Разбор подходов к анонимизации

Описание

Все тесты проводились с одной и той же встречей. Я сделал повторную транскрибацию (без диоризации) для корректной имплементации анонимизации цисленных и количественных данных. Это было сделано по причине того, что цифры во временных отрезках транскрипта также распознавались как чувствительные данные.

Во всех случаях использовался Qwen3:30b.

Среди 15 гипотез были выявлены 5 наиболее перспективных:

1. Каскадная обработка с помощью библиотек NER + LLM.
2. Интеграция словаря/правил и LLM.
3. Генеративная проверка (Adversarial Testing)
4. Шаблонизация ролей и сущностей - не импелементированно
5. Fine-tune локальной модели - не имплементированно

Ниже указаны описания результатов имплементации, а также сама имплементация для каждого подхода. Полный код с имплементациями каждго похода находятся на гх:

https://github.com/ErnieCoding/text_summarizer

Кол-во чувствительный данных (# упоминаний сущностей, а также цифровых данных) в транскрипте: 69

1. Каскадная обработка

1. Описание: Извелечение сущностей с помощью NER бибилиотеки Natasha в связке с pymorphy для нормализации текста. Прогон результата через локальные модели (в нашем случае - Qwen3) для проверки контекста и исправления ошибок
2. Инстурменты:
    a. Natasha
    b. pymorphy2
    c. Qwen3:30b
3. Файлы с результатами:
    a. Анонимизированный текст: anonymized_text_{1}.txt
    b. Карта замен: mappings_{1}.txt
4. Результаты:
    a. Успешность: 88%
    b. Пропущенно (кол-во): 8
    c. Пропущенно (данные): телеграме, Свет, светланой, максим, ВКС, пиф, пять, два

---

2. Интеграция Правил/Словаря и NER+LLM

1. Описание: Составление изначального словаря (либо вручную, либо через NER; также может быть пустым), прогон словаря через NER (извлечение сущностей) + LLM (дополнительный анализ для поиска доменных терминов), прогон словаря через petrovich (для PERSON) и pymorphy (для ORG и LOC) для генерации всех падежных, заглавных и обычных форм, прямая замена в транскрипте, сохранение обновленного словаря.
2. Примечание: Можно давать наш изначальный словарь с набором различных общих терминов, платформ, а также стандартизированных значений для дат, номеров и других цифровых значений. Это упрощает

изначальный проход по тексту - заменяются данные, которые для NER определить сложнее.

   a. В анонимизированные данные попадают абсолютно все цифровые данные, которые упоминаются в транскрипте.

3. Инструменты:
   a. Natasha
   b. Petrovich
   c. pymorphy2 или 3 (может возникнуть пробема с таскерами - не проверял)
   d. Qwen3:30b
   e. initial_dict.json
4. Файлы с результатами:
   a. Анонимизированный текст: anonymized_text_{2}.txt
   b. Карта замен: initial_dict.json
5. Результаты:
   a. Успешность: 98.6%
   b. Пропущенно (кол-во): 1
   c. Пропущенно (данные): телеграме

---

3. Adversarial Testing

1. Описание: По составленному транскрипту с предыдущим подходом, мы отправялем его в LLM для попытки деанонимизации через трехступенчатую проверку. В случая угадывания каких-либо из сущностей, отправляем текст обратно на анонимизацию с поиском конкретных слов, которые были замечены. Первая ступень - лобовая проверка (угадывание сущностей напрямую), вторая ступень - креативная проверка (модель пытается угадать сущности по контексту или ассоциациям), третья ступень - проверка с подсказками (в промпте

задаются конкретные подсказки о типах данных, которые фигурируют в тексте).

2. Файлы с результатами:
    a. Анонимизированный текст: anonymized_text_{3}.txt
    b. Карта замен: initial_dict_{3}.json
    c. Полный отчет с проверками: adversarial_report.json
3. Результаты:
    a. Успешность: (также как и в 2)
    b. Пропущенно (кол-во): (также как и в 2)
    c. Пропущенно (данные): (также как и в 2)
    d. Отчет adversarial testing:

```
==== ADVERSARIAL SUMMARY ====
Number of tags: 43
Attack: direct_guess
accuracy: 0.000, partial_rate: 0.000
Attack: creative_guess
accuracy: 0.000, partial_rate: 0.000
Attack: hinted_guess
accuracy: 0.000, partial_rate: 0.000
```