COMS 6998-7, Spring 2010
Statistical Methods for NLP

Homework 3

Due: April 9 (Friday), 2010 (11:59pm)

Total Points: 100

Summarization and Topic Clustering using Bayesian Network and Expectation Maximization

In this homework you will get a chance to implement a clustering algorithm and see how it can be used for summarization. You will also get to learn how to use a commonly used ML tool for speech summarization. For the simplicity of the homework we define extractive summarization as finding the most 'significant' sentences in a document. You are given a corpus of spoken documents with their extracted features. The extracted features contain lexical, acoustic, structural and discourse features. You are supposed to use these features for your experiments. This is research style homework where you get a chance to play with real data and answer real research problems. The research question we are trying to address is how to best summarize a spoken Broadcast News (CNN) document. You are not allowed to use the given data for any other purpose than this homework.

Q1. Data Analysis and Statistical Learning with Weka for Summarization [50]

Data directory: /home/smaskey/CS6998/hw3/q1_q2

For all the Homework so far you have implemented your own learning and inference algorithms. For the real world research it is sometimes important to be able to use statistical tools that other researchers have built. Weka is one such Machine Learning tool that is useful for many NLP related learning tasks. In the first part of the homework you will learn how to use Weka to analyze, visualize and learn from the data. Download and install Weka from http://www.cs.waikato.ac.nz/ml/weka/

(1) Load train.arff and look at the relationship of features with classes by visualizing the feature label relationship. Which feature do you think is the best indicator of summary (INSUMMARY) label? Justify your answer. [10]

(2) Train a classifier with 10-fold cross validation using Bayes Net algorithm. Report the Precision, Recall and F-measure. Visualize the Bayes Net. What does this graphical model structure represent? Can you improve the model by different sets of dependencies across features? What kind of changes do you suggest for the model structure of Bayes Net Model? Justify your answers. [10]

(3) Now label all your sentences in test.arff using the model you have trained. You must test the classifier from the commandline using the model saved from 1.2. Report Precision, Recall and F-measure. For all the sentences that have been labeled as significant (INSUMMARY) find the corresponding sentences from test.data file and concatenate them together. How does your summary look like? Does your analysis of best feature in

1.1 seems valid for test data? Do you think redundancy of sentences could be a problem? [10]

(4) Instead of training a classifier in supervised fashion we can also try to cluster the sentences and use the clusters as two separate classes. Remove all the features except TIMELENA for train.aff. Cluster the data using K-Means algorithm. Visualize the cluster, what do you believe should be the threshold for separating the clusters? Now re-run K-Means with all features. What is the clustering accuracy? What is the ratio of cluster assignments between two clusters? [10]
(5) Re-run 1.3 with EM algorithm. Do you see any difference in accuracy and cluster assignments? Why does it improve or does not improve? Justify your answers [5].
(6) Feature selection can help improve clustering or classification accuracy that you obtained above. Run a feature selection algorithm to automatically select the features. How many features are selected? Do these features make sense for summarization? [5]

Q2. Expectation Maximization for Topic Clustering [50]

Data directory: /home/smaskey/CS6998/hw3/q1_q2

One of the important problems in summarization is redundancy computation across sentences. Even though a sentence may be important for a summary if we have already included another sentence from the same topic we may want to discard that sentence from the summary. The train and test set sentences can be found in train.data and test.data file within <SENTL></SENTL> tags. You want to cluster the sentences and see if the clusters make any topical sense.

(1) Implement E-step of EM algorithm. [15]
(2) Implement M-step of EM algorithm. [15]
(3) Run EM algorithm to cluster the train sentences. Report you log-likelihood with number of iterations. Explain the kind of features you are using and the number of clusters. (You are not required to implement new feature extraction code and can just use the provided features if you want; but they may not be optimal features for topic clustering) Do the provided features work for topic clustering? What can you do to improve the clustering for topic? [10]
(4) Run EM algorithm to cluster test sentences. Extract the corresponding sentences for each cluster from test.data. Do you think your EM algorithm is useful for finding redundant information? Justify your answer. [10]

Extra Credit [5]
Build automatic show level summary for the test data.