CSC format

hjiang

Oct, 2024

1 Introduction

This document provides an explanation of Figures 2 and 3 from the Efficient Inference Engine (EIE) on Compressed Deep Neural Networks. These figures illustrate the structure and functionality of the EIE.

2 Figure 2: Sparse Matrix Representation

Figure 2 depicts the sparse matrix representation used in EIE. The key components are:

- Non-Zero Elements: The figure highlights the positions of non-zero weights in the matrix.
- Row and Column Pointers: These pointers are essential for accessing non-zero elements efficiently.
- Compression Techniques: The figure illustrates how compression techniques reduce storage requirements by focusing only on non-zero elements.

3 Figure 3: Column Pointer Structure

Figure 3 shows the column pointer structure used to access non-zero elements in the compressed matrix. The main points include:

- Relative Row Index: The relative row index in the context of Figure 3 refers to the count of zero elements in the original weight matrix between the previous non-zero element and the current non-zero element for each Processing Element (PE). For example, there is no zero elements before $w_{0,0}$, so the relative row index for $w_{0,0}$ is 0. Between $w_{0,0}$ and $w_{8,0}$, so the relative row index for $w_{8,0}$ is 1. There is no zero elments between $w_{8,0}$ and $w_{12,0}$, so the relative row index for $w_{12,0}$ is 0.

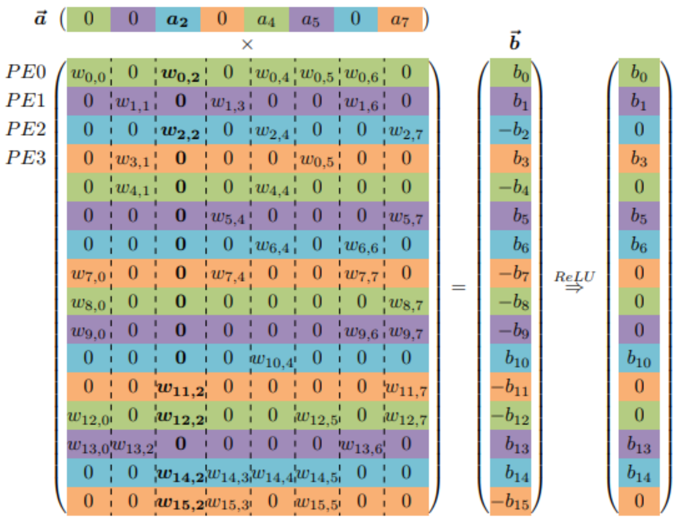

Figure 2. Matrix W and vectors a and b are interleaved over 4 PEs. Elements of the same color are stored in the same PE.

Virtual Weight	$W_{0,0}$	$W_{8,0}$	$W_{12,0}$	$W_{4,1}$	$W_{0,2}$	$W_{12,2}$	$W_{0,4}$	$W_{4,4}$	$W_{0,5}$	$W_{12,5}$	$W_{0,6}$	$W_{8,7}$	$W_{12,7}$
Relative Row Index	0	1	0	1	0	2	0	0	0	2	0	2	0
Column Pointer	0	3	4	6	6	8	10	11	13				

Figure 3. Memory layout for the relative indexed, indirect weighted and interleaved CSC format, corresponding to PE_0 in Figure 2.

Figure 1: Sparse Matrix Representation in EIE

- Column Pointers: The figure lists nine column pointers that indicate the starting position of non-zero elements for each column. For example, the 1st column has three non-zero weights $w_{0,0}$, $w_{8,0}$, and $w_{12,0}$; thus, the first entry in the column pointer is 0 and the second entry is 3. The difference between 0 and 3 represents the number of non-zero weights in the 1st column. The 2nd column has only one non-zero weight $w_{4,1}$; thus, the third entry in the column pointer is 4. The difference between 3 and 4 indicates the number of non-zero weights in the 2nd column. The term "column" here refers to the original weight matrix column.

- Distribution of Non-Zero Elements: Each pointer reflects how non-zero elements are distributed across columns, which is crucial for efficient data access.

- Implications for Performance: This structure allows for quick access and processing of relevant data, enhancing inference speed.

4 Conclusion

Figures 2 and 3 illustrate the core concepts behind the Efficient Inference Engine's approach to handling compressed deep neural networks. By leveraging sparse matrix representations and optimized access patterns, EIE significantly improves inference efficiency.