

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра теории функций и стохастического анализа

**РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ
АНАЛИЗА ТОРГОВЫХ ДАННЫХ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студентки 4 курса 451 группы
направления 38.03.05 — Бизнес-информатика

механико-математического факультета

Власовой Елены Константиновны

Научный руководитель

доцент, к. ф.-м. н.

О. А. Мыльцина

Заведующий кафедрой

д. ф.-м. н., доцент

С. П. Сидоров

Саратов 2024

ВВЕДЕНИЕ

Актуальность темы. В мире финансов главную роль играют деньги, но когда речь заходит об их увеличении, то актуальными становятся и другие инструменты. Пример этому – инвестиции в акции, и сам процесс вложения капитала в ценные бумаги с целью получения прибыли можно уверенно называть классикой биржевой торговли. Если в отношении золота и остальных видов драгметаллов среди инвесторов преобладает взгляд как на «тихую гавань» в период кризиса, то акции рассматриваются в качестве инструмента, который требует активности.

Основным преимуществом акций является большое разнообразие выбора ценных бумаг, что позволяет новичкам, только вышедшим на рынок, не потерять большое количество денежных средств в случае неудачной сделки, а опытным участникам рынка - во много раз преумножить свой капитал.

От взлетов и падений в мире финансов никто не застрахован. Принимать решение о целесообразности использовать именно этот инструмент для увеличения прибыли каждый инвестор должен сам. Для помощи в принятии решения есть множество инструментов и показателей, а также методы машинного обучения.

Целью дипломной работы является разработка информационной системы для анализа торговых данных.

Для достижения поставленной цели необходимо решить следующие задачи:

- изучить возможные варианты обучения моделей для наиболее точного предсказания цены акций;
- написать программный код для разного вида обучения моделей;
- написать сайт для визуального представления работы кода и взаимодействия пользователя с программой;
- проанализировать работу и точность предсказания для каждой из моделей и отобразить рекомендации на сайте;
- оценить результаты работы и ввести на сайте разделы для взаимодействия с пользователем.

Дипломная работа состоит из введения, 4 разделов, заключения, списка

использованных источников и приложений.

В первом разделе рассматривается теория о фондовом рынке и процессах, происходящих на нем.

Во втором разделе описаны методы машинного обучения и анализ их применения в данной работе.

В третьем разделе приводится процесс разработки дизайна сайта.

В четвертом разделе описывается процесс разработки информационной системы.

В заключении приведены основные результаты и выводы дипломной работы.

Список источников содержит 20 наименований, на которые в тексте работы присутствуют ссылки.

Основное содержание работы

В **первом** разделе раскрываются основные понятия фондового рынка. Фондовый рынок — это место, где происходит торговля акциями, облигациями, валютами и прочими активами.

Рынок ценных бумаг имеет определенные признаки:

- у него всегда есть фиксированная торговая площадка, например фондовый рынок Московской биржи;
- обязательно наличие специализированного механизма отбора товаров (активов), отвечающих определенным требованиям;
- установлены торговые процедуры по времени и стандартам;
- все оформление сделок централизовано;
- деятельность всех участников рынка контролируется уполномоченными органами;
- существуют официальные котировки активов.

Капитализация фондового рынка — это стоимость всех выпущенных на нем ценных бумаг. Этот показатель является основной оценкой рынка. Ее можно посчитать, умножив количество всех эмитированных бумаг на их рыночные цены. Эта величина не является фиксированной и постоянно изменяется с движением котировок.

Оборот рынка — это общая стоимость ценных бумаг, умноженная на

количество тех бумаг, с которыми совершены сделки. Этот показатель может быть выражен в процентах от уровня капитализации рынка.

Котировки ценных бумаг, курсы на начало и на конец дня, коэффициенты по разным видам активов также могут демонстрировать состояние рынка и учитываются при анализе.

Технический анализ — это анализ на основе исследования изменений цен бумаг в прошлом и используется для определения моментов покупки или продажи. Трейдеры визуализируют цены в виде различных графиков (линии, «свечи» и пр.). На этих графиках можно обнаружить фигуры или закономерности движения цен. Появление таких фигур говорит о дальнейшем движении цен и помогает принять решение о заключении сделок.

Кроме фигур (паттернов), в теханализе применяются различные расчетные показатели — индикаторы. Индикаторы — это усреднение показателей цены или ее динамики. Индикаторы могут накладываться на график цены, например скользящая средняя или линии Боллинджера, а могут отображаться на отдельной шкале и, например, принимать значения от 0 до 100 (стохастический осциллятор, MACD). Индикаторы, которые показывают на отдельной шкале и чьи значения представляют собой коэффициенты и соотношения, называют осцилляторами. Пересечение индикатором определенного уровня дает сигналы к покупке или продаже. Осциллятор также позволяет увидеть уровни перекупленности и перепроданности. В целом технический анализ работает с котировками ценных бумаг, не затрагивая рынок и экономические показатели в стране и мире.

Фундаментальный анализ — это анализ финансового состояния отраслей ценных бумаг, экономического положения стран, благополучия и капитализации компаний. Можно сказать, что фундаментальный анализ показывает реальную стоимость активов. В отношении компаний фундаментальный анализ фактически оценивает финансовое здоровье бизнеса, рынок в целом, сделки конкурентов, прогнозирует возможные инвестиции и дивиденды. Он очень полезен для долгосрочных стратегий.

Во **втором** разделе приведена теория машинного обучения.

Машинное обучение — это наука, изучающая алгоритмы, автоматически улучшающиеся благодаря опыту. Оно является одной из областей искус-

ственного интеллекта (ИИ).

С помощью машинного обучения ИИ может анализировать данные, запоминать информацию, строить прогнозы, воспроизводить готовые модели и выбирать наиболее подходящий вариант из предложенных.

В сфере инвестиций алгоритмы на базе машинного обучения анализируют рынок, отслеживают новости и подбирают активы, которые выгоднее всего покупать именно сейчас. При этом с помощью предикативной аналитики система может предсказать, как будет меняться стоимость тех или иных акций за конкретный период и корректирует свои данные после каждого важного события в отрасли.

В мире Data science задача классификации является одной из самых важных и распространенных задач. Ее основная цель заключается в разделении данных на классы в соответствии с заданными признаками. Линейные модели являются одним из наиболее популярных подходов к решению задачи классификации.

Линейная модель — это математическая модель, которая представляет собой линейную комбинацию входных признаков. В задаче классификации линейная модель используется для разделения данных на два или более классов.

Python предоставляет множество библиотек для работы с линейными моделями. Одна из самых популярных библиотек для работы с линейными моделями в Python — это scikit-learn. Scikit-learn предоставляет множество алгоритмов машинного обучения, в том числе и линейные модели для задачи классификации.

StandardScaler — это важный метод, который в основном выполняется как этап предварительной обработки перед многими моделями машинного обучения, чтобы стандартизировать диапазон функциональных возможностей входного набора данных. Разделим набор данных на обучающую и тестовую выборки, которые в дальнейшем будут использоваться при обучении моделей. StandardScaler используется, когда характеристики входного набора данных сильно различаются между диапазонами или когда они измеряются в разных единицах измерения.

Разделение выборки на обучающую (train) и тестовую (test) является

ключевым этапом в процессе разработки и оценки моделей машинного обучения. Этот процесс позволяет оценить, насколько хорошо модель будет работать на данных, которые она ранее не видела, что является важным аспектом для обеспечения ее обобщающей способности.

Процесс разделения выборки на обучающую и тестовую подвыборки включает в себя следующие шаги:

- разделение на обучающую и тестовую подвыборки: данные случайным образом разделяются на две подвыборки. Обучающая подвыборка используется для обучения модели машинного обучения, а тестовая подвыборка — для оценки ее производительности;
- обучение модели: модель обучается на обучающей подвыборке, используя признаки и соответствующие метки;
- оценка модели: после обучения модель применяется к тестовой подвыборке для предсказания меток. Затем предсказанные метки сравниваются с реальными метками тестовой подвыборки, что позволяет оценить точность и эффективность модели.

Разделение датафрейма на подвыборки происходит при помощи метода `train_test_split()` из библиотеки обработки данных `scikit-learn`. Это помогает свести к минимуму возможность смещения в процессе оценки и проверки.

В **третьем** разделе описывается процесс разработки дизайна сайта. Веб-дизайн имеет очень большие отличия от обычного полиграфического дизайна. Создание дизайна сайта обычно представляет собой определенный процесс, в котором учитываются различные требования не только художественного, но и технического характера. Такой процесс может включать в себя несколько этапов:

- разработка легко воспринимаемого дизайна стартовой страницы, а также других разделов и последующих уровней (в том числе разнообразных форм регистраций, корзин и т.д.);
- разработка грамотно продуманной структуры всего сайта;
- проектирование удобной, достаточно простой навигации по сайту;
- расположение текста на страницах;
- наличие возможностей для несложной поддержки и расширения сайта.

При создании дизайна будущего сайта важно помнить, прежде всего,

о функциональности и тех задачах, что должен будет решать будущий сайт. Излишне вычурный дизайн или неудобное меню, сложность нахождения нужной информации на сайте может привести к тому, что пользователь уйдёт на другие сайты.

Конечным продуктом разработки дизайна является дизайн-макет: картинка, представляющая предполагаемый будущий внешний вид страниц сайта. Данная картинка является многослойной, где почти каждая деталь — отдельный слой, приложенный к другим слоям-картинкам, за счёт чего может легко выполняться доработка, замена, перекомпоновка и другие задачи. В зависимости от идеи и целей макет может включать фотографии, сложные коллажи, иллюстрации, текстовые слои, уникальные иконки. Для главной страницы и внутренних иногда рисуются отдельные макеты с дополнениями или изменениями в соответствии с тематикой страницы.

В **четвертом** разделе приведена реализация информационной системы.

Для разработки сайта был выбран язык программирования Python, фреймворк Flask. Python – это высокоуровневый язык программирования, который был разработан в конце 1980-х годов.

Преимущества Python:

- простой и читаемый код. Python предлагает понятный синтаксис, что делает его привлекательным для опытных разработчиков и доступным для новичков;
- большое число полезных библиотек и модулей для Python позволяет быстро и легко решать различные задачи, такие как обработка данных, машинное обучение, работа с базами данных;
- язык подходит для большинства операционных систем. Код, написанный при помощи Python, может быть запущен на популярных ОС: Windows, macOS, Linux;
- python позволяет легко интегрировать код на других языках, таких как C++ и Java. Это позволяет использовать уже существующий код и библиотеки на этих языках, чтобы расширять функциональность Python;
- активное сообщество разработчиков помогающее и поддерживающее новичков. Это значит, что всегда можно получить ответы на возни-

кающие вопросы или найти готовый код для решения своих задач.

Преимущества работы на фреймворке Flask:

- простота и минимализм. Flask предоставляет основной набор инструментов для веб-разработки, делая его идеальным для начинающих;
- гибкость и расширяемость. Возможность легко интегрировать с другими библиотеками и сервисами;
- подход "микро-фреймворк". Flask позволяет разработчикам сохранять контроль над своим приложением, выбирая только необходимые инструменты и библиотеки;
- большое сообщество и поддержка. Благодаря своей популярности, Flask имеет обширное сообщество и множество доступных ресурсов для обучения и поддержки.

Сайт реализован с использованием трехслойной архитектуры. Трех-уровневая архитектура приложений - это модульная клиент-серверная архитектура, которая состоит из уровня представления, уровня приложения и уровня данных. Уровень данных обеспечивает хранение информации, уровень приложений обрабатывает логику, а уровень представления является собой графический интерфейс пользователя (GUI), который взаимодействует с двумя другими уровнями. Эти три уровня являются логическими, а не физическими, и могут работать как на одном физическом сервере, так и на разных машинах.

Уровень представления - уровень, созданный с использованием HTML5, JavaScript и каскадных таблиц стилей (CSS), развертывается на вычислительном устройстве через веб-браузер или веб-приложение. Уровень представления связывается с другими уровнями посредством вызовов интерфейса прикладных программ (API).

Уровень приложения - уровень приложения, который также можно назвать логическим уровнем, написан на языке программирования, таком как Java, Python или Ruby, и содержит бизнес-логику, которая поддерживает основные функции приложения. Базовый уровень приложений может быть размещен на распределенных серверах в облаке или на выделенном внутреннем сервере, в зависимости от того, сколько вычислительной мощности требуется приложению.

Уровень данных - уровень данных состоит из базы данных и программы для управления доступом для чтения и записи в базе данных. Этот уровень также может называться уровнем хранения и может быть размещен локально или в облаке. Популярные системы баз данных для управления доступом для чтения / записи включают MySQL, Oracle, PostgreSQL, Microsoft SQL Server и MongoDB.

Преимущества использования 3-уровневой архитектуры включают улучшенную масштабируемость, производительность и доступность. При использовании подхода к разработке приложений с тремя уровнями или частями, все эти части могут разрабатываться одновременно несколькими командами программистов, кодирующих на разных языках, при этом каждая из команд не зависит от других разработчиков, которые занимаются созданием другого уровня. Поскольку процесс создания программного кода для каждого уровня может претерпевать изменения, не затрагивая другие уровни, 3-уровневая модель облегчает непрерывное развитие приложения для предприятия или программного пакета по мере появления новых потребностей и возможностей. Существующие приложения или критические части могут быть постоянно или временно сохранены и инкапсулированы в новый уровень, компонентом которого они становятся.

Работа фреймворка Flask в проекте происходит через установленное виртуальное окружение. Именно оно задает определенную структуру проекта, а именно:

- все файлы html должны храниться в папке «templates», так как установленное окружение считывает файлы этого формата только из указанной папки;
- все «статичные файлы» (js, css, img) хранятся в папке «static» (их чтение может производиться только из нее);
- запуск приложения регулирует файл с содержанием backend-структуры (app.py).

В заключении приведены результаты бакалаврской работы.

Основные результаты

1. Изучены возможные варианты обучения моделей для наиболее точного предсказания цены акций.
2. Написан программный код для разного вида обучения моделей.
3. Написано web-приложение для визуального представления работы кода и взаимодействия пользователя с программой.
4. Проанализирована работа и точность предсказания для каждой из моделей и отображены рекомендации на сайте.
5. Проведена оценка результатов работы и введены разделы на сайте для взаимодействия с пользователем.