5) Уберите лишние данные – это может показаться вам странным, но данных подход довольно часто применим. Так, для обучения классификатора для определения профессиональной направленности на данном этапе мне была важна классификация IT профессий. Было правильным решением удалить классы, представленные недостаточной выборкой, и не относящиеся к IT.

6) Создайте синтетические данные – этот процесс относится к расширению выборки, но происходит искусственным путем. Например, если у вас нет возможности получить дополнительных данных, вы можете применить систематические алгоритмы для генерации синтетических образцов. Так, при помощи Weka, вы можете использовать SMOTE контролируемый фильтр. SMOTE – это метод передискретизации, который создает синтетические образцы [3].

Это не единственные способы по решению данной проблемы, но они являются самыми, на мой взгляд, понятными и наиболее применяемыми. Помните, что нет единственно верного подхода для решения проблемы несбалансированности. Все подходы можно применять в комплексе или выбрать один, и избежать проблем. Данная работа демонстрирует подходы к обнаружению и исправлению несбалансированности классов, а также показывает на решении конкретной задачи как их стоит применять.

Список литературы

1. 8 тактик для борьбы с несбалансированными классами в вашем наборе данных машинного обучения – режим доступа: https://www.machinelearningmastery.ru/tactics-to-combat-imbalanced-classes-in-your-machine-learning-dataset/

2. Samir, A. Transfer learning for class imbalance problems with inadequate data / A. Samir, K. R. Chandan // Knowl Inf Syst 48. — 2015. — P. 201–228.

3. Weka Wiki – режим доступа: https://waikato.github.io/weka-wiki/

4. Махсотова, Ц. В. Исследование методов классификации при несбалансированности классов / Ц. В. Махсотова // Научный журнал. – 2017. – № 5(18). – С. 35-36.

5. Старовойтов В. В. Об оценке результатов классификации несбалансированных данных по матрице ошибок / В. В. Старовойтов, Ю. И. Голуб // Информатика. – 2021. − Т. 18, № 1. – С. 61–71.

ПРИМЕНЕНИЕ LSTM-СЕТИ В РЕШЕНИИ ЗАДАЧИ ПРОГНОЗИРОВАНИЯ МНОГОМЕРНЫХ ВРЕМЕННЫХ РЯДОВ

Обрубов Максим Олегович
Магистрант 2 курса кафедры информационных систем и программной инженерии "Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых" (ВлГУ), город Владимир
Кириллова Светлана Юрьевна
кандидат технических наук, профессор "Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых" (ВлГУ), город Владимир

USING LSTM NETWORK FOR SOLVING THE MULTIDIMENTIONAL TIME SERIES FORECASTING PROBLEM

Obrubov Maxim Olegovich
2nd year master's student of the Department of Information Systems and Software Engineering "Vladimir State University named after Alexander Grigorievich and Nikolai Grigorievich Stoletovs" (VlSU), Vladimir
Kirillova Svetlana Yurievna
candidate of technical sciences, professor "Vladimir State University named after Alexander Grigorievich and Nikolai Grigorievich Stoletovs" (VlSU), Vladimir
DOI: 10.31618/nas.2413-5291.2021.2.68.450

АННОТАЦИЯ

В статье рассматривается применение технологии рекуррентных нейронных сетей к задаче прогнозирования многомерных временных рядов. Выполнено экспериментальное определение архитектуры нейронной сети и основных гиперпараметров для достижения минимальной погрешности. Выявленная структура сети будет использоваться далее для определения аномалий в многомерных временных рядах.

ABSTRACT

The article discusses using of the recurrent neural networks technology to the multidimensional time series prediction problem. There is an experimental determination of the neural network architecture and its main hyperparameters carried out to achieve the minimum error. The revealed network structure going to be used further to detect anomalies in multidimensional time series.

Ключевые слова: нейронные сети; прогнозирование; временные ряды.

Keywords: neural network; prediction; multidimensional time series.

В процессе исследовании задачи определения аномалий поведения учетных систем предприятий при взаимодействии с крупной федеральной информационной системой возникла необходимость прогнозировать корректное поведение. Данные о поведении представлены записями о заявках на исполнение некой операции. Данные содержат набор признаков, только некоторые из которых имеют значимость для идентификации поведения. Наряду с этими признаками существуют временные отметки поступления заявок. Таким образом данные о заявках можно позиционировать как многомерный временной ряд.

Иными словами, существует временной ряд $X = \{x_1, x_2, \ldots, x_L\}$, где x – это вектор признаков о заявке, L – длина временного ряда. Необходимо найти такую матрицу X_f` (спрогнозированные будущие значения) чтобы ошибка между $X_f$ (реальными будущими значениями) и $X_f$` была минимальной.

Рассмотрим применение LSTM модификации рекуррентной нейронной сети для решения данной задачи. В данной работе предполагается найти такие параметры для построения нейронной сети, чтобы она давала наилучший результат прогнозирования.

Реализация выполняется на языке Python с использованием фреймворка Keras, с Tensorflow в качестве бэкенда. Для выполнения вспомогательных операций используются пакеты numpy и pandas.

Подготовка данных к обучению нейронной сети осуществляется следующим образом:

1. выгружаются из хранилища данных при помощи подготовленных запросов;
2. очищаются от предварительно выявленных аномалий;
3. группируются в равномерный временной ряд, в котором временные отметки находятся на равных расстояниях (в 15 секунд), а к значениям ряда применяются агрегатные функции;
4. нормализуются в интервал [0, 1], чтобы иметь естественную форму для нейронных сетей;
5. делятся на тренировочную и проверочную выборки в соотношении 70/30.

Из исходных данных выбраны значащие признаки: количество заявок, размер запроса, размер ответа. Признаки сгруппированы в равномерный временной ряд с применением функций группировки: сумма (к количеству заявок) и среднее (к размерам запросов и ответов). Исходные данные можно изобразить на графике (рисунок 1).

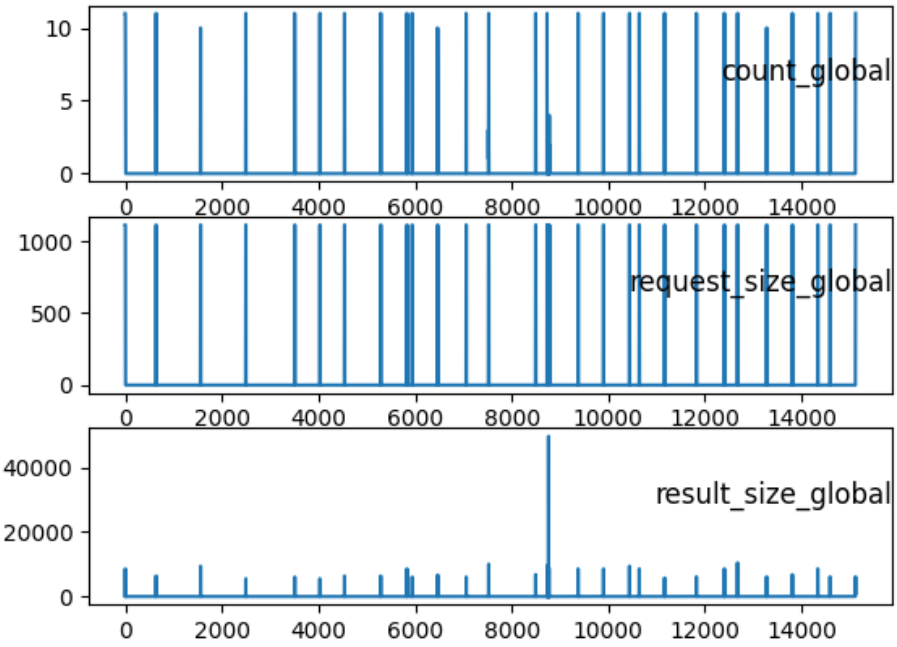

Рисунок 1. Исходные данные

Начнём с минимальной архитектуры: 1 слой с 10 LSTM модулями, 1 полносвязный слой с 3 нейронами (на каждый признак). Эксперименты с усложнением архитектуры не привели к улучшению результатов. Базовые гиперпараметры: оптимизатор – adam, количество эпох обучения – 20, размер пакета - 5, временной лаг – 4, прогнозируемых значений – 1. В качестве функции

потерь используется среднеквадратичная ошибка (MSE), так как она чувствительна к большим отклонениям, что необходимо в задаче определения аномалий:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Y_i')^2,$$

где n – количество наблюдений,
Y_i – реальное значение ряда,
Y_i` – прогнозное значение ряда.

Все эксперименты выполняются многократно (10 раз), в сводных таблицах приводится усредненная ошибка по всем попыткам.

Для начала определимся с функцией-оптимизатором. Оптимизатор определяет скорость и качество обучения модели. С его непосредственным участием модель в правильном направлении корректирует свои веса. В таблице 1 представлен список основных оптимизаторов, входящих в состав Keras.

Наименьшая ошибка наблюдается у нейронных сетей с оптимизаторами adam и nadam. Алгоритм adam алгоритм вычисляет экспоненциальное скользящее среднее градиента и квадратичный градиент [3], а nadam является улучшенной версией adam с применением ускоренного градиента Нестерова [1].

Таблица 1.

Выбор оптимизатора

Оптимизатор	Ошибка		
	Количество заявок	Размер запроса	Размер ответа
adadelta	1.047	146.597	403.508
adagrad	0.774	98.443	836.599
adam	0.425	61.332	250.652
adamax	0.461	65.199	268.231
nadam	0.423	61.358	232.017
rmsprop	0.414	61.424	249.774
sgd	0.725	83.588	407.229

Далее определимся со структурой нейронной сети. Так как усложнение архитектуры сети не привело к улучшениям, то будем экспериментировать с количеством модулей LSTM (таблица 2).

Из эксперимента видно, что оптимальный размер – 20 модулей. При меньшем количестве сеть недообучается, а при большем – переобучается. Для дальнейших исследований будем использовать 20 модулей LSTM.

Таблица 2.

Выбор количества LSTM модулей

Кол-во LSTM модулей	Ошибка		
	Количество заявок	Размер запроса	Размер ответа
10	0.426	61.274	266.215
20	0.425	60.364	220.144
30	0.430	60.880	230.181
40	0.432	61.087	232.616
50	0.438	61.071	263.565

Далее выберем количество эпох обучения. В таблице 3, что, как и в случае с количеством модулей LSTM, сначала наблюдается снижение ошибки, а после 75 эпох – заметный рост. С возрастанием количества эпох сеть начинает переобучаться. Выберем 50 эпох, как компромисс между ошибкой и временем обучения.

Таблица 3.

Выбор количества эпох обучения

Количество эпох обучения	Ошибка		
	Количество заявок	Размер запроса	Размер ответа
20	0.452	63.138	315.273
30	0.423	61.358	232.017
40	0.421	60.814	221.899
50	0.418	60.349	216.582
75	0.411	60.675	207.078
100	0.415	60.866	260.664
200	0.425	61.169	228.096

В рекуррентные сети данные на вход подаются пакетами, чтобы учитывать порядок следования [2]. Для оптимизации нужно также выбрать этот параметр.

Оптимальным является размер пакета – 10 (таблица 4). Такая ситуация возникла потому что с маленьким размером пакета сети не хватает данных для качественного нахождения взаимосвязей между последовательными данными, с другой стороны – увеличение размера пакета уменьшает количество шагов, которые делает сеть при обучении в рамках одной эпохи.

Таблица 4.

Выбор размера пакета

Размер пакета	Ошибка		
	Количество заявок	Размер запроса	Размер ответа
5	0.418	60.349	216.582
10	0.420	60.597	207.884
20	0.430	60.778	205.438
50	0.430	61.052	200.920
100	0.441	61.837	218.675

Наконец, необходимо определиться с размером диапазона значений, на которых будет основываться прогноз и собственно размер прогнозируемого интервала. На рисунке 2 схематично изображен процесс прогнозирования диапазонов по диапазонам. Здесь r – размер диапазона входных данных, или иначе называемый временной лаг, а f – размер прогнозируемого диапазона.

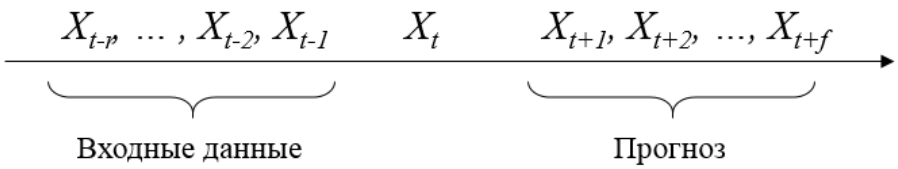

Рисунок 2. Процесс прогнозирования диапазонов по диапазонам

В таблице 5 отражен выбор временного лага. Наилучшие и примерно равные показатели имеют значения равные 4 и 6. Для значений выше 8 показатели ошибки резко возрастают. В таблице 6 отображен выбор размера прогнозируемого диапазона, там аналогично выбрано два значения – 4 и 6. Значения выше 6 во внимание не берем, так как по требованиям процесса, диапазоны более 90 секунд являются слишком большими. Чтобы выбрать единственный лучший результат выполним перекрестное сравнение сетей с этими показателями (таблица 7). В таблице видно, что наилучшие результаты показывают варианты с размером прогнозируемого диапазона равным 6. Выберем вариант 6-6, так как ограничений по временному лагу процесс не ставит.

Таблица 5.

Выбор временного лага

Временной лаг	Ошибка		
	Количество заявок	Размер запроса	Размер ответа
2	0.440	65.128	222.932
4	0.420	60.597	207.884
6	0.407	60.743	225.981
8	0.425	62.233	243.772

Таблица 6.

Выбор размера прогнозируемого диапазона

Размер прогнозируемого диапазона	Ошибка		
	Количество заявок	Размер запроса	Размер ответа
1	0.420	60.597	207.884
2	0.419	61.255	207.072
3	0.428	60.744	219.996
4	0.422	61.127	185.808
6	0.405	58.802	180.376

Таблица 7.

Совмещенный выбор размера диапазонов

Временной лаг	Размер прогнозируемого диапазона	Ошибка		
		Количество заявок	Размер запроса	Размер ответа
4	4	0.422	61.127	185.808
4	6	0.405	58.802	180.376
6	4	0.431	61.682	187.447
6	6	0.385	57.575	184.988

В итоге, оптимальными для данного процесса и исходных данных являются следующие гиперпараметры:

- оптимизатор – nadam;
- количество LSTM модулей – 20;
- количество эпох обучения – 50;
- размер пакета – 10;
- временной лаг – 6;
- размера прогнозируемого диапазона – 6.

Выполним прогнозирование при помощи оптимизированной нейронной сети на проверочных данных. На рисунке 3 отображен график изменения потерь во время обучения на тренировочных и проверочных данных. На рисунке 4. График предсказанных данных наложен на исходные.

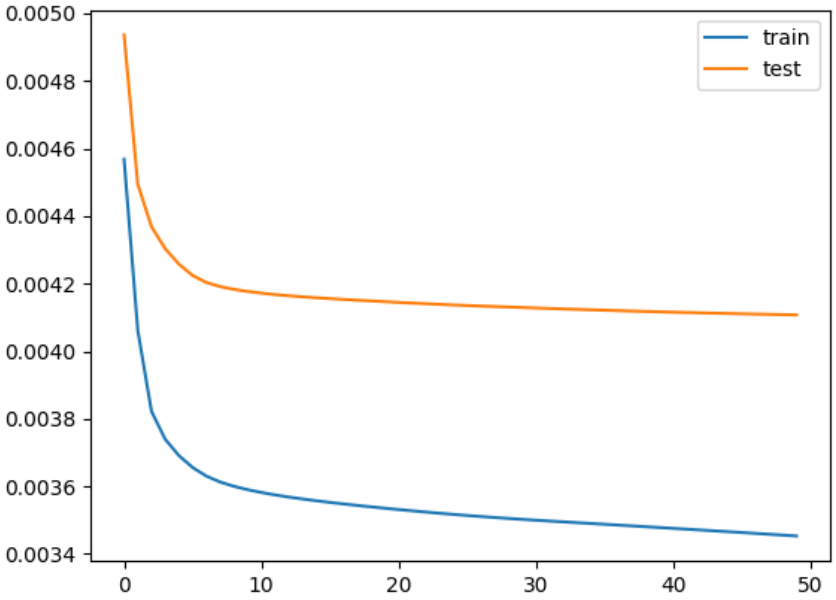

Рисунок 3 – Потери во время обучения

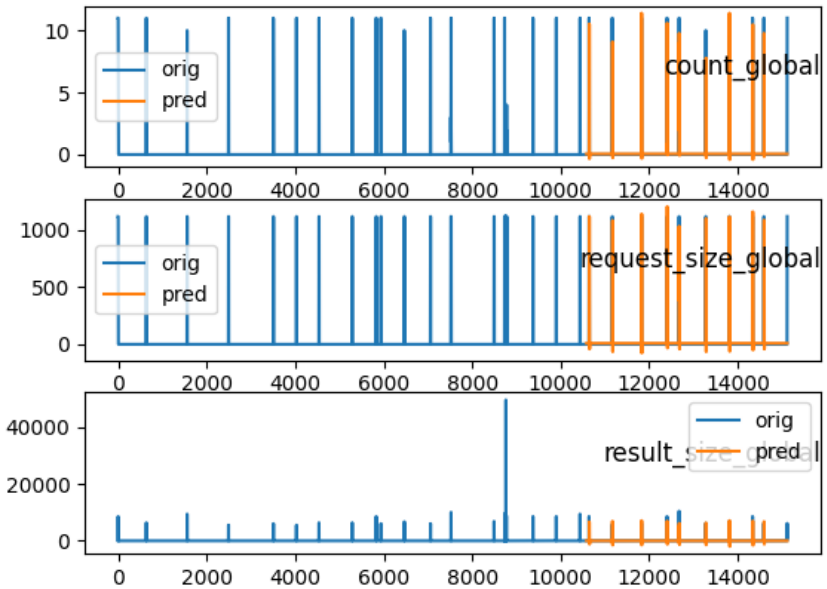

Рисунок 4 – Итоговый прогноз оптимизированной моделью

В результате построенная сеть достаточно качественно выполняет прогноз и удовлетворяет требованиям задачи.

Список литературы:

1. Dozat, T. Incorporating Nesterov Momentum into Adam / T. Dozat // ICLR Workshop. — 2016.
2. Dupond, S. A thorough review on the current advance of neural network structures / S. Dupond // Annual Reviews in Control. — 2019. — Vol. 14. — P. 200-230.
3. Kingma, D. Adam: A Method for Stochastic Optimization / D. Kingma, J. Ba // ICLR. — 2014.

КЛАССИФИКАЦИЯ ПОЛЬЗОВАТЕЛЕЙ СОЦИАЛЬНЫХ СЕТЕЙ ПО ПРОФЕССИОНАЛЬНОЙ ОРИЕНТАЦИИ

Обрубова Василиса Денисовна
Магистрант 2 курса кафедры информационных систем и программной инженерии
“Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых” (ВлГУ),
город Владимир
Озерова Марина Игоревна
кандидат технических наук, доцент
“Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых” (ВлГУ),
город Владимир

SOCIAL NETWORKS USER CLASSIFICATION FOR PROFESSIONAL ORIENTATION

Obrubova Vasilisa Denisovna
2nd year master's student of the Department of Information Systems and Software Engineering
"Vladimir State University named after Alexander Grigorievich and Nikolai Grigorievich Stoletovs" (VlSU),
Vladimir
Ozerova Marina Igorevna
candidate of technical sciences, associate professor
"Vladimir State University named after Alexander Grigorievich and Nikolai Grigorievich Stoletovs" (VlSU),
Vladimir
DOI: 10.31618/nas.2413-5291.2021.2.68.451

АННОТАЦИЯ

В статье рассматривается комплексная постановка темы классификации пользователей социальных сетей для определения профессиональной ориентации.

ABSTRACT

The article deals with a complex formulation of the topic social networks users classification to determine professional orientation.

Ключевые слова: классификация; алгоритмы классификации; профориентология; социальные сети.
Keywords: classification; classification algorithms; vocational guidance; social networks.

Предметная область по данной теме находится на стыке трех направлений (рисунок 1). Первое направление – это профессиональное ориентирование или так называемая наука профориентология. Второе направление – это исследование социальных сетей. И третье направление – это сам подход по решению, а именно машинное обучение.