Contents

1 Loss Interpretation Quick Reference Card

For: SKOL RNN 3-Class Text Classification (Nomenclature, Description, Misc)

1.1 Loss Value Interpretation

Loss	Meaning	Confidence	Action
0.0 - 0.1	Perfect	Extremely confident	⚠️ Check for overfitting
0.1 - 0.4	Excellent	Very confident	✅ Great model!
0.4 - 0.7	Good	Reasonably confident	✅ Good for production
0.7 - 1.1	Fair	Uncertain	□ Train more
1.1 - 1.5	Poor	Close to random	⚠️ Check model/data

Loss	Meaning	Confidence	Action
1.5+	Very Poor	Confidently wrong	❌ Something is broken

Random Baseline for 3 Classes: Loss ≈ 1.099

1.2 Quick Examples

1.2.1 Confident & Correct (Best)

```
True: Nomenclature
Predicted: [0.95, 0.03, 0.02] → 95% confident in Nomenclature
Loss = -log(0.95) = 0.05 ✅ Very low!
```

1.2.2 Uncertain & Correct (OK)

```
True: Nomenclature
Predicted: [0.50, 0.30, 0.20] → 50% confident
Loss = -log(0.50) = 0.69 ⚠️ Higher loss
```

1.2.3 Confident & Wrong (Bad)

```
True: Nomenclature
Predicted: [0.05, 0.90, 0.05] → 90% confident in wrong class!
Loss = -log(0.05) = 3.0 ❌ Very high loss
```

1.3 Training Health Checklist

1.3.1 ✅ Healthy Training

- ☐ Loss decreases each epoch
- ☐ Accuracy increases each epoch
- ☐ Final loss < 0.5
- ☐ Final accuracy > 0.75
- ☐ Training and validation loss both decrease

1.3.2 ⚠️ Warning Signs

Overfitting: - [ ] Training loss keeps decreasing - [ ] Validation loss starts increasing - [ ] Training accuracy » validation accuracy

Underfitting: - [ ] Loss stuck near 1.1 (random baseline) - [ ] Minimal improvement per epoch - [ ] Both train and val loss remain high

Instability: - [ ] Loss jumps up and down - [ ] Large variations between epochs - [ ] Loss occasionally increases

---

1.4 Target Metrics (Your Model)

Metric	Minimum	Good	Excellent
Loss	< 0.7	< 0.5	< 0.3
Accuracy	> 0.70	> 0.80	> 0.90
F1 Score	> 0.65	> 0.75	> 0.85

Your Current Performance: 79.9% accuracy ✅ (Good!)

1.5 When to Stop Training

1.5.1 Stop Now

- Final loss < 0.4 and accuracy > 0.80
- Loss hasn't improved for 2+ epochs
- Validation loss increasing
- Accuracy reached target

1.5.2 Keep Training

- Loss still decreasing steadily
- Accuracy still improving
- Final loss > 0.5
- No signs of overfitting

1.6 Loss vs Accuracy Decision Matrix

Loss	Accuracy	Interpretation	Action
0.4	0.80	✅ Confident, correct	Use model
0.8	0.80	⚠️ Uncertain, correct	Train more

Loss	Accuracy	Interpretation	Action
0.2	0.95	⚠️ May be overfitting	Check validation
0.5	0.50	❌ Confident, wrong	Fix model/data
1.0	0.40	❌ Random guessing	Restart training

1.7 What Loss Tells You

Beyond "Lower is Better":

1. **Confidence Level**: How sure the model is
   - Loss 0.05 = 95% confident
   - Loss 0.69 = 50% confident
   - Loss 3.0 = 5% confident (but wrong class!)
2. **Learning Progress**: Fine-grained tracking
   - Accuracy: 0.79 → 0.80 (small change)
   - Loss: 0.65 → 0.48 (significant improvement in confidence)
3. **Overfitting Detection**: Early warning
   - Training loss: 0.2 ✅
   - Validation loss: 0.8 ⚠️ Overfitting!
4. **Per-Class Performance**:
   - Nomenclature loss: 0.35 ✅ Easy class
   - Description loss: 0.42 ✅ Easy class
   - Misc loss: 0.85 ⚠️ Hard class

1.8 Common Questions

Q: I have 80% accuracy, is that good? A: Check the loss! - Loss < 0.5 → Yes, confident and correct ✅ - Loss > 0.8 → Not really, uncertain predictions ⚠️

Q: Loss went from 0.5 to 0.55, should I worry? A: Minor fluctuations are normal. Worry if it keeps increasing for 2+ epochs.

Q: Can I have low loss but low accuracy? A: Rare, but possible if: - Evaluation is broken - Class labels are wrong - Model outputs are not being interpreted correctly

Q: My loss is 0.3 after 2 epochs, should I keep training? A: Probably overfitting soon. Check validation loss!

1.9 At a Glance

Excellent Model:

```
Epoch 4/4 - loss: 0.35 - accuracy: 0.85
Test: loss: 0.38 - accuracy: 0.82
→ ✅ Deploy this model!
```

Good Model (Your Case):

```
Epoch 4/4 - loss: 0.48 - accuracy: 0.80
Test: accuracy: 0.7990
→ ✅ Ready for production
```

Needs Work:

```
Epoch 4/4 - loss: 0.85 - accuracy: 0.60
Test: accuracy: 0.55
→ ⚠️ Train longer or adjust model
```

Overfitting:

```
Train: loss: 0.15 - accuracy: 0.95
Test:  loss: 0.75 - accuracy: 0.70
→ ⚠️ Add dropout, reduce epochs
```

Keep this card handy during training!

For detailed explanation, see: understanding_rnn_training_metrics.md