Stochastic Gradient Descent

This reading provides you with the opportunity to check your understanding of stochastic gradient descent (SGD). We will look at word embeddings as an example and see how word embeddings can be trained using skip-gram with negative sampling with stochastic gradient descent. We will provide you with the information you need to compute a step of SGD and suggest you try to compute the output by hand first, before looking at the correct solution on the next page. Have fun!

Given Parameters and Input

Target–Context Pairs As a tiny example, we consider a corpus consisting of the following single sentence:

"The cat sat on the mat"

To compute word embeddings, we obtain information for each word from its context. With a window size of one – i.e., considering one word to the left and one word to the right of the target word as our context –, the target–context pairs (T, C) are:

Position	Word	Context Window	Pairs
0	the	next: cat	$(T = \text{the}, C = \text{cat})$
1	cat	prev: the, next: sat	$(T = \text{cat}, C = \text{the})$, $(T = \text{cat}, C = \text{sat})$
2	sat	prev: cat, next: on	$(T = \text{sat}, C = \text{cat})$, $(T = \text{sat}, C = \text{on})$
3	on	prev: sat, next: the	$(T = \text{on}, C = \text{sat})$, $(T = \text{on}, C = \text{the})$
4	the	prev: on, next: mat	$(T = \text{the}, C = \text{on})$, $(T = \text{the}, C = \text{mat})$
5	mat	prev: the	$(T = \text{mat}, C = \text{the})$

Total pairs: (the, cat), (cat, the), (cat, sat), (sat, cat), (sat, on), (on, sat), (on, the), (the, on), (the, mat), (mat, the).

Model Parameters (Embeddings) We want to compute 2-dimensional embeddings for our words. We initialize them as follows:

$$\mathbf{v}_{\text{the}} = \begin{bmatrix} 0.1 \\ -0.2 \end{bmatrix}, \quad \mathbf{v}_{\text{cat}} = \begin{bmatrix} -0.3 \\ 0.4 \end{bmatrix}, \quad \mathbf{v}_{\text{sat}} = \begin{bmatrix} 0.5 \\ 0.1 \end{bmatrix},$$
$$\mathbf{v}_{\text{on}} = \begin{bmatrix} -0.1 \\ 0.3 \end{bmatrix}, \quad \mathbf{v}_{\text{mat}} = \begin{bmatrix} 0.2 \\ -0.4 \end{bmatrix}.$$

Negative Sampling For skip-gram with negative sampling, we need to pick a negative sample $\mathbf{v}_{\text{neg}}$ (a "wrong" context) for each (T, C) pair. (This is likely new to you as we haven't discussed it in the videos.) For example, if $(T = \text{cat}, C = \text{the})$, we choose $N = \text{on}$.

Loss Function The negative sampling loss for a triple $(\mathbf{v}_{\text{target}}, \mathbf{v}_{\text{context}}, \mathbf{v}_{\text{neg}})$ – the embeddings of T, C, and N – is:

$$L = -\log \sigma(\mathbf{v}_{\text{target}} \cdot \mathbf{v}_{\text{context}}) - \log \sigma(-\mathbf{v}_{\text{target}} \cdot \mathbf{v}_{\text{neg}}),$$

where:

- $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid function, and
- $\cdot$ denotes the dot product.

Gradients Let $u = \mathbf{v}_{\text{target}} \cdot \mathbf{v}_{\text{context}}$ and $w = \mathbf{v}_{\text{target}} \cdot \mathbf{v}_{\text{neg}}$. Since PyTorch can compute the gradients for you, we provide them here. They are as follows:

- Gradient w.r.t. $\mathbf{v}_{\text{target}}$:

$$\frac{\partial L}{\partial \mathbf{v}_{\text{target}}} = (\sigma(u) - 1)\mathbf{v}_{\text{context}} + \sigma(w)\mathbf{v}_{\text{neg}}.$$

- Gradient w.r.t. $\mathbf{v}_{\text{context}}$:

$$\frac{\partial L}{\partial \mathbf{v}_{\text{context}}} = (\sigma(u) - 1)\mathbf{v}_{\text{target}}.$$

- Gradient w.r.t. $\mathbf{v}_{\text{neg}}$:

$$\frac{\partial L}{\partial \mathbf{v}_{\text{neg}}} = \sigma(w)\mathbf{v}_{\text{target}}.$$

Parameter Updates (SGD) Let η be the learning rate. The parameter updates are:

$$\begin{aligned}
\mathbf{v}_{\text{target}} &\leftarrow \mathbf{v}_{\text{target}} - \eta \frac{\partial L}{\partial \mathbf{v}_{\text{target}}}, \\
\mathbf{v}_{\text{context}} &\leftarrow \mathbf{v}_{\text{context}} - \eta \frac{\partial L}{\partial \mathbf{v}_{\text{context}}}, \\
\mathbf{v}_{\text{neg}} &\leftarrow \mathbf{v}_{\text{neg}} - \eta \frac{\partial L}{\partial \mathbf{v}_{\text{neg}}}.
\end{aligned}$$

Now it's your turn! Compute one step of SGD for $\mathbf{v}_{\text{cat}}$ with $(T = \text{cat}, C = \text{the}, N = \text{on})$ by yourself, before looking at the solution on the next page.

Solution

We've discussed in the videos that, when using PyTorch, you need the following four steps to update your model parameters: (1) compute the output (i.e., the forward pass), (2) compute the loss, (3) compute the gradients, and (4) update the parameters. This is necessary in order to let PyTorch know which parts of the model are involved and which loss you're using. However, since in our example we already know the formula for the gradients, we can skip the first two steps and start with Step 3, the computation of the gradients.

Consider $(T = \text{cat}, C = \text{the}, \text{N} = \text{on})$. The embeddings are:

$$\mathbf{v}_{\text{cat}} = \begin{bmatrix} -0.3 \\ 0.4 \end{bmatrix}, \quad \mathbf{v}_{\text{the}} = \begin{bmatrix} 0.1 \\ -0.2 \end{bmatrix}, \quad \mathbf{v}_{\text{on}} = \begin{bmatrix} -0.1 \\ 0.3 \end{bmatrix}.$$

Step 3: Compute the Gradients

Let's compute the dot products first:

$$\begin{aligned} u &= \mathbf{v}_{\text{cat}} \cdot \mathbf{v}_{\text{the}} = -0.3 \times 0.1 + 0.4 \times -0.2 = -0.11, \\ w &= \mathbf{v}_{\text{cat}} \cdot \mathbf{v}_{\text{on}} = -0.3 \times -0.1 + 0.4 \times 0.3 = 0.15. \end{aligned}$$

Next, we pass them through the sigmoid function:

$$\begin{aligned} \sigma(u) &= \frac{1}{1 + e^{0.11}} \approx 0.4725, \\ \sigma(w) &= \frac{1}{1 + e^{-0.15}} \approx 0.5375. \end{aligned}$$

Now, we use what we have to compute the final gradients:

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{v}_{\text{cat}}} &= (\sigma(u) - 1)\mathbf{v}_{\text{the}} + \sigma(w)\mathbf{v}_{\text{on}}, \\ &= \begin{bmatrix} -0.5275 \end{bmatrix} \begin{bmatrix} 0.1 \\ -0.2 \end{bmatrix} + \begin{bmatrix} 0.5375 \end{bmatrix} \begin{bmatrix} -0.1 \\ 0.3 \end{bmatrix}, \\ &= \begin{bmatrix} -0.1065 \\ 0.26675 \end{bmatrix}. \end{aligned}$$

Step 4: Update the Parameters

Using $\eta = 0.1$:

$$\begin{aligned}
\mathbf{v}_{\text{cat}} &\leftarrow \mathbf{v}_{\text{cat}} - 0.1 \begin{bmatrix} -0.1065 \\ 0.26675 \end{bmatrix} \\
&= \begin{bmatrix} -0.3 \\ 0.4 \end{bmatrix} - \begin{bmatrix} -0.01065 \\ 0.026675 \end{bmatrix} \\
&= \begin{bmatrix} -0.28935 \\ 0.373325 \end{bmatrix}.
\end{aligned}$$