Russian Text Detoxification Based on Parallel Corpora

Феоктистова Эмма
Щурова Елизавета

https://github.com/Mefeoss/Final_project_4year_NNLP

Мотивация

- В социальных сетях многие пользователи часто используют обсценную лексику, которая неприятна для большинства людей.
- Сегодня социальные сети, такие как Facebook, Instagram, VK, пытаются решить проблему токсичности путем удаления подобных текстов.
- Наше решение - представить нейтральную версию пользовательского сообщения, которая сохраняет значимый контент, - детоксикация.

Данные

1. **Исходные предложения** - русские токсичные сообщения с платформ Одноклассники, Пикабу и Твиттер.
2. **Параллельный датасет:** токсичное предложение на русском языке и 1-3 нетоксичных аналога.

Что пытались сделать?

1. Пытались сделать nlp решение, которое бы осуществляло style transfer для предложений - исправление токсичных частей в предложении
2. Цель - получить nlp решение, как можно более лучшее относительно тестовых метрик

Метрики

1. **Style transfer accuracy (STA) - Точность передачи стиля.**

Комментарии должны потерять свою токсичность и мат, т.е. значительно изменить свой стиль.

2. **Meaning preservation score (SIM) - Оценка сохранения значения.**

Необходимо, чтобы комментарий без оскорблений передавал смысл изначального комментария.

3. **Fluency score (FL) - Оценка беглости.**

Модель детоксикации должна выдавать текст, не уступающий по беглости исходному сообщению.

4. **Joint score (J) - Совместная оценка, (STA * SIM * FL).**

Совокупный результат по всем трем метрикам

Поставленные задачи

1. Опробовать baseline решения, параллельно разобравшись в их идейной сути и практической реализации:

1a. Разобрать, запустить и замерить метрики для решения с фильтрацией токсичных слов (Delete)

2b. Разобрать, запустить, получить модель и замерить метрики для supervised решения (на основе T5 модели и библиотеки transformers)

3c. Разобрать, запустить, получить модель и замерить метрики для решения на основе prompt подхода (библиотека RuPrompt) (но получить хорошую модель не вышло из-за огромного времени на обучение)

2. На основе лучшего (лучших) по метрикам из baseline решений осуществить подбор гиперпараметров (не успели)

3. Изучить и интегрировать решения репозитория https://github.com/s-nlp/rudetoxifier (не успели)

Как предобрабатывали данные?

1. **Delete**
   - Токенизация
   - Лемматизация
2. **T5**
   - Избавление от None в data frame
   - Группировка по парам: токсичное высказывание <===> нейтральное высказывание
   - Токенизация (такая же, как по pipeline модели)
   - Torch dataloader с подачей в него DataCollatorWithPadding
3. **ruPrompt**
   - Подготовка data frame (немного поменяли структуру)
   - Токенизация (такая же, как по pipeline модели)

Что получилось?

1. В подходе T5 модель обучилась и дает приемлемые результаты.
2. Модель в подходе ruPrompt не обучилась корректно.

Метрики для подхода Delete:

Style transfer accuracy (STA): 0.528

Meaning preservation (SIM): 0.874

Fluency score (FL): 0.824

Joint score (J): 0.356

Метрики для подхода T5:

Style transfer accuracy (STA): 0.731

Meaning preservation (SIM): 0.790

Fluency score (FL): 0.792

Joint score (J): 0.461

Что не получилось?

Хоть обучение в подходе ruPrompt удалось воспроизвести, однако модель не показывает удовлетворительного результата.

Что можно было бы еще попробовать сделать?

- Замерить метрики у более ранних весов T5
- Поиграться с гиперпараметрами для T5 подхода
- Детально понять, почему не обучилась модель в подходе ruPrompt
- Изучить эти решения: https://github.com/s-nlp/rudetoxifier
- Попробовать ещё какую-нибудь архитектуру из https://huggingface.co/docs/transformers/index

Спасибо за внимание!