Отчет об обучении ASR модели

Воспроизведение модели

Для того, чтобы воспроизвести модель необходимо запустить обучение с конфигурацией из файла *src/configs/deepspeech_char_colab.yaml.* Данный конфигурационный файл производит обучение deepspeech2 модели на наборе данных *train-clean-360*. Лучшая модель была получена после 23 эпох обучения.

Полные логи обучения доступны по ссылке wandb:
https://wandb.ai/professor322/asr_model/workspace

Обучение итоговой модели

Итоговая модель имеет архитектуру deepspeech2 с 5 GRU слоями, где размер скрытого состояния был равен 1024. Итоговая модель использует символьную кодировку, так как bpe (byte pair encoding) не продемонстрировал никаких улучшений.

Тренировочные данные проходили ряд преобразований на уровне спектрограмм и звуковых волн. Для звуковых волн было выбрано добавление гауссовского шума и изменение громкости, а для спектрограмм частотное и временное маскирование.

Обучение производилось на google colab с помощью видеокарты T4. Примерно было потрачено 11 часов на обучение модели.

Эксперименты

Был проведен ряд экспериментов, которые позволили определиться с итоговой конфигурацией модели.

Learning rate

Самым первым и, вероятно, самым главным шагом была настройка скорости обучения, так как при первоначальной конфигурации планировщика скорости обучения (OneCycleLR) модель выходила на плато и обучение останавливалось вне зависимости от batch size. Модель стала обучаться, когда диапазон увеличения learning rate был ограничен в районе от 3e-5 до 3e-4. В целом, стоило попробовать избавиться от OneCycleLR совсем, так как изначально он только мешал.

Вывод: в финальной модели был использован OneCycleLR планировщик с learning rate [3e-5, 3e-4]

Byte pair encoding vs char encoding

Был проведен эксперимент с использованием BPE и символьной кодировки.

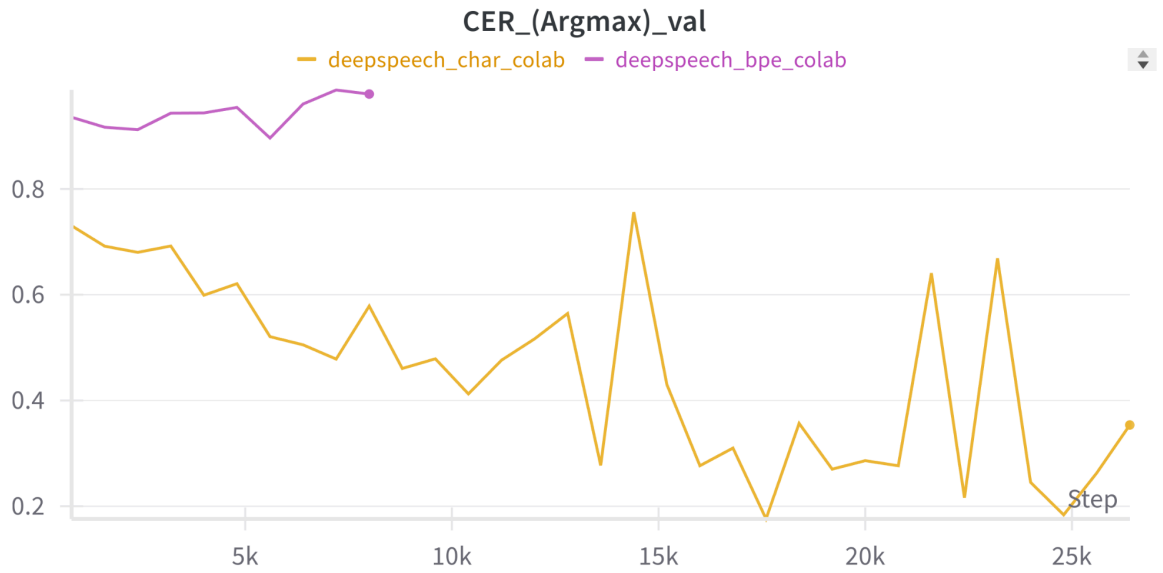

Рисунок 1. CER при обучении на BPE и посимвольной кодировке. На графике можно заметить, что модель с BPE за 10000 итераций не смогла существенно улучшить качество предсказаний. В свою очередь посимвольная кодировка демонстрировала улучшение метрики и именно поэтому она была выбрана для дальнейшего обучения модели.

В результате вариант модели с BPE кодировкой была отвергнут, так как модель не улучшала качество предсказаний.
Вывод: финальная модель использует символьную кодировку.

Beam search

Использование Beam Search давало небольшой прирост к обоим метрикам качества WER/CER. Важное замечание, что beam search используется только во время применения модели, так как во время обучения это довольно сильное замедление.

Metric Name	Value (lower - better)
WER ArgMAX	0.442
CER ArgMAX	0.158
WER BeamSearch	**0.433**
CER BeamSearch	**0.153**

Вывод: Beam Search используется для подсчета финальных метрик WER/CER

Beam search + LM

Использование языковой модели дало самый существенный прирост к качеству предсказаний. Изначально использование языковой модели только ухудшало качество предсказаний так как символьный словарь не был сопоставим с n-gram, которые были выучены языковой моделью. Соответственно важным шагом для того, чтобы заставить языковую модель работать, было приведение обоих словарей к одному регистру. Также были опробованы несколько языковых моделей.

Metric Name	Value (lower - better)
WER ArgMAX	0.442
CER ArgMAX	0.158
WER BeamSearch	0.433
CER BeamSearch	0.153
WER BeamSearch + 4-gram.arpa	**0.237**
CER BeamSearch + 4-gram.arpa	**0.108**
WER BeamSearch + 3-gram.pruned.3e-7.arpa	0.264
CER BeamSearch + 3-gram.pruned.3e-7.arpa	0.117

Вывод: LM *4-gram.arpa* используется для подсчета финальных метрик WER/CER

Обзор бонусных заданий

- Реализована BPE кодировка, но она не использована в финальной модели, так как не давала прироста к метрикам качества
- Реализован Beam Search
- Реализовано использование LM модели