# Opinion Mining

**Overview**

In this project, you will learn how to find aspects and opinions in product reviews. You will also learn how to perform basic text pre-processing before you can do opinion mining. We are providing you a digital camera dataset in two forms: tagged and untagged (see readme). Reference paper "[Mining and summarizing customer reviews](http://www.cs.uic.edu/~liub/publications/kdd04-revSummary.pdf)".

**Getting Started**

The coding is based on Java & python 2.7. Download and install python 2.7: sudo apt-get install python2.7.

You need to use the [Opinion Findertool](http://mpqa.cs.pitt.edu/opinionfinder/opinionfinder_2/) to do pre-processing steps (you also try out the [GATE tool](https://gate.ac.uk/download/)

For basic text pre-processing using python see [here](https://www.kaggle.com/c/word2vec-nlp-tutorial/details/part-1-for-beginners-bag-of-words). Some more info about text mining see [here](http://text-analytics101.rxnlp.com/p/compiled-tutorials.html).

Natural language processing with Python and NLTK p.1 ([here](https://www.youtube.com/watch?v=FLZvOKSCkxY&list=PLQVvvaa0QuDf2JswnfiGkliBInZnIC4HL))

For association rule mining using FP-Growth algorithm

https://github.com/enaeseth/python-fp-growth

http://www.borgelt.net/fpgrowth.htmlFP-growthalgorithm

Following are some relevant links.

http://rwet.decontextualize.com/book/textblob/

http://textblob.readthedocs.io/en/dev/install.html

http://textblob.readthedocs.io/en/dev/quickstart.html

https://textblob.readthedocs.io/en/dev/

Result obtained through Opinion Finder is recommended. It is optional if you want to give a comparison of the result using Opinion Finder vs TextBlob in the report.

**Task 1** Take the untagged dataset and perform POS tagging using the Opinion Finder tool.

**Task 2** From the POS tagged dataset extract all the Nouns. Call this the list of candidate aspects. (Hint: Need to write a Python script)

Note: While extracting aspects considered the lemma of the nouns instead of the nouns. Lemma will remove unnecessary duplicates. For instance, “picture” and “pictures” are treated as a different noun by Opinion Finder although they represent the same feature “picture”. Lemmas are used to solve this problem

**Task 3** Select all the frequent aspects using Association Rule Mining algorithm with minimum 0.5% and 1% support. (Submit the top-10 most frequent aspects in decreasing order of support.). What difference do you see in using 0.5% vs 1% minimum support?

Note: You need to consider each sentence as a transaction and not the whole review. Same feature might be mentioned multiple number of times in a single review. As a result, the count of frequent features may be more than the total number of reviews.

If two or more nouns are occurring consecutively, then you should treat it as follows: for a bigram “battery life”, pass three nouns to the association rule mining algorithm: “battery”, “life” and “battery life”. With this approach if “battery life” is a frequently occurring phrase then that will remain after association rule mining whereas less frequently occurring phrases such as “battery duration” will only contribute to the feature “battery”.

**Task 4** Use the identified frequent aspects in Exercise 3 to find candidate opinion words. These are the adjectives or adverbs which are near the frequent aspects identified in Exercise 3 (within five words on both sides of a frequent aspect). (Submit the list of top-10 most frequently used opinion words for both 0.5% and 1% support.)

Note: If the same opinion word is near to one or more frequent aspect in same sentence, then you should count it only once.

**Task 5** Compute precision, recall and F1 score for the aspects identified obtained step in Exercise 3. Use the tagged dataset to know the ground-truth aspects.

**Task 6** Compute the number of positive and negative opinion words you obtained in Exercise 4. You need to use the provided list of standard opinion words in English.