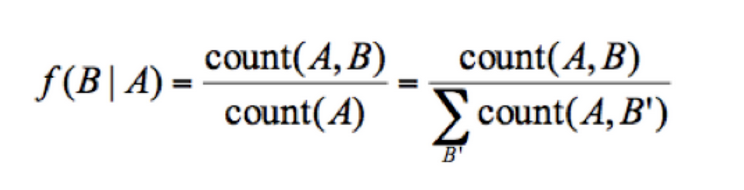
In this assignment, you will explore a set of 100,000 Wikipedia documents: [100KWikiText.txt](https://web.njit.edu/~chasewu/Courses/Fall2019/CS644BigData/HW/100KWikiText.txt), in which each line consists of the plain text extracted from an individual Wikipedia document. On the AWS VM instances you created or on your laptop, do the following:

1. Develop a MapReduce-based approach in your Hadoop system to compute the relative frequencies of each word that occurs in all the documents in 100KWikiText.txt, and output the top 100 word pairs sorted in a decreasing order of relative frequency. Note that the relative frequency (RF) of word B given word A is defined as follows:



where count(A,B) is the number of times A and B co-occur in the entire document collection, and count(A) is the number of times A occurs with anything else. Intuitively, given a document collection, the relative frequency captures the proportion of time the word B appears in the same document as A.

Submission requirements:

All the following files must be submitted in a zipped file:

* + A commands.txt text file that lists all the commands you used to run your code and produce the required results in both pseudo and fully distributed modes
  + A top100.txt text file that stores the final results (only the top 100 word pairs sorted in a decreasing order of relative frequency)
  + The source code of your MapReduce solution (including the JAR file)
  + An algorithm.txt text file that describes the algorithm you used to solve the problem
  + A settings.txt text file that describes:
  + i) the input/output format in each Hadoop task, i.e., the keys for the mappers and reducers
  + ii) the Hadoop cluster settings you used, i.e., number of VM instances, number of mappers and reducers, etc.

2. Use Spark to solve the above problem. You may choose to program use java, scala, or python.