Discussion on the Project of Investigating Python Profilers

Last updated: June 6, 2021

Comments I

- We specifically ask you to compare the running time and the theoretical complexity
- Most did that though few did not
- The way to present your results is something you need to think about
- Now you have
  - time vs. complexity
  - convolution vs. pooling vs. linear
  - single vs. multiple cores
- How to clearly summarize these things needs a careful design

Comments II

- For linear layers most observed no speedup after using multiple cores
- One possible reason is that matrices here are small and the running time for linear layers is short

Padding I

- An issue we did not mention in the project slides is the padding operation before the convolutional operation
- In CNN, images are assumed to have the same size
- We do padding for reasons such as making the output image of the current layer not too small or ensuring that images can be exactly split to sub-images
- For texts, we can do similar things
- But for texts, there is another place where padding is needed.

Padding II

- We assumed that each document is represented by the following word embeddings

$$X = \begin{bmatrix} x_1 & \dots & x_N \end{bmatrix} \in R^{d_e \times N},$$

  where d_e is the word-embedding dimension and N is the document length.
- Here we assume that each document has N tokens (words) and N is a constant
- This is of course not true as documents have different lengths.

Padding III

- One way is to pad all documents to have the same length
- But this may add too many zeros on some documents
- In LibMultiLabel, what we did is to make documents in the same batch have the same length. This setting has also been used in other software
- Thus in your complexity analysis, N is bigger than the average document lengths

Padding IV

- An issue of this setting is that when the batch size changes, padding is a contributing factor to affect the behavior of the optimization algorithm

Discussion I

- In this project you have had some experiences on the package LibMultiLabel
- We are actively developing this package. If you are interested in this research direction, you are welcome to work with us