MCIS 6263 Big Data

Project 2

From This To That

In this project, you will be asked to work on the ETL concept. ETL stands for Extract, Transform, and Load process. The ***extract phase*** may refer to the process of obtaining and mine the data required for the analysis project. This may include some cleaning and combining of the data.
The ***transform phase*** is the process to make the acquired data comply with the format you are planning to use in the future steps. The ***load phase*** is about shipping the data into the enterprise systems (e.g. the database).

In this project our focus is on the first two stages, namely Extract and Transform. *We may see the last stage in a future assignment.*

You will be given a group of files. These files represent the customer reviews of a group of products. These products are: Canon G3 camera, Dvd player, Jukebox, Nikon Coolpix, and Nokia 6610.

These review files are semi-structured in a format very specific to the website that generated this data. Usually this format limits the gains that can be attained from this data. To overcome this, you need to change this format into a popular format which is the JSON format.

Please see below:

1. General Information: **Data Description**
    a. You will be given a folder that has the review data files. Each file is fully about one product.
    b. Each file is semi-structured in the following manner:
        i. The top of the file has a block of information related to the source of the data. This part is important but it is not of an interest for us in the moment.
            1. This block will be ignored.
        ii. Any review starts with **[t]** followed by the review title
            1. *Ex:* [t]great camera
        iii. Any positive aspects will start with **[+n]**, where **n** can be 1, 2, 3….or any number of points representing a good score. Followed by ## and then the text of the review.
            1. *Ex:* [+2]##i have only had this camera for one full day and i have to say that it is wonderful .
        iv. Any negative aspects will start with **[-m]**, where **m** can be 1, 2, 3….or any number of points representing a bad score. Followed by ## and then the text of the review.
            1. *Ex:* [-1]##* main dial is not backlit .

2. **First part: Extract, Clean, and Combine**
    a. In this part you will extract the data from the files given in point 1 above.
    b. In the extract process, you have to identify the product that is being reviewed.
    c. After identifying the product, each review has to be identified by the title.
    d. In a review you will identify the positive and negative aspects. To distinguish both, you can look and point 1 above.
        i. Extract the positive aspects of a review
        ii. Extract the negative aspects of a review.
    e. You should know that the text of any review, title and content, will not be clean. The cleaning should be done in the following manner:
        i. The text should not contain any special characters (!,@,#,$,%,^,&,",',<,>,/,? and *)
        ii. Also speech punctuation symbols have to be removed (, ; : \t)
        iii. The reviews are written in English. If for any reason a review has a language other than English, you can ignore such a review and do not include it in the output.
    f. After cleaning all the positive aspects, these will be combined in one single text block of positive criticism.
    g. The same will be done to the negative aspects.
3. **Second part: Data Transformation**
    a. After extracting, cleaning, and combining every review, the data will be put into the output JSON file. Every review will have its own JSON block.
    b. The JSON block has 4 fields.
        i. The first field is the id which holds a sequential value.
        ii. The second field holds the title of the review
        iii. The third field contains all the positive criticism
        iv. The forth field includes all the negative criticism
            1. *Ex*: The reviews stored in the JSON file would look like:
            {"id":1 , "title": "*the title text*", "positive": "*positive block…*", "negative": "*negative block…*"}
            {"id":2 , "title": "*the title text*", "positive": "*positive block…*", "negative": "*negative block…*"}
            {"id":3 , "title": "*the title text*", "positive": "*positive block…*", "negative": "*negative block…*"}
            .
            .
            .
    c. The output file should be named with the product name and the type is JSON. Make sure the file type is **json**.
        i. *Ex*: Canon G3 camera.**json**

d. If your code is given several files as input, then the output **json** files should have the same count as the input files.
    i. *Ex*:
        1. If input is: Canon_G3_Camera.txt Dvd_player.txt
        2. The output will be: Canon_G3_Camera.json Dvd_player.json

Notes:

1- The project is to be done in groups of 3 or less. **Groups must be from same section**. Forming groups, if you want to have a group, is the responsibility of students. Therefore, not finding a group in your section is not an excuse not to do the project; you still can do it on your own.
2- You should be developing this project under the Linux machine (the Cloudera virtual machine) you should have installed at the beginning of this semester, without the need to install any special packages or libraries except the default compilers and libraries.
3- Name the solution file ETL.py, or ETL.java.
4- Only one code file should be submitted per group. Your code should start with a block of comment.
    a. This comment block has:
        i. Students names, ids, and sections
5- You have to make sure that your code runs error-free, especially compilation errors. **We will not debug or fix any errors.** Very low score is expected in this case.
6- Be careful about the Path names. Always assume current folder/directory.
7- **How to run the code**:
    a. Python: python2.6 ETL.py file1 file2 file3 …..
    b. Java:
        i. Compile: **javac ETL.java**
        Run: **java ETL file1 file2 file3 …..**
    c. Your code should accept one file or more
    d. Do not hard-code the review/input file names inside your code.
8- Copying and cheating will have serious consequences. So, avoid that.
9- Due date is: 10/1/2016 , 11:59 CDT.

Good Luck!