Оптимизация критерия заданного нейросетевой моделью в задаче детоксификации текста

А. А. Пилькевич

Московский физико-технический институт

Консультант: А. С. Попов

2022

TODO: Новизна

В NLP есть тысячи задач. Под каждую задачу скорее всего есть отдельная модель, которая её решила. Сейчас большинство SOTA моделей очень дорого переобучать или вообще не возможно.
При этом существуют задачи, качество решения которых оценивают при помощи таких SOTA моделей. Например: качество сгенерированных текстов (смысл, связность, орфография), схожесть исходного и нового текста (перефразировка) и так далее.
И возникает естественное желание использовать эти модели в качестве функции потерь. Но есть одно НО...

Детоксификация предложений

Задача

Детоксификация токсичных входных предложений к нейтральному варианту.

Проблема

Для оценки качества детоксификации используется **нейросетевая модель** оценки токсичности. Но эту модель нельзя использовать в качестве функции потерь из-за отличного токенайзера, так как **теряется дифференцируемость**.

Решение

В данной работе предлагается переобучать эмбеддинг слой лосс-модели, который будет принимать логиты модели детоксификатора.

Данные и оценка качества

Задано множество пар: $\{T_i, D_i\}_{i=1}^{N}$, где T_i — токсичное предложение, D_i — нейтральная версия.
Пример:

T_i = *'Её муженька козла на кол надо посадить'*,
D_i = *'Её мужа нужно наказать'*.

Требуется: по токсичному предложению построить его нейтральный вариант.
Метрики:

- Style Transfer Accuracy (STA) — выход предобученного Conversational RuBERT дообученного на задачу определения токсичности,
- chrF — F-мера на основе символьных *n*-грамм.[1]

[1] chrF: character n-gram F-score for automatic MT evaluation (Popović, 2015)

STA, chrF

STA

Вероятность токсичности предложения. (?)

chrF

chrP — доля символьных n-грамм из предлагаемого предложения, которые имеются в оригинальном.
chrR — доля символьных n-грамм из оригинального предложения, которые представлены в предлагаемом.
Тогда финальная формула:

$$\text{chrF}\beta = (1 + \beta^2)\frac{\text{chrP} \cdot \text{chrR}}{\beta^2\text{chrP} + \text{chrR}},$$

где chrP, chrR считаются как среднее по всем предложениям.

Baseline и первые шаги

Берётся предобученный русскоязычный трансформер T5[2] (*детоксификатор*). Он дообучается на задаче seq-to-seq для пар $x, y \in \{T_i, D_i\}_{i=1}^{N}$:

$$Loss(p^*, p) = -\sum_{t=1}^{n}\sum_{v=1}^{|V|} p^*_{t,v} \log p_{t,v} = -\sum_{t=1}^{n} \log\left(p(y_t|y_{<t}, x)\right),$$

где p^*— истинная вероятность, p — предсказанная.

Предложение и проблемы

Хотим использовать **STA в качестве функции потерь**. Это позволит учитывать токсичность получаемых предложений и корректировать их.
Но в данном случае нет дифференцируемости, так как **нет отображения между токенами различных токенайзеров**.

[2] `https://huggingface.co/sberbank-ai/ruT5-base`

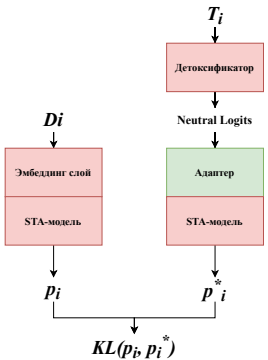
Адаптер

Вместо эмбеддингов STA-модели используется адаптер, который принимает на вход логиты детоксификатора и приближает истинные эмбеддинги.

Выход STA-модели: p_i — вероятность токсичности предложения D_i.
Функция потерь адаптера: $KL(p_i, p_i^*)$, где p_i^* — вероятность, полученная с использованием адаптера.

Алгоритм обучения

1. Дообучается T5 на задаче seq-to-seq.
2. Обучаем адаптер при фиксированных весах детоксификатора и STA-модели.



Дообучение детоксификатора

При дообучении детоксификатора используется схожая идея с обучением порождающих моделей при использовании adversarial loss[3].
Пусть CE, TP — значение кросс-энтропии и выход STA-модели (вероятность токсичности) на батче. И пусть $f(*,*)$ — произвольная функция, такая что $f : \mathbb{R}^2 \to \mathbb{R}$. Тогда

Алгоритм[4]

1. N батчей обучается детоксификатор на $f(\text{CE}, \text{TP})$, при фиксированном адаптаре.
2. M батчей обучается адаптер на KL-divergence, при фиксированном детоксификаторе.

[3] Goodfellow, I. et al., 2014. Generative adversarial nets. In Advances in neural information processing systems. pp. 2672–2680.

[4] В обоих случаях веса STA-модели фиксированы, как и везде до этого

Эксперименты и результаты

Размер обучающего датасета: $N = 11136$ пар (T_i, D_i), 10% использовались для валидации во время обучения. Размер тестового датасета: $N_t = 800$ предложений.

В качестве $f(\text{CE}, \text{TP})$ брались следующие комбинации:

$$f(\text{CE}, \text{TP}) = \text{CE} \cdot \text{TP},$$
$$f(\text{CE}, \text{TP}) = w_1\text{CE} + w_2\text{TP}.$$

Результаты экспериментов в различных сетапах обучения:

Эксперимент	STA	chrF1	STA*chrF1
CE only	0.739	0.578	0.427
GAN style, $\text{CE} + 10 \cdot \text{TP}$	0.813	0.569	0.462
GAN style, $\text{CE} \cdot \text{TP}$	0.754	0.574	0.439
Model loss only	0.998	0.119	0.119

Эксперименты и результаты

Подбор способа обучения и проверка стат. значимости:[5]

Эксперимент	STA	chrF1	STA*chrF1
CE only	0.744 ± 0.010	0.575 ± 0.002	0.430 ± 0.042
Same batch[6], $\mathrm{CE} \cdot \mathrm{TP}$	0.776 ± 0.015	0.569 ± 0.004	0.442 ± 0.006
Same batch, $\mathrm{CE} + 10 \cdot \mathrm{TP}$	0.774 ± 0.011	0.561 ± 0.012	0.435 ± 0.013

TODO: Эксперимент с одновременным обучением адаптера и модели
Обучение адаптера на эмбеддингах детоксификатора вместо логитов.

Эксперимент	STA	chrF1	STA*chrF1
CE only	0.742	0.577	0.428
Adapter on embs,	0.639	0.544	0.348

[5] При запусках экспериментов использовались различные seed

[6] На одном батче сперва оптимизировался детоксификатор, затем адаптер

Заключение

1. Предложен алгоритм использования модели в качестве функции потерь в условиях отсутствия дифференцируемость из-за различных токенайзеров.
2. Продемонстрирована работоспособность и эффективность предложенного метода.
3. Подобраны оптимальный параметры обучения.