Details of our ASG algorithm

Ziyang Hu

April 10, 2019

Contents

1 The log semi-ring **2**
- 1.1 Arithmetic . . . 2
- 1.2 Calculus . . . 2
- 1.3 Generalization . . . 3

2 Sequence to sequence model **3**
- 2.1 The input tensor . . . 3
- 2.2 The output tensor . . . 3
- 2.3 The transition matrix . . . 3
- 2.4 Lattice and paths . . . 4
- 2.5 The fully-connected lattice . . . 5
- 2.6 The force-alignment lattice . . . 7
- 2.7 The Auto-Segmentation Criterion . . . 9

1 The log semi-ring

1.1 Arithmetic

On the extended real numbers $\bar{\mathbb{R}} = \mathbb{R} \cup \{-\infty\}$, define the operators $\oplus$ where

$$x \oplus y = \log(e^x + e^y) \tag{1}$$

which is usually called "log-sum-exp" in the literature, and $\otimes$ where

$$x \otimes y = x + y. \tag{2}$$

We can verify that these two operations are associative and commutative, and they satisfy the distributive law:

$$x \otimes (y \oplus z) = (x \otimes y) \oplus (x \otimes z). \tag{3}$$

We also define

$$\bar{0} = -\infty, \qquad \bar{1} = 0 \tag{4}$$

and they have the properties

$$\bar{0} \oplus x = x, \qquad \bar{0} \otimes x = \bar{0}, \qquad \bar{1} \otimes x = x. \tag{5}$$

The structure $\oplus, \otimes, \bar{0}, \bar{1}$ forms a commutative semi-ring (in fact a semi-field).

As we have associativity and commutativity at our disposal, we will define the big operators

$$\bigoplus_{i=0}^{N-1} x_i = x_0 \oplus x_1 \oplus \cdots \oplus x_{N-1} \tag{6}$$

and

$$\bigotimes_{i=0}^{N-1} x_i = x_0 \otimes x_1 \otimes \cdots \otimes x_{N-1}. \tag{7}$$

1.2 Calculus

Let

$$y = y(x_j), \qquad z = z(x_j), \tag{8}$$

meaning that y, z are both functions of $x_0, x_1, \ldots$, then

$$\frac{\partial}{\partial x_i}(y \otimes z) = \frac{\partial y}{\partial x_i} \otimes \frac{\partial z}{\partial x_i} \tag{9}$$

since $\otimes$ is just ordinary $+$. We can generalize further:

$$\frac{\partial}{\partial x_i} \bigotimes_j w_j = \bigotimes_j \frac{\partial w_j}{\partial x_i}. \tag{10}$$

The generalized addition is more complicated:

$$\frac{\partial}{\partial x_i}(y \oplus z) = \frac{1}{e^y + e^z}\left(e^y \frac{\partial y}{\partial x_i} + e^z \frac{\partial z}{\partial x_i}\right) \tag{11}$$

note the ordinary addition. For generalization:

$$\frac{\partial}{\partial x_i} \bigoplus_j w_j = \frac{\sum_j e^{w_j} \frac{\partial w_j}{\partial x_i}}{\sum_j e^{w_j}} \tag{12}$$

1.3 Generalization

More generally, we can define the generalized addition using base-B exponential and logarithms as

$$x \oplus_B y = \log_B(B^x + B^x) = \frac{1}{\log B} \log(e^{x \log B} + e^{y \log B}) \tag{13}$$

then every formula we have so far derived remains valid, except that for (11) and (12) we need to replace e by B. Note that in the limit $B \to +\infty$, we get the tropical semi-ring

$$x \oplus_{\mathrm{T}} y = \max(x, y) \tag{14}$$

where calculus is obviously problematic now. In general, in our models we could make $\log B$ a tunable parameter, corresponding to something like the "temperature" of the model.

2 Sequence to sequence model

2.1 The input tensor

Assume that we have a tensor of sequence inputs $I_{tbi} \in \bar{\mathbb{R}}$ where $b \in [0, B)$ is the batch index, $t \in [0, T)$ is the frame index, and $i \in [0, N)$ is the label index. We are also given a vector L^I_b of positive integers denoting the number of active frames in each batch. Further, we are told that $I_{tbi} = \bar{0}$ for all $t \geq L^I_b$.

For example, this could be features extracted from a batch of B audio data, where each batch has at most T frames of data, and each frame contains a N-vector of real-numbers. L^I_b then gives the actual number of frames in each batch. Getting ahead of ourselves, the condition for $t \geq L^I_b$ then says that these out-of-bound frames should have zero contributions.

2.2 The output tensor

Assume that we are given, together with the input tensor, $O_{bs} \in [0, N)$ where $b \in [0, B)$ is the batch index and $s \in [0, S)$ is the output index. We are also given a vector L^O_b of the number of active outputs, analogous to L^I_b.

Continuing our example, the output tensor could represent a transcription of the audio data where S is the batch length of the transcriptions, L^I_b are the individual lengths, and N is the size of the extended alphabet used for the transcription (see later for what "extended" means here). Recall that N is also the bound of the label index for I_{tbi}, so that the labels in I_{tbi} are related to the outputs in a way that we will specify later.

2.3 The transition matrix

We are also given $T_{ij} \in \bar{\mathbb{R}}$ for $i, j \in [0, N)$. The interpretation of T_{ij} is the transition score from j to i to be used in a generalized multiplicative sense, i.e., for a vector v_i, after we apply the transition, it becomes $v'_i = T_{ij} \otimes v_j$. T_{ij} is in general not symmetric.

For our running example, T_{ij} could represent the transition scores from state j to state i. The diagonal entries are the self transition scores which will be

crucial for warping the inputs to outputs of different lengths, as we will see later. These scores could come from, e.g., the logarithm of the transition probabilities of a bigram language model.

2.4 Lattice and paths

Now we can finally construct our sequence to sequence model. First, let's talk about notation. In the following, bold letters are used for all symbols which are used as place-markers and which have no numerical values per se. Upper case symbols denote states, whereas lower case symbols denote edges. Italics will continue to denote tensors with numerical values or indices.

The model is realized as a batch of weighted finite state acceptors labelled by the batch index $b \in [0, B)$. Within each batch, the states are denoted with symbols $\mathbf{S}^b_{ti}$ (note that the index structure is the same as I_{tbi}). There are also an initial state $\mathbf{I}^b$ and a final state $\mathbf{F}^b$. Between the states, there are edges denoted by the symbols $\mathbf{e}^b_{tij}$ from $\mathbf{S}^b_{ti}$ to $\mathbf{S}^b_{t+1,j}$ for all $t < T-1$. Finally, there are the initial edges $\mathbf{i}^b_i$ from $\mathbf{I}^b$ to $\mathbf{S}^b_{0,s}$ and the final edges $\mathbf{f}^b_i$ from $\mathbf{S}^b_{T-1,i}$ to $\mathbf{f}^b$. So, in total we have $B(TN+2)$ states and $BN((T-1)N+2)$ edges.

Now we have defined a lattice with states and edges. To make the lattice a finite state acceptor, we need to put labels on the edges. We associate with each edge $\mathbf{i}^b_i$ the label $i \in [0, N)$, with each edge $\mathbf{e}^b_{tij}$ the label $j \in [0, N)$, and with each edge $\mathbf{f}^b_i$ the special null label ϵ.

To further enhance our lattice to a weighted acceptor, we need to associate weights, or scores, on the edges as well. For $\mathbf{i}^b_i$, the weights are $W^{\mathbf{i}}_{bi} = I_{0bi}$, for $\mathbf{e}^b_{tij}$ the weights are $W^t_{bij} = T_{ji} \otimes I_{t+1,b,j}$, for $\mathbf{d}^b_{ti}$ the weights are $W^{\mathbf{f}}_{bi} = \bar{1} \equiv 0$.

Our completed weighted finite state acceptor then represents the set of all possible scored transcriptions from a given tensor I_{tbi} to the set of all possible outputs O_{bs} for max output lengths $S \leq T$. Let's decode what this means. For a particular interpretation, for batch b, frame t of the input is interpreted as the symbol i. Then state $\mathbf{S}^b_{ti}$ is active for this particular combination of b, t, i. Now we see that within each batch, for every t, only one of the states is active. Then there is a single active path $\pi^b : [0, T) \to [0, N)$ such that $\pi^b(t) = i$ from $\mathbf{I}^b$ to $\mathbf{F}^b$ for this interpretation. If we do the generalized product over all weights on this path, we get the score for this particular interpretation:

$$S_{\pi^b} = W^{\mathbf{i}}_{b,\pi^b(0)} \otimes \left(\bigotimes_{t=0}^{T-2} W^t_{b,\pi^b(t),\pi^b(t+1)} \right) \otimes W^{\mathbf{f}}_{b,\pi^b(t-1)} \tag{15}$$

What is the output of this interpretation? Now we stipulate that to arrive at the output for this interpretation, we collapse all repeated labels, e.g. $iijjjjkl \to ijkl$. In this way we deal with inputs and outputs with different lengths. What about repeated labels in the outputs? We simply eliminate them by introducing even more symbols. For example, if we want an output of $ijjklll$, we replace it with ijr_2klr_3, where r_k can be read as "repeat the previous label k times". If there is an upper bound on the potential number of repeated indices in the output, then this procedure does not forgo any expression power. Now our running audio data example should become crystal clear.

Note that we have a certain inelegant asymmetry between the "forward" direction $t \to t+1$ and the "backward" direction $t \to t-1$ due to the placement of weights. We could restore symmetry by placing the weights I_{tbi} directly on

the states $\mathbf{S}_{ti}^b$ instead, but this complicates the computation, so we will just live with this inelegance.

2.5 The fully-connected lattice

Ignoring O_{bs} for a moment, for given I_{tbi} and T_{ij}, can we calculate the total score, meaning the contribution from all possible paths, for our lattice? Sure, it is as simple as

$$S_{\text{full}}^b = \bigoplus_{\text{all } \pi^b} S_{\pi^b} \tag{16}$$

where S_{π^b} is given by (15). Well, the problem is the "all π^b" part. How many paths are there? For each b, there are exactly N^T paths. For example, usually in speech recognition we have about the order $N = 30, T = 100$, then we need to deal with more than 5×10^{147} possible paths. Clearly we need to do something more clever. Dynamic programming, of course.

Suppose that we many paths π_k^b such that $\pi_k^b(t) = \pi_{k'}^b(t) \equiv \pi^b(t)$ for $t \leq T_0$. Then, applying distributivity,

$$\bigoplus_k S_{\pi_k^b} = W_{b,\pi^b(0)}^{\mathbf{i}} \otimes \left(\bigotimes_{t=0}^{T_0-1} W_{b,\pi^b(t),\pi^b(t+1)}^t \right) \otimes \tag{17}$$

$$\bigoplus_k \left(\left(\bigotimes_{t=T_0}^{T-2} W_{b,\pi^b(t),\pi^b(t+1)}^t \right) \otimes W_{b,\pi^b(t-1)}^{\mathbf{f}} \right). \tag{18}$$

This represents a huge saving in computation since for $t \leq T_0$, we avoided an exponential number of identical calculations.

Carrying this saving to the extreme, let us recursively define

$$\alpha_{bi}^0 = W_{bi}^{\mathbf{i}} = I_{0bi} \tag{19}$$

$$\alpha_{bi}^t = \bigoplus_{j=0}^{N-1} W_{bji}^{t-1} \otimes \alpha_{bj}^{t-1} \tag{20}$$

$$= \bigoplus_{j=0}^{N-1} (I_{tbi} + T_{ij} + \alpha_{bj}^{t-1}) \tag{21}$$

$$= I_{tbi} + \bigoplus_{j=0}^{N-1} (T_{ij} + \alpha_{bj}^{t-1}) \qquad 0 < t < T \tag{22}$$

$$\alpha_b = \bigoplus_{j=0}^{N-1} W_{bj}^{\mathbf{f}} \otimes \alpha_{bj}^{T-1} \tag{23}$$

$$= \bigoplus_{j=0}^{N-1} \alpha_{bj}^{T-1}, \tag{24}$$

then α_b is the total score for batch b. But of course we can go from the other

direction as well: define

$$\beta_{bi}^{T-1} = W_{bi}^{\mathbf{f}} = 0 \tag{25}$$

$$\beta_{bi}^{t} = \bigoplus_{j=0}^{N-1} W_{bij}^{t} \otimes \beta_{bj}^{t+1} \tag{26}$$

$$= \bigoplus_{j=0}^{N-1} T_{ji} + I_{t+1,b,j} + \beta_{bj}^{t+1} \qquad 0 < t < T \tag{27}$$

$$\beta_{b} = \bigoplus_{j=0}^{N-1} W_{bj}^{\mathbf{i}} \otimes \beta_{bj}^{0} \tag{28}$$

$$= \bigoplus_{j=0}^{N-1} I_{0bj} + \beta_{bj}^{0}, \tag{29}$$

then β_b is the total score as well. It gets even better: we have, in general:

$$S_{\text{full}}^{b} = \alpha_b = \beta_b = \bigoplus_{j=0}^{N-1} \alpha_{bi}^{t} \otimes \beta_{bi}^{t} \equiv \bigoplus_{j=0}^{N-1} \gamma_{bi}^{t}, \qquad \text{for all } 0 \le t < T, \tag{30}$$

because $\gamma_{bi}^{t} \equiv \alpha_{bi}^{t} \otimes \beta_{bi}^{t}$ is the generalized score sum for all paths going through $\mathbf{S}_{ti}^{b}$. Now, by naïve counting, to get the total score, we need only do about $\mathcal{O}(BTN^2)$ generalized sums and $\mathcal{O}(BTN^2)$ generalized products. But we can actually still do better. Remember that generalized products are actually plain additions, which is very fast, so we would expect that for most problems the dominating factor is the generalized additions. But if we look at (20) and (26) carefully, we see that they are in the form of generalized matrix product, and more efficient algorithms for calculating generalized matrix product exist.

Observe that each step in the recursion of α (resp. β) depends only on the previous (resp. next) frame. In particular, the calculation of α_{bi}^{t} for fixed t, b and different i are completely independent. The same goes for β_{bi}^{t}. This will become important when we want to parallelize the computation.

Next we want the derivatives of S_{full}^{b} with respect to I_{tbj} and T_{ij}. First we have

$$\Delta I_{tbi} \equiv \frac{\partial S_{\text{full}}^{b}}{\partial \gamma_{bi}^{t}} = \frac{\partial S_{\text{full}}^{b}}{\partial \alpha_{bi}^{t}} = \frac{\partial S_{\text{full}}^{b}}{\partial \beta_{bi}^{t}} = \frac{\partial S_{\text{full}}^{b}}{\partial I_{tbi}} = \frac{e^{\gamma_{bi}^{t}}}{\sum_k e^{\gamma_{bk}^{t}}} \tag{31}$$

Next,

$$\frac{\partial \alpha_{bi}^{t}}{\partial W_{bji}^{t-1}} = \frac{e^{W_{bji}^{t-1} + \alpha_{bj}^{t-1}}}{\sum_k e^{W_{bki}^{t-1} + \alpha_{bk}^{t-1}}}, \tag{32}$$

so we have[1],

$$\Delta T_{ij} \equiv \sum_{b=0}^{B-1} \frac{\partial S_{\text{full}}^{b}}{\partial T_{ij}} = \sum_{b=0}^{B-1} \sum_{t=1}^{T-1} \Delta I_{tbi} \frac{e^{W_{bji}^{t-1} + \alpha_{bj}^{t-1}}}{\sum_k e^{W_{bki}^{t-1} + \alpha_{bk}^{t-1}}}. \tag{36}$$

[1]Note that

$$\frac{\partial \alpha_{bi}^{t}}{\partial T_{ij}} \neq \frac{\partial \alpha_{bi}^{t}}{\partial W_{bji}^{t-1}} \tag{33}$$

because there is also contribution from α_{bi}^{t-1}. To actually derive the stated result, introduce, for every $t > 0$, $T_{ij}^{t} = T_{ij}$, and use it in the definition of α_{bi}^{t}. Then since T_{ij}^{t} decouples with

2.6 The force-alignment lattice

Now we bring O_{bs} into the picture. Ultimately, we want to calculate

$$S^b_{\text{align}} = \bigoplus_{\text{valid } \pi^b} S_{\pi^b}. \tag{37}$$

In the equation above, paths in the original lattice are valid with respect to O_{bs} if they collapse to O_{bs} by our contraction scheme discussed before. This has the effect of vastly reducing the size of the lattice by removing all the inconsistent states and edges. Still, by judicious relabelling, we can get a very clean picture of the lattice after the reduction.

Consider a valid path π^b. If up to frame t the decoded labels correspond to O_{bs} for $s \in [0, S_0)$, what could $\pi^b(t+1)$ be? There are only two possibilities: it could either be O_{bs} in which case the decoded labels still correspond to $s \in [0, S_0)$ due to collapse of identical labels, or $O_{b,s+1}$ in which case the decoded labels now corresponds to $s \in [0, S_0 + 1)$ by moving onto the next label.

Motivated by this observation, we now define the force-alignment lattice, which can be obtained by removing states and edges (but sometimes also duplicating states and edges) as follows. The reduced input is now $\bar{I}_{tbs} = I_{t,b,\pi^b(s)}$ for $s \in [0, S)$, the identity transitions are $\bar{H}_{bs} = T_{\pi^b(s),\pi^b(s)}$ for $s \in [0, S)$, and the next transitions are $\bar{D}_{bs} = T_{\pi^b(s),\pi^b(s+1)}$ for $s \in [0, S-1)$. These three reduced tensors are the only ones that matter in the reduced lattice.

For states, $\bar{\mathbf{S}}^b_{ts}$ is the state that at frame t the path through it decodes to $\pi^b(s)$. For edges, $\bar{\mathbf{h}}^b_{ts}$ goes from $\bar{\mathbf{S}}^b_{ts}$ to $\bar{\mathbf{S}}^b_{t+1,s}$ (the collapse, or horizontal, route), and $\bar{\mathbf{d}}^b_{ts}$ goes from $\bar{\mathbf{S}}^b_{ts}$ to $\bar{\mathbf{S}}^b_{t+1,s+1}$ (the next, or diagonal route).

Note that some of the states are inaccessible for valid paths. For example, we can never get to state $\bar{\mathbf{S}}^b_{01}$. A state $\bar{\mathbf{S}}^b_{ts}$ is inaccessible if either $t - s < 0$ or $t + S >= T$. The accessible states form a parallelogram leaning to the left.

It remains to link the initial and final states. We could have an edge $\bar{\mathbf{i}}^b_0$ going from the initial state $\mathbf{I}^b$ to $\bar{\mathbf{S}}^b_{00}$, and another edge $\bar{\mathbf{f}}^b_{S-1}$ going from $\bar{\mathbf{S}}^b_{T-1,S-1}$ to the final state $\mathbf{F}^b$. This has the advantage of successfully making all the inaccessible states unreachable. The problem is that this requires computation to deal with many special cases. So instead, we will also link all the states $\bar{\mathbf{S}}^t_{0s}$, even inaccessible ones, to the initial state via $\bar{\mathbf{i}}^b_s$, and link all the states $\bar{\mathbf{S}}^b_{T-1,s}$ to the final state via $\bar{\mathbf{f}}^b_s$. We will deal with inaccessible states using weights.

Next we put weights on the edges: on $\bar{\mathbf{i}}^b_0$ we have $\bar{W}^{\mathbf{i}}_{b0} = \bar{I}_{0b0}$, on $\bar{\mathbf{i}}^b_s$ for $s > 0$ we have $\bar{W}^{\mathbf{i}}_{bs} = \bar{0} = -\infty$, on $\bar{\mathbf{f}}^b_{S-1}$ we have $\bar{W}^{\mathbf{f}}_{b,S-1} = \bar{1} = 0$, on $\bar{\mathbf{f}}^b_s$ for $s < S-1$ we have $\bar{W}^{\mathbf{f}}_{bs} = \bar{0} = -\infty$, on $\bar{\mathbf{h}}^t_{ts}$ we have $\bar{U}^t_{bs} = \bar{H}_{bs} \otimes \bar{I}_{t+1,b,s}$, and on $\bar{\mathbf{d}}^t_{ts}$ we have $\bar{V}^t_{bs} = \bar{D}_{bs} \otimes \bar{I}_{t+1,b,s+1}$.

---

respect to t, we have

$$\frac{\partial \alpha^t_{bi}}{\partial T^t_{ij}} = \frac{\partial \alpha^t_{bi}}{\partial W^{t-1}_{bji}}. \tag{34}$$

Finally, by the chain rule,

$$\frac{\partial S^b_{\text{full}}}{\partial T_{ij}} = \sum_{t=1}^{T-1} \frac{\partial S^b_{\text{full}}}{\partial T^t_{ij}}. \tag{35}$$

Similar to the fully connected case, we define

$$\bar{\alpha}_{b0}^0 = \bar{W}_{b0}^{\mathbf{i}} = \bar{I}_{0b0} \tag{38}$$

$$\bar{\alpha}_{bs}^0 = \bar{W}_{bs}^{\mathbf{i}} = -\infty \qquad s > 0 \tag{39}$$

$$\bar{\alpha}_{b0}^t = \bar{U}_{b0}^{t-1} \otimes \bar{\alpha}_{b0}^{t-1} \tag{40}$$

$$= \bar{H}_{b0} + \bar{I}_{t,b,0} + \bar{\alpha}_{b0}^{t-1} \qquad 0 < t < T \tag{41}$$

$$\bar{\alpha}_{bs}^t = (\bar{U}_{bs}^{t-1} \otimes \bar{\alpha}_{bs}^{t-1}) \oplus (\bar{V}_{b,s-1}^{t-1} \otimes \bar{\alpha}_{b,s-1}^{t-1}) \tag{42}$$

$$= \bar{I}_{tbs} + (\bar{H}_{bs} + \bar{\alpha}_{bs}^{t-1}) \oplus (\bar{D}_{b,s-1} + \bar{\alpha}_{b,s-1}^{t-1}) \qquad 0 < t < T, 0 < s < S \tag{43}$$

$$\bar{\alpha}_b = \bigoplus_{s=0}^{S-1} \bar{W}_{bs}^{\mathbf{f}} \otimes \bar{\alpha}_{b,s}^{T-1} = \bar{\alpha}_{b,S-1}^{T-1}, \tag{44}$$

and also

$$\bar{\beta}_{b,S-1}^{T-1} = \bar{W}_{b,S-1}^{\mathbf{f}} = 0 \tag{45}$$

$$\bar{\beta}_{bs}^{T-1} = \bar{W}_{bs}^{\mathbf{f}} = -\infty \qquad s < S-1 \tag{46}$$

$$\bar{\beta}_{b,S-1}^t = \bar{U}_{b,S-1}^t \otimes \bar{\beta}_{b,S-1}^{t+1} \tag{47}$$

$$= \bar{H}_{b,S-1} + \bar{I}_{t+1,b,S-1} + \bar{\beta}_{b,S-1}^{t+1} \qquad 0 \le t < T-1 \tag{48}$$

$$\bar{\beta}_{bs}^t = (\bar{U}_{bs}^t \otimes \bar{\beta}_{bs}^{t+1}) \oplus (\bar{V}_{bs}^t \otimes \bar{\beta}_{b,s+1}^{t+1}) \tag{49}$$

$$= (\bar{H}_{bs} + \bar{I}_{t+1,b,s} + \bar{\beta}_{bs}^{t+1}) \oplus (\bar{D}_{bs} + \bar{I}_{t+1,b,s+1} + \bar{\beta}_{b,s+1}^{t+1}) \qquad 0 \le t < T-1, 0 \le s < S-1 \tag{50}$$

$$\bar{\beta}_b = \bigoplus_{s=0}^{S+1} \bar{W}_{bs}^{\mathbf{f}} \otimes \bar{\beta}_{bs}^0 = \bar{I}_{0b0} + \bar{\beta}_0^0, \tag{51}$$

and so we have

$$S_{\text{aligned}}^b = \bar{\alpha}_b = \bar{\beta}_b = \bigoplus_{s=0}^{S-1} \bar{\alpha}_{bs}^t \otimes \bar{\beta}_{bs}^t \equiv \bigoplus_{s=0}^{S-1} \bar{\gamma}_{bs}^t, \qquad \text{for all } 0 \le t < T, \tag{52}$$

where $\bar{\gamma}_{bs}^t \equiv \bar{\alpha}_{bs}^t \otimes \bar{\beta}_{bs}^t$. We can check that our judicious placement of weights make the contribution of paths going through inaccessible states as if they were not there, since for every inaccessible path, at least one edge alone the path has weight $\bar{0}$. For reference, the number of generalized sums and products are in this case both $\mathcal{O}(BTS)$.

Now we derive the derivatives. This is more complicated than the fully connected case because the lattice is more complicated. First

$$\Delta \bar{I}_{tbs} \equiv \frac{\partial S_{\text{aligned}}^b}{\partial \bar{\gamma}_{bs}^t} = \frac{\partial S_{\text{aligned}}^b}{\partial \bar{\alpha}_{bs}^t} = \frac{\partial S_{\text{aligned}}^b}{\partial \bar{\beta}_{bs}^t} = \frac{\partial S_{\text{aligned}}^b}{\partial \bar{I}_{tbs}} = \frac{e^{\bar{\gamma}_{bs}^t}}{\sum_r e^{\bar{\gamma}_{br}^t}}, \tag{53}$$

note that this is 0 if $\bar{\gamma}_{bs}^t = -\infty$, so inaccessible states do not contribute. Then,

for $\bar{U}^t_{bs}$ and $\bar{V}^t_{bs}$,

$$\frac{\partial \bar{\alpha}^t_{b0}}{\partial \bar{U}^{t-1}_{b0}} = 1, \qquad t > 0 \tag{54}$$

$$\frac{\partial \bar{\alpha}^t_{bs}}{\partial \bar{U}^{t-1}_{bs}} = \frac{e^{\bar{H}_{bs}+\bar{\alpha}^{t-1}_{bs}}}{e^{\bar{H}_{bs}+\bar{\alpha}^{t-1}_{bs}} + e^{\bar{D}_{b,s-1}+\bar{\alpha}^{t-1}_{b,s-1}}}, \qquad t > 0, s > 0 \tag{55}$$

$$\frac{\partial \bar{\alpha}^t_{bs}}{\partial \bar{V}^{t-1}_{b,s-1}} = \frac{e^{\bar{D}_{b,s-1}+\bar{\alpha}^{t-1}_{b-1,s}}}{e^{\bar{H}_{bs}+\bar{\alpha}^{t-1}_{bs}} + e^{\bar{D}_{b,s-1}+\bar{\alpha}^{t-1}_{b,s-1}}}, \qquad t > 0, s > 0 \tag{56}$$

therefore,

$$\Delta \bar{H}_{b0} \equiv \frac{\partial S^b_{\text{aligned}}}{\partial \bar{H}_{b0}} = \sum_{t=1}^{T-1} \Delta \bar{I}_{tb0} \tag{57}$$

$$\Delta \bar{H}_{bs} \equiv \frac{\partial S^b_{\text{aligned}}}{\partial \bar{H}_{bs}} = \sum_{t=1}^{T-1} \frac{\partial S^b_{\text{aligned}}}{\partial \bar{\alpha}^t_{bs}} \frac{\partial \bar{\alpha}^t_{bs}}{\partial \bar{U}^{t-1}_{bs}} \tag{58}$$

$$= \sum_{t=1}^{T-1} \Delta \bar{I}_{tbs} \frac{e^{\bar{H}_{bs}+\bar{\alpha}^{t-1}_{bs}}}{e^{\bar{H}_{bs}+\bar{\alpha}^{t-1}_{bs}} + e^{\bar{D}_{b,s-1}+\bar{\alpha}^{t-1}_{b,s-1}}} \tag{59}$$

$$\Delta \bar{D}_{bs} \equiv \frac{\partial S^b_{\text{aligned}}}{\partial \bar{D}_{bs}} = \sum_{t=1}^{T-1} \frac{\partial S^b_{\text{aligned}}}{\partial \bar{\alpha}^t_{bs}} \frac{\partial \bar{\alpha}^t_{bs}}{\partial \bar{V}^{t-1}_{b,s-1}} \tag{60}$$

$$= \sum_{t=1}^{T-1} \Delta \bar{I}_{tbs} \frac{e^{\bar{D}_{b,s-1}+\bar{\alpha}^{t-1}_{b,s-1}}}{e^{\bar{H}_{bs}+\bar{\alpha}^{t-1}_{bs}} + e^{\bar{D}_{b,s-1}+\bar{\alpha}^{t-1}_{b,s-1}}}. \tag{61}$$

What we actually want, however, is derivatives with respect to the original inputs I_{tbi} and T_{ij}. We have

$$\Delta I_{tbi} = \sum_{s=0}^{S-1} \Delta \bar{I}_{tbs} \delta_{\pi^b(s),i} \tag{62}$$

$$\Delta T_{ii} = \sum_{b=0}^{B-1} \sum_{s=0}^{S-1} \Delta \bar{H}_{bs} \delta_{\pi^b(s),i} \tag{63}$$

$$\Delta T_{ij} = \sum_{b=0}^{B-1} \sum_{s=0}^{S-2} \Delta \bar{D}_{bs} \delta_{\pi^b(s),i} \delta_{\pi^b(s+1),j} \qquad i \neq j, \tag{64}$$

where δ_{ij} is Kronecker delta function.

2.7 The Auto-Segmentation Criterion

We want an overall score that encourages alignments compatible with the given O_{bs} and discourages all other alignments. Such a score is given by

$$S^b_{\text{ASG}} = S^b_{\text{aligned}} - S^b_{\text{full}}. \tag{65}$$

This can be motivated by, e.g., taking I_{tbi} to be log-probabilities with respect to fixed t and b, but note that our model does not enforce the normalization that is implied by probabilities.

In doing gradient descent, we want a loss instead of a score, which is easily obtained by reversing the sign:

$$L^b_{\text{ASG}} = -S^b_{\text{ASG}} = S^b_{\text{full}} - S^b_{\text{aligned}}. \tag{66}$$