Оптимизация критерия заданного нейросетевой моделью в задаче детоксификации текста

А. А. Пилькевич

Московский физико-технический институт

Научный руководитель: д.ф.-м.н. К. В. Воронцов, д.ф.-м.н. В. В. Стрижов

Консультант: А. С. Попов

2022

Детоксификация предложений

Задача

Стилизация токсичных (частично обсценных) входных предложений к нейтральному варианту. Для оценки качества используется нейросетевая модель степени токсичности.

Проблема

Модель невозможно использовать в качестве функции потерь из-за различия входных данных, так как теряется её дифференцируемость по параметрам.

Решение

Предлагается «адаптировать» оценочную модель, чтобы она принимала распределения вероятностей токенов, выданных детоксификатором.

Данные и оценка качества детоксификации

Дано множество пар (s_t, s_d):

s_t — токсичное предложение	s_d — нейтральная версия
«Её муженька козла на кол надо посадить.»	*«Её мужа нужно наказать.»*
«Это твари а не люди.»	*«Это плохие люди.»*

Требуется: по токсичному предложению построить нейтральное.

Качество стилизация текста оценивается:

- точностью соответствия заданному стилю,
- качеством сохранения смысла после перефраза.

Автоматическая оценка качества детоксификации

Style Transfer Accuracy (STA) — Conversational RuBERT[1]. Модель классификации, предсказывающая вероятность токсичности предложения. Отвечает за стилизацию текста.

chrF — F-score на основе символьных *n*-грамм[2]. Отвечает за сохранение смысла:

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

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

[1] https://huggingface.co/DeepPavlov/rubert-base-cased-conversational

[2] Popović M. *chrF: character n-gram F-score for automatic MT evaluation* – 2015.

Детоксификация как машинный перевод

Поставим задачу детоксификации как задачу машинного перевода (seq-to-seq) для пар предложений (t, d):

- $t = \tau_f(s_t)$, $d = \tau_f(s_d)$,
- τ_f — токенизатор, переводящий текст в последовательность BPE-токенов из словаря V_f.

Архитектура задаётся моделью кодировщик-декодировщик f_θ на основе предобученного ruT5-base[3].

$$\mathcal{L}_{\text{CE}} = -\sum_{i=1}^{n} \log f_\theta(d_i | d_{<i}, t) \longrightarrow \min_\theta,$$

$f_\theta(*|d_{<i}, t) \in [0,1]^{|V_f|}$ — распределение вероятностей.

[3] Raffel C. et al. *Exploring the limits of transfer learning with a unified text-to-text transformer* – 2019.

Детоксификация как задача стилизации

Предлагается использовать STA-модель g_{toxic} в качестве функции потерь:

$$\mathcal{L}_{\text{TP}} = g_{\text{toxic}}\big(\tau_g(s_{detox})\big) \to \min_{\theta}.$$

- τ_g — токенизатор, переводящий текст в последовательность BPE-токенов из словаря V_g,
- s_{detox} — результат работы детоксификатора:

$$s_{\text{detox}} = \{\arg\max_{d_i} f_\theta(d_i | d_{<i}, t)\}_{i=1}^{n}.$$

$\mathcal{L}_{\text{TP}}$ не дифференцируема по параметрам детоксификатора θ, в силу недифференцируемости функции τ_g и arg max!

$$\textit{ты слишком токсичный} \longrightarrow \begin{cases} [\textit{ты},\ \textit{слишком},\ \textit{токс},\ \textit{ич},\ \textit{ный}] & \text{от } \tau_f \\ [\textit{ты},\ \#UNK,\ \textit{токсичн},\ \textit{ый}] & \text{от } \tau_g. \end{cases}$$

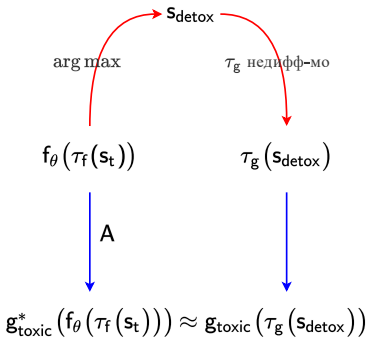
Адаптер — аппроксимация векторного представления

Хотим при фиксированных параметрах STA-модели g_{toxic} выполнения:

$$g_{\text{toxic}}\big(\tau_g(s_d)\big) \approx g^*_{\text{toxic}}\big(f_\theta\big(\tau_f(s_t)\big)\big).$$

g^*_{toxic} — STA-модель, в которой заменили входной эмбеддинг слой *адаптером* $A \in \mathbb{R}^{|V_f| \times e}$, где e — размерность векторного представления токенов.

Причём g^*_{toxic} принимает $f_\theta(*) \in [0,1]^{n \times |V_f|}$ — «зашумлённые one-hot вектора».



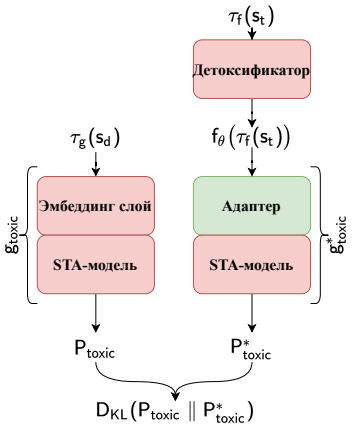
Обучение адаптера

Выход STA-модели:
$P_{\text{toxic}} = g_{\text{toxic}}\big(\tau_g(s_d)\big)$ — вероятность токсичности предложения s_d,
$P^*_{\text{toxic}} = g^*_{\text{toxic}}\big(f_\theta\big(\tau_f(s_t)\big)\big)$ — вероятность токсичности с использованием адаптера для s_t.

Функция потерь адаптера:
$D_{\text{KL}}(P_{\text{toxic}} \parallel P^*_{\text{toxic}}) \longrightarrow \min_A$.

Алгоритм обучения A

1. Обучается детоксификатор f_θ на задаче seq-to-seq.
2. Обучается адаптер A при фиксированных параметрах f_θ и STA-модели.



Дообучение детоксификатора

При дообучении детоксификатора f_θ используется схожая идея с обучением порождающих моделей[4].

Пусть CE, TP — значение кросс-энтропии $\mathcal{L}_{\text{CE}}$ и выход STA-модели $\mathcal{L}_{\text{TP}}$.
$F(*,*)$ — произвольная функция для агрегации функционалов, $F : \mathbb{R}^2 \to \mathbb{R}$.

Алгоритм дообучения f_θ

1. N батчей обучается f_θ на $F(\text{CE}, \text{TP})$, при фиксированных параметрах A.
2. M батчей обучается A на D_{KL}, при фиксированных параметрах f_θ.

[4] Goodfellow I. et al. *Generative adversarial nets* – 2014.

Эксперименты по детоксификации

Обучающая выборка: 11136 пар (s_t, s_d), 10% использовались для валидации во время обучения. Тестовой выборка: 800 предложений.

Результаты экспериментов в различных конфигурациях обучения:

Подход	F(CE, TP)	STA	chrF1	STA*chrF1
seq-to-seq	CE	0.739	0.578	0.427
GAN style	$CE \cdot TP$	0.754	0.574	0.439
GAN style	$CE + w_2 TP$	0.813	0.569	0.462
seq-to-seq	TP	0.998	0.119	0.119

Наилучшее качество показывает линейное взвешивание функций потерь, что позволяет задать приоритет для оптимизируемого функционала.

Проверенные подходы обучения

Проверка статистической значимости, когда на одном батче сперва оптимизировался детоксификатор, затем адаптер:

Подход	F(CE, TP)	STA	chrF1	STA*chrF1
seq-to-seq	CE	0.744 ± 0.010	0.575 ± 0.002	0.430 ± 0.042
Same batch	$\mathrm{CE} \cdot \mathrm{TP}$	0.776 ± 0.015	0.569 ± 0.004	0.442 ± 0.006
Same batch	$\mathrm{CE} + w_2\mathrm{TP}$	0.774 ± 0.011	0.561 ± 0.012	0.435 ± 0.013

Эксперимент с одновременным обучением детоксификатора и адаптера на одинаковую функцию потерь:

Подход	STA	chrF1	STA*chrF1
seq-to-seq CE	0.742	0.577	0.428
Adapter on embs	0.639	0.544	0.348
Adapter on logits	0.708	0.569	0.403

Выносится на защиту

1. Предложен алгоритм использования нейросетевой модели в качестве функции потерь в условиях отсутствия дифференцируемость, так как нет отображения между токенами различных токенайзеров.
2. Продемонстрирована работоспособность и эффективность предложенного метода.
3. Подобраны оптимальные параметры совместного обучения детоксификатора и адаптера.
4. Реализован и опубликован код для воспроизведения экспериментов из представленной работы[5].

[5] https://github.com/Intelligent-Systems-Phystech/Pilkevich-BS-Thesis