NLU course projects

Amir Gheser (247173)

University of Trento

amir.gheser@studenti.unitn.it

1. Introduction (approx. 100 words)

I replaced the RNN layer with a single LSTM and two dropout layers, testing different values, concluding with a stronger dropout after the embedding layer and a weaker dropout before the final output layer. Lastly, multiple test were conducted with varying learning rates and varying batch sizes. Applying weight tying and variational dropout permitted higher learning rate producing greater results. Lastly, I integrated non-monotonically triggered AvSGD but it did not provide significantly better results.

2. Implementation details (max approx. 200-300 words)

I tested the LSTM with different dropout values, initially 0.5 for both dropout layers, after running multiple tests I concluded by setting an embedding dropout probability of 0.65 and a output dropout probability of 0.2 and used linear schedule with warm-up. Greater dropout values reduce the impact of overfitting, allowing for greater learning rates during training, hence the learning rate for SGD was changed from 1 to 3 yielding greater results with a perplexity of 148 (delta of -8). After replacing SGD with the AdamW optimizer with a learning rate of 1e-3 I noticed that the loss dropped more rapidly in the first epochs. Also the role of a scheduler is less relevant since the optimizer is already changing the learning rate. The application of the weight tying regularization technique caused a reduction in perplexity to 106 (-13).

In part 2 of the assignment, I applied weight tying, another regularization technique which makes the model share the output and embedding weights. Then I replaced standard dropout with its variational kind which uses the same mask across all timesteps during the forward pass, making it more suitable for LSTMs resulting in greater temporal coherence. Lastly, as suggested by Merity S. et al. [1].
I included non-monotically triggered AvSGD. This approach trains using SGD and monitors the perplexity in a time window T, and if the perplexity doesn't improve with respect to the minimum of such window we start training with AvSGD.

3. Results

In Part1 sub-task 1 we have no regularization method implemented. In fact the plot shows that the model is clearly overfitting. After adding dropout and slightly increasing the size of the model I achieved a PPL of 148. This is less than the previous stage but the model hasn't overfit. Changing optimizer to AdamW with a learning rate of 1e-3 yields much faster convergence and better results achieving a perplexity of 119, although showing some sign of overfitting.

In part2 this issue is addressed and with the addition of weight tying the models validation and train losses show less divergence (see Figures 3,4) and it achieves a minimum of 106.

	sub1	sub2	sub3
Part 1	140	148	119
Part 2	106	101	100

Table 1: *Perplexity of the best achieved model for each subtask*

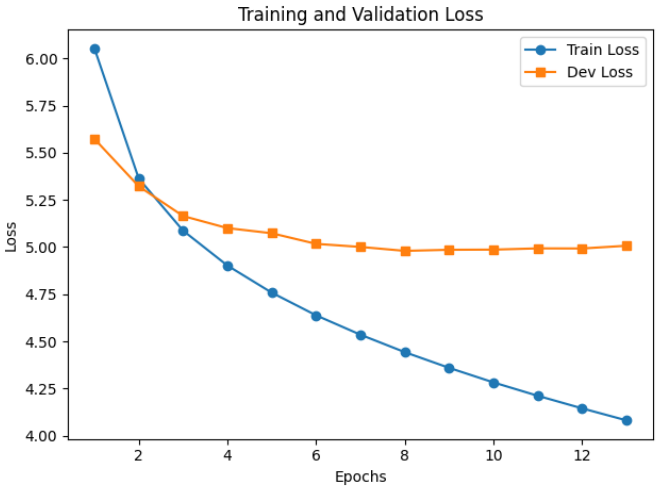

Figure 1: *Plot for task 11*

After the integration of variational dropout the model achieves a perplexity score of 100. Despite performing different tests and changing window size the model wasn't able to trigger AvSGD and get sensibly better results and achieved a score of 101. Very similar to the previous step.

4. References

[1] S. Merity, N. S. Keskar, and R. Socher, "Regularizing and Optimizing LSTM Language Models," *arXiv preprint arXiv:1708.02182*, 2017.

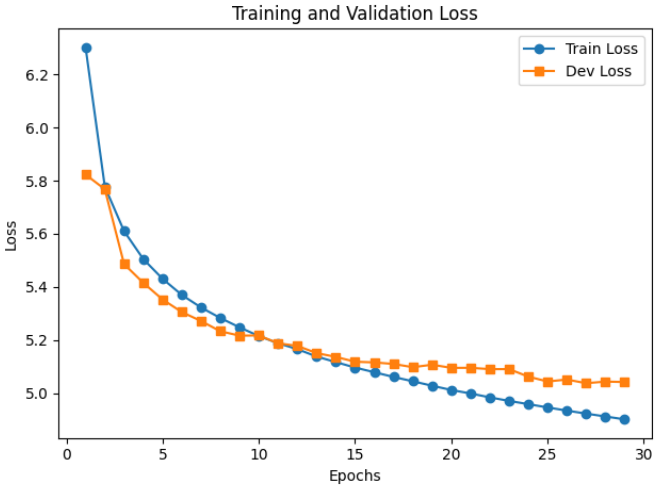

Figure 2: *Plot for task 12*

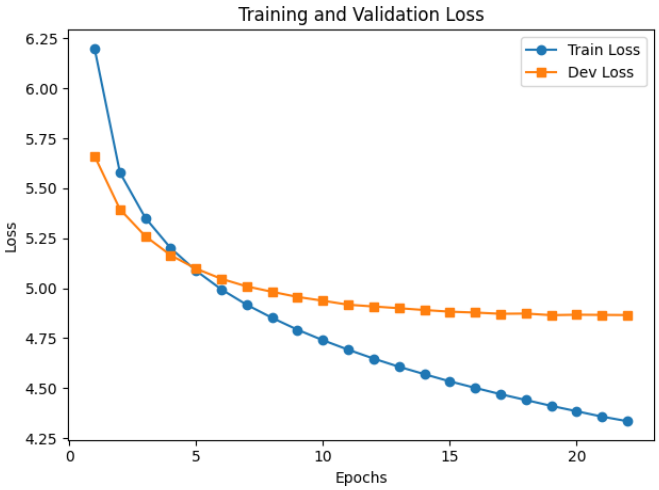

Figure 3: *Plot for task 13*

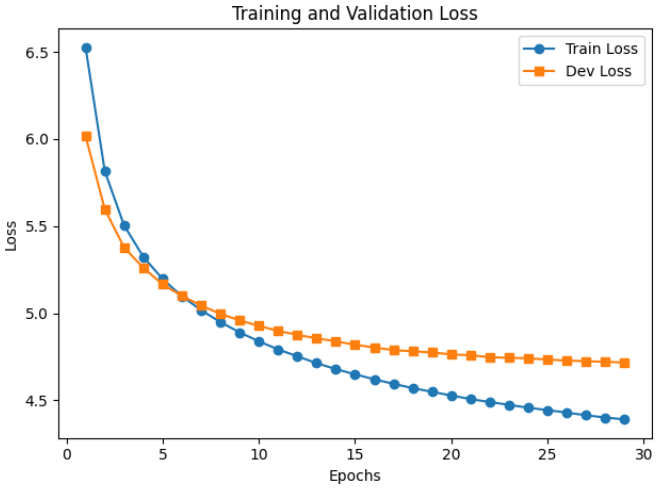

Figure 4: *Plot for task 21*

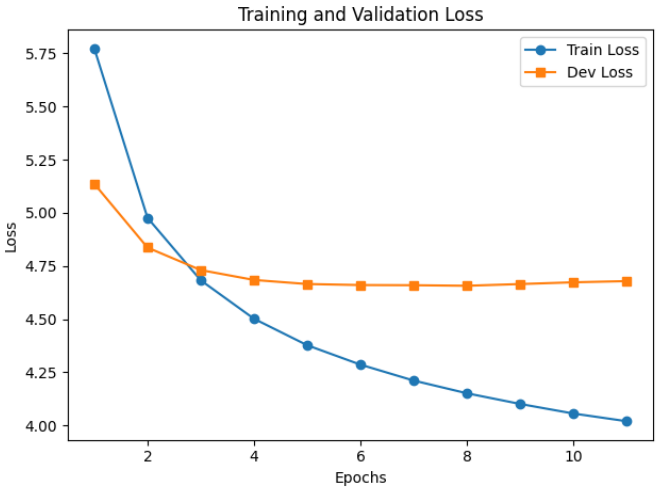

Figure 5: *Plot for task 22*

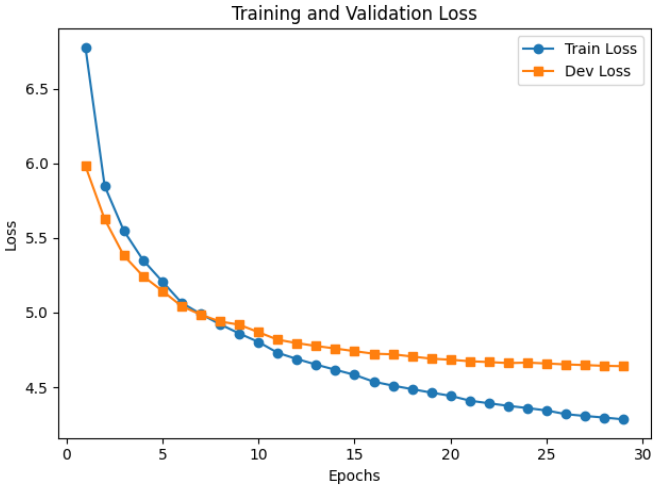

Figure 6: *Plot for task 23*