Technical Review
Sentiment Analysis using LingPipe

What is Sentiment Analysis: It is a method to identify and extract subjective content usually an opinion of text. It classifies opinions in text into categories like 'positive' or 'negative'. There can be an implicit category of 'neutral'. A good example of sentiment analysis is to categorize a movie review as positive or negative.

What is LingPipe: LingPipe is a JAVA toolkit for text processing. It uses JAVA api's to do various tasks related to text mining like-

a) searching names, organizations and locations in news,
b) categorizing twitter search results and
c) spell correction of queries.

For sentiment analysis LingPipe uses various algorithms and language models to classify text based on various sentiments in text. LingPipe's library is multi-lingual, multi-domain and multi-genre models.

Sentiment Analysis and LingPipe: Sentiment analysis is a special case of classifying text where text is categorized into: positive and negative sentiments. There are many approaches to this to do classification. We can use LingPipe's language classification framework. It hierarchically classifies by composing basic sentiment analysis algorithm such as logistic regression. Basic sentiment classifiers handle 2 level classification tasks-

1) Separating subjective from objective sentences
2) Separating positive from negative movie reviews

The one that is used often is sentence based sentiment with logistic regression classifier. LingPipe is re-implementation of hierarchical classification technique described in Bo Pang and Lillian Lee ACL paper in 2004. They introduced a hierarchical polarity analysis approach to classification. This technique helped improve polarity classification by removing objective sentences from the text and then apply a standard machine learning classifier to the resulting extract.

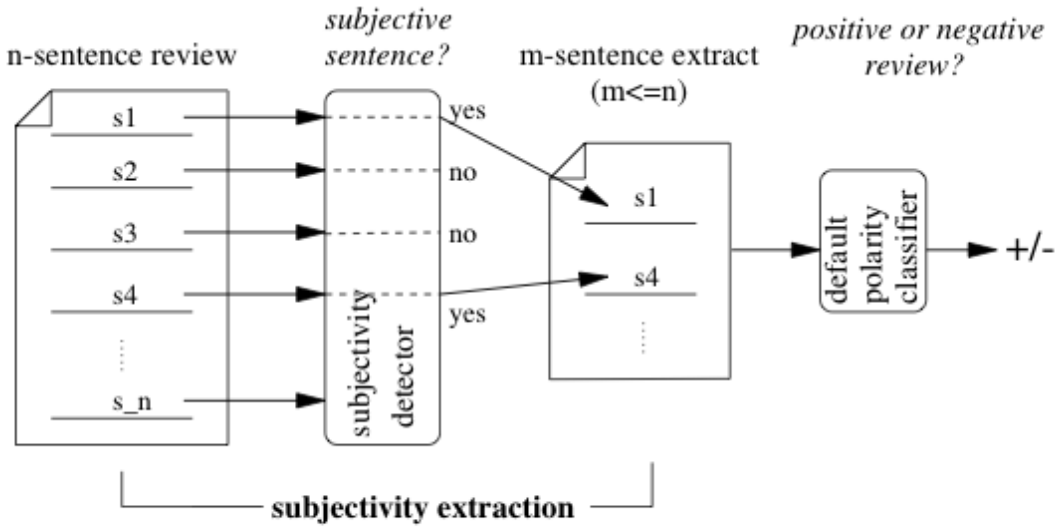
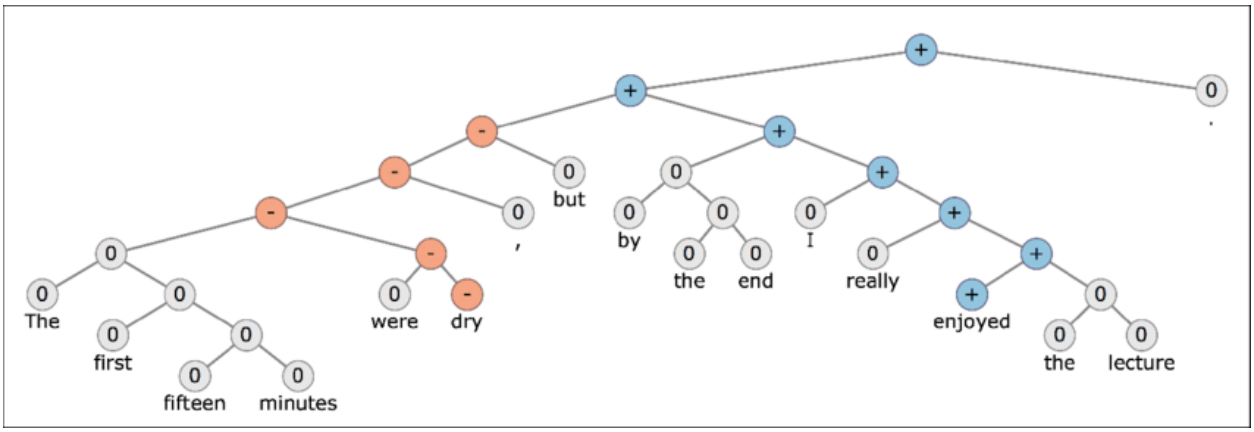


Figure 1: Polarity classification via subjectivity detection.

As shown in the above diagram – once the input text which we assume are movie reviews are fed into the classifier it first eliminates the objective sentence and then use remaining sentences to classify document polarity.
The subjectivity detector is employed first to find out if the sentences are subjective or not and discarding the objective ones which creates an extract that should better represent a review's subjective content to a default polarity classifier like support vector machines (SVM) or Naïve Bayes (NB). It reduces each movie review from top 5 to N sentences which gives performance improvement. There are 5 more subjective sentences as ranked by conditional probability of subjectivity model as well as N more sentences which are 50% or more likely to be subjective. LingPipe uses unigram features extracted from movie review data. It also assumes that similar sentences are likely to have similar subjective and objective polarity (SO). LingPipe can efficiently extract subjective sentences by using a min-cut algorithm which iterates over the sentences and classify them with subjective classifier.

A Basic Flow to use LingPipe for Sentiment Analysis: Unlike CoreNL, LingPipe does not provide any available model. We first train the classifier with answer sets. We pre-identify some data as negative data. And some other data sets as positive data. We place the negative and positive reviews in separate folders. The manually classified scores of the review are used to basically train the model. The LingPipe central model takes positive and negative folders and passes them into a classifier, dynamic LM classifier. Then train the classifier based on correct predefined answer sets and then create sentiment model. Once the classifier is serialized into a sentiment model we load it into memory. For each sentence we can then predict the polarity.

LingPipe vs CoreNLP: CoreNLP is top performing Natural Language Processing classifier in JAVA. It does not use bag of words approach but rather stores sentences in parsed tree format. The sentence structure is taken into account while classifying sentences. In treebank or semantic tree bank the nodes of binary tree of each sentence including the root node of each sentence is given a sentiment score. While parsing through every level of tree annotates the sentiment of phrase. It uses 5 class scheme.



Conclusion: LingPipe have shown that subjectivity extracts are more effective as inputs to the polarity classifiers than originating documents because it completely removes the objective sentences. With LingPipe the subjectivity detection can compress reviews to shorter extracts but still retain polarity information better than full reviews. These extracts are not only short but also cleaner representations of polarity. In comparison to CoreNLP, LingPipe uses classic conditional probabilistic LM's. CoreNLP has pre trained deep learning models which can be used directly whereas with LingPipe we need to train data before using the classifiers.

References:

http://www.alias-i.com/lingpipe/demos/tutorial/sentiment/read-me.html
https://www.cs.cornell.edu/home/llee/papers/cutsent.pdf
https://www.researchgate.net/figure/A-sample-sentence-from-the-Stanford-Sentiment-Treebank3_fig1_334636635