

# Отчет о проделанной работе

## 1. Обучение модели

В качестве предобученной модели я выбрал [bert-base-uncased](#) и дообучил ее на предоставленном [датасете](#) классифицировать по 9-ти классам.

Статус отзыва (Положительный или отрицательный) рассчитывается как:

- Отрицательный, если предсказание модели  $< 5$ ;
- Положительный, если предсказание модели  $> 5$ ;

Использовал библиотеки transformers, pytorch

## 2-3. Веб-сервис в открытом доступе

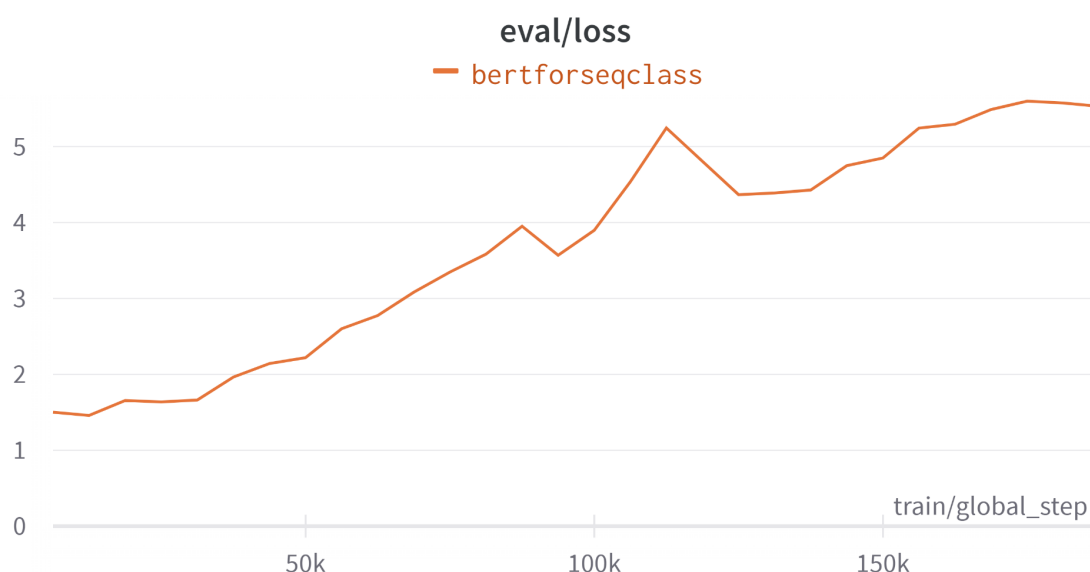
Веб-сервис разработан на Django, доступно окно для ввода текста отзыва и кнопка, по нажатию которой модель предсказывает оценку.

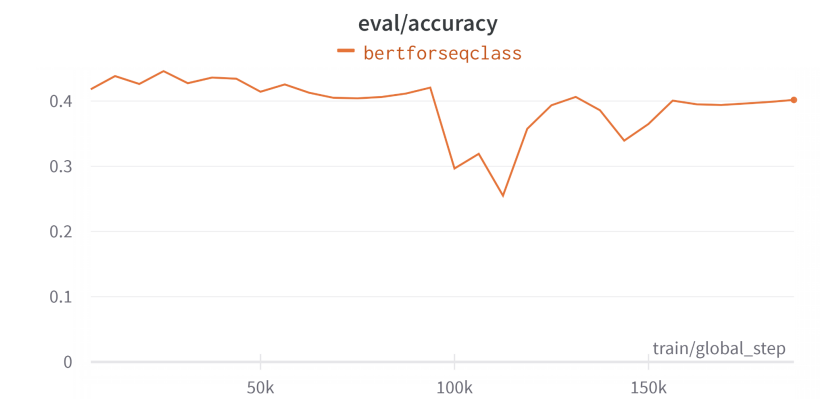
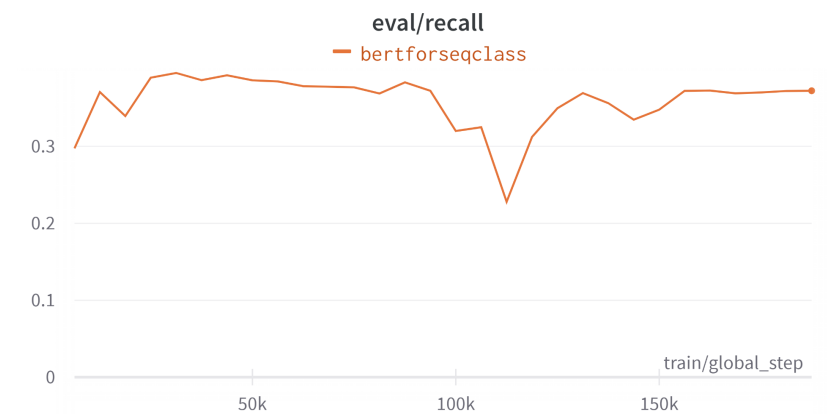
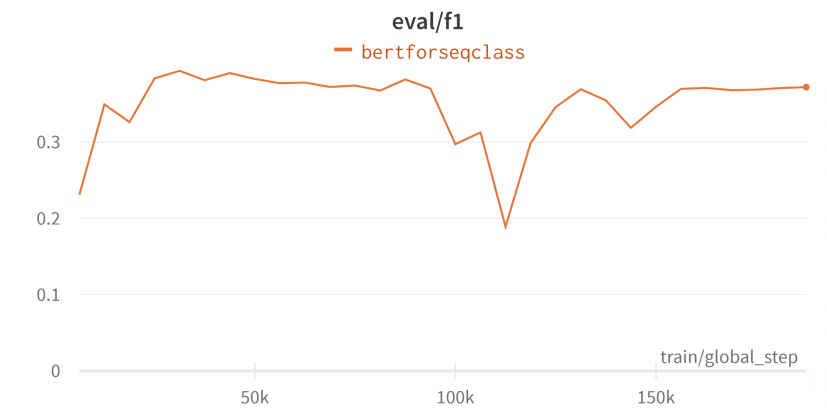
При запуске проекта локально\на различных хостингах веса дообученной модели с помощью функции [download\\_model](#) автоматически скачаются.

## 4. Оценка точности полученного результата

Трекинг моделей велся с помощью сервиса wandb.

Предоставляю графики метрик дообучения на 9 классах по эпохам (30) для валидационного датасета





Итоговой моделью, используемой в веб-сервисе была выбрана модель после 5-ой эпохи. Ее метрики на тестовом датасете при бинарной классификации (позитивный\негативный) таковы:

	precision	recall	f1-score
NEGATIVE	0.89	0.90	0.89
POSITIVE	0.89	0.88	0.89
accuracy			0.89
macro avg	0.89	0.89	0.89
weighted avg	0.89	0.89	0.89