

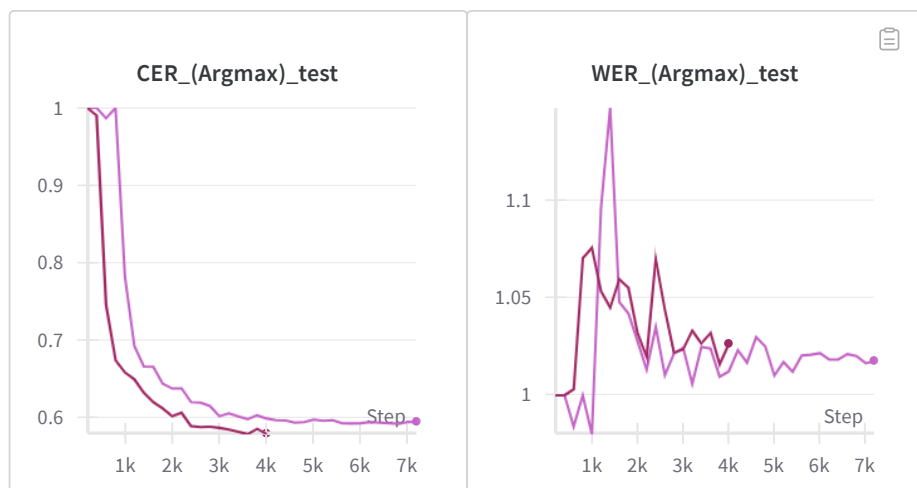
Report

Игорь Дмитриевич Рябков

Created on October 14 | Last edited on October 14

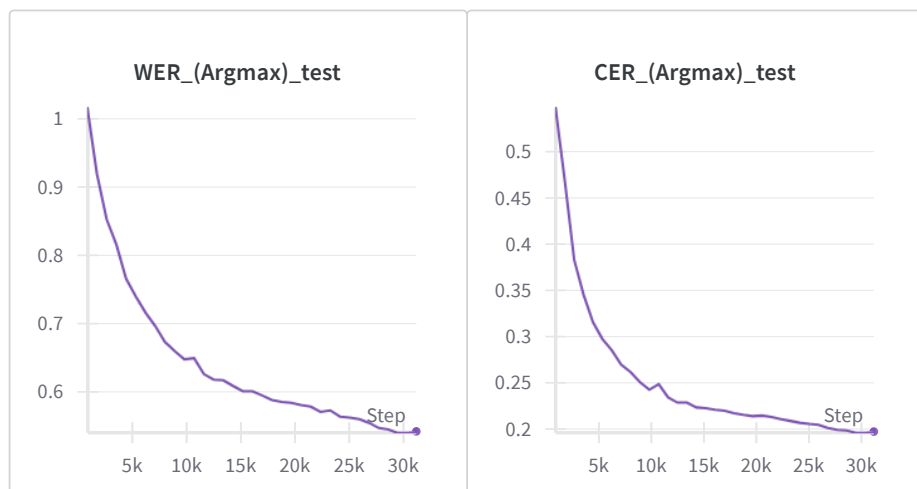
Первые модели

Первые модели, которые были испробованы - чистые RNN с константным $lr=0.003$, получилось добиться CER=0.6, WER не сходился

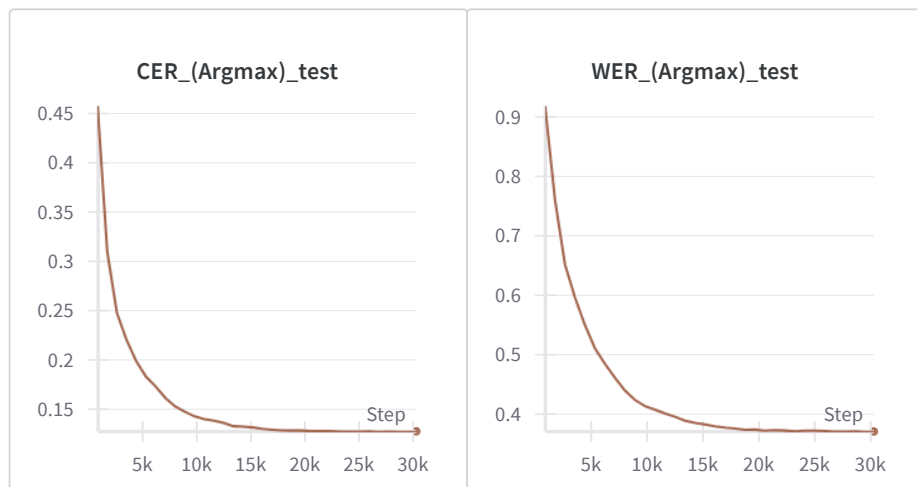


Такой плохой результат был вызван тем, что lr сильно затухал и модель переставала обучаться

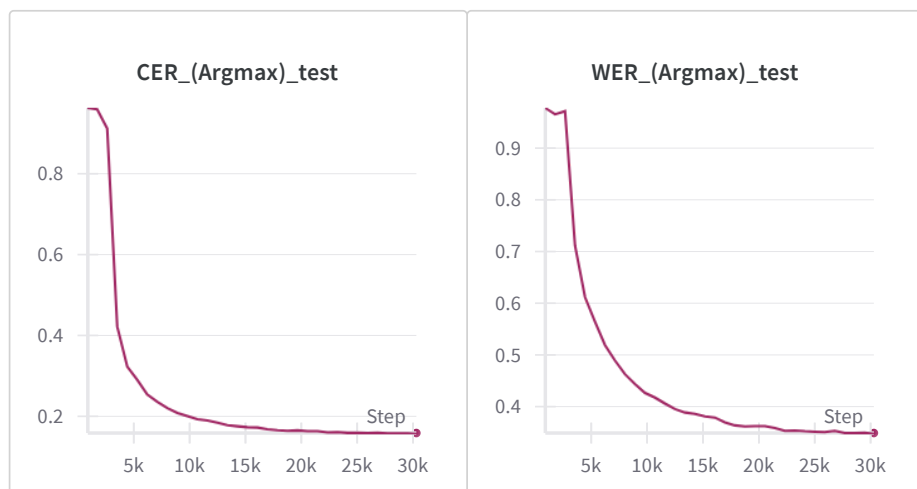
Значительно улучшить качество помогло увеличение кол-во эпох на обучение, lr раз в эпоху (сначала константный 0.003, потом экспоненциально затухающий)



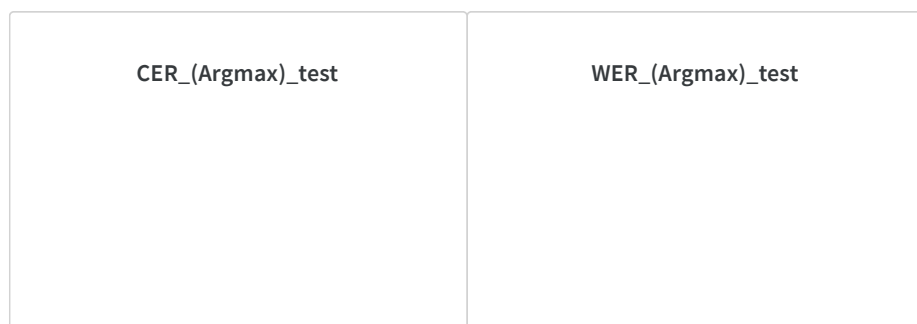
Следующий прирост в качестве произошёл после увеличения кол-ва слоёв RNN до 6, использование двунаправленной RNN, увеличение вдвое кол-ва скрытых слоёв. А также изменение lr до 0.001 с экспоненциальным затухание 0.9



Следом я добавил `bpe_vocab_size=1000`, данная прибавка изменила то, как модель обучается. Процесс обучения перестал быть однородным. Первые несколько эпох модель застревает, не показывая улучшение метрик. Попадание в локальный минимум происходит резко. (Данный ран был лучшим)



Дальше я решил добавить несколько слоёв конволюции с изменением каналов $1 \rightarrow 32 \rightarrow 1$. Как я понял, данное изменение было слишком резким, модель перестала сходиться:





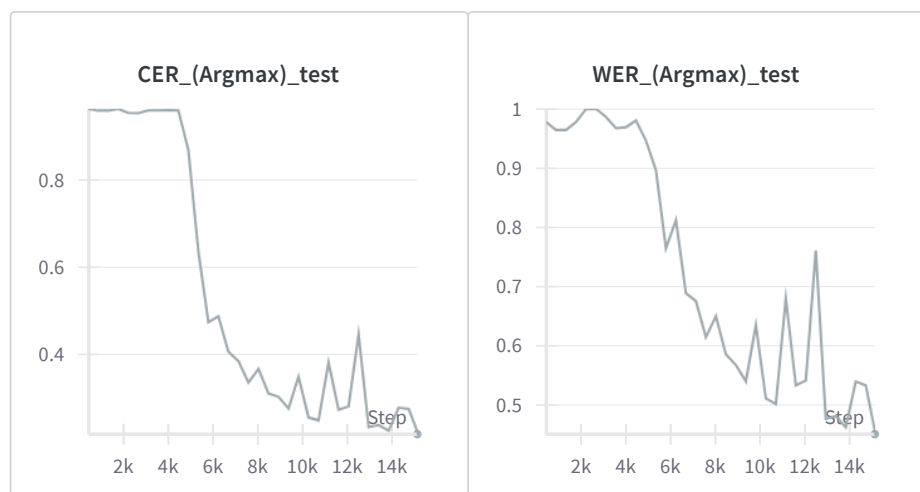
Что это пофиксить, я решил сделать правильный переход между каналами, сделать его более плавным с batch нормализациями и h между ними. Данное изменение заставило модель сходиться, одна, как бы я не изменял параметры консолидации, стабильно заставить их сходиться у меня не получилось



Также, были добавлены аугментации и beamsearch:

Целый ран я запускал только с шумом и маскированием из-за того, что остальные требовали слишком много времени.

Данный ран не принёс сильного результата, модель сходилась, но качество выходило хуже



Работоспособность всех аугментации можно увидеть в ране с соответствующим названием

BeamSearch был также реализован, на тестовом сете он немного улучшил качество (beam=3):

```
inference_CER_(Argmax): 0.12751327657476072  
inference_WER_(Argmax): 0.37070674129600656  
inference_WER_(Beam): 0.3673631140828037  
inference_CER_(Beam): 0.12647157295653758
```

Created with  on Weights & Biases.

https://wandb.ai/dm-_axel/pytorch_template_asr_example/reports/Report--Vmldzo5NzE4MTc5