми данными

```
figure
subplot(2,1,1)
plot(YTest)
hold on
plot(YPred,'-')
hold off
legend(["Observed" "Predicted"])
ylabel("Temperature")
title("Forecast with Updates")
subplot(2,1,2)
stem(YPred - YTest)
xlabel("years")
ylabel("Error")
title("RMSE = " + rmse)
```

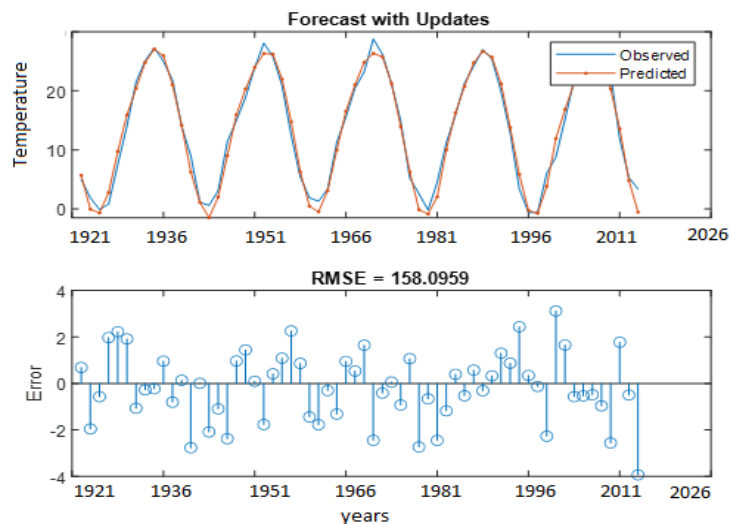


Рис.4. Прогнозируемые значения с тестовыми данными

Здесь прогнозы более точны при обновлении состояния сети наблюдаемыми значениями вместо прогнозируемых значений [2-6].

**Выводы.** Изучение климатических изменений Ошской области Кыргызской Республики продемонстрировало, что среднемесячная температура воздуха за период 1921-2021 гг. существенно возросла, при этом тенденция имеет нелинейный характер и в прошедшие десятки лет также значительно возросла. Анализ представленных данных и итогов, полученных при помощи модели LSTM нейросетевого прогнозирования владеют весьма большой обобщающей способностью

которые с большей точностью позволяют аппроксимировать и предсказывать значения временного ряда.

### Список литературы (References)

1. Абдрасакова А.Б., Сулайманова С.М. Анализ климатических данных с помощью нейронных сетей [Текст]/ Абдрасакова А.Б., Сулайманова С.М.//Вестник КГУСТА, 2022.- № 3(77). – С.140-146.
2. Deepti Arela, Shilpa Agrawal. Demand forecasting by using generalized regression neural network. - Open access international journal of science and engineering (OAIJSE). - Volume 03, Issue 02, Year: 2018, Pages:39-45.
3. С.С. Aggarwal. Neural Networks and Deep Learning. A Textbook. Springer International Publishing AG, 2018.
4. Л. Ю. Емалетдинова, З. И. Мухаметзянов, Д. В. Катасёва, А. Н. Кабирова. Метод построения прогнозной нейросетевой модели временного ряда. - Компьютерные исследования и моделирование 2020, Т. 12 № 4 С. 737–756.
5. Бильгаева Л. П., Власов К. Г. Исследование моделей нейросетевого прогнозирования в среде MATLAB // Приложение математики в экономических и технических исследованиях. — 2017. — № 1 (7). — С. 11–19.
6. [Глубокое обучение в MATLAB - MATLAB & Simulink \(mathworks.com\)](#)
7. <http://www.pogodaiklimat.ru/history/38353.htm>