

Легковесный детектор для токсичного текстового контента

Мы используем глубокие модели повсюду для решения наших задач. В последнее время есть тренд делать модели всё больше и больше, однако это делает их использование в реальном мире сложнее — они занимают много памяти и долго работают. Поэтому мы всё чаще начинаем думать о том, как бы уменьшать наши модели, чтобы запускать их, например, на смартфонах.

В этом задании мы предлагаем вам обучить легковесный классификатор токсичного контента. Вы вольны делать с моделью что угодно, главное, чтобы в результате она весила как можно меньше и работала как можно быстрее.

Задание

Датасет:

<https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge>

1. Прочитайте правила.
2. **Обучите легковесную модель** определять 6 лейблов: `toxic`, `severe_toxic`, `obscene`, `threat`, `insult`, `identity_hate`. Вы можете пытаться сделать модель легковесной любым способом, который вы считаете правильным. Главное получить маленький и быстрый классификатор.
3. Раскройте в отчёте следующие темы:
 1. Какая получилась точность у модели?
 2. Какие есть способы ускорения/уменьшения модели?
 3. Как выбрать баланс между качеством и скоростью?
 4. Что лучше — тяжелая модель и потом ее оптимизировать или сразу легкая? Поясните почему.
 5. Опишите ваш подход. Чем он лучше других возможных подходов? Какие у него могут быть недостатки?
 6. Что можно сделать, чтобы улучшить классификатор?
4. **Бонусные очки** — подумайте, как добавить интерпретируемость в работу модели.

Правила

1. **Нет правильного способа решить задачу.** Не стоит беспокоиться, что вы делаете что-то неправильно. Мы хотим увидеть ваши способности к исследованиям, а не какое-то конкретное решение задачи.
2. Используйте [colab](#), чтобы получить доступ к бесплатным вычислительным ресурсам.
3. Присылайте решение в виде репозитория на github, в котором будет вложен `.ipynb` с решением, чёткие инструкции, как запустить ваш код и отчёт по задаче. **Убедитесь, что мы сможем запустить ваше решение по этим инструкциям.**

4. Вы можете использовать любые библиотеки и фреймворки, которые вам могут быть необходимы. **Если вы заимствуете чужой код в решении** (в разумных пределах; например, для загрузки данных), **чётко обозначьте, где находится ваш код, а где чужой.**
5. **Сфокусируйтесь на том, чтобы код был чист и понятен.** Если вы считаете, что какая-то его часть может быть непонятна, то добавьте комментарии. Мы очень сильно ценим хорошо написанный код, поэтому **если решение задачи будет оформлено грязно, то мы можем отклонить заявку.**
6. Желательно сделать задание целиком, но вы можете пропустить какие-то части, если не чувствуете себя достаточно опытным. Но помните — кто-то другой его сделает целиком.
7. **Мы ждем и будем рады, если вы при выполнении этого задания узнаете что-то новое. Весь движ — ради новых знаний.**