Feedback-Augmented Loss Function: Worked Example

Batch Size: 2 | Sequence Length: 3 | Vocab Size: 4

Logits:

```
[[[2.  1.  0.1 0. ]
  [1.5 2.2 0.5 0.3]
  [0.3 0.2 2.  1.7]]

 [[0.1 2.  1.  0.5]
  [2.  0.1 0.1 2.1]
  [0.2 1.2 1.8 1.5]]]
```

Labels:

```
[[0 1 2]
 [1 3 2]]
```

Feedback: solution_score=[0.9 0.4], reasoning_score=[0.8 0.2], is_correct=[True False]

Step 1: Cross-Entropy per Sample:

Cross-entropy per token:

```
[[-1.49753926 -1.59631951 -1.26340996]
 [-1.44578263 -1.32273781 -0.88710448]]
```

Cross-entropy per sample (mean):

```
[-1.45242291 -1.21854164]
```

Step 2: Reward Loss per Sample:

Mean log-prob of correct tokens: [1.45242291 1.21854164]

Feedback score: [0.85 0.3]

Reward loss: [-1.23455947 -0.36556249]

Step 3: Penalty Loss per Sample:

Mean max softmax prob per sample: [0.54352635 0.47851496]

Penalty loss: [0.   0.33496047]

Step 4: Total Loss per Sample & Batch Mean:

Total loss per sample: [-2.68698238 -1.24914366]

Batch mean loss: -1.9681

Mathematical Formulation:

L_aug = L_CE + lambda1 * (-feedback_score * mean_log_prob) + lambda2 * ((1-feedback_score) * mean_max_prob * (1-is_correct))

Where L_CE is mean cross-entropy per sample, mean_log_prob is mean log-prob of correct tokens, mean_max_prob is average max-prob per sample, feedback_score is in [0,1], is_correct is boolean.