



ТЕХНИЧЕСКОЕ ЗАДАНИЕ

ЗАДАЧА 6

Сервис для отслеживания динамики клиентских настроений и проблем по конкретным банковским продуктам в режиме реального времени



1. Контекст и актуальность задачи

Актуальность сервиса обусловлена тем, что в современном банковском секторе успех напрямую зависит от качества клиентского опыта и способности быстро реагировать на меняющиеся потребности клиентов. Отслеживание динамики отзывов и их тональности поможет своевременно решать возникающие проблемы и улучшать сервис, повышая лояльность клиентов.

2. Описание задачи. Подробная задача для участников, зарегистрированных на кейс. Отображается после начала хакатона

2.1. Описание задачи

Разработайте сервис для менеджеров Газпромбанка, который позволит эффективно отслеживать сильные и слабые стороны ключевых банковских продуктов и услуг. **Для этого необходимо собрать текстовые отзывы о работе Газпромбанка с популярных агрегаторов отзывов (например, [sravni.ru](https://www.sravni.ru), [banki.ru](https://www.banki.ru) и др.).** Отзывы собираются за период 01.01.2024 – 31.05.2025. Допускается использование парсинга, открытых API или готовых выгрузок, при условии соблюдения условий использования источников.

Далее необходимо выполнить тематическую кластеризацию текстов с целью выявления основных групп обсуждаемых продуктов и услуг. Кластеризация должна быть выполнена с использованием подходов машинного обучения, тематического моделирования, эмбедингов или других методов. На выходе ожидается:

- разбиение отзывов на семантические обоснованные кластеры (например, кредитные карты, вклады, дистанционное обслуживание и др)
- название и описание каждого кластера

Полученные кластеры будут использоваться в следующем этапе для построения классификатора.

Затем по полученным кластерам необходимо составить одноуровневый классификатор, определяющий, к каким темам относится каждый отзыв. Поскольку один

отзыв может затрагивать несколько продуктов и услуг одновременно, классификатор обязан поддерживать мультитейбл предсказание. Участникам предлагается самостоятельно выбрать архитектуру решения: от классических до нейросетевых или гибридных подходов. На выходе ожидается:

- список всех релевантных тем (из числа ранее выделенных)
- соблюдение согласованности с темами, полученными на этапе кластеризации.

После определения тематик отзывов, необходимо определить эмоциональную окраску каждой тематики. Возможные значения: положительно/нейтрально/отрицательно. Важно: Тональность будет учитываться только для тех тем, которые определены правильно. Это поможет избежать двойного наказания — если тема не была распознана, ошибка по тональности по ней не учитывается.

Оценка качества модели участников будет производиться по двум направлениям:

- определение тематики (мультитейбл)
- определение тональности (сентимент-анализ)

Каждый компонент вносит вклад в итоговый балл по следующей схеме:

Компонент	Метрика	Вес в итоговой оценке
Тематика	F1-micro	70%
Тональность	Accuracy	30%

2.2. Метрики и порядок оценки:

Перед расчетом метрик организаторы выполняют ручной маппинг тем, предложенных командой, к темам из эталонного (gold) датасета.

Маппинг необходим, так как разные команды могут назвать одну и ту же тему по-разному, а иногда и объединить несколько тем в одну или, наоборот, разделить. Чтобы обеспечить корректное и справедливое сравнение, все предсказанные темы сначала приводятся к единой системе названий из эталона. При этом, темы, которым нет соответствия в эталоне, объединяются в категорию «Прочее» и не будут учитываться в метриках.

Затем считается $F1_{micro}$ и Accuracy.

	Истинные темы	Предсказанные темы
Отзыв 1	Ипотека, мобильное приложение	Ипотека, Дистанционное обслуживание
Отзыв 2	Кредитная карты, Вклад	Кредитная карта, Вклад

$$F1_{micro} = 0.75$$

Тема	Истинная тональность	Предсказанная тональность
Ипотека (отзыв 1)	положительно	положительно
Кредитная карта (отзыв 2)	нейтрально	нейтрально
Вклад (отзыв 1)	отрицательно	нейтрально

$$\text{Accuracy} = 0.667$$

$$\text{Финальная оценка: } 0.75 \cdot 0.7 + 0.667 \cdot 0.3 = 0.725$$

2.3. Требования к дашборду:

Сервис должен быть реализован в виде дашборда.

Требования к дашборду:

1. Отображение списка продуктов/услуг, полученных в результате кластеризации
2. Для каждого продукта/услуги:

- процентное распределение отзывов по тональностям

(положительно/нейтрально/отрицательно).

- абсолютное количество отзывов по каждой тональности.

- дополнительно (этот пункт не является обязательным) можно показать, какие именно аспекты продуктов особенно нравятся или не нравятся клиентам.

3. Возможность выбора временного интервала (01.01.2024–31.05.2025 или меньший период). Работа в режиме реального времени не требуется.

4. Динамика во времени:

- Динамика тональностей по каждому продукту/услуге — изменение долей положительных/нейтральных/отрицательных отзывов за выбранный период времени.

- Динамика количества отзывов по выбранным продуктам/услугам — изменение абсолютного числа упоминаний за выбранный период времени.

2.4. Порядок проверки решений:

К 3 октября у команды должен быть развернут веб-сервис с доступом по публичной ссылке (HTTP/HTTPS), предоставляющий API для получения предсказаний. Сервис работает только в режиме inference (предсказания), обучение в момент проверки запрещено. Время обработки запроса — не более 3 минут для датасета из 250 записей.

2.4.1. Формат входного запроса (JSON)

Запрос отправляется методом POST на endpoint /predict.

Пример тела запроса:

```
{
  "data": [
    {"id": 1, "text": "Очень понравилось обслуживание в отделении, но мобильное приложение часто зависает."},
    {"id": 2, "text": "Кредитную карту одобрили быстро, но лимит слишком маленький."}
  ]
}
```

id — уникальный идентификатор отзыва (целое число).

text — текст отзыва (строка в UTF-8).

2.4.2. Формат ответа (JSON)

После обработки файла модель должна вернуть файл в формате:

```
{
  "predictions": [
    {"id": 1, "topics": ["Обслуживание", "Мобильное приложение"], "sentiments":
    ["положительно", "отрицательно"]},
    {"id": 2, "topics": ["Кредитная карта"], "sentiments": ["нейтрально"]}
  ]
}
```

id — тот же, что во входном файле.

topics — одна или несколько тем (через запятую), выделенные моделью.

sentiments — массив тональностей для каждой темы в том же порядке.

2.4.3. Функциональные требования

Сервис должен быть доступен по HTTP/HTTPS.

Поддержка формата JSON (UTF-8).

Обработка до 250 отзывов за один запрос ≤ 3 минут.

В случае ошибки (неверный формат, пустые данные) сервис должен возвращать структурированный ответ с сообщением об ошибке.

2.4.4. Сценарий проверки

1. Организатор отправляет на API команд запрос с тестовым датасетом (без меток).
2. Получает предсказания в формате JSON.
3. Сравнивает результат с эталонным датасетом (gold).
4. Рассчитывает метрики F1-micro и Accuracy.

3. Программно-аппаратные требования:

3.1. Аппаратные требования при необходимости (например: тип устройства, на котором должно работать решение)

Решение должно быть развёрнуто на сервере или хостинге, обеспечивающем стабильный доступ по HTTP/HTTPS для проверки.

3.2. Программные требования (ожидаемый стек)

Веб-сервис с дашбордом и REST API для загрузки данных и получения предсказаний. Возможность обработки входного файла объёмом до N записей за ≤ 3 минуты.

Модель для кластеризации:

Использование методов машинного обучения, тематического моделирования, эмбедингов или гибридных подходов. Результат — тематическая группировка отзывов с описанием каждой темы (продукта/услуги).

Модель для классификации (мультилейбл):

Принимает текст отзыва и возвращает одну или несколько тем из списка кластеризации. Поддерживает мультилейбл-предсказание.

Для каждой темы возвращает тональность (положительная / нейтральная / отрицательная).

Дашборд:

Отображает список продуктов/услуг, их распределение по тональностям и динамику показателей.

- Интерфейс решения должен быть доступен и удобен в использовании.
- Сценарий и путь пользователя должны быть интуитивно понятными.
- Каждый элемент интерфейса должен решать определенную задачу и присутствовать на экране только если он необходим для решения задач пользователя.

4. Требования к презентации/демонстрации

Презентация представляется в формате pptx или pdf.

5. Требования к сопроводительной документации

К решению прилагается комплект сопроводительных документов включающий:

Описание решения:

- Краткое описание функционала сервиса и его назначения
- Схема/блок-диаграмма архитектуры (модель, backend, frontend, взаимодействие компонентов)
- Описание использованных методов и подходов к предобработке данных и построению моделей кластеризации и классификации

6. Ресурсы:

6.1. Для аппаратных средств это может быть, например, ОС устройств, диагональ экрана, ОЗУ устройства и др.

Организаторы не предоставляют отдельное оборудование, команды используют собственные устройства для разработки и развертывания решения.

7. Требования к сдаче решений

7.1. Требования для промежуточной сдачи решения

1. Ссылка на прототип для проверки решения в виде дашборда
2. Ссылка на endpoint (например, <https://team1-solution.ru/api/predict>) и доступ, если API закрыто: токен, ключ авторизации или тестовый логин/пароль.
3. Ссылка на репозиторий с исходным кодом (должен быть размещен на GitHub/GitLab или аналогичной платформе с доступом для организаторов)
4. Ссылка на презентацию
5. Ссылка на сопроводительную документацию
6. Контакт кого-либо из участников (телеграм), если endpoint не отвечает.

7.2. Требования для финальной сдачи решения

1. Ссылка на прототип для проверки решения в виде дашборда
2. Ссылка на endpoint (например, <https://team1-solution.ru/api/predict>) и доступ, если API закрыто: токен, ключ авторизации или тестовый логин/пароль.
3. Ссылка на репозиторий с исходным кодом (должен быть размещен на GitHub/GitLab или аналогичной платформе с доступом для организаторов)
4. Ссылка на презентацию
5. Ссылка на сопроводительную документацию
6. Контакт кого-либо из участников (телеграм), если endpoint не отвечает.

8. Критерии оценки

8.1. Подход коллектива к решению задачи

- Ясность и логичность идеи (насколько команда четко понимает задачу и сформулировала стратегию ее решения)
- Оригинальность подхода (наличие нестандартных идей в сборе данных, обработке текста, кластеризации, классификации и визуализации)
- Обоснованность архитектуры решения
- Выбор технологий и инструментов

8.2. Техническая проработка решения

- Качество кода
- Скорость работы решения (время выполнения предсказания при отправке через API)
- Подход к кластеризации данных (обоснованность выбранного метода, качество группировки — осмысленные и интерпретируемые темы)
- Подход к классификации (обоснованность выбранного метода, корректная реализация мультитейбл-предсказания)

- Работа с тональностью — наличие предсказания тональности, обоснованность выбранного метода

- Корректность работы сервиса (возможность получить предсказание по API)

Оценка дашборда:

- наличие базового интерфейса (страница доступна, данные прогружаются)
- Отображение списка тем
- Отображение тональностей
- Фильтр по временному периоду
- Интерактивность (корректная работа элементов управления)
- Визуальная читаемость (данные оформлены так, что их легко и удобно интерпретировать)

8.3. Соответствие решения поставленной задачи

- Полнота выполнения требований ТЗ (реализованы все обязательные функциональные блоки: кластеризация, мультитейбл классификация, определение тональности, дашборд)
- Согласованность результатов с этапами решения (классификация опирается на темы, выделенные при кластеризации, определена тональность)
- Качество описания решения (наличие понятной сопроводительной документации)

8.4. Эффективность решения в рамках поставленной задачи

- Качество определения тематик (мультитейбл классификация) — оценка по метрике $F1_{micro}$ на gold датасете
- Качество определения тональности — оценка по метрике Accuracy

8.5. Выступление коллектива на питч-сессии (только для финальной экспертизы)

- Четкое изложение идеи (команда ясно формулирует проблему, которую решает сервис, и ключевые преимущества своего подхода)
- Демонстрация работы сервиса
- Структурированность презентации
- Умение отвечать на вопросы