УДК 004

РАЗРАБОТКА ВЕБ-СЕРВИСА ДЛЯ АРБИТРАЖА КРИПТОВАЛЮТНЫХ АКТИВОВ И КОНТРАКТОВ

Канунников Б.С., студент гр. ПИб-212, IV курс,
Савельев С.А., студент гр. ПИб-212, IV курс,
Крутский Д.Л., старший преподаватель

Кузбасский государственный технический университет

имени Т.Ф. Горбачева
г. Кемерово

В современном мире рынок криптовалютных активов развивается стремительными темпами и становится неотъемлемой частью глобальной финансовой системы. С каждым годом количество бирж, предлагающих торговлю цифровыми активами, увеличивается, а объемы торгов достигают рекордных значений. Вместе с ростом интереса к криптовалютам возрастает и потребность в инструментах, позволяющих эффективно анализировать рынок, выявлять выгодные торговые возможности и минимизировать риски при проведении сделок.

Одна из ключевых стратегий, применяемых профессиональными трейдерами и инвесторами в криптовалютной индустрии, – это арбитраж между различными торговыми площадками. Арбитраж (криптовалют) — это процесс покупки и продажи криптовалюты на разных биржах с целью извлечения прибыли из разницы в ценах. Однако, в условиях высокой волатильности криптовалютных рынков и большого количества доступных торговых пар, поиск и анализ подобных возможностей вручную становится невыполнимой задачей.

Именно для этого решения нами был разработан веб-сервис, предназначенный для мониторинга и анализа цен криптовалютных активов и бессрочных фьючерсных контрактов на популярных биржах. Система динамически собирает актуальные данные с различных торговых площадок, предоставляя пользователю удобный инструмент для сравнения цен, а также оценки рыночной ситуации в режиме реального времени.

Данные отображаются в формате интерактивной таблицы с возможностью фильтрации и сортировки по ключевым параметрам, таким как цена актива, объем торгов, биржа и другие характеристики. Это позволяет пользователям оперативно выявлять расхождения в стоимости активов между биржами и находить потенциальные возможности торговли.

Помимо анализа цен, веб-сервис предоставляет оценку ликвидности активов на основе суточного объема торгов и разностью между лучшими ценами

заявок на продажу и покупку в один и тот же момент времени. Пользователь может задавать собственные критерии оценки ликвидности, по которым система автоматически присваивает активам соответствующую категорию.

Для рынка бессрочных фьючерсных контрактов дополнительно отображается информация о текущих ставках финансирования на всех поддерживаемых биржах. Эти данные позволяют пользователю учитывать расходы или потенциальную прибыль при удержании позиций по контрактам, а также оценивать степень отклонения цены контракта от базового актива.

Такой подход обеспечивает комплексный анализ спотового и фьючерсного рынков, предоставляя пользователям инструменты для эффективного принятия решений в рамках арбитражных стратегий.

Для разработки нашего сервиса используется язык программирования Python [1]. Он выбран по причине того, что позволяет быстро и эффективно создавать веб-приложения, а также предоставляет широкие возможности по работе с данными, сетевыми запросами и API различных бирж.

В качестве основного веб-фреймворка выбран Django. Это современный и мощный инструмент, который обеспечивает высокую скорость разработки, безопасность и масштабируемость. Django [2] предоставляет встроенные средства для работы с базами данных, авторизацией пользователей, а также упрощает реализацию REST API, а это жизненно важный аспект для созданного нами проекта, по причине активного обмена данными между фронтендом и сервером.

Для сбора информации с бирж в нашем сервисе мы используем отдельные классы, которые, с помощью API запросов, берут информацию с бирж. Реализация классов представлена на рисунке 1.

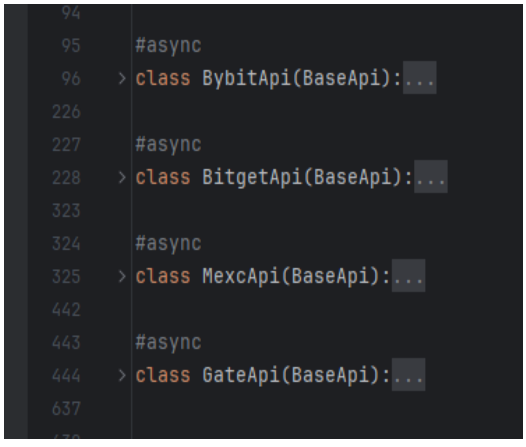


Рисунок 1

После сбора данных с различных бирж, они поступают в единый обработчик данных — это отдельный класс, отвечающий за фильтрацию и предварительную обработку информации.

Основные функции обработчика:

- Фильтрация полученных данных по необходимым параметрам.
- Очистка и нормализация данных, проверка на актуальность и целостность.
- Агрегация данных при необходимости (например, расчет среднего «спреда» по биржам).

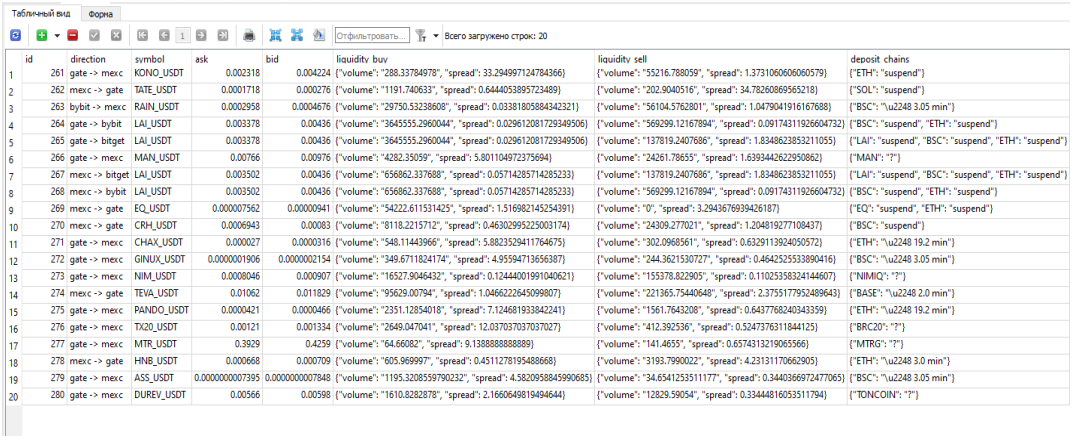
Частичная реализация класса представлена на рисунке 2.

```python
#return final df between all exchanges, sorted by percent_diff and contain a certain number of records
def get_total_data(df_list):  1 usage
    df_records = 20
    spot_total_df  = pd.DataFrame()
    futures_total_df = pd.DataFrame()

    for first, second in combinations(df_list, r: 2):
        spot_total_df = pd.concat([spot_total_df, compare_exchange_data(first, second)])
        futures_total_df = pd.concat([futures_total_df, compare_exchange_future_data(first, second)])

    spot_total_df = spot_total_df[['direction', 'symbol', 'ask', 'bid', 'percent_diff','withdraw_chains', 'deposit_chains', 'liquidity_buy', 'liquidity_sell']]
    futures_total_df = futures_total_df[['direction', 'symbol', 'ask', 'bid', 'percent_diff', 'funding_buy', 'funding_sell', 'liquidity_buy', 'liquidity_sell']]

    spot_total_df = spot_total_df.sort_values(by='percent_diff', ascending=False).head(df_records)
    futures_total_df = futures_total_df.sort_values(by='percent_diff', ascending=False).head(df_records)

    return {
        'spot': spot_total_df,
        'futures': futures_total_df
```

Рисунок 2

После сбора и предварительной обработки вся актуальная информация о торгах, ценах, объемах и ликвидности криптовалютных активов сохраняется в базу данных для дальнейшего использования.

В качестве системы управления базами данных (СУБД) используется SQLite. Это легковесная, встроенная реляционная СУБД, которая отлично интегрируется с Python и Django, обеспечивая надежное и эффективное хранение больших объемов информации при минимальных затратах на настройку и обслуживание.



Табличный вид | Форма

Отфильтровать... | Всего загружено строк: 20

	id	direction	symbol	ask	bid	liquidity buy	liquidity sell	deposit chains
1	261	gate -> mexc	KONO_USDT	0.002318	0.004224	{"volume": "288.33784978", "spread": 33.294997124784366}	{"volume": "55216.788059", "spread": 1.3731060606060579}	{"ETH": "suspend"}
2	262	mexc -> gate	TATE_USDT	0.0001718	0.000276	{"volume": "1191.740633", "spread": 0.6444053895723489}	{"volume": "202.9040516", "spread": 34.78260869565218}	{"SOL": "suspend"}
3	263	bybit -> mexc	RAIN_USDT	0.0002958	0.0004676	{"volume": "29750.53238608", "spread": 0.03381805884342321}	{"volume": "56104.5762801", "spread": 1.0479041916167688}	{"BSC": "\u2248 3.05 min"}
4	264	gate -> bybit	LAI_USDT	0.003378	0.00436	{"volume": "3645555.2960044", "spread": 0.029612081729349506}	{"volume": "569299.12167894", "spread": 0.09174311926604732}	{"BSC": "suspend", "ETH": "suspend"}
5	265	gate -> bitget	LAI_USDT	0.003378	0.00436	{"volume": "3645555.2960044", "spread": 0.029612081729349506}	{"volume": "137819.2407686", "spread": 1.8348623853211055}	{"LAI": "suspend", "BSC": "suspend", "ETH": "suspend"}
6	266	gate -> mexc	MAN_USDT	0.00766	0.00976	{"volume": "4282.35059", "spread": 5.801104972375694}	{"volume": "24261.78655", "spread": 1.6393442622950862}	{"MAN": "?"}
7	267	mexc -> bitget	LAI_USDT	0.003502	0.00436	{"volume": "656862.337688", "spread": 0.05714285714285233}	{"volume": "137819.2407686", "spread": 1.8348623853211055}	{"LAI": "suspend", "BSC": "suspend", "ETH": "suspend"}
8	268	mexc -> bybit	LAI_USDT	0.003502	0.00436	{"volume": "656862.337688", "spread": 0.05714285714285233}	{"volume": "569299.12167894", "spread": 0.09174311926604732}	{"BSC": "suspend", "ETH": "suspend"}
9	269	mexc -> gate	EQ_USDT	0.000007562	0.00000941	{"volume": "54222.611531425", "spread": 1.5169821452543911}	{"volume": "0", "spread": 3.2943676939426187}	{"EQ": "suspend", "ETH": "suspend"}
10	270	mexc -> gate	CRH_USDT	0.0006943	0.00083	{"volume": "8118.2215712", "spread": 0.46302995225003174}	{"volume": "24309.277021", "spread": 1.204819277108437}	{"BSC": "suspend"}
11	271	gate -> mexc	CHAX_USDT	0.000027	0.0000316	{"volume": "548.11443966", "spread": 5.882352941176475}	{"volume": "302.0968561", "spread": 0.6329113924050572}	{"ETH": "\u2248 19.2 min"}
12	272	gate -> mexc	GINUX_USDT	0.0000001906	0.0000002154	{"volume": "349.6711824174", "spread": 4.955947136563873}	{"volume": "244.3621530727", "spread": 0.4642525533890416}	{"BSC": "\u2248 3.05 min"}
13	273	gate -> mexc	NIM_USDT	0.0008046	0.000907	{"volume": "16527.9046432", "spread": 0.12444001991040621}	{"volume": "155378.822905", "spread": 0.11025358324144607}	{"NIMIQ": "?"}
14	274	mexc -> gate	TEVA_USDT	0.01062	0.011829	{"volume": "95629.00794", "spread": 1.0466222645099807}	{"volume": "221365.75440648", "spread": 2.3755177952489643}	{"BASE": "\u2248 2.0 min"}
15	275	gate -> mexc	PANDO_USDT	0.0000421	0.0000466	{"volume": "2351.12854018", "spread": 7.124681933842241}	{"volume": "1561.7643208", "spread": 0.6437768240343359}	{"ETH": "\u2248 19.2 min"}
16	276	gate -> mexc	TX20_USDT	0.00121	0.001334	{"volume": "2649.047041", "spread": 12.037037037037027}	{"volume": "412.392536", "spread": 0.5247376311844125}	{"BRC20": "?"}
17	277	gate -> mexc	MTR_USDT	0.3929	0.4259	{"volume": "64.66082", "spread": 9.138888888889}	{"volume": "141.4655", "spread": 0.6574313219065566}	{"MTRG": "?"}
18	278	mexc -> gate	HNB_USDT	0.000668	0.000709	{"volume": "605.969997", "spread": 0.4511278195488668}	{"volume": "3193.7990022", "spread": 4.23131170662905}	{"ETH": "\u2248 3.0 min"}
19	279	gate -> mexc	ASS_USDT	0.0000000007395	0.0000000007848	{"volume": "1195.3208559790232", "spread": 4.582095884590685}	{"volume": "34.65412535511177", "spread": 0.3440366972477065}	{"BSC": "\u2248 3.05 min"}
20	280	gate -> mexc	DUREV_USDT	0.00566	0.00598	{"volume": "1610.8282878", "spread": 2.1660649819494644}	{"volume": "12829.59054", "spread": 0.3344481605351794}	{"TONCOIN": "?"}

Рисунок 3

Для асинхронной обработки задач, таких как регулярный сбор данных с API бирж, используется Celery [3] в связке с брокером сообщений Redis. Это решение позволяет нашему сервису обеспечивать актуальность данных, обновления происходят каждую минуту.

После сбора, фильтрации и обработки данные предоставляются пользователю через удобный и интуитивно понятный веб-интерфейс в виде таблиц. Таблицы содержат актуальную информацию о возможностях арбитражных сделок, ценах, ликвидности и доступных способах перевода активов между биржами. На рисунке 4 приведен пример отображения данных пользователю в виде таблицы, визуальное оформление находится в доработке, но примерно это будет выглядеть так.

Категории ликвидности

	Низкая	Средняя	Высокая
Объём:	1000	10000	50000
Спред:	1.5	1.0	0.5

ID	Направление	Пара	Ask / Bid	Разница, %	Ликвидность (покупка)	Ликвидность (продажа)	Вывод	Ввод
261	gate -> mexc	KONO_USDT	0.002318 / 0.004224	82.2260569456428	Средняя	Низкая	Сеть: ETH; Стоимость: ~ 4.60834626 usd	Сеть: ETH; Время: suspend
262	mexc -> gate	TATE_USDT	0.0001718 / 0.000276	60.6519208383926	—	—	Сеть: SOL; Стоимость: ~ 1.49879 usd	Сеть: SOL; Время: suspend
263	bybit -> mexc	RAIN_USDT	0.0002958 / 0.0004676	58.079783637592975	—	—	Сеть: BSC; Стоимость: suspend; Сеть: ETH; Стоимость: ~ 3.5496 usd	Сеть: BSC; Время: ~ 3.05 min
264	gate -> bybit	LAI_USDT	0.003378 / 0.00436	29.07045589059805	—	—	Сеть: LAI; Стоимость: ~ 0.48872904 usd; Сеть: BSC; Стоимость: suspend; Сеть: ETH; Стоимость: suspend	Сеть: BSC; Время: suspend; Сеть: ETH; Время: suspend

Рисунок 4

В результате проделанной работы был создан веб-сервис для мониторинга и анализа цен криптовалютных активов на различных биржах. Система автоматически собирает, обрабатывает и хранит актуальные данные, предоставляя пользователям удобные инструменты для выявления арбитражных возможностей. Использование Python, Django и SQLite позволило добиться высокой производительности и удобства работы с данными. Данный сервис значительно упростит процесс анализа рынка и поможет трейдерам принимать более обоснованные торговые решения.

Список литературы:

1. Биссек, П. Django. Разработка веб-приложений на Python / - Пер. с англ. - Санкт-Петербург «СимволПлюс», 2010. – 456 с.
2. Дронов, В.А. Django. Практика создания веб-сайтов на Python / Санкт-Петербург «БХВ-Петербург», 2019. – 672 с.
3. Материалы сайта «Celery для новичков» [Электронный ресурс]. – Режим доступа: https://habr.com/ru/companies/otus/articles/796413// (дата обращения 21.03.2025)