Lab 2.

N.B: finish at least the mandatory part of Lab 1 before attempting this lab.

In this lab, you will write on your own a complete Hadoop application. For your ease you can rely, as a template, on one of the solutions we have met before in lab or in classroom, like the simple WordCount program, and modify it for your scope.

From now on, keep in mind always (even if we do not explicitely ask for it):

- the complexity of your program: try to understand, before even submitting a job, what will be the effort you will require to the cluster in terms of time, network and I/0. How many bytes will be read from HDFS? How many bytes will be shuffled (sent) between the nodes of a cluster? How many will be written back to disk? You can then check on HUE if you guessed correctly (more or less).
- the utility of your program: is the analysis we're asking in the text interesting to anybody, and to you? Can you modify it (slightly) to extract some not trivial piece of knowledge, or answer a question you personally have about this data?

1. Filter a file

If you completed Lab 1, you should now have (at least one) huge files with the word frequencies in the amazon food reviews, in the format `word\tnumber`, where number is an int (a copy of the output of Lab 1 is available in the HDFS shared folder /data/students/bigdata-01QYD/Lab2/). You should also have realized that inspecting these results manually is not feasible.
Your task is to write an Hadoop application to filter these results, and analize the filtered data. The filter we propose to you is the following: try to keep only the words that start with "ho". How large is the result of this filter? Do you need to filter more?

Can you specify the beginning string ("ho") as a command-line parameter?

Bonus task

If you completed the bonus task of lab 1, try your filter on the n-grams you have generated. What is the size of your input dataset, compared to the simple word counts (1-grams)? Did you need the cluster to filter the 1-grams? What about the 2-grams (or the n-grams for the n of your choice)?
As a sample filter, you can choose all the 2-grams that contain, at any position, the word "like". What do you think will be, most likely, the other word?