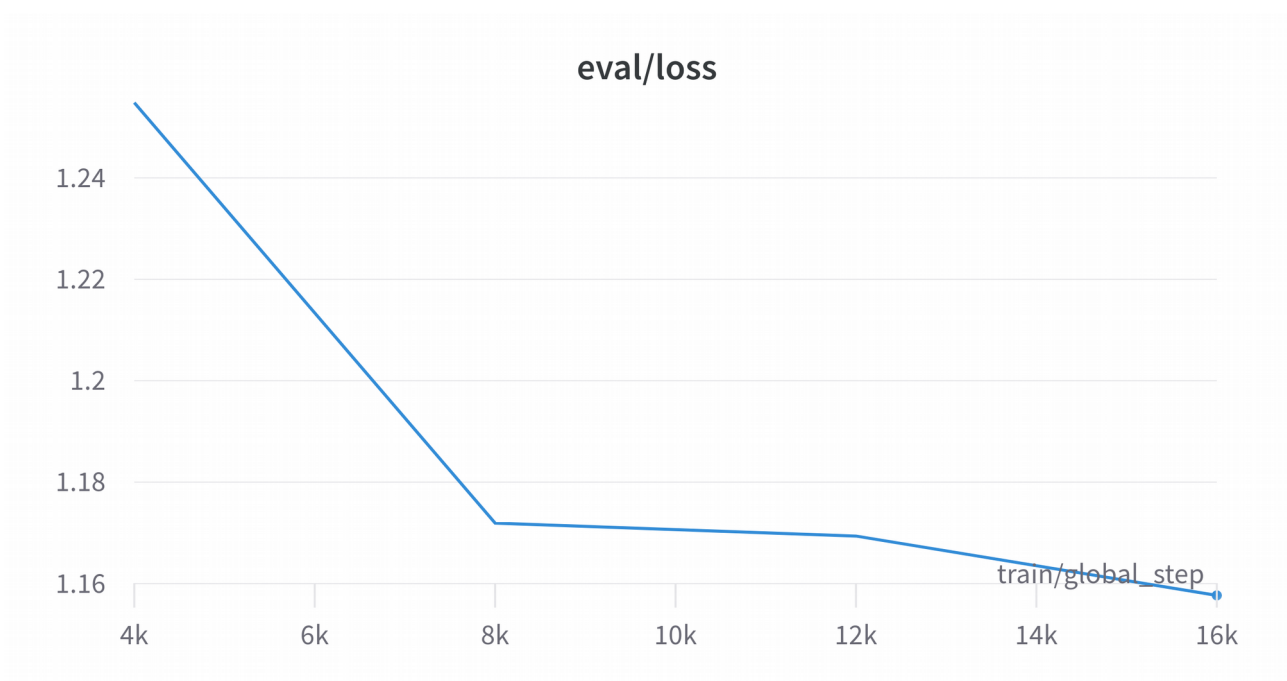


Постановка задачи: обучение большой языковой модели (LLM) для решения задачи классификации (можно представить в качестве классификации на 8 классов)

Как проводилось обучение: локально, использовался графический ускоритель AMD Radeon 6700, на бэкэнде Pytorch для ROCm (HIP) на Linux ОС

Выбор модели: microsoft/deberta-base (<https://huggingface.co/microsoft/deberta-base>). Была выбрана эта модель, т.к ее можно тренировать с комфортной скоростью (с учетом вычислительных ограничений) и были достигнуты удовлетворительные результаты.

Результаты: В качестве функции потерь была выбрана кросс-энтропия (график тестовой потери см. ниже). Был достигнут результат 1.158, что для 8 классов был определен как приемлемый. Тестовая выборка составляет 10000 элементов.



Ниже приведена выдержка из логов об обучении модели. Из них видно, что последующее обучение могло привести к переобучению (тренировочная функция потерь стабильно заметно меньше тестировочной)

```
301 {'loss': 0.9126, 'learning_rate': 1.4475000000000001e-06, 'epoch': 1.86}
302 {'loss': 0.8831, 'learning_rate': 1.3850000000000003e-06, 'epoch': 1.86}
303 {'loss': 1.0593, 'learning_rate': 1.3225000000000003e-06, 'epoch': 1.87}
304 {'loss': 1.0568, 'learning_rate': 1.26125e-06, 'epoch': 1.88}
305 {'loss': 0.9477, 'learning_rate': 1.19875e-06, 'epoch': 1.88}
306 {'loss': 0.9403, 'learning_rate': 1.1362500000000001e-06, 'epoch': 1.89}
307 {'loss': 0.9372, 'learning_rate': 1.07375e-06, 'epoch': 1.89}
308 {'loss': 1.022, 'learning_rate': 1.01125e-06, 'epoch': 1.9}
309 {'loss': 0.9378, 'learning_rate': 9.487500000000001e-07, 'epoch': 1.91}
310 {'loss': 0.9427, 'learning_rate': 8.8625e-07, 'epoch': 1.91}
311 {'loss': 0.9537, 'learning_rate': 8.237500000000001e-07, 'epoch': 1.92}
312 {'loss': 0.9104, 'learning_rate': 7.6125e-07, 'epoch': 1.93}
313 {'loss': 1.0046, 'learning_rate': 6.987500000000001e-07, 'epoch': 1.93}
314 {'loss': 1.0518, 'learning_rate': 6.3625e-07, 'epoch': 1.94}
315 {'loss': 0.8995, 'learning_rate': 5.737500000000001e-07, 'epoch': 1.94}
316 {'loss': 0.9119, 'learning_rate': 5.1125e-07, 'epoch': 1.95}
317 {'loss': 0.9222, 'learning_rate': 4.4875000000000004e-07, 'epoch': 1.96}
318 {'loss': 1.0566, 'learning_rate': 3.8625e-07, 'epoch': 1.96}
319 {'loss': 0.9287, 'learning_rate': 3.2375e-07, 'epoch': 1.97}
320 {'loss': 0.9872, 'learning_rate': 2.6125e-07, 'epoch': 1.98}
321 {'loss': 0.9699, 'learning_rate': 1.9875000000000003e-07, 'epoch': 1.98}
322 {'loss': 0.9906, 'learning_rate': 1.3625000000000002e-07, 'epoch': 1.99}
323 {'loss': 0.9666, 'learning_rate': 7.375000000000001e-08, 'epoch': 1.99}
324 {'loss': 0.9182, 'learning_rate': 1.125e-08, 'epoch': 2.0}
325 {'eval_loss': 1.1576775312423706, 'eval_runtime': 222.7632, 'eval_sample
326 {'train_runtime': 7335.2492, 'train_samples_per_second': 10.906, 'train_
327 [{'label': 'LABEL_3', 'score': 0.4052000045776367}]}
328 LABEL_3
```