SEQUENCE METRIC LEARNING AS SYNCHRONIZATION OF RECURRENT NEURAL NETWORKS : SUPPLEMENTARY MATERIAL

Anonymous authors
Paper under double-blind review

1 DIFFERENTIATION OF COUPLED GRU

$\frac{\partial E}{\partial h_t}$ is known, we call dx derivatives of the form $\frac{\partial E}{\partial x}$ with the pair of inputs concatenated (named hereafter left and right inputs). It is sometimes necessary to use separately the left and right sides of the derivative, in which case respectively indicated by $\leftarrow$ and $\rightarrow$ over the indices. Finally, $[x, y, ...]$ designates a concatenation along the suitable axe. Post-synaptic potentials (before activation function) are represented by the letter a with indices i for input and h for hidden and the associated gate letter. We seek to compute :

- dx_t
- dh_{t-1}
- dW_i where $W_i = [W_{iz}, W_{ir}, W_{in}]$
- dW_h where $W_h = [W_{hz}, W_{hr}, W_{hn}, W_{hc}]$,
- db_i where $b_i = [b_{iz}, b_{ir}, b_{in}]$
- db_h where $b_h = [b_{hz}, b_{hr}, b_{hn}, b_{hc}]$

$$
\begin{aligned}
dh_{t-1,o} &= dh_t \circ z_t && (1)\\
dz_t &= dh_t \circ (h_{t-1} - \bar{h}_t) && (2)\\
d\bar{h}_t &= dh_t \circ (1 - z_t) && (3)\\
dc_t &= d\bar{h}_{\overleftarrow{t}} \circ (\tilde{h}_{\overrightarrow{t}} - \tilde{h}_{\overleftarrow{t}}) + d\bar{h}_{\overrightarrow{t}} \circ (\tilde{h}_{\overleftarrow{t}} - \tilde{h}_{\overrightarrow{t}}) && (4)\\
d\tilde{h}_t &= d\bar{h}_t + [c_t, c_t] \circ ([d\bar{h}_{\overrightarrow{t}}, d\bar{h}_{\overleftarrow{t}}] - d\bar{h}_t) && (5)\\
da_z &= dz_t \circ ((1 - \sigma(a_z)) \circ \sigma(a_z)) && (6)\\
da_n &= d\tilde{h}_t \circ (1 - \tanh(a_{n,i} + r_t \circ a_{n,h})^2) && (7)\\
da_{n,h} &= da_n \circ r_t && (8)\\
dr_t &= da_n \circ a_{n,h} && (9)\\
da_r &= dr_t \circ ((1 - \sigma(a_r)) \circ \sigma(a_r)) && (10)\\
da_c &= dc_t \circ ((1 - \sigma(a_{\overleftarrow{c}} + a_{\overrightarrow{c}})) \circ \sigma(a_{\overleftarrow{c}} + a_{\overrightarrow{c}})) && (11)\\
da_{\overleftarrow{i}} &= [da_{\overleftarrow{z}}, da_{\overleftarrow{r}}, da_{\overleftarrow{n}}] && (12)\\
da_{\overrightarrow{i}} &= [da_{\overrightarrow{z}}, da_{\overrightarrow{r}}, da_{\overrightarrow{n}}] && (13)\\
da_{\overleftarrow{h}} &= [da_{\overleftarrow{z}}, da_{\overleftarrow{r}}, da_{\overleftarrow{n,h}}, da_c] && (14)\\
da_{\overrightarrow{h}} &= [da_{\overrightarrow{z}}, da_{\overrightarrow{r}}, da_{\overrightarrow{n,h}}, da_c] && (15)\\
dW_{iz} &= da_{\overleftarrow{z}} x_{\overleftarrow{t}} + da_{\overrightarrow{z}} x_{\overrightarrow{t}} && (16)\\
dW_{ir} &= da_{\overleftarrow{r}} x_{\overleftarrow{t}} + da_{\overrightarrow{r}} x_{\overrightarrow{t}} && (17)\\
dW_{in} &= da_{\overleftarrow{n}} x_{\overleftarrow{t}} + da_{\overrightarrow{n}} x_{\overrightarrow{t}} && (18)\\
dW_i &= [dW_{iz}, dW_{ir}, dW_{in}] \ \square && (19)\\
dW_{hz} &= da_{\overleftarrow{z}} h_{\overleftarrow{t}} + da_{\overrightarrow{z}} h_{\overrightarrow{t}} && (20)\\
dW_{hr} &= da_{\overleftarrow{r}} h_{\overleftarrow{t}} + da_{\overrightarrow{r}} h_{\overrightarrow{t}} && (21)\\
dW_{hn} &= da_{\overleftarrow{n,h}} h_{\overleftarrow{t}} + da_{\overrightarrow{n,h}} h_{\overrightarrow{t}} && (22)\\
dW_{hc} &= da_c (h_{\overleftarrow{t}} + h_{\overrightarrow{t}}) && (23)\\
dW_h &= [dW_{hz}, dW_{hr}, dW_{hn}, dW_{hc}] \ \square && (24)\\
db_i &= da_{\overleftarrow{i}} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} + da_{\overrightarrow{i}} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \ \square && (25)\\
db_h &= da_{\overleftarrow{h}} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} + da_{\overrightarrow{h}} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \ \square && (26)\\
dx_t &= [da_{\overleftarrow{i}} W_i, da_{\overrightarrow{i}} W_i] \ \square && (27)\\
dh_{t-1} &= dh_{t-1,o} + [da_{\overleftarrow{h}} W_h, da_{\overrightarrow{h}} W_h] \ \square && (28)
\end{aligned}
$$

2 STRUCTURAL LOSS

This paper makes use of the loss proposed by Yang et al. Yang et al. (2018) called a *structural loss*. It is a hardness-aware structural loss, composed of two terms a local one and a global one. The local one integrates a system of pair weighting to emphasize the learning on hard-positive samples (see Eq. 31 and 32): all positive distances above a certain class threshold τ_c will contribute more to the learning than the others. The second term is a global loss term used to prevent the similar samples to be disseminated in different places of the feature space and ultimately improve regularization and generalization. To do so, this term acts on the second order statistics of the distances to diminish the variance. The following notations are used: $\mathcal{B}$, the batch containing the outputs y of the neural network, $\mathcal{P}$ and $\mathcal{N}$ respectively the positive and negative samples of the batch, $\mathcal{P}_c$ the samples of

the class c. The values $\mu^t_{\mathcal{P}}$ and $\mu^t_{\mathcal{N}}$ are the average distances between respectively the positive and negative samples, γ controls the smoothness of the evolution of theses values between the previous batch $\mathcal{B}_{t-1}$ and the present one $\mathcal{B}_t$. Finally, m, m^+ and m^- are margin parameters, η is a scaling parameters and λ controls the magnitude of the global loss term in order to avoid it outweighing the local one.

$$\mathcal{L}_{\text{structural}}(f(\mathcal{B})) = \frac{1}{B} \sum_{(i,j)\in\mathcal{P}} \beta_{ij} \log\left(1 + \sum_{n=1}^{|\mathcal{N}|} \exp(d(y_i, y_j) - d(y_i, y_n) + m)/\eta\right) + \frac{\lambda}{2}([\sigma_p^2 - m^+]_+ + [\sigma_n^2 - m^-]_+) \quad (29)$$

$$B = \sum_{(i,j)\in\mathcal{P}} \beta_{ij} \quad (30)$$

$$\beta_{ij} = \exp(d(y_i, y_j)) - \tau_c \quad (31)$$

$$\tau_c = \frac{2}{|\mathcal{P}_c|} \sum_{(i,j)\in\mathcal{P}_c} d(y_i, y_j) - \min_{(i,j)\in\mathcal{P}_c} (d(y_i, y_j)) \quad (32)$$

$$\sigma_p^2 = \frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} (d(y_i, y_j) - \mu^{\mathcal{B}_t}_{\mathcal{P}})^2 \quad (33)$$

$$\sigma_n^2 = \frac{1}{|\mathcal{N}|} \sum_{(i,n)\in\mathcal{N}} (d(y_i, y_n) - \mu^{\mathcal{B}_t}_{\mathcal{N}})^2 \quad (34)$$

$$\mu^{\mathcal{B}_t}_{\mathcal{P}} = \gamma \mu^{\mathcal{B}_{t-1}}_{\mathcal{P}} + (1-\gamma)\mu^{\mathcal{B}_t}_{\mathcal{P}} \quad (35)$$

$$\mu^{\mathcal{B}_t}_{\mathcal{N}} = \gamma \mu^{\mathcal{B}_{t-1}}_{\mathcal{N}} + (1-\gamma)\mu^{\mathcal{B}_t}_{\mathcal{N}}. \quad (36)$$

3 AVERAGE NORMALIZED DISTANCE TO NEAREST NEIGHBORS

We present in Figure 1 the evolution of the average distance to the nearest neighbors depending on the number of nearest neighbors. Each distance has been scaled between 0 and 1 prior to the average. For SGRU, we observe that the average distance grows slowly at first until some point between 150 and 200 (around the average number of samples per class in this validation set, which is about 165) and from there starts to grow faster. This seems to indicate that the training of SGRU has allowed the model to create an important margin between similar and dissimilar samples. The curve for CGRU is very different with a very smooth progression and no apparent margin. This can be explained by the coupling which always combines the New States in some extend even if the input are dissimilar and therefore produces closer outputs. Despite the absence of margin, CGRU gives comparable results as SGRU on UCI HAR dataset which indicates that nevertheless, similar samples are closer than dissimilar samples even if the coupling produces overall lower distances. We attribute this compensation to better performances of CGRU on hard samples.

REFERENCES

Xun Yang, Peicheng Zhou, and Meng Wang. Person reidentification via structural deep metric learning. *IEEE Transactions on Neural Networks and Learning Systems*, 30(10):2987–2998, 2018.

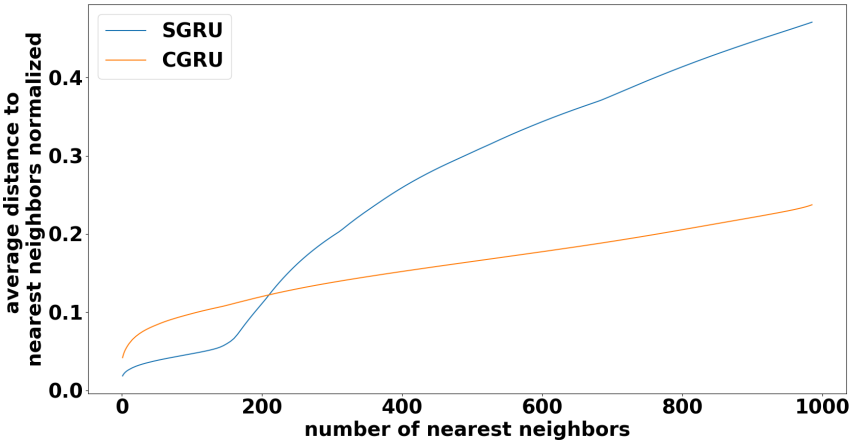

Figure 1: Average normalized distance to nearest neighbors for SGRU and CGRU computed on the validation set (users 1, 3 and 5)