Data Mining, PS 2, 2ⁿᵈ Part

David Harar, 203670872

1. We have two elements, $A \in R^{n \times n}$ and $b \in R^n$ and we want to calculate $c = Ab$ using a MapReduce procedure.

    The input to our Map function is:

    $\{[a_{i,*}, a_{i+1,*}, \dots], i, k, b\}$ where $a_{i,*}$ is the ith row of the matrix A, b is the whole vector b, i is the index of the first row we have, and k is the last one (that is, the rows we have are $l \in \{i, \dots, k\}$).

    a. The (key, values) couples would be $(i, [a_{i1}b_1, a_{i2}b_2, \dots, a_{in}b_n])$. The first element is the index for row, and the second, $[a_{i1}b_1, a_{i2}b_2, \dots, a_{in}b_n]$ is a tuple of n elements that are the products of the a_{ij} element with b_j.

    b. In our example, the key is i, which denotes the row of the vector c, the product of A and b. Eventually, to each key we would like to assign the sum of these v_j values, which are correspond to the multiplication of the elements a_{ij} and b_j. Given the input $(key; [v_1, \dots, v_n])$ the output from the Reduce function would be $(key; \Sigma_{j=1}^n v_j)$, which is the value of row key in the resulted vector c.

2. Now, we would like to compute the product AB, which are both belong to $R^{n \times n}$.

    a. We have that that input for our Map function is

    $$\{(a_{i*}, \dots, a_{k*}), (b_{*i}, \dots, b_{*k}), i, k\}$$

    In that case, our desired (key, values) would be:

    $$(key, valeus) = ((i, j), (a_{i1}b_{1j}, a_{i2}b_{2j}, \dots, a_{in}b_{n,j})$$

    Where i (j) stands for the row (column) index of C, the produced matrix.

    b. In contrast to the previous subsection, here key is a tuple of length 2; specifically, $key = (i, j)$. This pair of indices denotes the element of matrix $C = AB$ which we will calculate the value for, using our Reduce function. Similarly to the previous subsection, here, the Map function will produce for each key a tuple of n different values. Here, $[v_1, \dots, v_n] = [a_{i1}b_{1j}, a_{i2}b_{2j}, \dots, a_{in}b_{n,j}]$ and the Reduce function will sum them up, so the results will be $(key, value) = ((i, j), \Sigma_{j=1}^n a_{ij}b_{jk})$.

3. We want to use a different method to calculate the matrix $C = AB$.

    a. Our input is the tuple (x, isA, i, j), where $isA := 1_{\{x \in A\}}$. If $isA = 0$ that means that x is actually b_{ij}. In contrast, if $isA = 1$ so $x = a_{ij}$.

    We write our matrices in that form

$$A_{ij} = (x, 1, i, j), B_{ij} = (x, 0, i, j).$$

$C_{ij} = \Sigma_{j=1}^{n} a_{ij} b_{jk}$, so out Map function will map the inputs to keys and values as in the following:

$$(keys, values) = \left((i, j), \left((x, 1, i, 1), \dots, (x, 1, i, n), (x, 0, 1, j), \dots, (x, 0, n, j)\right)\right).$$

The keys correspond to the element that we wish to calculate; the corresponding values are all the elements from the two matrices that affect its value.

b. After grouping, we get $(key; v_1, \dots, v_n, \dots, v_{2n})$. The key if (i, j) and these v_j are elements from which it is built of $(a_{ij}$ and $b_{ij})$. Assuming that v_1, v_{n+1} corresponding to $\left((x, 1, i, 1), (x, 0, 1, j)\right)$ then the reduce function would calculate the dot product of v_j and v_{n+j} and then summarize it all together.