А у меня следующая корректировка. Распишу ее более подробно, чтобы был виден большой отрезок данной дороги, а не только до самого ближайашего поворота )). На карте мне хотелось бы иметь фильтры по группам, чтобы можно было оставлять на ней только объекты выбранных групп (для каждого из двух наборов групп свои фильтры). Далее уже оставшиеся метки объектов выбранных групп выделять цветом в зависимости от сентимента (помним, что у нас могут быть разные способы обработки данных, например, разные LLM, разные классические методы, также разные оценки пользователей), за базовые значения которого берем {"хорошо", "плохо", "удовлетворительно"}. Начальные оценки пользователей ("rating") после подгрузки файла преобразуем (базовый вариант) следующим образом: 5-"хорошо"; 3,4-"удовлетворительно"; 1,2-"плохо" и эти настройки нужно разместить где-нибудь в коде где мы определяем начальные данные и установки, чтобы можно было при необходимости сразу изменить. Итак, в идеале хорошо бы иметь диаграмму или таблицу, где по горизонтали откладываются записи, а по вертикали "различные указанные выше способы обработки исходных данных" с получением сентимента ("плохо"-оранжевый, "хорошо"-зеленый и "удовлетворительно"- желтый) таким образом для каждого выбранного набора записей (у нас это в сумме называется "архив") и способов обработки мы получим наглядную картинку разницы в обработке данных различными способами). Далее, по поводу архитектуры данных: по-хорошему у нас должен быть основная таблица с данными и уникальными хэшами, а также вторая таблица, связанная по хэшам отношением один-ко\_многим. причем на стороне "многие" у нас для каждого хэша будет находиться определенное количество сделанных обработок основной записи различными описанными выше методами. Архивный csv файл нужно в этом случае подумать как лучше сделать, но при этом нам нужно изменить загрузку на входе csv-файлов и мы должны здесь загружать именно наши сохраненные архивные csv-данные - других csv нам подгружать не нужно. Так при желании мы сможем накапливать и не терять результаты обработки исходных данных. Причем мы помним, что если у нас заполнены поля координат, то они вторично не определяются, равно как и название округа из координат не определяется, если оно уже заполнено.

**Этап 3: Визуализация сравнения методов**

**Задачи:**

1. **Таблица/диаграмма сравнения методов** - записи по горизонтали, методы по вертикали
2. **Цветовая схема** - зеленый/желтый/оранжевый для сентимента
3. **Унификация шкалы оценок** - приведение к единой шкале

**Унификация шкалы оценок:**

**Сентимент:** "положительный", "нейтральный", "отрицательный"

**Рейтинг:** "хорошо", "нейтрально", "плохо" → приведем к шкале сентимента

Нужно немного подкорректировать фронтенд: из-за добавления фильтров и другой информации сверху над картой в блок "карта объектов" сам подблок карты потерял заданное первоначально соотношение сторон (примерно лист А4 в режиме книга). Отодвигам левую границу блока "результаты анализа" в ущерб "карте" на необходимую ширину после вставки поля «determined\_group» и переименовываем этот заголовок в «опред\_групп». Поле «опред\_групп» лучше всего вставить справа от поля «Название». Далее Легенду на карте переносим из положения "над картой" в положение "справа от карты". Еще у нас есть две неиспользуемые кнопки: «архив» и «новые» - это уже атавизмы и предлагаю их убрать.

Поле «**determined\_group**» мы определяем автоматически по наличию ключевых слов групп объектов (проверяем сначала поле **review\_text**, а потом **name**) по принципу: если поле «**determined\_group**» пустое, то автоматически определяем, а если значение есть, то оставляем его без изменения.

Напомню, что для определения поля **«группа»** у нас правило: если поле пустое, то вызываем модальное окно для выбора из списка, а если значение есть, то тоже оставляем его без изменения.

Все кнопки (кроме фильтров) для карты предлагаю сделать в едином цветовом стиле – как для кнопки «По сентименту»: нажатая кнопка темным цветом с белым текстом, а неактивная кнопка с темнозелеными рамкой и текстом, а заполнение лучше сделать не без цвета, а слабым оттенком темнозеленого. И еще нужно в блоке оставить место для выбора кнопок «По сентименту», когда у нас появляются дополнительные кнопки, чтобы объекты в блоке не дергались при вставке новых кнопок.

Также правильным будет отформатировать в блоке «Карта объектов» левые края кнопок по одной линии.

Теперь еще по сути: у нас «Методы анализа» находятся в блоке «Загрузка файла», а по идее это все нужно перенести в «Результаты анализа» - там должен быть и выбор методов и сами результаты, а также там нужно добавить «индикатор выполнения», который принимает эстафету после того как данные загрузились в приложение. Если мы это не сделаем, то все, что связано с анализом данных происходит справа от блока карты, а индикатор обработки данных находится совершенно в другом месте слева от блока карты.

**Логика определения группы:**

Теперь система автоматически определяет группу из содержимого данных:

* **Название объекта** (поле name)
* **Текст отзыва** (поле review\_text)

Предлагаю для автоматического определения группы добавить следующие ключевые слова:

university: семинар, лекция, преподаватель;

kindergarden: дошкольное;

hospital:палата, операционная;

shopmall: ТЦ, молл, гипермаркет

Этап 3: Визуализация сравнения методов  
Диаграмма/таблица сравнения методов обработки  
Цветовая схема по сентименту  
По записям и методам обработки

Этап 4: Архитектура данных  
Основная таблица с уникальными хэшами  
Связанная таблица обработок (один-ко-многим)  
Загрузка только архивных CSV  
Этап 5: Умная обработка координат  
Не переопределять существующие координаты  
Не переопределять существующие округа  
Предлагаю из этого плана сейчас к сравнению методов - таблица или диаграмма с записями по горизонтали и методами по вертикали  
Цветовая схема - зеленый/желтый/оранжевый для отображения сентимента. Здесь еще должен заметить, что правильно было бы привести к единой шкале оценку отзывов - сентимент у нас определяется как ("положительный", "нейтральный", "хороший" - поправь меня если я ошибаюсь), а для рейтинга в целях сравнения вчера я предлагал ("хорошо", "нейтрально", "плохо") - предлагаю рейтинг измерять по шкале, идентичной сентименту, чтобы у нас была согласованность в оценках. Далее, у нас по архитектуре в будущих исправлениях запланированы 2 таблицы со связью "один ко многим", так вот для оценок пользователей предлагаю на стороне "один" так и оставить поле с оценкой отзыва "rating", в вот в таблице на стороне "многие" уже для случая "оценка пользовтеля" для сравнения с обработкой отзывов разными методами и разместить как итог преобразованную оценку "rating" по шкале сентимента

в части "тепловой карты", и "диаграмма" функционирует. Теперь что хотелось бы изменить в "тепловой карте": вместо "Сентимент-анализ" логичнее было бы написать "Сентимент"+метод, ведь у нас здесь возможно появление нескольких различных методов определения сентимента, далее - 10 записей мало в этой тепловой карте, нужно чтобы можно было посмотреть все записи с прокруткой и, наверное, будет в этом случае целесообразно название объекта размещать только на одной строке, хотя в нее все название может и не войти. Теперь по поводу "диаграммы" - я, конечно, извиняюсь, но это не только не по заданию, но и удивительно неинформативно и неудобно получилось. Цель диаграммы в нашем случае не вчитываться в буквы и значки, а с одного взгляда (на то она и диаграмма) увидеть общую картину и понять не только сколько в объектах зеленого, желтого и оранжевого цветов, но и как их выдают различные методы - вот только названия этих методов и должны быть доступны для прочтения, а остальное конкретно какая запись каким методом как была определена - не интересно, только общая картина. В этом случае мне видится, подчеркиваю еще раз, примерно следующая диаграмма: по горизонтальной оси должны располагаться отзывы (вдоль оси можно написать лишь обозначение "отзывы" и никаких номеров или обозначений отзывов - это не нужно, можно на оси только деления разместить) и если у нас размер бара отзыва в ширину будет на экране 0,5мм, это устроит, а в высоту его размер будет таким, чтобы слева на вертикальной оси можно было написать название метода и высота такого бара вполне может быть 5-10мм, в этом случае на одной диаграмме мы легко сможем увидеть информацию по 200-300 записям. Заметным улучшением восприятия диаграммы стала бы еще сортировка, чтобы одинаковые цвета располагались рядом: сначала на нижнем ярусе диаграммы, затем в пределах имеющейся сортировки нижнего слоя на втором слое в пределах одноцветных зон нижнего слоя и так далее вверх по всем имеющимся уровням. Могу, если сможешь преобразовать, даже скинуть прототип такой диаграммы как это будет смотреться, правда формат такой диаграммы может быть для Тебя не информативным

22.7

Предложение по изменению архитектуры: сейчас у нас есть одна таблица (архив), где содержатся как исходные, так и добавленные данные. Если посмотреть внимательно, то увидим, что здесь напрашивается разбиение данных на 5 таблиц: первая таблица (Объекты) содержит данные по объектам и для одного объекта у нас в данных от одного до нескольких десятков отзывов. Эти таблицы мы можем связать как один (объект)-ко многим (отзывы). В Отзывах разместим имеющиеся поля - все, что с ними связано). В качестве ключа я бы взял суммарное поле: "имя"+"адрес" - эта комбинация, насколько я понимаю имеет признаки уникальности по отношению к объектам. Также таблица Объекты должна быть связана отношением один-ко многим с двумя таблицами: Группа объектов (многие) и ключом для связи со стороной "многие" является целое число - номер группы. Вторая таблица "Определяемые\_группы" (многие) имеет аналогичную связь один-ко многим тоже с таблицей Объекты по второму аналогичному ключу. В свою очередь таблица "Отзывы" (один) должна связываться как один-ко многим с таблицей, в которой содержатся результаты обработки отзывов различными методами. Я бы назвал ее Сентименты, хотя там данные не только для них. В этой таблице, мне кажется, целесообразно разместить следующие поля (ключ - целое число, определяющее "метод обработки" или предложи свой вариант ключа; сентимент - в случае метода обработки "оценка юзера" мы договорились, что у нас rating преобразуется как раз в множество трех значений сентимента, а в Отзывах останется исходный столбец "rating" 1...5; тип отзыва (благодарность, жалоба, предложение, информационный); + необходимые технические поля если нужно. И у нас еще образовалась шестая таблица методы\_обработки - {"оценка\_юзера", "NLP\_VADER", "LLM\_Yandex", ..., } и эта таблица связывается как один (Сентименты)-ко многим (методы обработки). Просьба дать критические замечания по предлагаемой возможной замене архитектуры. При этом БД на Твое усмотрение, хоть SQLite. Также предполагается, что при появлении БД никакие архивы мы стирать не будем и у нас будет отображение по кнопкам (входные/архив) визуальных отображений. Также нам нужно будет преобразовывать входные файлы с разнесением по таблицам, а когда выводим, например, обработанную входную таблицу, то собирать ее нужно будет через запрос к таблицам и т.д.

23.7

**✅ Что мы сделали:**

**Ветка №3 - Архитектура БД и определение групп:**

1. **✅ SQLite база данных** с 6-табличной архитектурой
2. **✅ Автоматическое определение групп** в Excel и JSON процессорах
3. **✅ API эндпоинты БД** (/database/stats, /database/migrate, etc.)
4. **✅ Переключение групп на карте** (поставщик vs определяемые)
5. **✅ Миграция и экспорт данных**
6. **✅ Документация новой архитектуры**

**Коммиты:**

* ba95940 - feat: implement database architecture and group determination
* a72c8b1 - feat: add database file and sentiment summary

**Завтра:**

1. **Смержить ветку 2** с main
2. **Смержить ветку 3** с main
3. **Начать реализацию** нового плана:

* Загрузка файлов → БД
* Карта читает из БД
* Срезы данных
* Переключатель "Архив" / "БД"

**Завтра (развитие):**

1. **При загрузке файла** - сохранять в БД
2. **БД не очищать** - накапливать результаты обработки
3. **Карта читает из БД** с возможностью срезов
4. **CSV экспорт** - только выбранных срезов
5. **Двойное сохранение** - архив + БД (для отладки)
6. **Переключатель** "Архив" / "БД"

Предлагаю далее выбрать немного другой вариант: 1) При загрузке файла - сохранять в БД - согласен; 2) БД очищать не будем - там результаты обработки, в том числе и по апи; 3) Карта читает из БД, а срезами мы выделяем нужный для просмотра/анализа кусок данных (про дополнительные срезы - завтра); 4) импорт в CSV оставим, но только того, что мы выберем срезами (кнопками); 5) иногда для просмотра при отладке нужный кусок данных из БД, может быть, будет долго выбирать кнопками, поэтому пока новые данные будем добавлять и в архив, и в БД, а почистить можем только архив; 6) для отображения данных сделать кнопку ("архив"/"БД"). Но это все завтра, а сегодня надо сделать комит, причем сделаем сейчас ветку №3, т.к. принципиальное изменение архитектуры. Позже сначала ветку 2 смержим с main, а потом уже к ним смержим ветку 3

Следующее планируемое изменение: изменить работу с данными в архиве и БД. Сейчас у нас загружаемые на вход данные обрабатываются и, в случае если они не дублируют данные в архиве, загружаются в него. Архив при этом представляет собой файл в каталоге. Для визуализации сейчас также используются данные из файла «архив».

БД SQLite мы вчера создали и теперь нужно подключать ее в работу. Теперь новые данные после проверки на дублирование и последующей обработки загружаются в БД. Отображение всех данных на картах и различных диаграммах тоже предполагает использование данных БД. Все фильтры и т.д. также продолжают работать в обычном режиме вне зависимости от того, откуда данные поступили на отображение.

Архивный файл как отдельный источник данных не нужен. Однако нам нужна «Выборка» - это подмножество существующей БД, которое мы выбираем, допустим, для получения какого-то конкретного анализа или отображения. Нам необходимо сделать кнопку «создать Выборку» и в заранее добавленное поле «в\_Выборке» таблицы «Отзывы» размещаем информацию (например: «да») для всех записей, которые в данный момент выбраны нами имеющимися фильтрами из существующей у нас БД. Вместо кнопок «скачать архив» и «очистить архив» мы на этом месте располагаем, соответственно, кнопки «скачать выборку» и «очистить выборку». Кнопку «создать выборку» размещаем здесь же в блоке «Управление архивом», а сам этот блок переименовываем в «Управление выборкой». Поля при выборе «скачать выборку» создаются в том же виде, как это сделано сейчас для «архива» (csv) - здесь ничего не меняем.

Теперь по алгоритму обработки входящих данных:

* 1. Начинаем работу с таблицы «Объекты». Если Объект уже у нас есть, то проверяем Группу Объекта и, в случае несовпадения, данные по этому объекту в БД не добавляем, а в терминал выдаем сообщение «Группа объекта (здесь имя и адрес объекта) (здесь существующая Группа) не совпадает с Группой объекта из входного файла (здесь Группа из входа)». Если Объект и его Группа совпали с входными данными, то «координаты», «округ города» у нас уже должны быть и их снова не запрашиваем – эти поля у нас одинаковые для каждого Объекта и должны быть (надеюсь) в таблице «Объекты». Далее для каждой записи Объекта (если мы загружаем на входе json, то у него почти всегда для каждого Объекта более чем по одному отзыву) мы ищем «Определяемую группу» производя поиск по ключевым словам сначала в тексте Отзыва, потом в имени Объекта. Если Объект новый, то для него определяем все по списку, как говорится;
  2. Если для выбранного Объекта новый Отзыв уже у нас есть, то проверяем на совпадение «date», «user\_name», «rating» и, в случае полного совпадения, данные по этому Отзыву в БД не добавляем, а в терминал выдаем сообщение «Отзыв (здесь имя и адрес объекта) (здесь «user\_name», «date», первые 40 символов Отзыва) совпадает с Отзывом из входного файла. Если отзыв новый, то определяем выбранными нами методами Сентимент и т.д.;

У нас должна быть конечно же отдельная кнопка для переключения между источниками данных: Выборка/БД - именно с этой целью мы ведь и добавляем Выборку.