Algorithm 1: CTC Loss alpha computation

Data: $out_{m \times n}$ (result of softmax), where m=$\bar{W}_{padded}/4$,
w=$\bar{W}_{unpadded}/4$, n=$|\hat{A}|$,
label (encoded by alphabet),
bl=0 (blank index)

begin

- $Loss = 0$
- $l = len(label)$
- $L = 2 \times len(label) + 1$
- $T = w$
- $T_{padded} = m$
- $a = zeros(T_{padded}, L)$
- $a_0^0 = out_0^{bl}$
- $a_0^1 = out_0^{label_0}$
- $c = a_0^0 + a_0^1$
- **if** $c > 0$ **then**
  - $a_0^0 = \frac{a_0^0}{c}$
  - $a_0^1 = \frac{a_0^1}{c}$
- **else**
  - $c = 1.0$
- $Loss = Loss + log(c)$
- **for** $t := 1$ **to** T **do**
  - $start = \max(0, L - 2 \times (T - t))$
  - $end = \min(2 \times t + 2, L)$
  - **for** $s := start$ **to** L **do**
    - $i = \frac{s-1}{2}$
    - $red = a_{t-1}^s$
    - $blue = 0$
    - **if** $s > 0$ **then**
      - $blue = a_{t-1}^{s-1}$
    - **if** s ***mod*** $2 = 0$ **then**
      - $a_t^s = (red + blue) \times out_t^{bl}$
    - **else if** $s = 1$ ***or*** $label_i = label_{i-1}$ **then**
      - $a_t^s = (red + blue) \times out_t^{label_i}$
    - **else**
      - $orange = a_{t-1}^{s-2}$
      - $a_t^s = (red + blue + orange) \times out_t^{label_i}$
  - $c = \sum_{i=start}^{end} a_t^i$
  - **if** $c > 0$ **then**
    - **for** $i := start$ **to** end **do**
      - $a_t^i = \frac{a_t^i}{c}$
  - **else**
    - $c = 1.0$
  - $Loss = Loss + log(c)$
- **return** Loss, a

Algorithm 2: CTC Loss alpha computation

Data: $out_{m \times n}$ (result of softmax), where m=$\bar{W}_{padded}/4$,
w=$\bar{W}_{unpadded}/4$, n=$|\hat{A}|$,
label (encoded by alphabet),
bl=0 (blank index)

begin

- $l = len(label)$
- $L = 2 \times len(label) + 1$
- $T = w$
- $T_{padded} = m$
- $b = zeros(T_{padded}, L)$
- $b_{T-1}^{L-1} = out_{T-1}^{bl}$
- $b_{T-1}^{L-2} = out_{T-1}^{label_{l-1}}$
- $d = b_{T-1}^{L-1} + b_{T-1}^{L-2}$
- **if** $d > 0$ **then**
  - $b_{T-1}^{L-1} = \frac{b_{T-1}^{L-1}}{d}$
  - $b_{T-1}^{L-2} = \frac{b_{T-1}^{L-2}}{d}$
- **for** $t := T - 2$ **to** -1 **do**
  - $start = \max(0, L - 2 \times (T - t))$
  - $end = \min(2 \times t + 2, L)$
  - **for** $s := end - 1$ **to** -1 **do**
    - $i = \frac{s-1}{2}$
    - $red = b_{t+1}^{s}$
    - $blue = 0$
    - **if** $s > 0$ **then**
      - $blue = b_{t+1}^{s+1}$
    - **if** s ***mod*** $2 = 0$ **then**
      - $b_t^s = (red + blue) \times out_t^{bl}$
    - **else if** $s = L$ - 2 ***or*** $label_i = label_{i+1}$ **then**
      - $b_t^s = (red + blue) \times out_t^{label_i}$
    - **else**
      - $orange = b_{t+1}^{s+2}$
      - $b_t^s = (red + blue + orange) \times out_t^{label_i}$
  - $d = \sum_{i=start}^{end} d_t^i$
  - **if** $d > 0$ **then**
    - **for** $i := start$ **to** end **do**
      - $b_t^i = \frac{b_t^i}{d}$
- **return** b

Algorithm 3: CTC Loss gradient computation

Data: $out_{m \times n}$ (result of softmax), where m=$\bar{W}_{padded}/4$,
w=$\bar{W}_{unpadded}/4$, n=$|\hat{A}|$,
label (encoded by alphabet),
bl=0 (blank index),
a, b

begin

- $L = 2 \times len(label) + 1$
- $T = w$
- $T_{padded} = m$
- $softmaxgrad = zeros(T_{padded}, n)$
- $ab = a * b$
- **for** $s := 0$ **to** L **do**
  - **if** s ***mod*** $2 = 0$ **then**
    - **for** $t := 0$ **to** T **do**
      - $softmaxgrad_t^{bl} += ab_t^s$
      - $ab_t^s = \frac{ab_t^s}{out_t^{bl}}$
  - **else**
    - **for** $t := 0$ **to** T **do**
      - $i = \frac{s-1}{2}$
      - $softmaxgrad_t^i += ab_t^s$
      - $ab_t^s = \frac{ab_t^s}{out_t^i}$
- $absum = zeros(T)$
- **for** $t := 0$ **to** T **do**
  - $absum_t = \sum_{s=1}^{L} ab_t^s$
- **for** $t := 0$ **to** T **do**
  - **for** $i := 0$ **to** n **do**
    - $softmaxgrad_t^i = out_t^i - \frac{softmaxgrad_t^i}{out_t^i * absum_t}$
- **return** softmaxgrad