Some color variation, needing treatment in dark mode themes:
A black number on mint green background: 33.35

1 Tabular

Dataset	Att.	h	d_m	# Param.	Epoch Time	Acc. (%)	Loss	Val Acc. (%)	Val Loss
Gray 12									
Gray 6									
Gray 0									
My Green									
White smoke									
Timberwolf									
IMDB	Stn.	4	32	4,224 (4)	0.315 (4)	95.70 (4)	0.086 (3)	77.62 (4)	0.474 (4)
	Opt.	4	32	3,168 (2)	0.305 (3)	96.31 (3)	0.095 (4)	77.85 (2)	0.472 (2)
	Eff.	4	32	**2,112 (1)**	**0.280 (1)**	96.41 (2)	**0.064 (1)**	77.77 (3)	**0.468 (1)**
	Sup.	4	32	3,168 (2)	0.299 (2)	**97.45 (1)**	0.070 (2)	**78.34 (1)**	0.472 (2)
Amazon	Stn.	4	128	66,048 (4)	66.97 (4)	88.49 (3)	0.25 (3)	65.55 (4)	0.77 (3)
	Opt.	4	128	49,536 (3)	61.75 (3)	**89.56 (1)**	**0.23 (1)**	65.67 (2)	0.75 (2)
	Eff.	4	128	**33,024 (1)**	**56.44 (1)**	86.63 (4)	0.29 (4)	65.58 (3)	0.77 (3)
	Sup.	4	128	42,336 (2)	59.86 (2)	88.56 (2)	0.24 (2)	**68.10 (1)**	**0.71 (1)**

2 Math

$$x + \sqrt{y} = 0$$

$x +$ y $= 0$ and black text blue text.

$$\kappa^\star := \inf_{w \in Y} \quad \langle T, w \rangle$$
$$A\,w - g \in C(X)_+ \tag{1}$$

2mm

3 Listings

```
1   component expr(a:32,b:32,c:32,d:32)->(out:32) {
2     cells {
3       add = std_add(32);   // 32-bit adder
4       mult = std_mult(32); // 32-bit multiplier
5       div = std_div(32);   // 32-bit divider
6     }
7     wires {
8       static<1> group do_add {
9         add.left = %[0:1] ? a;
10        add.right = %[0:1] ? b;
11        // do_add[done] = add.done;
12      }
```

```
13        static<3> group do_mult {
14          mult.left = %[0:3] ? add.out;
15          mult.right = c; // implicit %[0:3] guard
16          // do_mult[done] = mult.done;
17        }
18        group do_div {
19          div.go = 1'd1;
20          div.left = mult.out;
21          div.right = d;
22          do_div[done] = div.done;
23        }
24        out = div.out;
25      }
26      control {
27        seq { static seq { do_add; do_mult; }
28              do_div;
29        }
30      }
31    }%
```

Algorithm 1 Pytorch code

```python
class Matryoshka_CE_Loss(nn.Module):
    def __init__(self, relative_importance, **kwargs):
    super(Matryoshka_CE_Loss, self).__init__()
    self.criterion = nn.CrossEntropyLoss(**kwargs)
    self.relative_importance = relative_importance # usually
        set to all ones

  def forward(self, output, target):
    loss=0
    for i in range(len(output)):
      loss+= self.relative_importance[i] * self.criterion(
          output[i], target)
    return loss
```

Table C.5: Prompt to Choose Representative Landmarks for Each Cluster

You have been provided with a set of similar documents, each indexed by a number. Your task is to identify the most representative example from this cluster of documents. Please carefully analyze the given documents and select one document that best captures the common essence and characteristics of the samples. The selection should emphasize the representativeness and relevance of the chosen sample to the category, so that it can serve as a reference for labeling the entire cluster.

Documents in the cluster:

1-. {Document 1}......

2-. {Document 2}......

3-. {Document 3}......

4-. {Document 4}......

⋮

Please choose one document that could best serve as a reference for labeling the entire cluster, and return only the index number of your selection, in format such as [0], [1], etc.

Answer: