

УДК 004.048

***ПРОГНОЗИРОВАНИЕ СТОИМОСТИ КРИПТОВАЛЮТ С ПОМОЩЬЮ
БИБЛИОТЕКИ KERAS***

Науменко В.А.

Магистр

Национальный Исследовательский Ядерный Университет «МИФИ»

Москва, Россия

Филиппов С.А.

к.т.н., доцент

Национальный Исследовательский Ядерный Университет «МИФИ»

Москва, Россия

Аннотация: в данной статье рассмотрено применение глубоких нейросетей для прогнозирования курса криптовалюты Bitcoin. Представлен процесс получения исторических данных о курсах этой криптовалюты, указан формат полученных данных. Кроме того, описан процесс формирования обучающей выборки в машинном обучении. В процессе разработки и обучения нейросети использовались библиотеки языка Python: pandas, numpy, matplotlib, keras, tensorflow.

Ключевые слова: глубокие нейронные сети, криптовалюта, прогнозирование, машинное обучение, библиотека Keras

***PRICE FORECASTING CRYPTOCURRENCIES USING THE KERAS
LIBRARY***

Naumenko V.A.

Master

National Research Nuclear University “MEPhI”

Moscow, Russia

Filippov S.A.

PhD, Associate Professor,

National Research Nuclear University “MEPhI”

Moscow, Russia

Annotation: this article provides the use of deep neural networks for predicting the exchange rate of the Bitcoin cryptocurrency. The process of obtaining historical data on the exchange rates of this cryptocurrency is presented, and the format of the received data is specified. In addition, the process of forming a training sample in machine learning is described. During the development and training of the neural network, Python libraries were used: pandas, numpy, matplotlib, keras, and tensorflow.

Keywords: deep neural networks, cryptocurrency, forecasting, machine learning, Keras library.

Одним из главных направлений искусственного интеллекта является машинное обучение. Традиционные алгоритмы проигрывают машинному обучению в возможности обучения. На текущий момент машинное обучение можно назвать одним из самых востребованных инструментов для ведения бизнеса.

Широкий функционал машинного обучения позволяет использовать его в различных задачах обработки информации: распознавание лиц, объектов, речи, перевод и так далее. В сравнении с традиционными алгоритмами у машинного обучения имеется возможность самостоятельно определять шаблоны и уже на их основе делать прогнозы [5]. Применение методов машинного обучения требует сбора данных за предшествующий период [4]. Программный сбор котировок криптовалют позволяет получить широкие возможности для анализа и дальнейшего прогнозирования.

Целью данной работы является проверка гипотезы о применимости сверточных сетей для прогнозирования временных рядов.

Для реализации цели ставятся следующие задачи:

1. Сбор данных;
2. Разработка архитектуры нейросети;
3. Реализация архитектуры нейросети на языке Python;
4. Обучение сети;
5. Вычисление средней ошибки для сформированного прогноза при различных функциях потерь;

Нейронные сети будут использоваться как базовая модель. Особенность обучения нейронной сети – это способность сети находить зависимости между входными и выходными данными и производить обобщение. Кроме того, нейросеть может предсказывать будущие значения некоторой последовательности на основе исторических данных.

Компания «Финам» предоставляет на своем сайте доступ к котировкам за различные периоды. С сайта компании для анализа были получены котировки за указанный на рис. 1 период. Данные, загруженные с сайта представляют собой дату торгов, время открытия торгов, цену на момент открытия, максимальную

цену на текущую дату, минимальную цену на текущую дату, цену на момент закрытия, объем. Интервал, за который взяты данные, составляет 4 года, а периодичность – 5 минут [6].

Криптовалюты ▼ **BTC-USD** ▼ ☆

Интервал и периодичность: 01.01.2016 [📅] – 01.01.2020 [📅] 5 мин. ▼

Имя выходного файла: GDAX.BTC-USD_160101_200101 .csv ▼

Имя контракта: GDAX.BTC-USD

Формат: даты: ггггммдд ▼ времени: ЧЧММСС ▼

Выдавать время: начала свечи окончания свечи московское

Разделитель: полей: запятая (,) ▼ разрядов: нет ▼

Формат записи в файл: DATE, TIME, OPEN, HIGH, LOW, CLOSE, VOL ▼

Добавить заголовок файла:

Заполнять периоды без сделок:

Рис. 1 – Получение данных для анализа

Данные принято делить на тестовую, валидационную и обучающую выборки. Обучающая выборка нужна для обучения алгоритма, валидационная – для подбора параметров, а тестовая – для того, чтобы оценить качество работы алгоритма. Раньше считалось, что тестовая выборка большого размера необходима для подтверждения факта высокого качества алгоритма. Однако сегодня данная практика становится менее используемой, в связи с тем, что перед машинным обучением анализа свыше миллиарда примеров. Поэтому доли тестовых и валидационных данных сокращаются. Нет никакой серьезной

необходимости использовать чрезмерно большую тестовую выборку, чтобы оценить качество работы алгоритма [2].

Полученные с сайта компании «Финам» котировки криптовалют представляют собой временной ряд, с которым нейросети прекрасно справляются. На текущий момент одним из часто используемых подходов к решению разнообразных задач является применение глубоких нейронных сетей. До сих пор не существует определения, с помощью которого можно отличить сеть данного типа от других. В этой работе глубокой нейронной сетью будем называть нейросеть, содержащую два и более скрытых слоев.

Сверточные сети являются отдельным видом нейросетей прямого типа распространения. Прямое распространение подразумевает собой разбиение переменных нейронов на группы, называемые слоями. Во время применения такой сети к данным значения этих переменных, то есть активация слоев считается последовательно: начиная значением активации самого первого слоя он же является входным, затем каждого следующего и заканчивая значением активации выходного слоя [3]. Причем в каждом слое сети есть параметры, которые влияют на то, как активация слоя зависит от активации предшествующего слоя. Кроме того, возможность параллельного вычисления значения активации внутри одного слоя позволяет эффективно обчислять такие нейронные сети на графических платах.

Keras заслуженно является одной из наиболее мощных библиотек Python, используемых при оценке и разработке моделей глубокого обучения. Keras предоставляет возможность разрабатывать нейронные сети благодаря простому функционалу [1, 7]. Создание нейронной сети со слоями, идущими последовательно, возможно благодаря функции Sequential (рис.2).

```
model = Sequential()
model.add(Conv1D(50, 5, input_shape = (xTrain.shape[1], xTrain.shape[2]), activation="linear"))
model.add(Flatten())
model.add(Dense(10, activation="linear"))
model.add(Dense(yTrain.shape[1], activation="linear"))
model.compile(loss="mae", optimizer=Adam(lr=1e-4))
model.fit(xTrain, yTrain, epochs=50, batch_size=20, verbose=1, validation_data=(xVal, yVal))
```

Рис. 2 – Разработка сети с использованием библиотеки Keras в Python

Слои для сети добавляются через add(). В конфигурацию сети сначала добавляем Conv1D - сверточный слой, работающий со входными данными как с одномерным вектором, при этом 50 – это количество узлов в слое, kernel_size = 5 – размерность ядра, то есть размерность матрицы фильтра для сети, input_shape – размерность входного набора, activation="linear" – линейная активационная функция. Затем идет слой выравнивания Flatten, служащий связующим элементом между слоями.

Следующие два слоя – это слои пуллинга Dense с 10 узлами у первого и 1 у второго, функции активации у них также линейные. Для компиляции модели необходимо определить следующие параметры: оптимизатор Adam, функция потерь mean squared error (mse), то есть среднеквадратичная ошибка, скорость обучения learning rating.

Обучение модели будет происходить посредством функции fit() с необходимыми параметрами: данные обучения (x_train), целевые данные (y_train), количество эпох (epochs), размер каждого блока данных (batch_size) и валидационные данные (validation_data).

Результаты обучения нейронной сети показывают, что значения ошибки не превышает 0,009 на обучающем наборе и 0,002 на валидационной выборке. В качестве функции потерь была использована среднеквадратичная ошибка (рис.3-4).

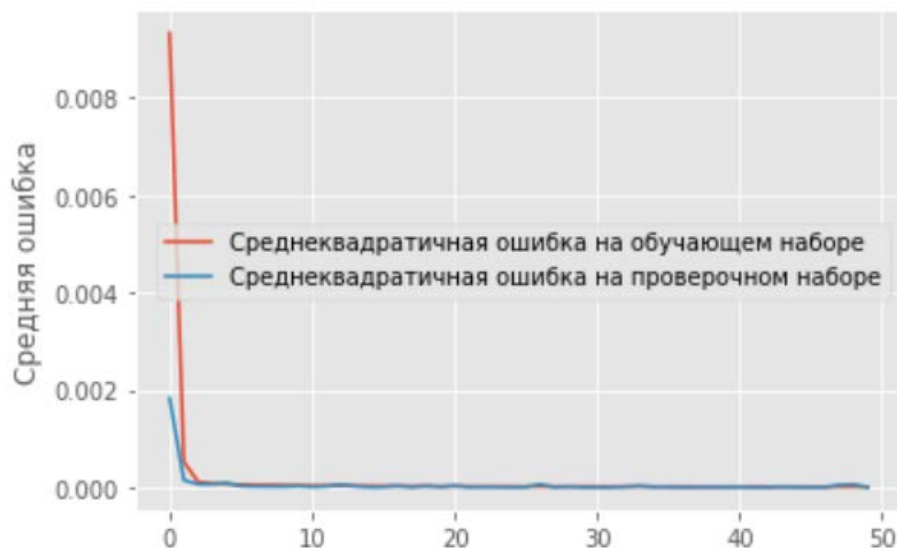


Рис. 3 – Величина средней ошибки от количества эпох на обучающей и тестовой выборках с функцией потерь в виде среднеквадратичной ошибки

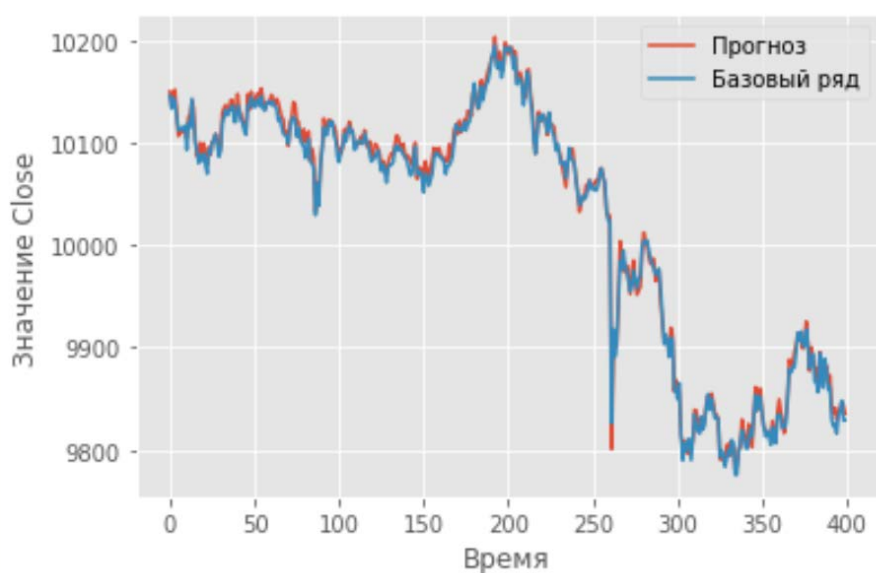


Рис. 4 – Результаты прогноза на обучающей и тестовой выборках с функцией потерь в виде среднеквадратичной ошибки

При этом в сравнении с нейросетью, в которой в качестве функции потерь используется средняя абсолютная ошибка, описанная выше модель дает менее точные результаты обучения на тех же выборках (рис.5-6).



Рис. 5 – Величина средней ошибки от количества эпох на обучающей и тестовой выборках с функцией потерь в виде средней абсолютной ошибки

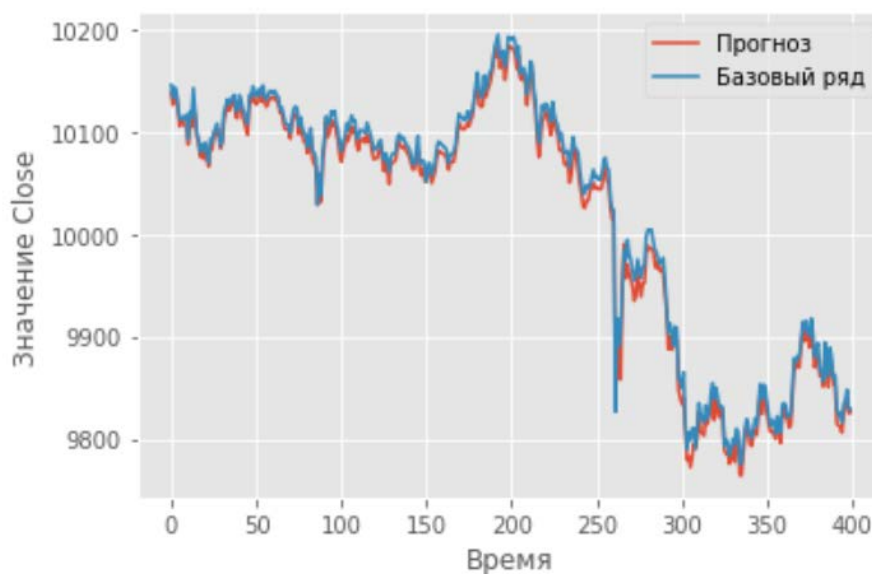


Рис. 6 – Результаты прогноза на обучающей и тестовой выборках с функцией потерь в виде средней абсолютной ошибки

Результаты, полученные в ходе работы, позволяют сделать вывод о высокой точности прогноза, сформированного нейросетью, и подтверждают гипотезу о применимости глубоких нейросетей к временным рядам в задачах прогнозирования.

В работе рассмотрено понятие глубоких нейронных сетей, их отличие от других типов нейросетей. Помимо этого, также был рассмотрен принцип формирования выборок, была разработана архитектура нейронной сети с помощью библиотеки Keras. Проведенные эксперименты позволили оценить величину ошибки на обучающем и тестовом наборах и сделать вывод о пригодности данной архитектуры нейронной сети для анализа котировок криптовалюты Bitcoin. Дальнейшие исследования будут направлены на усовершенствование архитектуры сети и применение на котировках других криптовалют.

Библиографический список

1. Майк МакГрат. Программирование на Python для начинающих. (перевод с английского М. А. Райтмон) Москва. – Эксмо 2015. – 192с.
2. Созыкин А. В. Обзор методов обучения глубоких нейронных сетей // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2017. Т. 6, № 3. С. 28-59. DOI: 10.14529/cmse170303
3. Connor J., Martin D. R., Atlas L. E. Recurrent Neural Networks and Robust Time Series Prediction // IEEE Transactions on Neural Networks. 1994. Vol. 5, No. 2. P. 240-254. DOI: 10.1109/72.279188
4. Peter J. Brockwell, Richard A. Davis: Introduction to Time Series and Forecasting. Second Edition. Springer-Verlag New York, Inc., 175 Fifth Avenue, New York, NY 10010, USA (2002)

5. Sergii, K., Yurii, S., Tatyana, V., Natalia, A.: Feature Selection for Time-Series Prediction in Case of Undetermined Estimation. In: Samsonovich A., Klimov V., Rybina G. (eds.) Biologically Inspired Cognitive Architectures (BICA) for Young Scientists, Advances in Intelligent Systems and Computing, vol. 449, pp. 85-97. Springer, Cham (2016). DOI: 10.1007 / 978-3-319-32554-5_12
6. Котировки на мировых биржах онлайн / [Электронный ресурс]. – Режим доступа: <https://www.finam.ru/quotes> (дата обращения 10.04.2020).
7. Keras: The Python Deep Learning library / [Электронный ресурс]. – Режим доступа: <https://keras.io/> (дата обращения 10.04.2020).

Оригинальность 83%