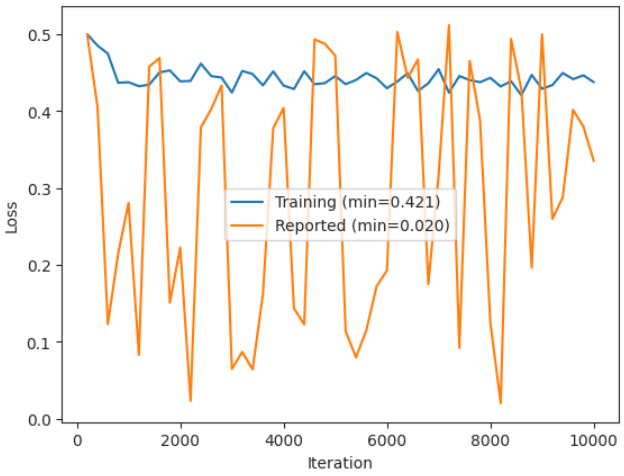
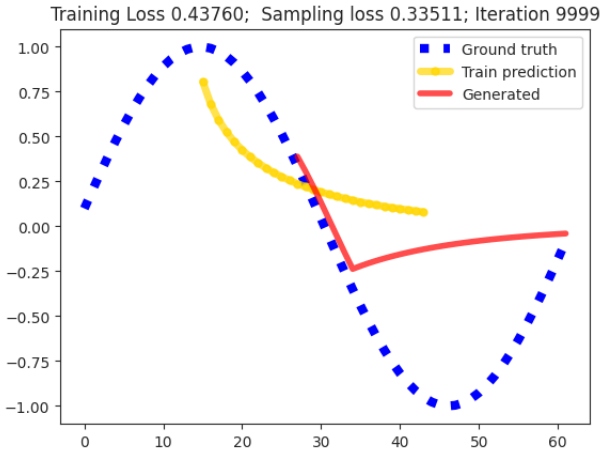
Homework Assignment 2
Adrian Dinu Urse

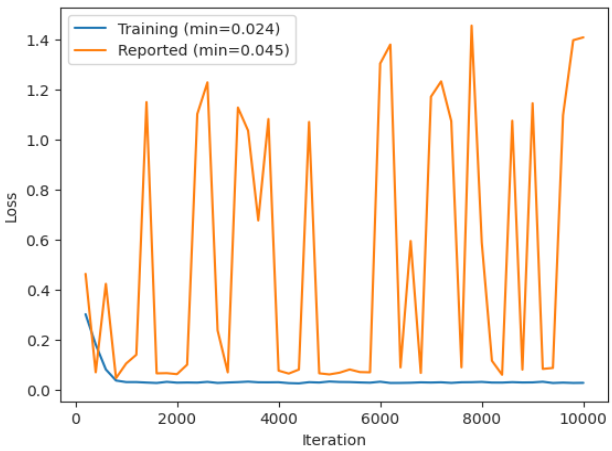
Task 1
Teacher forcing propability = 0

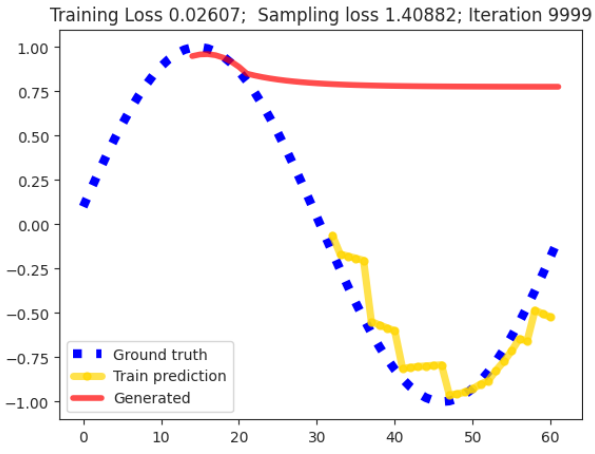




With **teacher forcing = 0**, the model struggles to learn effectively. Training loss stabilizes, but reported loss fluctuates, showing poor generalization. The generated sequence diverges significantly after the warm start, as the model fails to handle compounding errors when relying on its predictions.

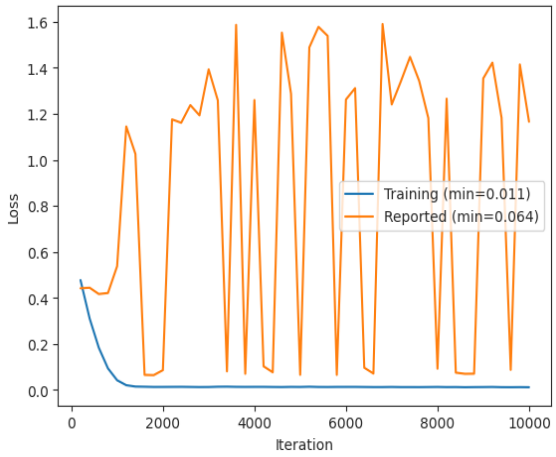
Teacher forcing probability 0.5

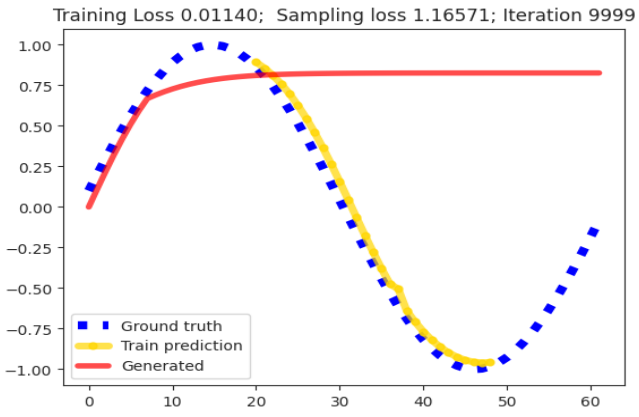




For **teacher forcing set to 0.5**, the model demonstrates significant improvement in learning. The training loss decreases rapidly and stabilizes at a low value, indicating effective short-term dependency learning. However, the reported loss fluctuates, suggesting that while the model performs well during training, it struggles to generalize consistently over long sequences. The generated sequence initially aligns with the ground truth, but divergence occurs after a few timesteps, highlighting the limitations of relying partially on teacher forcing.

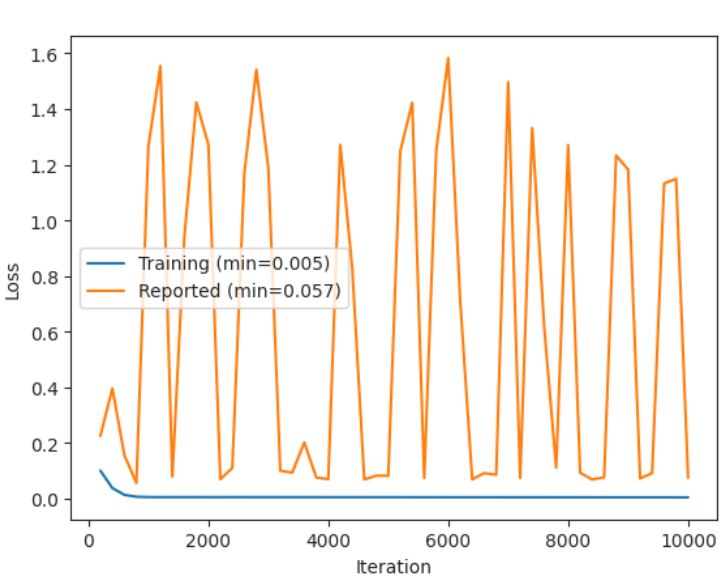
Teacher forcing probability 0.75

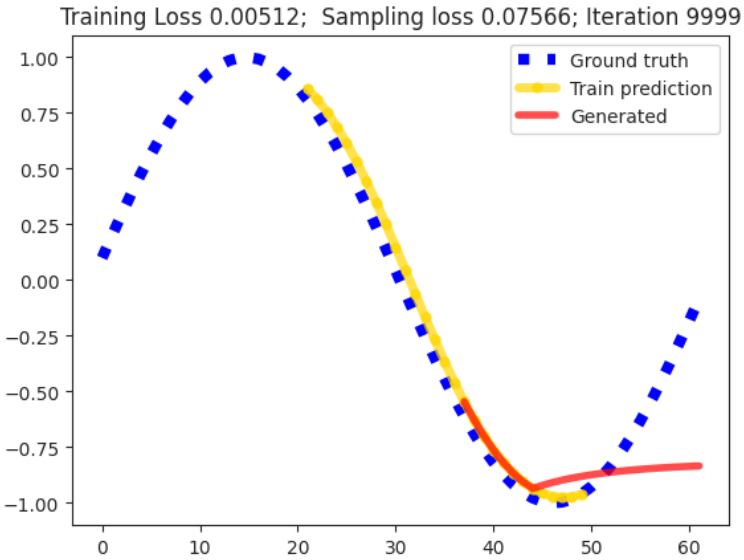




With **teacher forcing set to 0.75**, the model achieves very low training loss, indicating highly effective learning during training. However, the reported loss shows substantial fluctuations, reflecting difficulty in generalizing when relying on self-predictions. The generated sequence aligns closely with the ground truth initially but diverges over time, with compounding errors becoming evident as teacher forcing is reduced during evaluation. This setup heavily guides the model during training but still struggles to transition smoothly to independent predictions.

Teacher forcing propability = 1





With **teacher forcing set to 1**, the model achieves excellent training performance, with loss stabilizing at an extremely low value. However, during evaluation, the generated sequence diverges quickly after the warm start, showing significant reliance on ground truth during training. While the model predicts well for short intervals, it struggles to transition to independent predictions, leading to noticeable deviations as it relies solely on its outputs.

Difference between teacher forcing and learning on own samples: What are the pros and cons of teacher forcing?

Pros:

1. **Faster Convergence**: By using the ground truth, the model avoids compounding errors during training, allowing it to learn more quickly and effectively.
2. **Improved Short-Term Accuracy**: The model learns to closely match the ground truth within the training sequence, especially for short-term dependencies.
3. **Stable Training**: Teacher forcing ensures that the training process remains stable, reducing the risk of unstable gradients caused by cascading prediction errors.

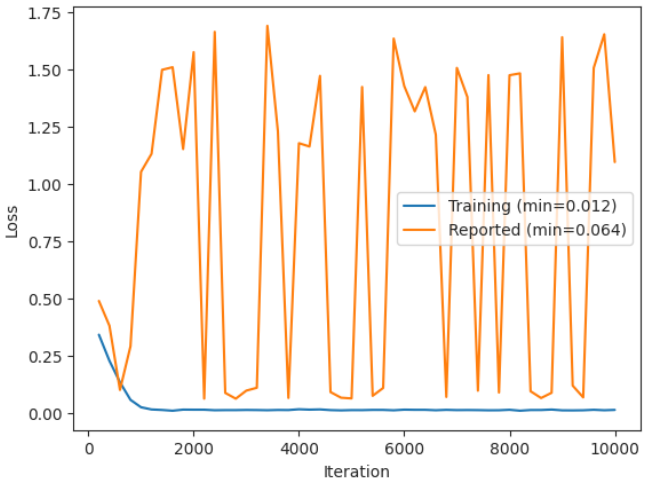
Cons:

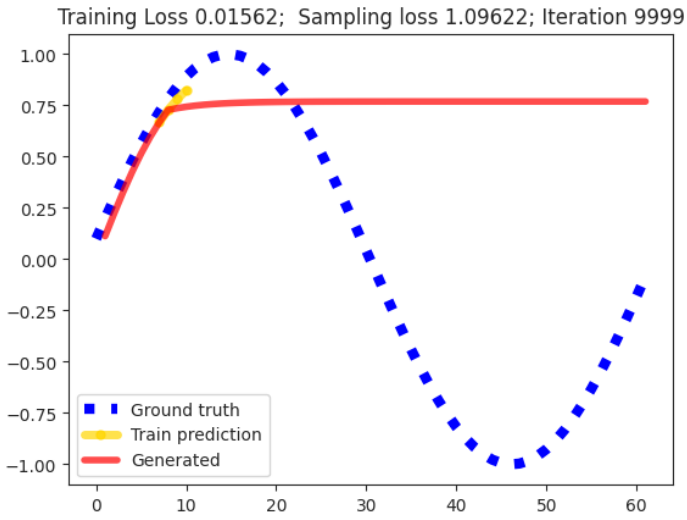
1. **Poor Generalization**: At test time, the model must rely on its own predictions, which can lead to compounding errors since it hasn't been trained to handle its own outputs.
2. **Overreliance on Ground Truth**: The model struggles with the transition from ground truth inputs during training to self-generated inputs during evaluation, leading to divergence over longer sequences.

3. **Limited Long-Term Dependency Learning**: Teacher forcing can mask issues related to the model's ability to capture long-term dependencies, as the errors during training are artificially suppressed.

Teacher forcing helps in faster and more stable training but often leads to poor generalization when the model is evaluated without ground truth inputs.

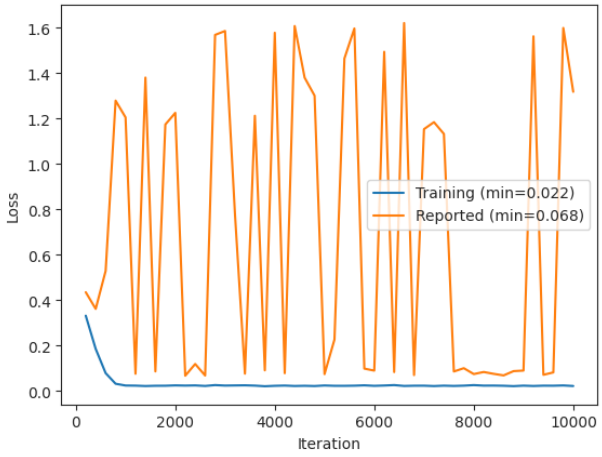
Teacher forcing probability = 0.5 UNROLL_LENGTH = 5

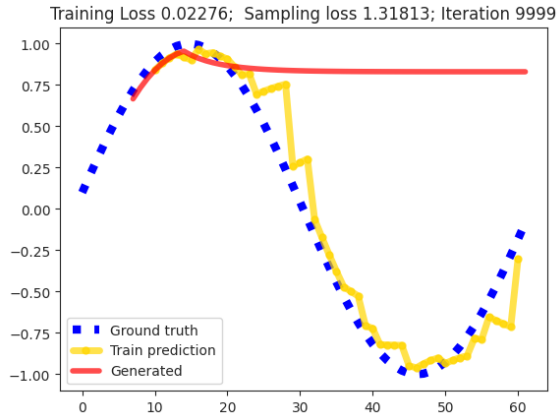




With **teacher forcing set to 0.5** and a very small **unroll length of 5**, the model achieves low training loss, indicating effective short-term learning. The short unroll length limits the context available to the model, preventing it from capturing broader patterns in the sine wave. This configuration highlights that while the model can perform well for short sequences, it fails to generalize effectively when tasked with generating longer ones.

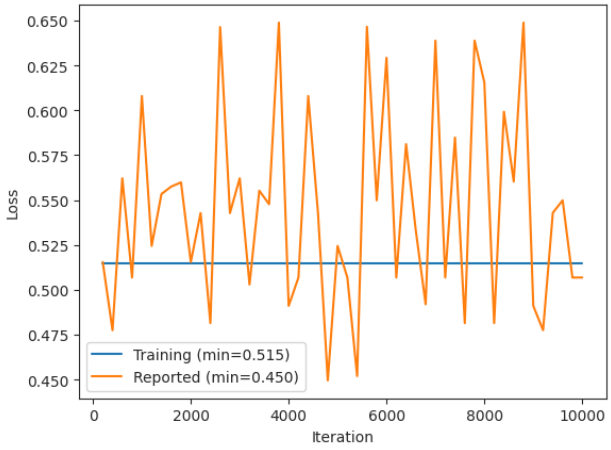
Teacher forcing probability = 0.5 UNROLL_LENGTH = 52

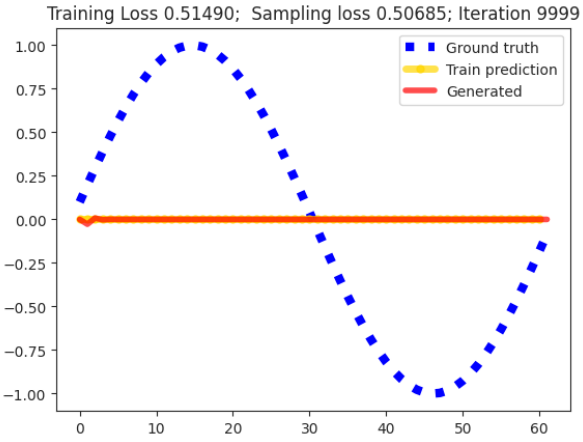




With **teacher forcing set to 0.5** and a **large unroll length of 52**, the model achieves low training loss, indicating that it can learn well over longer contexts during training. However, the generated sequence still diverges significantly after the warm start. The long unroll length allows the model to capture longer-term dependencies during training, but it struggles to generalize at evaluation time when it has to rely on its own predictions. This divergence is caused by compounding errors, as the model is not fully equipped to handle the transition from ground truth to self-predictions over extended sequences.

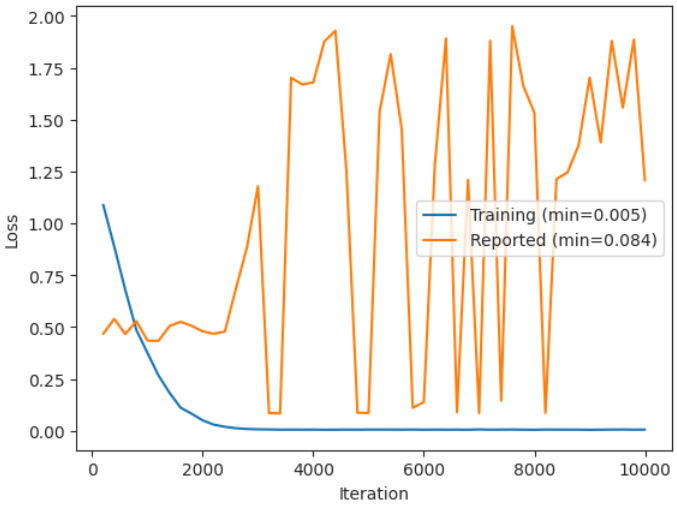
Teacher forcing probability = 0.0 UNROLL_LENGTH = 62 WARM START = 2

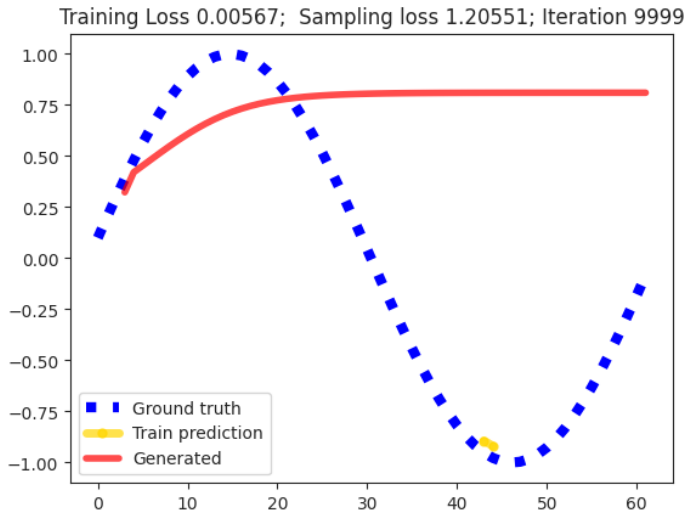




With **teacher forcing probability = 0.0**, **unroll length = 62**, and **warm start = 2**, the model struggles to learn effectively. Training loss remains relatively high and the generated sequence completely fails to track the ground truth, producing flat predictions. This indicates that without teacher forcing, the model becomes heavily reliant on its own predictions during training, leading to compounding errors and poor performance. The short warm start (2 timesteps) during evaluation does little to mitigate this issue, as the model quickly diverges from the ground truth after transitioning to its own predictions. This highlights the importance of balanced teacher forcing, especially for longer sequences.

Teacher forcing probability = 1 UNROLL_LENGTH = 3 WARM START = 2





With **teacher forcing probability = 1**, **unroll length = 3**, and **warm start = 2**, the model achieves very low training loss, indicating that it learns effectively over the short unroll length during training. However, the generated sequence diverges significantly after the warm start, as the model relies entirely on its predictions for longer sequences. The short unroll length limits the context the model can learn, and full teacher forcing prevents it from handling its own prediction errors, leading to compounding errors during evaluation. This configuration highlights the challenge of using a short unroll length and full teacher forcing, as the model cannot generalize well for long-term sequences.

In which setup is the model struggling to learn?

The model struggles the most in the setup with:

- **Teacher forcing probability = 0.0**, **unroll length = 62**, and **warm start = 2**.

In this configuration, the model relies entirely on its own predictions during training, leading to significant errors. Without teacher forcing, the model does not receive guidance from ground truth and fails to learn effectively, producing flat predictions even during training.

How does warm starting affect test time prediction? Why?

Warm starting helps the model by providing ground truth inputs for a fixed number of initial timesteps during evaluation. This can stabilize predictions at the beginning of the sequence and prevent immediate divergence. However:

- If the warm start period is too short, the model may quickly transition to self-generated inputs and deviate from the ground truth, as errors propagate and compound.

- The effectiveness of warm starting depends on how well the model handles its own predictions, which is heavily influenced by the training setup.

Warm starting provides a smoother transition from ground truth to model-generated inputs, but it cannot fully compensate for poor generalization or the inability to handle long-term dependencies.

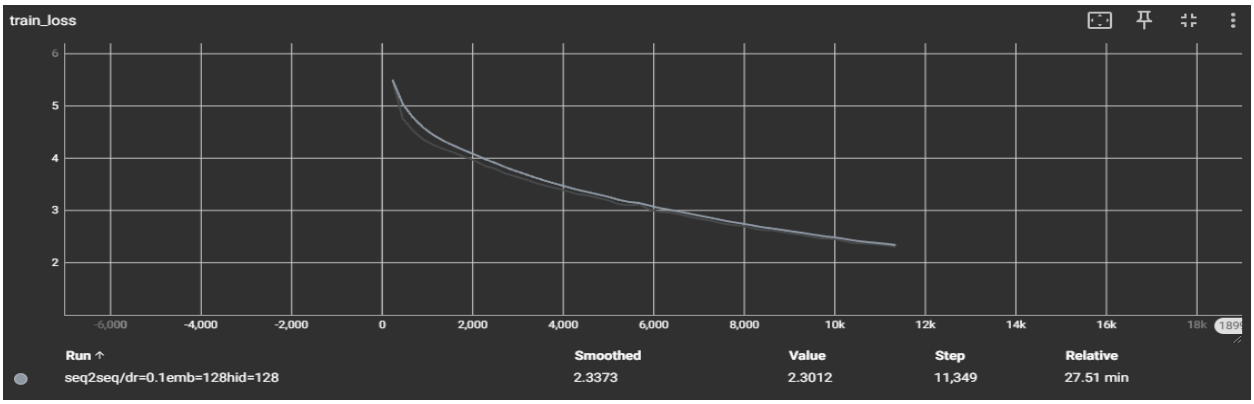
What happens if the structure of interest is much longer than the unroll length?

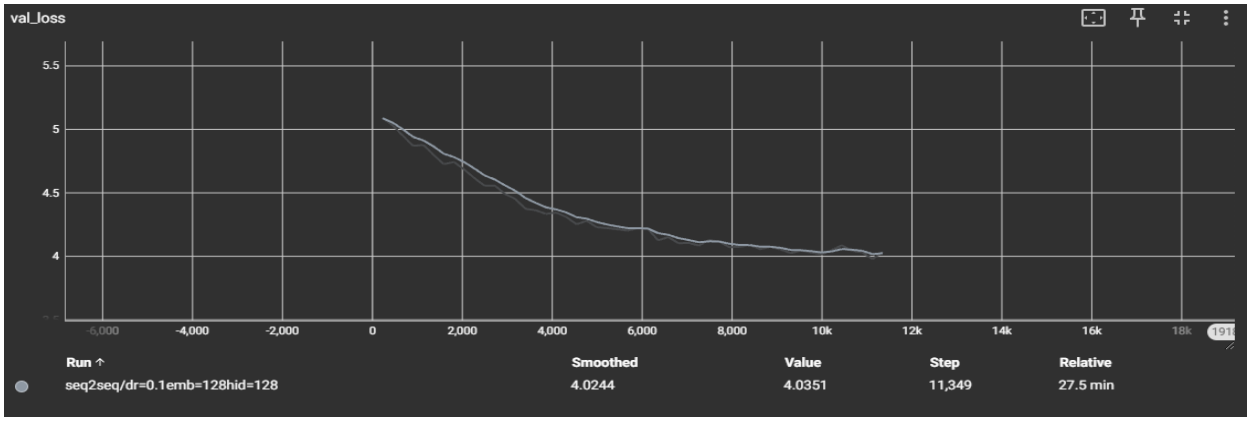
When the structure of interest is much longer than the unroll length:

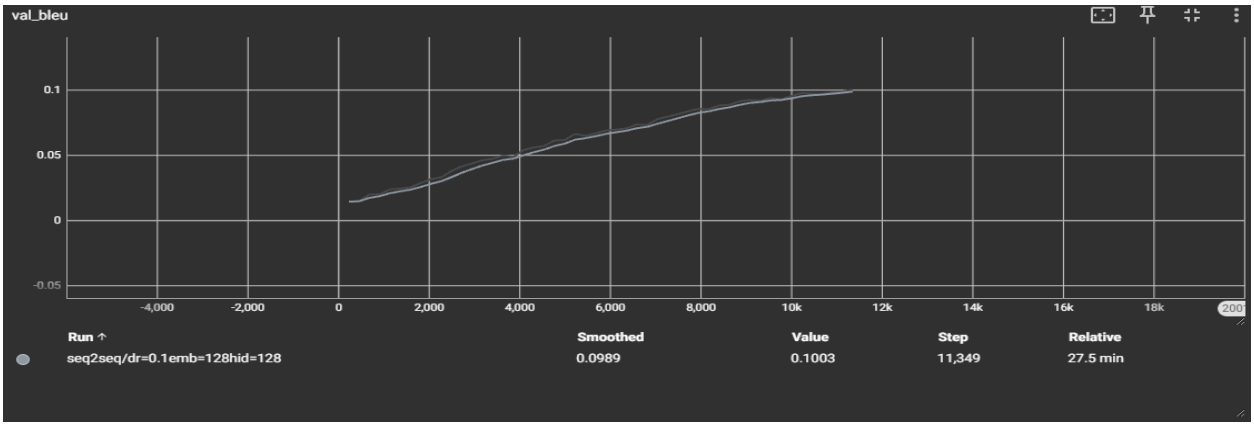
- **Limited Context**: The model cannot learn dependencies beyond the length of the unrolled sequence, leading to poor performance on tasks requiring long-term memory.
- **Error Accumulation**: The model struggles to connect information across timesteps that exceed the unroll length, causing compounding errors in predictions for longer sequences.
- **Overfitting to Short Patterns**: The model may overfit to short-term patterns within the unroll length, failing to generalize to the global structure of the sequence.

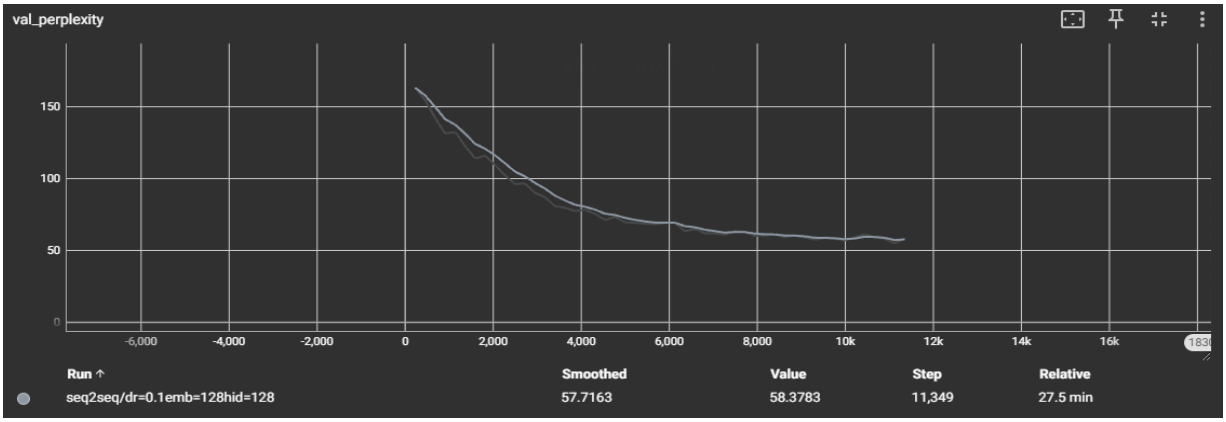
Task 2

1. Epochs = 50
Batch size = 128
Dropout = 0.1
Hidden size = 128
Embedding size = 128
Lr = 1e-3





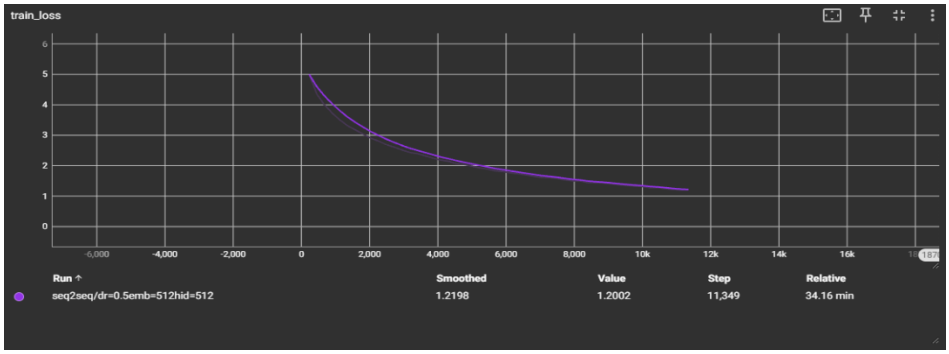


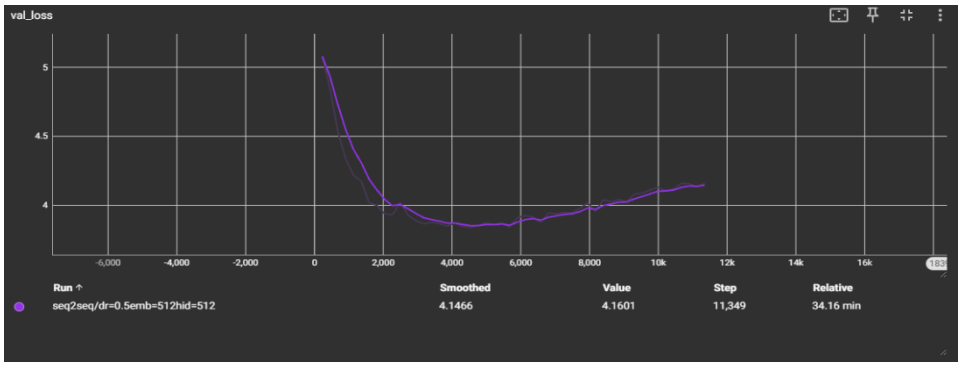


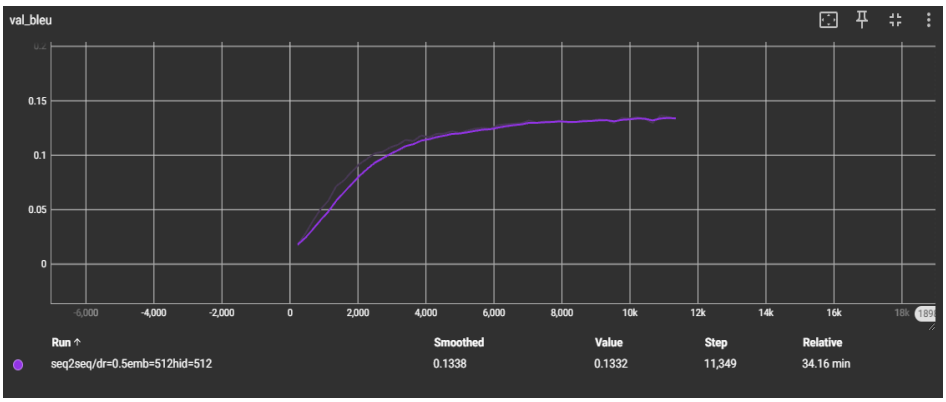
This configuration shows steady learning, with decreasing training and validation loss(train_loss=2.33, val_loss=4.03) and improving perplexity (~58). However, the BLEU score (~0.1) is low, indicating poor translation accuracy. The hidden and embedding size of 128 might be insufficient for this task.

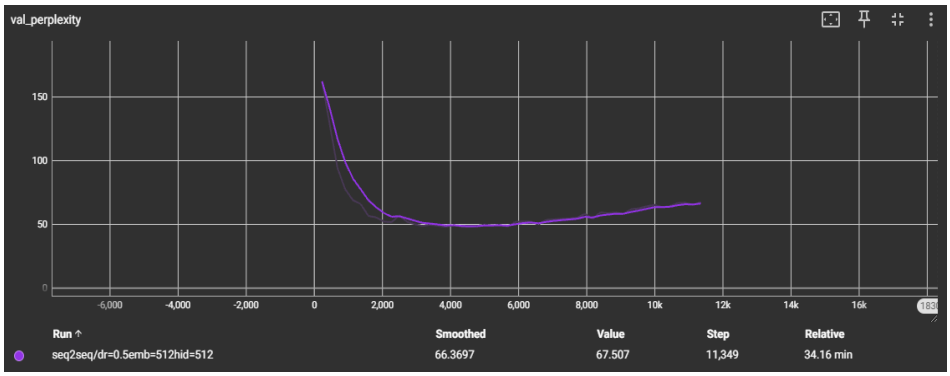
2.

Epochs = 50
Batch size = 128
Dropout = 0.5
Hidden size = 512
Embedding size = 512
Lr = 1e-3



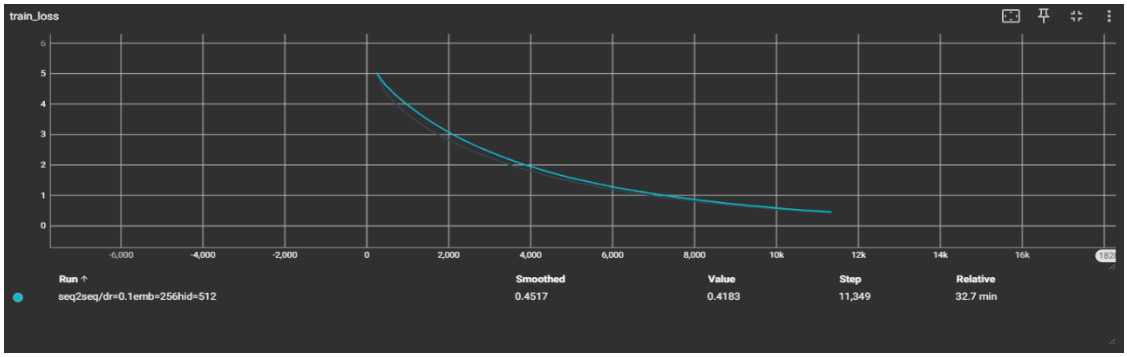


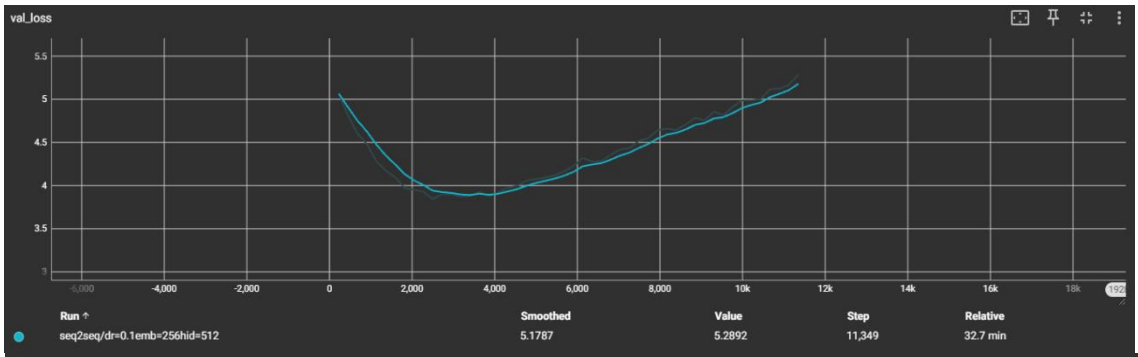


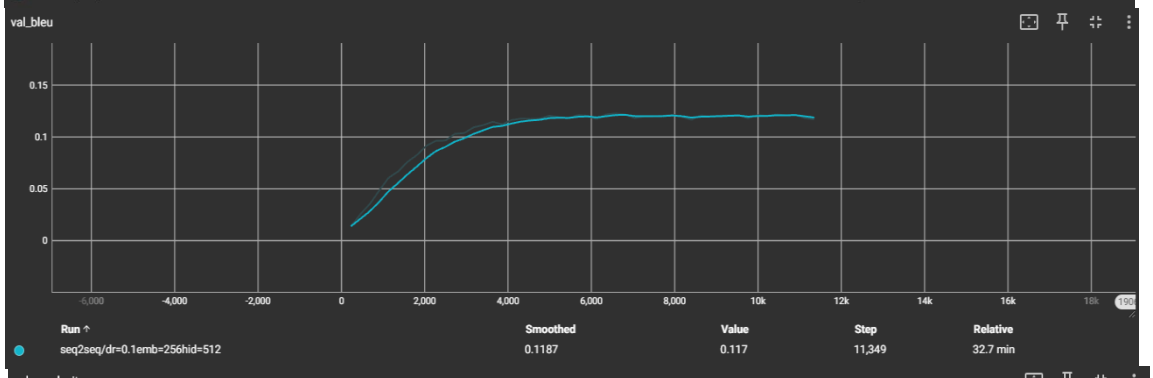


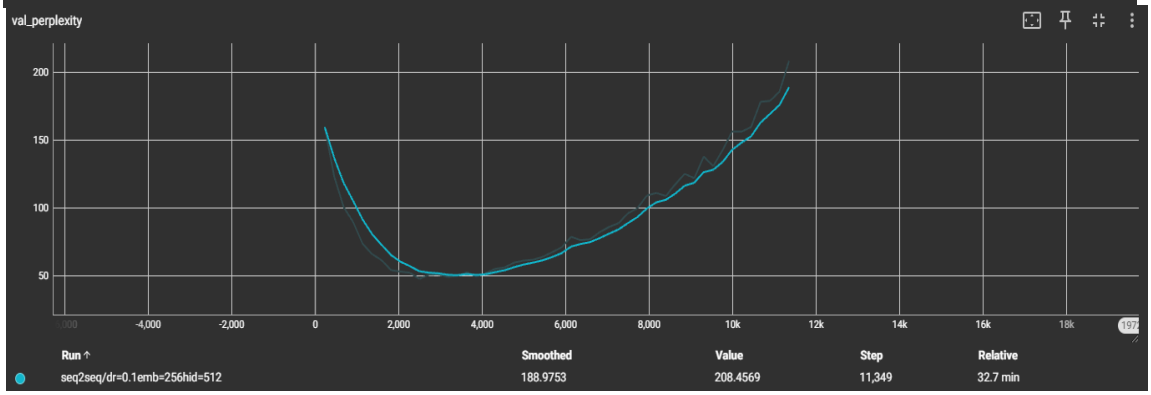
With larger hidden and embedding sizes (512) and a higher dropout of 0.5, this configuration showed improved learning, with training loss dropping to ~1.2 and BLEU score improving to ~0.13. Before overfitting set in, the validation loss reached a low of 3.84 and perplexity reached 48.04, indicating stronger generalization at that point. However, as training continued, the validation loss increased, signaling overfitting and limiting the model's performance.

3. Epochs = 50
Batch size = 128
Dropout = 0.1
Hidden size = 512
Embedding size = 256
Lr = 1e-3



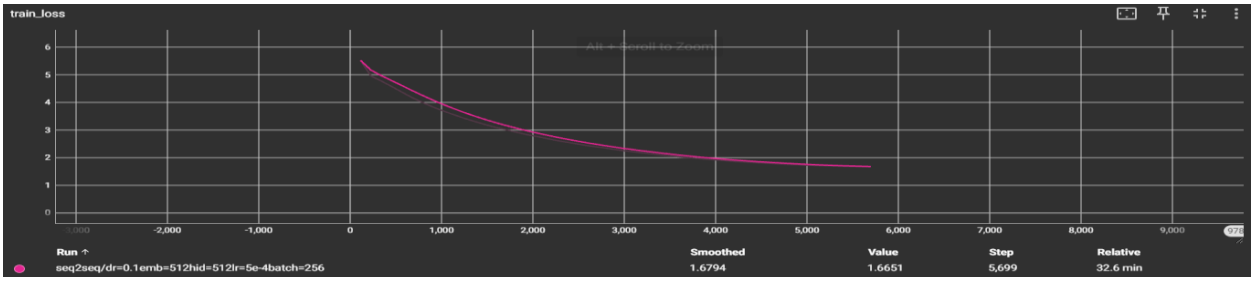


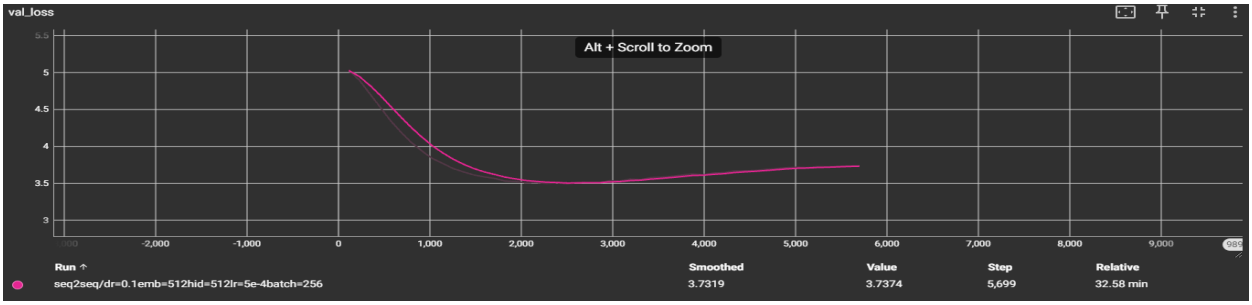


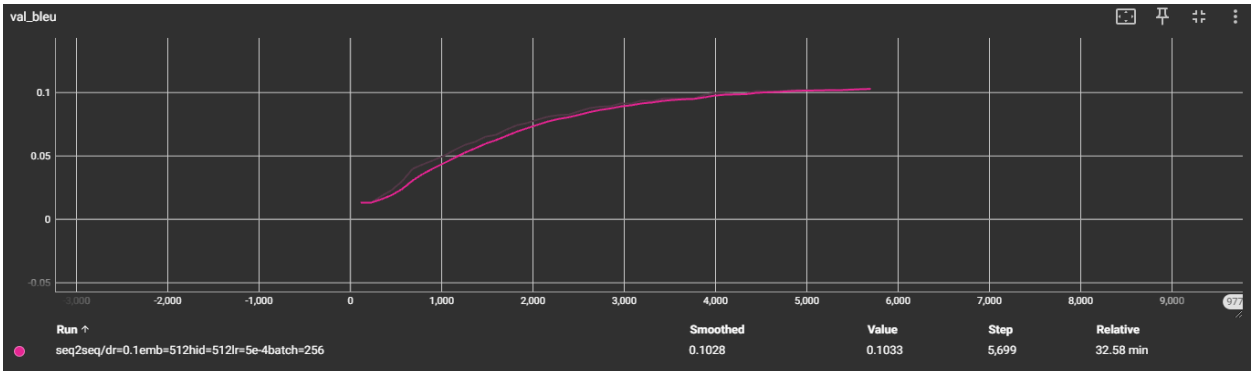


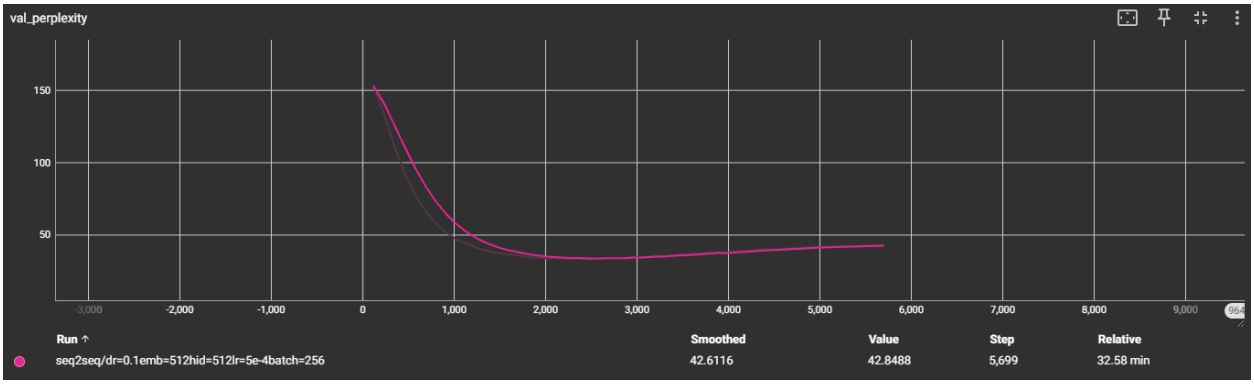
This configuration, with a hidden size of 512, embedding size of 256, and dropout of 0.1, showed strong initial learning but faced significant overfitting. The validation loss dropped to a minimum of **3.87** and perplexity to **50.46** before increasing, indicating overfitting as training continued. The BLEU score stabilized around **0.12**, suggesting moderate translation accuracy. The smaller embedding size compared to the hidden size might have limited performance, and the low dropout likely contributed to overfitting.

4. Epochs = 50
Batch size = 256
Dropout = 0.1
Hidden size = 512
Embedding size = 512
Lr = 5e-4
Linear decay scheduler



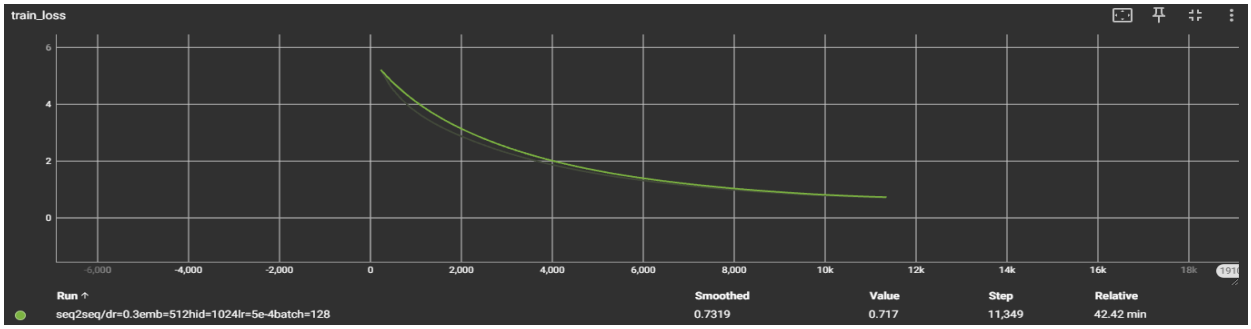


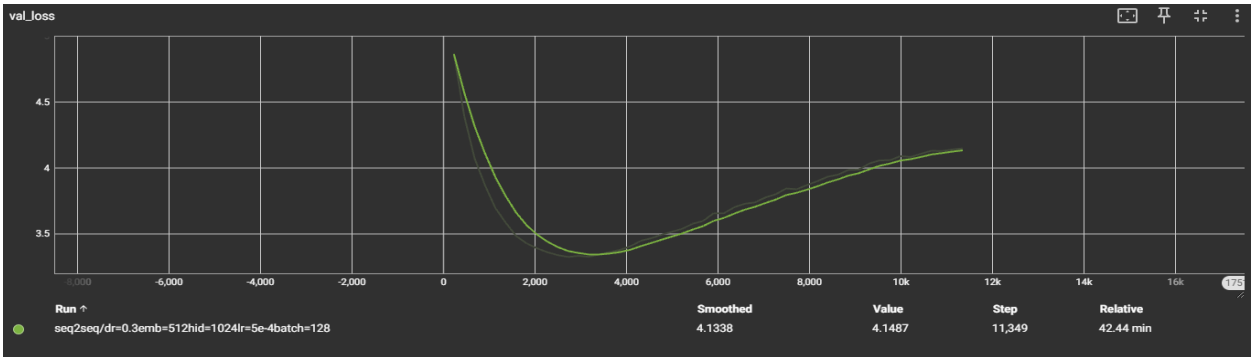


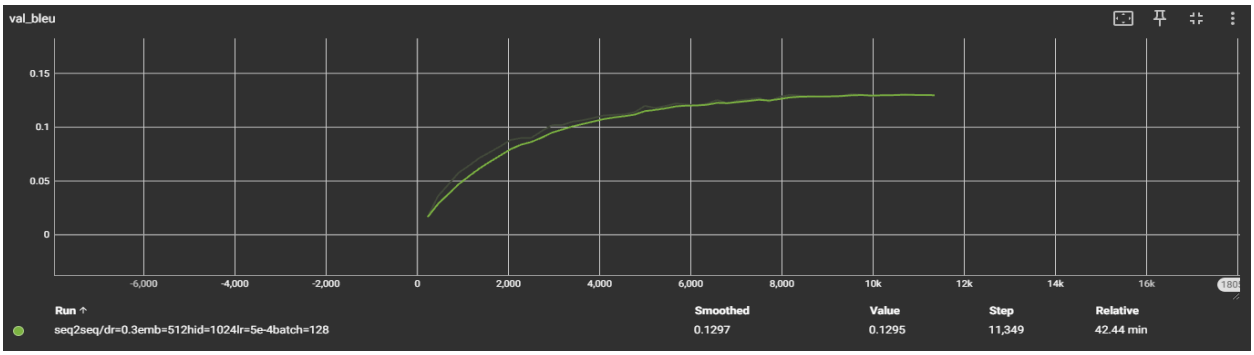


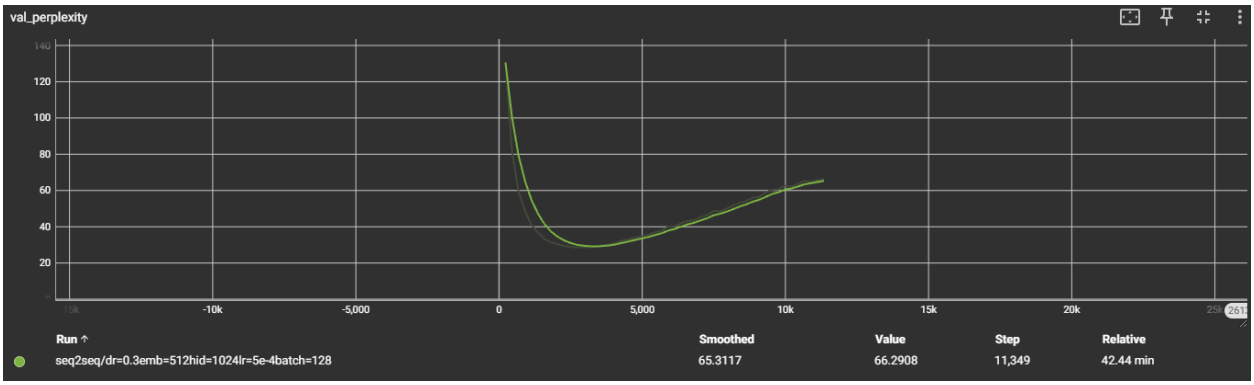
This configuration, with larger batch size (256), hidden and embedding sizes of 512, and a linear learning rate decay, showed steady progress with minimal overfitting. Validation loss reached **3.50** and perplexity dropped to **33.77**, indicating significant improvement. However, the BLEU score (~0.103) was relatively low, suggesting limited translation accuracy.

5. Epochs = 50
Batch size = 128
Dropout = 0.3
Hidden size = 1024
Embedding size = 512
Lr = 5e-4
Linear decay scheduler



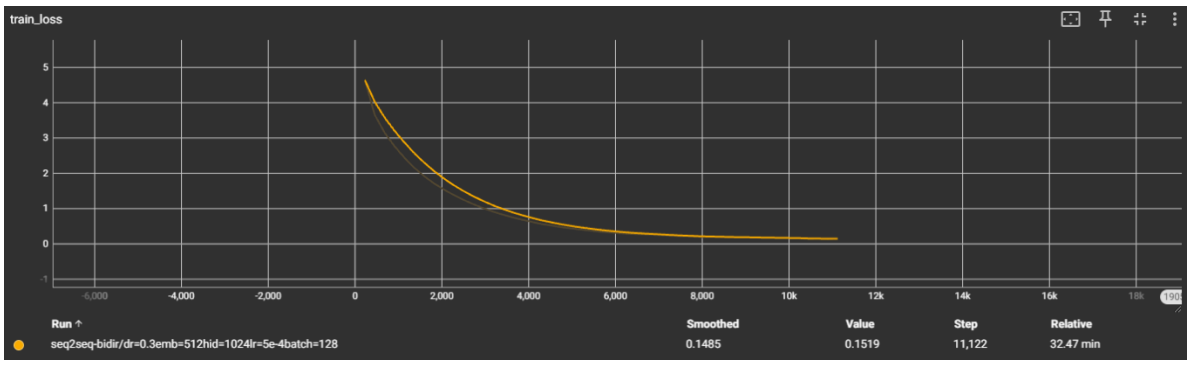


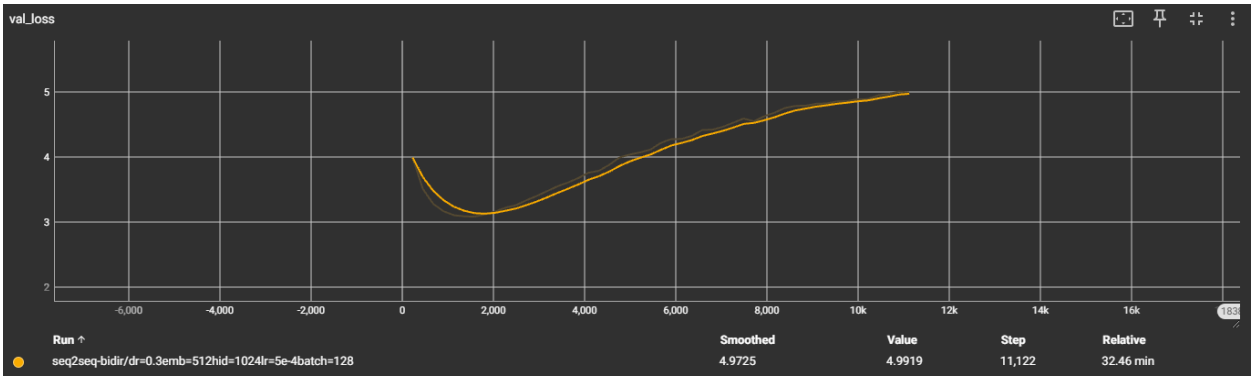


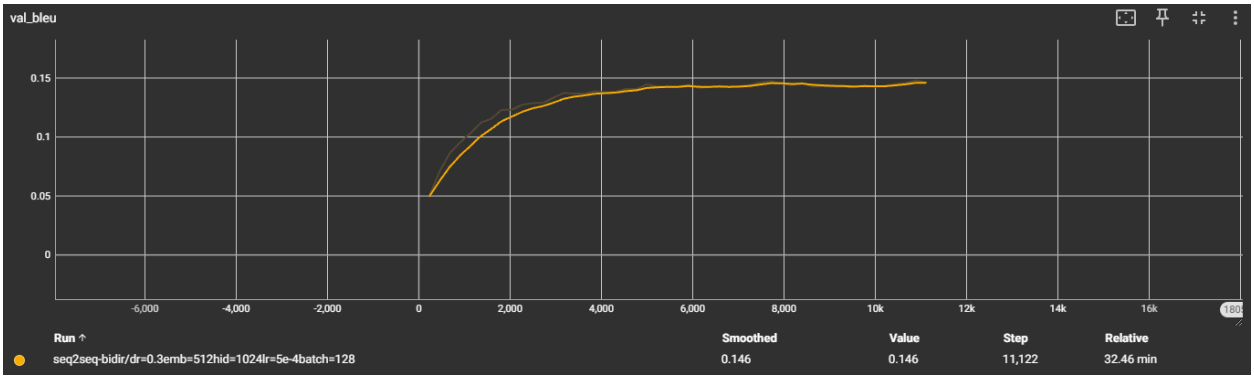


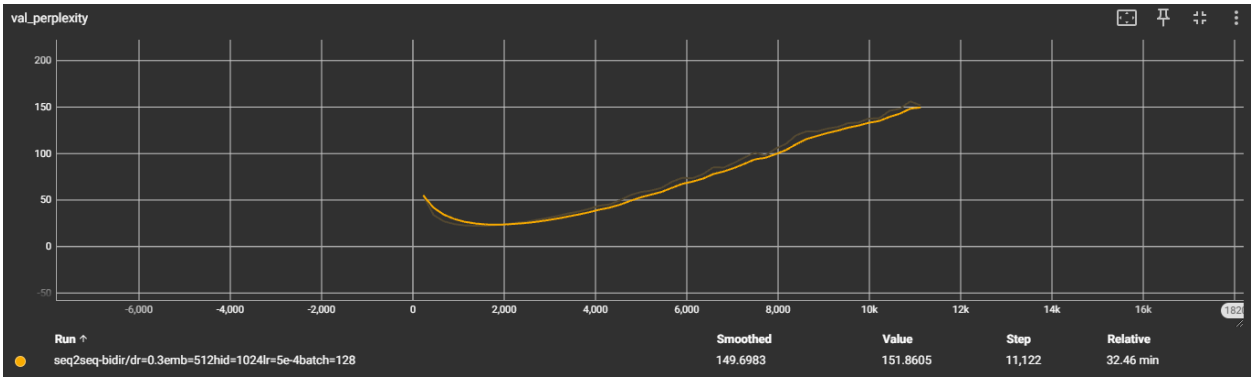
This configuration, with a higher hidden size of 1024, embedding size of 512, and dropout of 0.3, demonstrated effective learning but also overfitting. Validation loss initially dropped to **3.45** and perplexity to **29.16**, but both increased as training continued, indicating overfitting. The BLEU score stabilized at ~0.13. The larger hidden size helped capture complex patterns, while the dropout reduced overfitting to some extent.

6. Epochs = 50
Batch size = 128
Dropout = 0.3
Hidden size = 1024
Embedding size = 512
Lr = 5e-4
BiDirectional LSTM Encoder





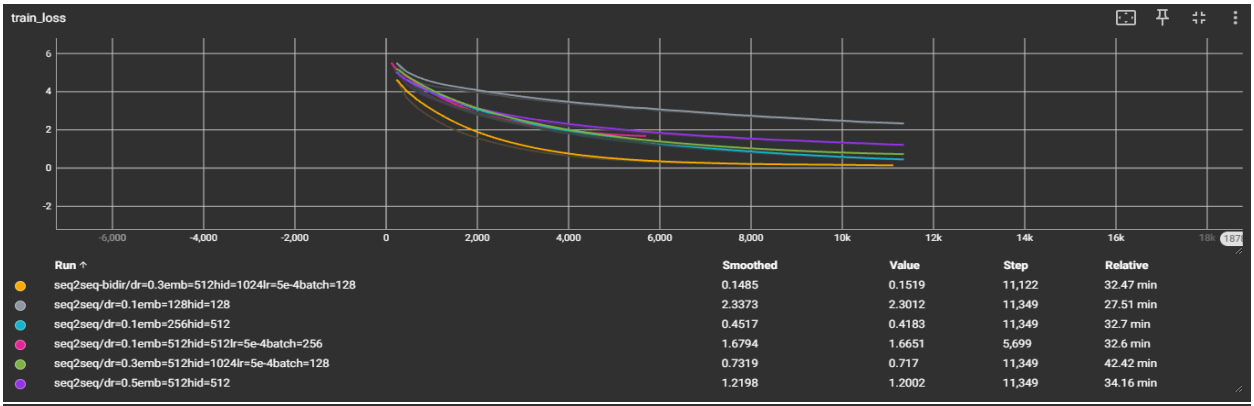




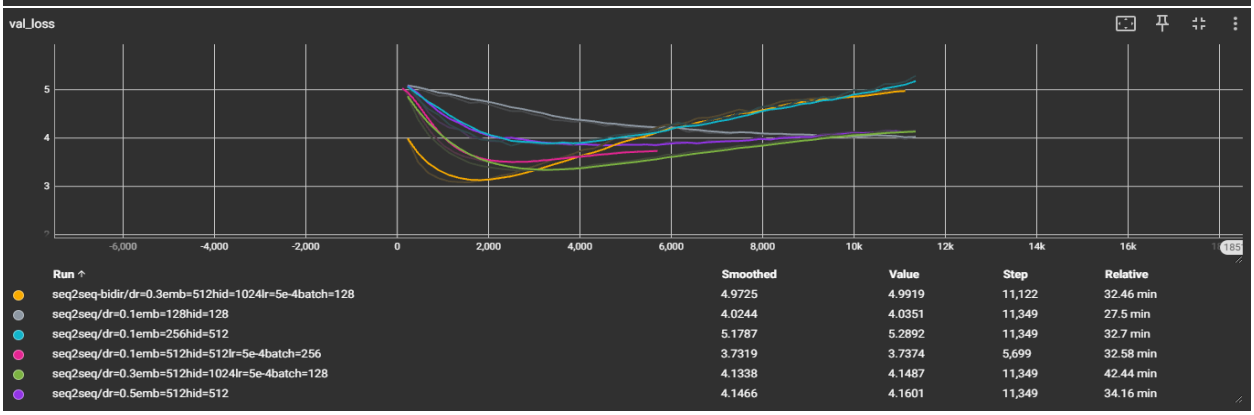
With a bidirectional LSTM encoder, the training loss decreased steadily, indicating effective learning. However, the validation loss initially dropped but started increasing, signaling overfitting. Before overfitting began, the validation loss reached a minimum of **3.08**, and the perplexity dropped to **23.48**. BLEU score plateaued at **0.146**, showing minor gains compared to the unidirectional model.

The bidirectional encoder leveraged both forward and backward contexts, improving sequence representation. However, the increased complexity likely required better regularization or more data to avoid overfitting.

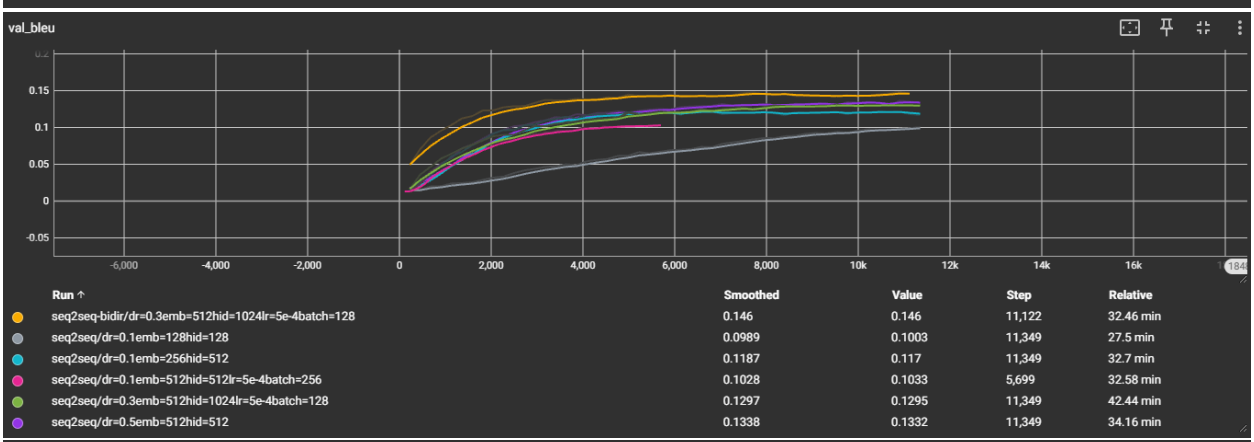
Overall comparison



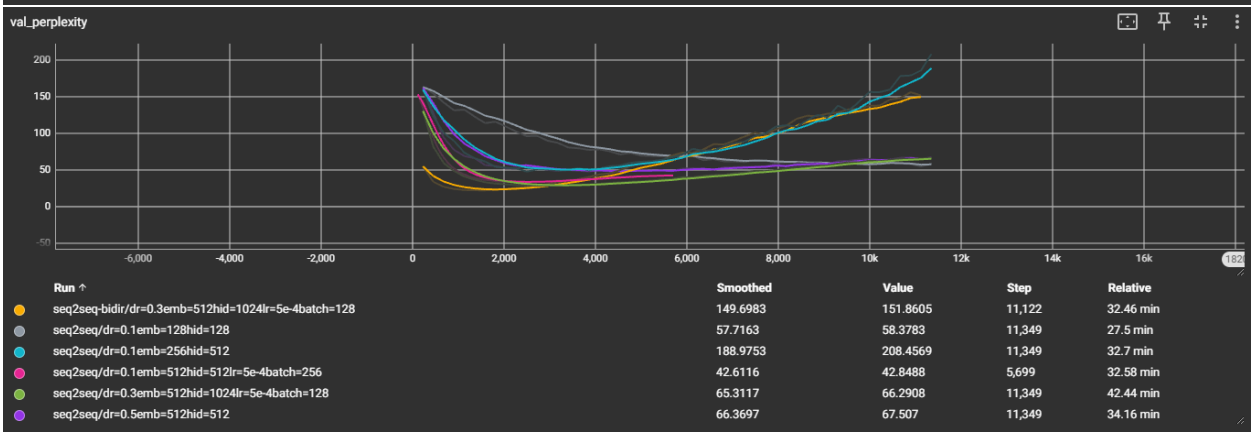
Run	Smoothed	Value	Step	Relative
seq2seq-bidir/dr=0.3emb=512hid=1024lr=5e-4batch=128	0.1485	0.1519	11,122	32.47 min
seq2seq/dr=0.1emb=128hid=128	2.3373	2.3012	11,349	27.51 min
seq2seq/dr=0.1emb=256hid=512	0.4517	0.4183	11,349	32.7 min
seq2seq/dr=0.1emb=512hid=512lr=5e-4batch=256	1.6794	1.6651	5,699	32.6 min
seq2seq/dr=0.3emb=512hid=1024lr=5e-4batch=128	0.7319	0.717	11,349	42.42 min
seq2seq/dr=0.5emb=512hid=512	1.2198	1.2002	11,349	34.16 min



Run	Smoothed	Value	Step	Relative
seq2seq-bidir/dr=0.3emb=512hid=1024lr=5e-4batch=128	4.9725	4.9919	11,122	32.46 min
seq2seq/dr=0.1emb=128hid=128	4.0244	4.0351	11,349	27.5 min
seq2seq/dr=0.1emb=256hid=512	5.1787	5.2892	11,349	32.7 min
seq2seq/dr=0.1emb=512hid=512lr=5e-4batch=256	3.7319	3.7374	5,699	32.58 min
seq2seq/dr=0.3emb=512hid=1024lr=5e-4batch=128	4.1338	4.1487	11,349	42.44 min
seq2seq/dr=0.5emb=512hid=512	4.1466	4.1601	11,349	34.16 min



Run	Smoothed	Value	Step	Relative
seq2seq-bidir/dr=0.3emb=512hid=1024lr=5e-4batch=128	0.146	0.146	11,122	32.46 min
seq2seq/dr=0.1emb=128hid=128	0.0989	0.1003	11,349	27.5 min
seq2seq/dr=0.1emb=256hid=512	0.1187	0.117	11,349	32.7 min
seq2seq/dr=0.1emb=512hid=512lr=5e-4batch=256	0.1028	0.1033	5,699	32.58 min
seq2seq/dr=0.3emb=512hid=1024lr=5e-4batch=128	0.1297	0.1295	11,349	42.44 min
seq2seq/dr=0.5emb=512hid=512	0.1338	0.1332	11,349	34.16 min



Run	Smoothed	Value	Step	Relative
seq2seq-bidir/dr=0.3emb=512hid=1024lr=5e-4batch=128	149.6983	151.8605	11,122	32.46 min
seq2seq/dr=0.1emb=128hid=128	57.7163	58.3783	11,349	27.5 min
seq2seq/dr=0.1emb=256hid=512	188.9753	208.4569	11,349	32.7 min
seq2seq/dr=0.1emb=512hid=512lr=5e-4batch=256	42.6116	42.8488	5,699	32.58 min
seq2seq/dr=0.3emb=512hid=1024lr=5e-4batch=128	65.3117	66.2908	11,349	42.44 min
seq2seq/dr=0.5emb=512hid=512	66.3697	67.507	11,349	34.16 min

The configuration with bidirectional LSTM encoder achieved the lowest final training loss and validation loss before overfitting began, indicating strong learning capacity.
Configurations with **higher dropout (0.5)** or **smaller sizes** (hidden size, embedding size) showed reduced performance.
Generally, configurations with a lower learning rate 5e-4 performed better across metrics.
The bidirectional LSTM encoder setup achieved the highest BLEU score among all configurations, showing superior translation quality. It also achived lower perplexity before overfitting started.
Overfitting was evident in most configurations after achieving minimum validation loss, highlighting the need for early stopping or regularization techniques like higher dropout or weight decay.