



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

НА ТЕМУ:

Гибридный интеллектуальный бот для
управления финансовыми активами

Студент ИУ5-32М
(Группа)

Овчинников С.С.
(Подпись, дата) (И.О.Фамилия)

Руководитель

Ю.Е. Гапанюк
(Подпись, дата) (И.О.Фамилия)

2023 г.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

УТВЕРЖДАЮ
Заведующий кафедрой ИУ5
(Индекс)
В.И. Терехов
(И.О.Фамилия)
« 04 » _____ сентября 2023 г.

ЗАДАНИЕ
на выполнение научно-исследовательской работы

по теме _____

Студент группы ИУ5-32М

Овчинников Степан Сергеевич
(Фамилия, имя, отчество)

Направленность НИР (учебная, исследовательская, практическая, производственная, др.)

ИССЛЕДОВАТЕЛЬСКАЯ

Источник тематики (кафедра, предприятие, НИР) КАФЕДРА

График выполнения НИР: 25% к ____ нед., 50% к ____ нед., 75% к ____ нед., 100% к ____ нед.

Техническое задание создать торгового бота и обучить его на нескольких моделях для предсказания последующего курса актива

Оформление научно-исследовательской работы:

Расчетно-пояснительная записка на 19 листах формата А4.

Перечень графического (иллюстративного) материала (чертежи, плакаты, слайды и т.п.)

Дата выдачи задания « 04 » _____ сентября 2023 г.

Руководитель НИР

Ю.Е. Гапанюк
(Подпись, дата) (И.О.Фамилия)

Студент

Овчинников С.С.
(Подпись, дата) (И.О.Фамилия)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

Оглавление

Оглавление	3
1. Постановка проектируемой задачи.....	4
2. Выбор СУБД и веб-сервера	8
3. Технический анализ свечей	12
4. Архитектура моделей.....	15
5. Обучение моделей	18
6. Сравнение и оценка работы моделей	21
Список использованных источников.....	22

1. Постановка проектируемой задачи

Для организации современного взаимодействия пользователя с торговым криптовалютным ботом предлагается обучить несколько моделей различными способами и реализовать развертывание на сервере. Торговый бот будет помогать пользователю предсказывать курс криптовалюты на коротких промежутках времени на коротких интервалах времени. Предлагаемые результаты будут основываться на данных, собранных с криптовалютной биржи Binance. Для этого нужно организовать правильный сбор данных, а также проанализировать какие из них представляют важность в получении результата.

Бот должен отвечать главному запросу: предсказывать наиболее вероятный через короткий промежуток курс криптовалюты. Результаты должны быть отображены наиболее удобным и понятным образом для пользователя. Взаимодействие пользователя с системой должно быть удобным и простым для понимания.

Помимо этого, с изменением промежутка времени, цены, обученных моделей бот должен делать предсказание заново.

Помимо этого, с изменением промежутка времени, цены, обученных моделей бот должен делать предсказание заново.

Таким образом, задача проектирования может быть сформулирована следующим образом:

- Изучить различные типы моделей для предсказаний.
- Проанализировать и собрать данные об изменениях цен криптовалюты.
- Подобрать подходящую модель для предсказаний, изучить возможные для нее алгоритмы.
- Разработать модель для предсказаний на основе выбранных алгоритмов, с помощью нейронной сети.
- Обучить систему с использованием собранных данных.
- Проанализировать результаты работы модели для предсказаний.
- Настроить визуальное отображение результатов.

Так как специфика торговли для всех пользователей разная и данные о криптовалютах также везде разные – необходимо разрабатывать под каждое

приложение своего бота, который будет учитывать специфику конкретного пользователя и его стратегий.

2. Выбор СУБД и веб-сервера

Для хранения данных, которые будут использоваться в построении моделей была выбрана СУБД ClickHouse.

ClickHouse – колоночная реляционная СУБД с открытым исходным кодом от компании Яндекс для быстрой обработки аналитических SQL-запросов на структурированных больших данных в режиме реального времени.

Ключевым преимуществом ClickHouse считается высокая скорость выполнения SQL-запросов на чтение (OLAP-сценарий), которая обеспечивается благодаря следующим архитектурным особенностям:

- Столбцовое хранение данных, что позволяет считывать данные только из нужных колонок и эффективно сжимать однотипную информацию
- Физическая сортировка данных по первичному ключу позволяет быстро получать конкретные значения или диапазонов
- Векторные вычисления по кусочкам столбцов снижают издержки на диспетчеризацию и позволяют более эффективно использовать CPU
- Кроссплатформенность
- Распараллеливание операций как в пределах одного сервера на несколько процессорных ядер, так и в рамках распределенных вычислений на кластере за счет механизма шардирования
- Поддержка приближенных вычислений на части выборки, что снижает число обращений к жесткому диску и еще больше повышает скорость обработки данных
- Простота развертывания и удобство эксплуатации
- Высокая доступность и отказоустойчивость

Вышеописанные преимущества обуславливают высокую популярность ClickHouse. Также в плане эксплуатации стоит отметить пониженную стоимость хранения данных по сравнению с другими столбцовыми СУБД, такими как SAP HANA и Google PowerDrill, которые работают только в RAM. ClickHouse ориентирован именно на эффективную работу с жесткими дисками, которые намного дешевле оперативной памяти.

VPS (Virtual Private Server) — это хостинг-услуга, где пользователю предоставляется виртуальный сервер с максимальными привилегиями. VDS или VPS эмулирует работу реального физического сервера — есть root-доступ, возможна установка своих операционных систем и программного обеспечения. На одном физическом сервере обычно работает несколько независимых виртуальных серверов.

Плюсы использования VDS/VPS по сравнению с виртуальным хостингом:

- Полный контроль: владелец виртуального выделенного сервера получает права доступа уровня root (для Unix) или Administrator (для Windows), и может тонко настраивать сервер под нужды конкретного проекта вплоть до установки необходимого ПО и смены операционной системы.
- Минимум ограничений: количество создаваемых сайтов, баз данных, доменных зон, ssh- ftp- email-пользователей ограничивается только ресурсами сервера.
- Независимость: виртуальные серверы-«соседи» по хост-машине работают изолированно и не «перетягивают» друг у друга ресурсы, а также используют отдельные IP-адреса, в то время как на обычном виртуальном хостинге несколько сайтов нередко используют один IP-адрес.

Плюсы использования VPS по сравнению с выделенным сервером:

- Невысокая стоимость: аренда виртуального выделенного сервера обходится пользователю значительно дешевле аренды сопоставимого по мощности физического сервера. При этом производительность VDS / VPS может даже оказаться выше (смотри наше исследование)
- Простая масштабируемость: виртуальному серверу очень легко добавить мощности в виде дополнительного объема оперативной памяти, процессорных ядер или объёма жесткого диска.

3. Технический анализ и анализ свечей

Данные загружаются с одной из крупнейших криптовалютных бирж - Binance. Данные были получены с помощью Binance API. API используется не только в криптовалютах, но и в любых других приложениях. Это аббревиатура английского `application programming interface` означает интерфейс взаимодействия приложений. API позволяет двум приложениям взаимодействовать друг с другом. API ключ позволяет двум приложениям идентифицировать друг друга. После того, как вы их представили — они смогут взаимодействовать без участия пользователя.

Далее после идентификации происходит получение данных о свечах. Свечи (японские свечи) - это графический способ отображения ценовой динамики, при котором используются вертикальные прямоугольники и линии. Этот метод изобретен на рисовых биржах Японии в 17-м веке, откуда и получил свое название. Каждая свеча показывает цены открытия и закрытия, максимум и минимум цены, также среднюю цену по свече. Именно эти данные о свечах будут использованы в дальнейшем.

Существуют несколько паттернов, которые будут влиять на предсказание, после обработки данных.

Двойная вершина напоминает букву «М». Это – паттерн разворота. Образуется следующим образом: график актива пытается продолжить движение вверх, дважды доходя до воображаемой линии сопротивления и дважды от нее с силой отталкиваясь. После такого маневра чаще всего начинает формироваться тренд вниз.

Когда трейдеры видят, что на графике появляется «Двойная вершина», стараются открывать сделки на понижение через 1-2 свечи после второго отскока вниз.

Двойное дно напоминает букву «W» и формируется после продолжительного нисходящего тренда. В отличие от двойной вершины, модель двойного дна показывает, что после второго столкновения с сопротивлением последует разворот вверх.

После второй неудачной попытки трейдеры обычно открывают сделки на покупку, подождав, пока сформируются 1-2 подтверждающие японские свечи.

Фигура голова и плечи состоит из первого пика цены (это левое плечо), за которым следует более высокий пик (это голова), и завершает композицию более

низкий пик (правое плечо). Такой паттерн сигнализирует о том, что после формирования правого плеча цена, скорее всего, отскочит и перейдет к снижению.

При этом, линия шеи – это уровень поддержки, от которого отталкивается цена. Когда график проходит эту «ватерлинию» сверху вниз, трейдеры ищут удачный момент для открытия сделки на продажу.

Восходящий клин указывает на паузу в текущем тренде и демонстрирует, что на рынке сейчас период нерешительности. После Клина тренд может как продолжиться, так и развернуться в обратном направлении.

Если во время нисходящего тренда возникает восходящий клин, он выступает в качестве модели продолжения тренда. Как показано на графике справа, это происходит потому, что цена пробивает линию сопротивления вниз и продолжает падение.

Однако если восходящий клин появляется во время восходящего тренда, он выступает в качестве модели разворота графика японских свечей. Это происходит потому, что цена доходит до линии сопротивления и начинает снижаться.

Нисходящий клин сигнализировать о развороте или продолжении тренда. В отличие от Восходящего Клина, Нисходящий чаще всего предшествует дальнейшему росту цены, что бы ни происходило до этого.

Как правило, Нисходящий клин после тренда вниз указывает на разворот вверх, а во время восходящего тренда подтверждает его направление.

4. Архитектура моделей

LSTM-сеть — это искусственная нейронная сеть, содержащая LSTM-модули вместо или в дополнение к другим сетевым модулям. LSTM-модуль — это рекуррентный модуль сети, способный запоминать значения как на короткие, так и на длинные промежутки времени. Ключом к данной возможности является то, что LSTM-модуль не использует функцию активации внутри своих рекуррентных компонентов. Таким образом, хранимое значение не размывается во времени, и градиент или штраф не исчезает при использовании метода

обратного распространения ошибки во времени при обучении искусственной нейронной сети. LSTM-модули часто группируются в «блоки», содержащие различные LSTM-модули. Подобное устройство характерно для «глубоких» многослойных нейронных сетей и способствует выполнению параллельных вычислений с применением соответствующего оборудования. В формулах ниже каждая переменная, записанная строчным курсивом, обозначает вектор размерности равной числу LSTM-модулей в блоке[10][11].

LSTM-блоки содержат три или четыре «вентили», которые используются для контроля потоков информации на входах и на выходах памяти данных блоков. Эти вентили реализованы в виде логистической функции для вычисления значения в диапазоне $[0; 1]$. Умножение на это значение используется для частичного допуска или запрещения потока информации внутрь и наружу памяти. Например, «входной вентиль» контролирует меру вхождения нового значения в память, а «вентиль забывания» контролирует меру сохранения значения в памяти. «Выходной вентиль» контролирует меру того, в какой степени значение, находящееся в памяти, используется при расчёте выходной функции активации для блока. На первом этапе LSTM нужно решить, какую информацию мы собираемся выбросить из состояния ячейки. Это решение принимается сигмовидным слоем, называемым «слоем гейта утраты». Он получает на вход h и x и выдает число от 0 до 1 для каждого номера в состоянии ячейки C . 1 означает «полностью сохранить», а 0 — «полностью удалить». На следующем шаге нужно решить, какую новую информацию сохранить в состоянии ячейки. Разобьем процесс на две части[12]. Сначала сигмоидный слой, называемый «слоем гейта входа», решает, какие значения требуется обновить. Затем слой \tanh создает вектор новых значений-кандидатов C , которые добавляются в состояние. На следующем шаге мы объединим эти два значения для обновления состояния. Теперь обновим предыдущее состояние ячейки для получения нового состояния C . Способ обновления выбран, теперь реализуем само обновление.

Умножим старое состояние на f , теряя информацию, которую решили забыть. Затем добавляем $i * C$. Это новые значения кандидатов, масштабируемые в зависимости от того, как мы решили обновить каждое значение состояния. Результат будет являться отфильтрованным состоянием ячейки. Сначала запускаем сигмоидный слой, который решает, какие части состояния ячейки выводить. Затем пропускаем состояние ячейки через \tanh (чтобы разместить все значения в интервале $[-1, 1]$) и умножаем его на выходной сигнал сигмовидного гейта[2,3,4].

На рисунке ниже изображена традиционная LSTM модель.

$$\begin{aligned}
 f_t &= \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \\
 i_t &= \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \\
 o_t &= \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \\
 c_t &= f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \\
 h_t &= o_t \circ \sigma_h(c_t)
 \end{aligned}$$

Традиционная LSTM модель

На рисунке 16 изображена LSTM модель со сверточным слоем.

$$\begin{aligned}
 f_t &= \sigma_g(W_f * x_t + U_f * h_{t-1} + V_f \circ c_{t-1} + b_f) \\
 i_t &= \sigma_g(W_i * x_t + U_i * h_{t-1} + V_i \circ c_{t-1} + b_i) \\
 o_t &= \sigma_g(W_o * x_t + U_o * h_{t-1} + V_o \circ c_{t-1} + b_o) \\
 c_t &= f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c * x_t + U_c * h_{t-1} + b_c) \\
 h_t &= o_t \circ \sigma_h(c_t)
 \end{aligned}$$

LSTM модель со сверточным слоем

$$\begin{aligned}
 f_t &= \sigma_g(W_f x_t + U_f c_{t-1} + b_f) \\
 i_t &= \sigma_g(W_i x_t + U_i c_{t-1} + b_i) \\
 o_t &= \sigma_g(W_o x_t + U_o c_{t-1} + b_o) \\
 c_t &= f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + b_c) \\
 h_t &= o_t \circ \sigma_h(c_t)
 \end{aligned}$$

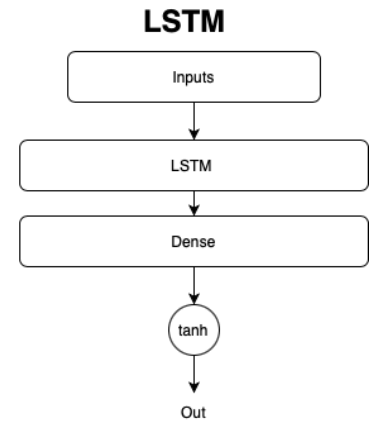
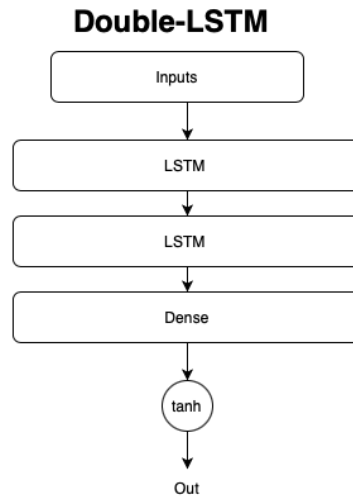
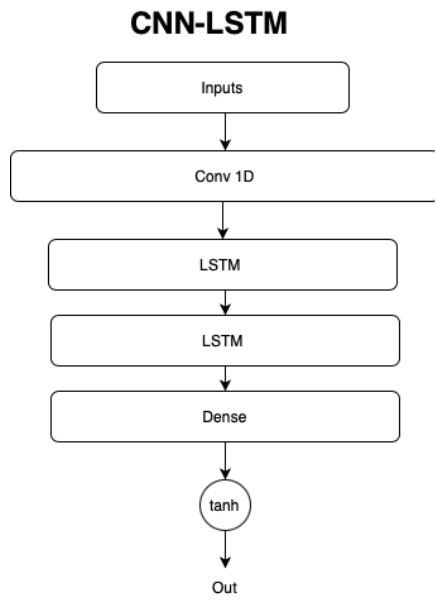
Двойная LSTM модель

Архитектура выбранных трех моделей для обучения представлена на рисунке ниже. На нем мы можем наблюдать подробную структуру изображения

каждой

МОДЕЛИ.

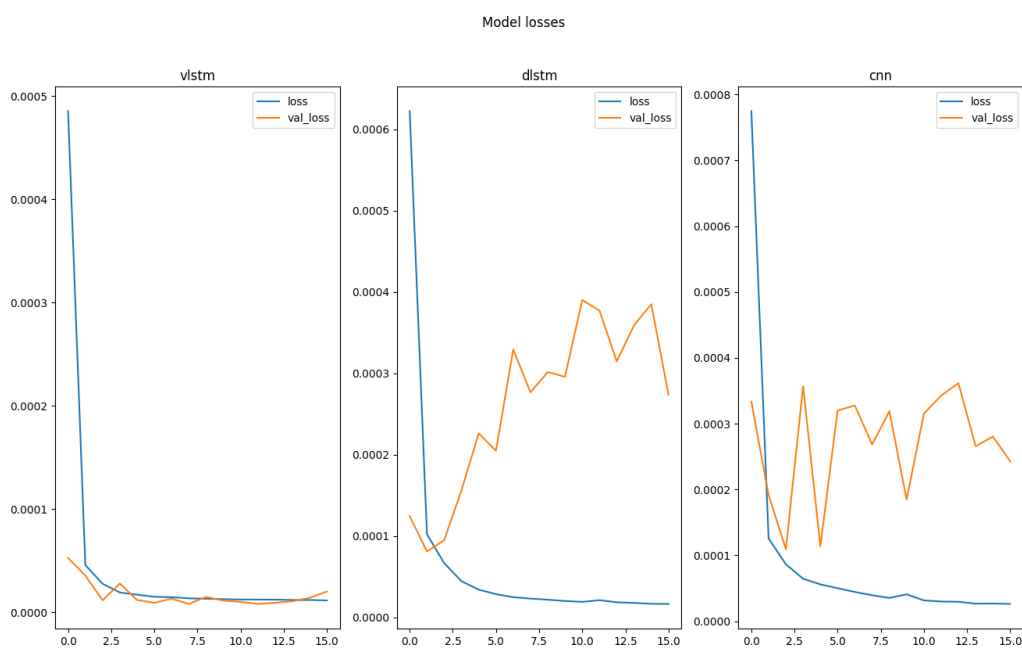
[6,7,9]



5. Обучение моделей

Модели обучали на протяжении 16 эпох, batch size взяли равным 16 - что давало приемлемое соотношение качества и скорости обучения. В конце каждой эпохи, для каждой модели оценивался loss на обучающей выборке, и loss на тестовой выборке. Чтобы избежать переобучения, сохранялась та модель, для которой loss на тестовой выборке был минимальным.

На рисунке ниже показаны значения loss и val_loss - ошибка на обучающей и тестовой выборке соответственно.



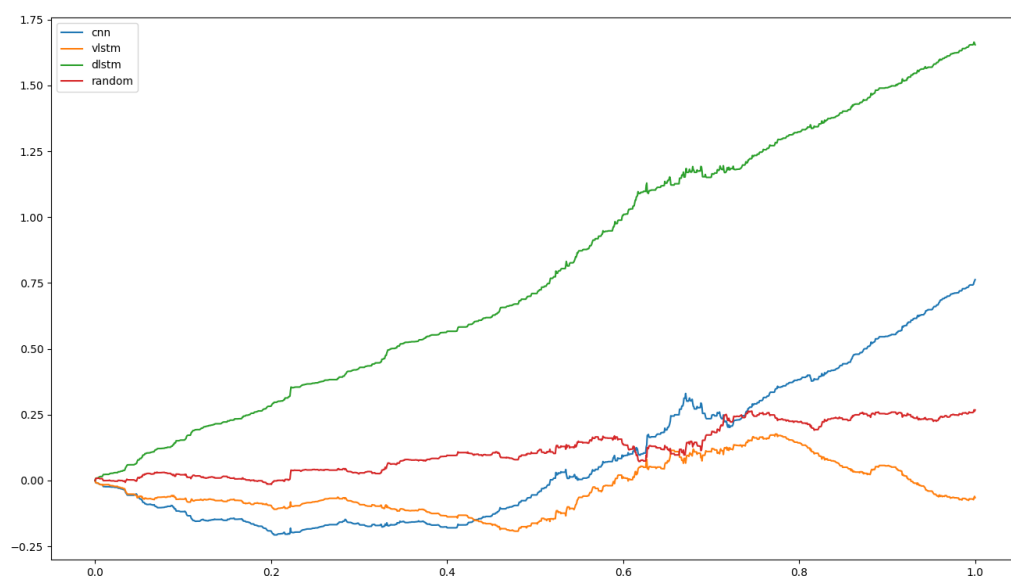
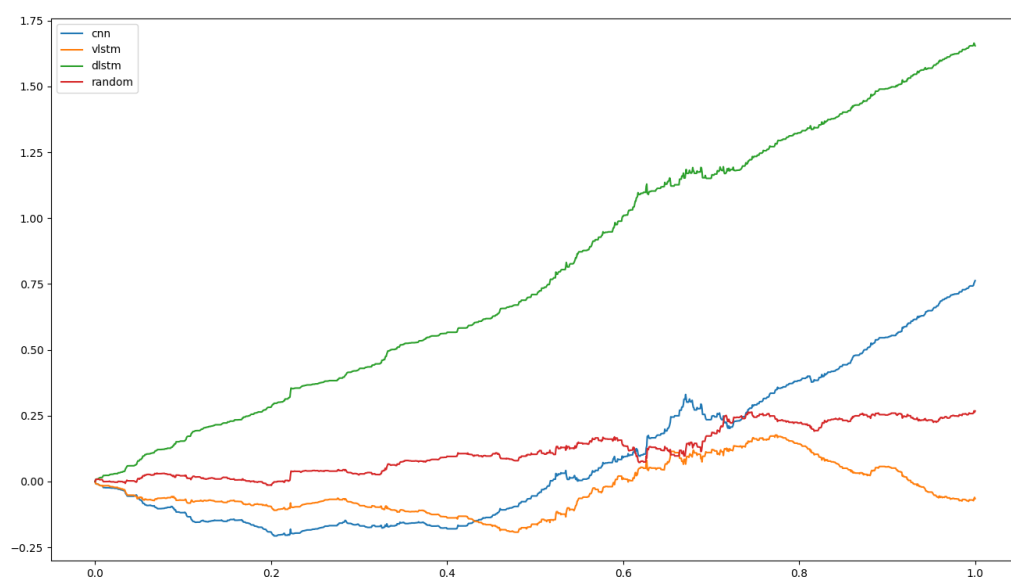
Обучим модель с этими параметрами и получим лучшие результаты по ранее выбранным метрикам.

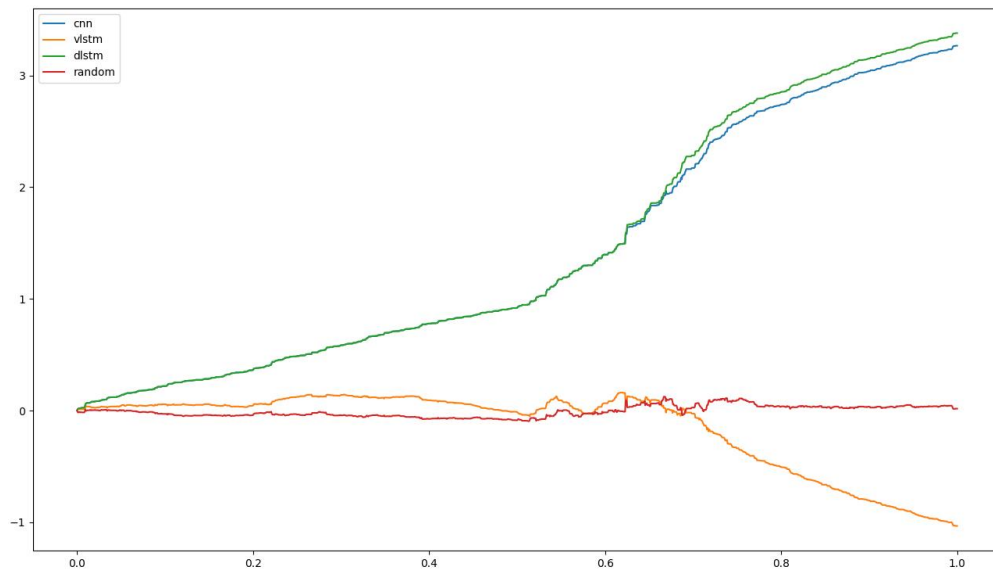
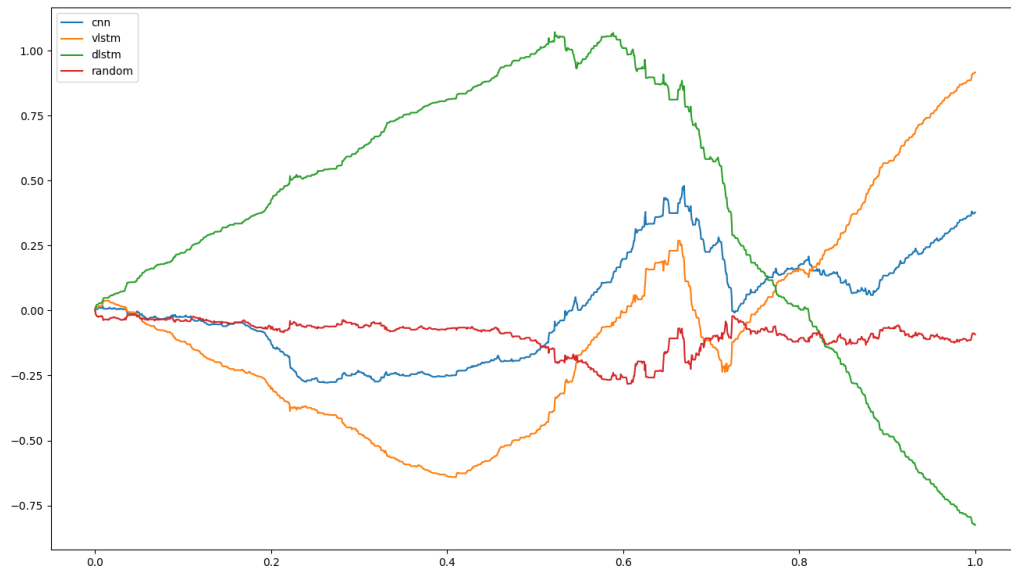
По результатам обучения моделей можно увидеть, что общая оценка по моделям не улучшится значительно. То есть для каждой модели средняя квадратичная ошибка соответствует определенным образом ценовому диапазону свечей. В итоге в веб-сервис будут интегрированы три модели машинного обучения, позволяющие предсказать курс криптовалюты в течение ближайших 15 минут.

6. Сравнение и оценка работы моделей

Для оценки эффективности моделей использовался следующий подход:

1. Модели обучили на данные за выбранный промежуток времени: с января 2020 года, по апрель 2022.
2. В качестве проверочных данных взяли данные за май 2022 г (20 дней)
3. На основе новых данных, которые не участвовали в процессе обучения модели, были сгенерированы предсказания - прогноз значений курсов криптовалют, на определенных промежутках времени.
4. Для каждого промежутка времени получили пары значений - действительный курс криптовалюты, и предсказанный курс.
5. В качестве торговой стратегии использовали простой принцип: если прогнозируется падение курса - нужно продать криптовалюту, если прогнозируется рост - купить/удерживать.
6. На основе вышеприведенной стратегии, был сгенерирован график относительного роста цены портфеля криптовалют, в процентах. (Рис. 21)
7. В качестве baseline'a были взяты предсказания на основе равномерно распределенных случайных значений, прогнозирующих падение или рост курса криптовалюты (значение отображено красным на графике)





На основе полученных значений, можно видеть, что модель double lstm ушла в большой плюс, cnn lstm незначительно опередила по показателям стратегию на основе случайных чисел. Худшие результаты показала простая lstm модель.

ЗАКЛЮЧЕНИЕ

Разработанная модель позволяет предсказать кратковременное движение криптовалюты. При необходимости в модели можно увеличить количество

входных данных, что поможет более детально обучить модель и давать более точные предсказания дальнейшего движения криптовалютного актива. При разработке гибридного торгового бота предсказания стоимости криптовалюты были получены следующие результаты:

- Была изучена предметная область, связанная с криптовалютным рынком;
- Были сформулированы требования к веб-сервису предсказания стоимости криптовалютного актива;
- Были собраны и подготовлены данные, на основе которых было проведено обучение модели;
- Разработанная модель была интегрирована в веб-сервис;

Данная система в дальнейшем может послужить хорошим инструментом торговли для людей, которые только начинают изучать тему криптовалюты и торговать активами на рынке, так и для людей, кто уже давно торгует.

Список использованных источников

1. Алжеев А. В., Кочкаров Р. А. Сравнительный анализ прогнозных моделей ARIMA и LSTM на примере акций российских компаний. Финансы: теория и практика. – 2020. – Т. 24. – №. 1.
2. Yoon, K. Convolutional Neural Networks for Sentence Classification / К. Yoon [Электронный ресурс] - URL: <https://adventuresinmachinelearning.com/recurrent-neural-networks-lstm-tutorial-tensorflow>, дата обращения 03.03.2022.
3. Кратович П. В. Нейронные сети и модели ARIMA для прогнозирования котировок Программные продукты и системы. – 2011. – №. 1. – С. 95-98.
4. Прогнозирование временных рядов с помощью рекуррентных нейронных сетей Хабр [Электронный ресурс] - URL: <https://habr.com/ru/post/495884/>, дата обращения 21.04.2022.
5. Траск Э. Грокаем глубокое обучение //СПб.: Питер. – 2019.
6. Как разработать модели LSTM для прогнозирования временных рядов Хабр [Электронный ресурс] - URL:<https://www.machinelearningmastery.ru/how-to-develop-lstm-models-for-time-series-forecasting/>, дата обращения 22.04.2022.
7. Математические методы обучения по прецедентам (теория обучения машин) К. В. Воронцов.
8. Каширина И. Л., Демченко М. В. Исследование и сравнительный анализ методов оптимизации, используемых при обучении нейронных сетей //Вестник ВГУ. Серия: Системный анализ и информационные технологии. – 2018. – №. 4. – С. 123-132.

9. Воронцов К.В. Комбинаторный подход к оценке качества обучаемых алгоритмов. Математические вопросы кибернетики / Под ред. О.Б.Лупанова. — М.: Физматлит, 2004. -Т.13.-С.5.
10. Андреас Мюллер, Сара Гвидо Введение в машинное обучение с помощью Python. Руководство для специалистов по работе с данными. / М.: Вильямс, 2017. — 480 с.
11. Флах П. Машинное обучение. / М.: ДМК Пресс, 2015. — 400 с.
12. Бринк Хенрик, Ричардс Джозеф, Феверлоф Марк Машинное обучение. — СПб.: Питер, 2017. — 336 - 400 с.