

Торговый алгоритм и сервис по построению торговых решений на основе данных **AlgoPack**

Инновационные решения для инвесторов — эффективный анализ данных, уверенные решения, успешные результаты

α AlfaAdept





Для себя



Задача:

Разработка торгового алгоритма на основе данных [AlgoPack](#)



Идея:

[Алгоритм](#) для теханализа, который [используя индикаторы](#), [находит точки входа](#) различных индикаторов, [анализирует](#) сработал ли тот или иной индикатор, в течении какого времени он давал [прибыль](#), в каком [проценте](#) случаев индикатор сработал и какой у него [потенциал роста](#)



Реализация:

Продукт использует **moexalgo** для получения исторических данных с таймфреймом в 1 день

Используя **pandas_ta**, рассчитываются индикаторы:

RSI, WILLR, Awesome Oscillator, AROON

Затем **две модели обучаются** на данных:

- Линейная регрессия для предсказания цены через i дней
- Логистическая регрессия для предсказания вероятности роста цены через i дней

Модели обучаются на данных за все дни, кроме последних 30

Далее, **для текущего дня**, на основе модели **предсказывается**:

- вероятность **роста**
- процент **изменения** цены
- **количество дней** для реализации изменения цены

Если вероятность роста больше 0,7, совершается **заявка на покупку**, а при достижении цены предсказанного значения совершается **продажа**. **Если** за время предсказанное для выполнения прогноза **цена не доходит** до нужного уровня, **сделка закрывается по текущей цене**



Backtest:

Алгоритм **тестировался на исторических данных** с:

- имитацией трейдинга
- установленным бюджетом
- комиссией брокера и указанием стопов

Подготавливаются данные, **инициализируется** стратегия, создается **бэктест** с помощью библиотеки **backtesting**,
а затем **анализируются результаты**, такие как прибыль, убытки и количество сделок.

Далее происходит **оптимизация** параметров **стратегии** для достижения лучших **результатов** и возможное **расширение** на другие активы



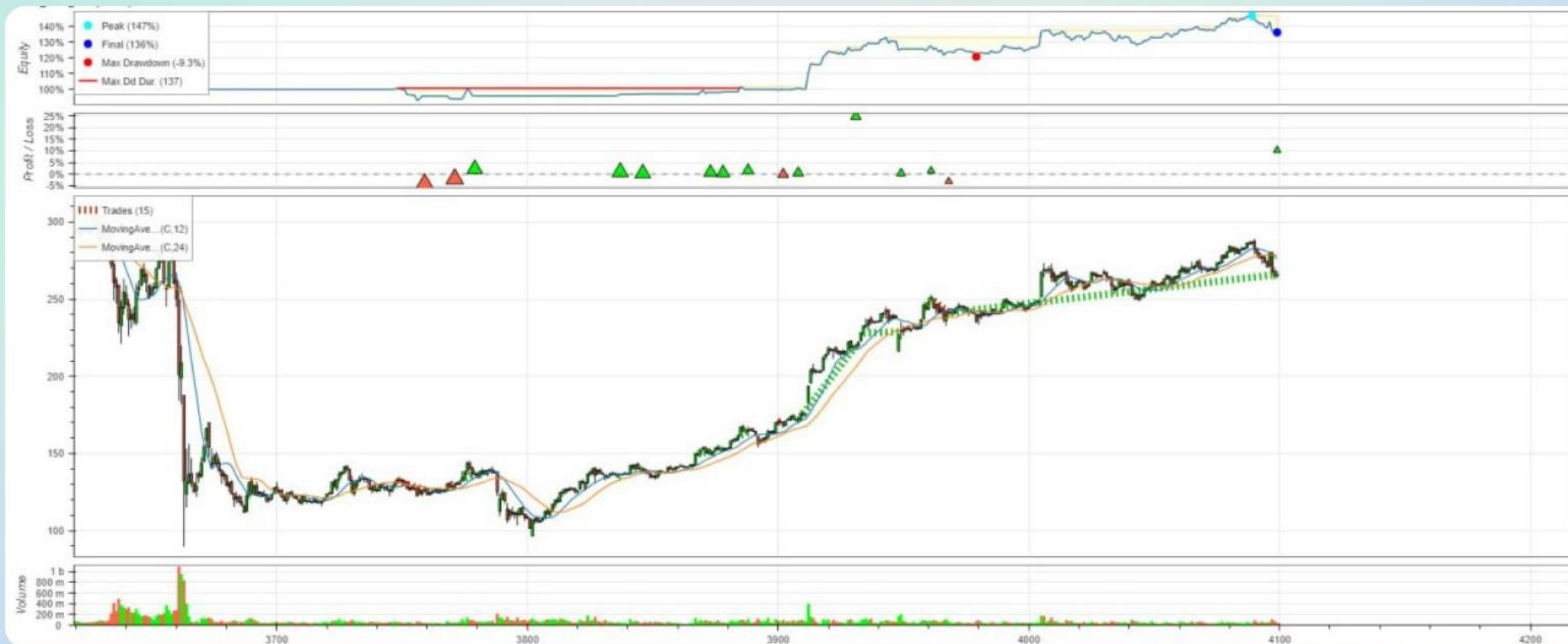


Результат:

Начальный капитал увеличился на 36.2%

Более чем 70% сделок были прибыльными

Общая прибыль была в 4.6 раза больше, чем общий убыток



Для инвесторов



Задача:

Разработка полезного сервиса для инвесторов на основе данных **AlgoPack**



Идея:

Сервис или приложение, которое в простом и удобном формате **выдает информацию о рисках вложения в акции**, на основе исторических данных



Аудитория:

Платежеспособная аудитория от 25 до 60 (мужчины и женщины)

Занятые люди, которым некогда погружаться в детали, но хотелось бы **иметь подсказку: как начать инвестировать?**





Описание:

Используя Наш сервис Пользователь **сможет видеть** информацию о том, **какой доход** наиболее вероятно он получит вложившись в ту или иную ценную бумагу и **какие будет иметь риски** при этом

Анализировать риски и доход **будет машинный алгоритм** на основе исторических данных индикаторов и торговых свеч



Польза и перспективность идеи для инвесторов:

Сервис разработан для людей, которые **не хотят разбираться во всех тонкостях трейдинга** — индикаторах и прочем, но хотят начать инвестировать, для этого **им достаточно в удобном формате получать данные** обо всех показателях риска и доходности



Реализация:

Разработан **сайт**, который **выдаёт информацию о сигналах** на покупку и продажу **с рисками** вложения в акцию, **анализируя исторические данные**.

Пользователь **может видеть информацию** о том, **какая вероятность роста или падения** акции **для** конкретного **временного отрезка**, на сколько дней стоит вкладываться и какие могут быть риски при этом





Техническая реализация:

Написано на JavaScript на фреймворке NextJS с использованием MaterialUI, Farmer Motion, MaterialCharts

Архитектура

2х-слойная клиент + сервер не зависящие друг от друга.

Dataflow

Клиент выбирает входящие данные - тикер и индикатор -> при нажатии на кнопку -> клиент обращается к серверу за данными:

1. Цена акции на протяжении 30 дней
2. Сигналы о продаже
3. Сигналы о покупке

-> после того как сервер собирает и обобщает необходимые данные -> клиент разбивает их на компоненты -> клиент определяет необходимый интерфейс для каждого компонента и отображает полученные данные

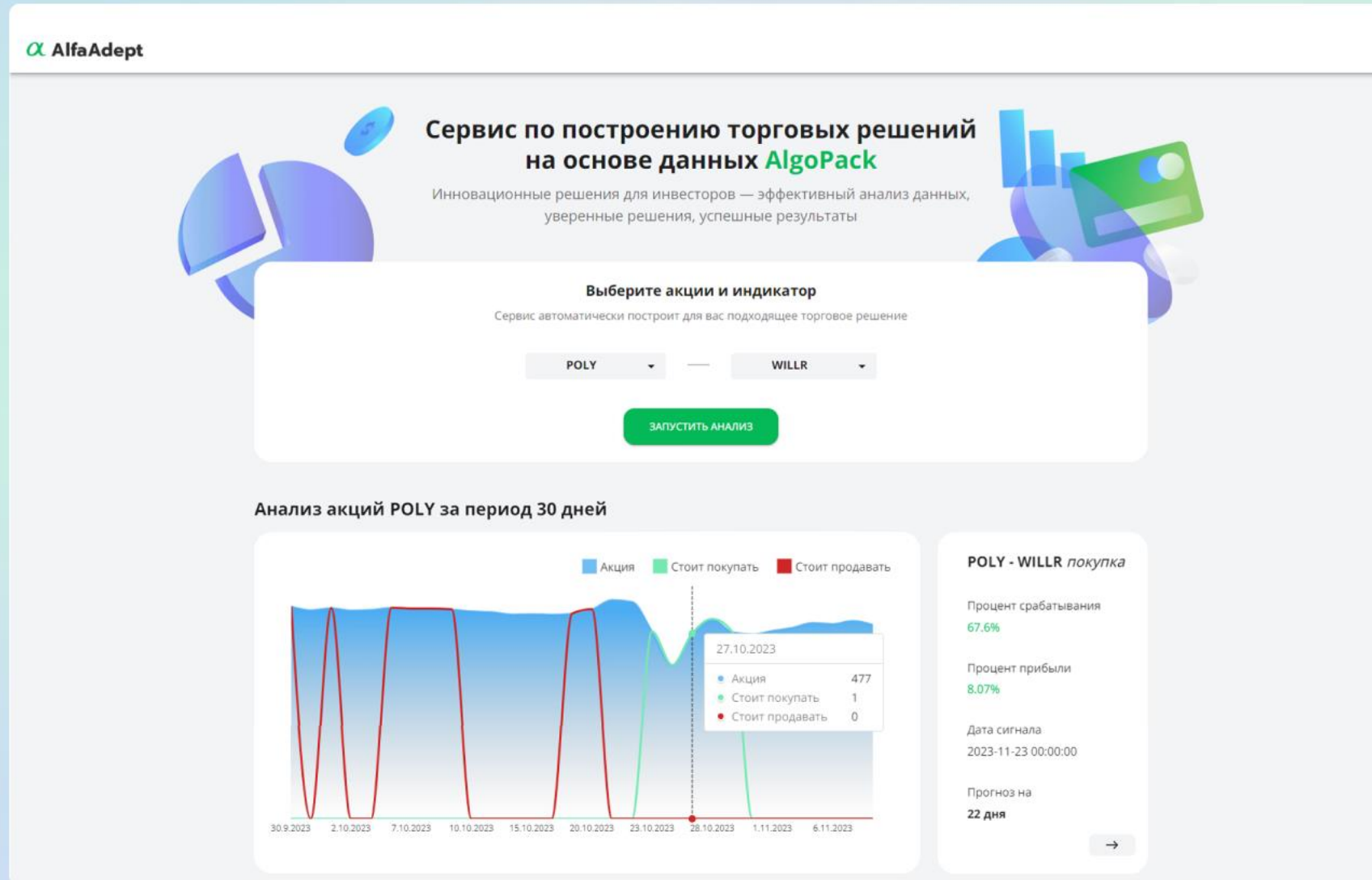


Результаты:



<https://trading-helper-seven.vercel.app>

Рекомендовано открывать с экрана 1920x1080



Над проектом работали:



Буторина Полина
@appa_bu

Бизнес-аналитик
Генерация идеи для сервиса



Лашкевич Даннил
@Daniil_5t4Rt

Обработка данных,
ML-специалист
Работа с алгоритмом
на pytorch, изучение
библиотеки torchalgo,
backtest алгоритма



Сафонов Артём
@artjom_safonoff

Fullstack разработчик
Реализация frontend,
backend сервиса
и их связывание



Исинбаев Артур
@Normalmd

backend-разработчик
Идея алгоритма,
реализация алгоритма
на python,
backend сервиса



Новокшенова Дарья
@nova_ds

Дизайнер
Разработка дизайна

С уважением команда



α AlfaAdept

Спасибо за внимание!

Будем благодарны за обратную связь на адрес:
appolinarija-butorina@yandex.ru