TTIC 31230 Fundamentals of Deep Learning, winter 2020

Quiz 2

In these problems capital letter indeces are used to indicate subtensors (slices) so that, for example, $M[I,J]$ denotes a matrix while $M[i,j]$ denotes one element of the matrix, $M[i,J]$ denotes the ith row, and $M[I,j]$ denotes the jth collumn.

Throughout these problems we assume a word embedding matrix $e[W,I]$ where $e[w,I]$ is the word vector for word w. We then have that $e[w,I]^\top h[t,I]$ is the inner product of the word vector $w[w,I]$ and the hidden state vector $h[t,I]$.

We will adopt the convention, similar to true Einstein notation, that repeated capital indeces in a product of tensors are implicitly summed. We can then write the inner product $e[w,I]^\top h[t,I]$ simply as $e[w,I]h[t,I]$ without the need for the (meaningless) transpose operation.

The batch index is ommitted in all equations in this quiz.

Problem 1. Image captioning as translation with attention. We consider a simple version of machine translation with attention. We first run a right-to-left (backward) RNN on the input sentence to get a sequence $\overleftarrow{h}[T_{\mathrm{in}},J]$ of hidden vectors $\overleftarrow{h}[t,J]$ for $1 \leq t \leq T_{\mathrm{in}}$ where T_{in} is the length of the input sentence. We then define an autoregressive conditional language model

$$P_\Phi(w_1,\ldots,w_{T_{\mathrm{out}}} \mid \overleftarrow{h}[T_{\mathrm{in}},J])$$

as follows.

$$\vec{h}[0,J] = \overleftarrow{h}[1,J]$$

for t from 1 to T_{out}

$$\begin{aligned}
P(w_t \mid w_0,\cdots,w_{t-1}) &= \operatorname*{softmax}_{w_t}\ e[w_t,I]W^{\mathrm{auto}}[I,J]\vec{h}[t-1,J] \\
\alpha[t_{\mathrm{in}}] &= \operatorname*{softmax}_{t_{\mathrm{in}}}\ h[t-1,J_1]W^{\mathrm{key}}[J_1,J_2]\overleftarrow{h}[t_{\mathrm{in}},J_2] \\
V[J] &= \sum_{t_{\mathrm{in}}} \alpha[t_{\mathrm{in}}]\overleftarrow{h}[t_{\mathrm{in}},J] \\
\vec{h}[t,J] &= \mathrm{CELL}_\Phi(\vec{h}[t-1,J],\ V[J],\ e[w_t,I])
\end{aligned}$$

Here CELL is some function taking (objects for) two vectors of dimensions J and one vector of dimension I and returning (an object for) a vector of dimension J.

Rewrite these equations for image captioning where instead of $\overleftarrow{h}[t_{\mathrm{in}},J]$ we are given an image feature tensor $L[x,y,J]$

Solution:

$$P_\Phi(w_1, \ldots, w_{T_{\rm out}} \mid L[X, Y, J])$$

$$\vec{h}[0, J] = \frac{1}{XY} \sum_{x,y} L[x, y, J] \text{ Any function of the image gets full credit}$$

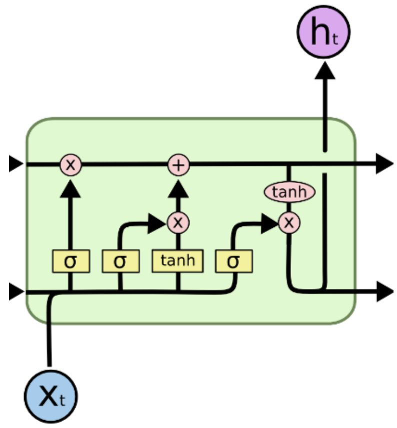
for t from 1 to $T_{\rm out}$

$$\begin{aligned}
P(w_t \mid w_0, \cdots, w_{t-1}) &= \operatorname*{softmax}_{w_t} \; e[w_t, I] \, W^{\rm auto}[I, J] \, \vec{h}[t-1, J] \\
\alpha[x, y] &= \operatorname*{softmax}_{x,y} \; h[t-1, J_1] W^{\rm key}[J_1, J_2] \, L[x, y, J_2] \\
V[J] &= \sum_{x,y} \alpha[x, y] L[x, y, J] \\
\vec{h}[t, J] &= \mathrm{CELL}_\Phi(\vec{h}[t-1, J], \; V[J], \; e[w_t, I])
\end{aligned}$$

Problem 2. Translating diagrams into equations. A UGRNN cell for computing $h[t, J]$ from $h[t-1, J]$ and $x[t, J]$ can be written as

$$\begin{aligned}
G[t, j] &= \sigma\left(W^{h,G}[j, \tilde{J}] h[t-1, \tilde{J}] + W^{x,G}[j, K] x[t, K] - B^G[j]\right) \\
R[t, j] &= \tanh\left(W^{h,R}[j, \tilde{J}] h[t-1, \tilde{J}] + W^{x,R}[j, K] x[t, K] - B^R[j]\right) \\
h[t, j] &= G[t, j] h[t-1, j] + (1 - G[t, j]) R[t, j]
\end{aligned}$$

Modify the above equations so that they correspond to the following diagram for an LSTM.



The top line in the diagram is the "carry vector" c_t. The equations for an LSTM should define $h[t, j]$ and $c[t, j]$ in terms of $h[t-1, J]$, $c[t-1, J]$ and $x[t, K]$.

Solution:

$$
\begin{aligned}
G_1[t,j] &= \sigma\left(W^{h,G_1}[j,\tilde{J}]h[t-1,\tilde{J}] + W^{x,G_1}[j,K]x[t,K] - B^{G_1}[j]\right)\\
G_2[t,j] &= \sigma\left(W^{h,G_2}[j,\tilde{J}]h[t-1,\tilde{J}] + W^{x,G_2}[j,K]x[t,K] - B^{G_2}[j]\right)\\
G_3[t,j] &= \sigma\left(W^{h,G_3}[j,\tilde{J}]h[t-1,\tilde{J}] + W^{x,G_3}[j,K]x[t,K] - B^{G_3}[j]\right)
\end{aligned}
$$

$$
\begin{aligned}
R_c[t,j] &= \tanh\left(W^{h,c}[j,\tilde{J}]h[t-1,\tilde{J}] + W^{x,c}[j,K]x[t,K] - B^{c}[j]\right)\\
c[t,j] &= G_1[t,j]c[t-1,j] + G_2[t,j]R_c[t,j]
\end{aligned}
$$

$$
\begin{aligned}
R_h[t,j] &= \tanh\left(W^{c,h}[j,\tilde{J}]c[t,\tilde{J}] - B^{h}[j]\right)\\
h[t,j] &= G_3[t,j]R_h[t,j]
\end{aligned}
$$

Problem 3. Gated CNNs

Again, A UGRNN is defined by the following equations.

$$G[t,j] = \sigma(W^{h,G}[j,\tilde{J}]h[t-1,\tilde{J}] + W^{x,G}[j,k]x_t[t,k] - B^G[j])$$

$$R[t,j] = \tanh(W^{h,R}[j,\tilde{J}]h[t-1,\tilde{J}] + W^{x,R}[j,K]x[t,K] - B^R[j])$$

$$h[t,j] = G[t,j]h[t-1,j] + (1 - G[t,j])R[t,j]$$

Modify these to form a data-dependent data-flow CNN for vision — an Update-Gate CNN (UGCNN). More specifically, give equations analogous to those for UGRNN for computing a CNN "box" $L_{\ell+1}[x,y,j]$ from $L_\ell[x,y,i]$ (stride 1) using a computed "gate box" $G_{\ell+1}[x,y,j]$ and an "update box" $R_{\ell+1}[x,y,j]$. In the CNN case there is no $x[t,I]$, just the previous layer $L_\ell[x,y,J]$.

Solution:

$$R_{\ell+1}[x,y,j] = \tanh(W^{L,R}_{\ell+1}[\Delta X,\Delta Y,I,j]\ L_\ell[x+\Delta X,\ y+\Delta Y,\ I] - B^R_{\ell+1}[j])$$

$$G_{\ell+1}[x,y,j] = \sigma(W^{L,G}_{\ell+1}[\Delta X,\Delta Y,I,j]\ L_\ell[x+\Delta X,\ y+\Delta Y,\ I] - B^G_{\ell+1}[j])$$

$$L_{\ell+1}[x,y,j] = G_{\ell+1}[x,y,j]L_\ell[x,y,j] + (1 - G_t[x,y,j])R_t[x,y,j]$$

Problem 4. Variance of running averages. For two independent random variables x and y and a weighted sum $s = ax + by$ we have

$$\sigma_s^2 = a^2\sigma_x^2 + b^2\sigma_y^2$$

Now consider a runing average for computing $\hat{\mu}_1,\ldots,\hat{\mu}_t$ from $x_1,\ldots,x_t$

$$\hat{\mu}_0 = 0$$

$$\hat{\mu}_t = \left(1 - \frac{1}{N}\right)\hat{\mu}_{t-1} + \frac{1}{N}x_t$$

(a) Assume that the values of x_t are independent and identically distributed with variance σ_x^2. We now have that $\hat{\mu}_t$ is a random variable depending on the draws of x_t. The random variable $\hat{\mu}_t$ has a variance $\sigma^2_{\hat{\mu},t}$. Assume that as $t \to \infty$ we have that $\sigma^2_{\hat{\mu},t}$ converges to a limit (it does). Solve for this limit $\sigma^2_{\hat{\mu},\infty}$. Your solution should yield that for $N = 1$ we have $\sigma^2_{\hat{\mu},\infty} = \sigma_x^2$ (a sanity check).

Solution: The limit must satisfy

$$\sigma^2_{\hat{\mu},\infty} = \left(1 - \frac{1}{N}\right)^2\sigma^2_{\hat{\mu},\infty} + \frac{1}{N^2}\sigma_x^2$$

We can then solve for $\sigma^2_{\hat{\mu},\infty}$

$$\begin{aligned}
\sigma^2_{\hat{\mu},\infty} &= \left(1 - \frac{2}{N} + \frac{1}{N^2}\right)\sigma^2_{\hat{\mu},\infty} + \frac{1}{N^2}\sigma^2_x \\
0 &= \left(\frac{-2}{N} + \frac{1}{N^2}\right)\sigma^2_{\hat{\mu},\infty} + \frac{1}{N^2}\sigma^2_x \\
&= \left((-2) + \frac{1}{N}\right)\sigma^2_{\hat{\mu},\infty} + \frac{1}{N}\sigma^2_x \\
\sigma^2_{\hat{\mu},\infty} &= \frac{1}{\left(2 - \frac{1}{N}\right)N}\ \sigma^2_x
\end{aligned}$$

(b) Compare your answer to (a) with the variance of an average of N values of x_t defined by

$$\hat{\mu} = \frac{1}{N}\sum_{t=1}^{N} x_t$$

Solution: For an average of N we have $\sigma^2_{\hat{\mu}} = \sigma^2_x/N$. For N large we have that the answer to part (a) is about half as large.