Assignment 1a – Due before class on January 29

Learning Objectives:

1. Gain more insight into what LDA variants are doing with text data
2. Gain experience working to validate whether a model is doing something reasonable
3. Gain experience weighing and balancing the trade offs between models and troubleshooting models

Description:

In this assignment you will be working with a data set where groups of 3 9th grade students worked together on an assignment related to the concept of diffusion. Students worked together in an online chat environment. Each discussion is treated like a separate discussion set in this data. In each discussion there are three students and one tutor agent (which is a computer agent). There are three different conditions (ns: no support, ind: indirect support, and dir: direct support), which describe different strategies used by the computer agents. There were 4 different classrooms of students involved in the experiment, labeled A – D. You can find the whole annotated dataset in the file called AnnotatedData.xls.

You will pick 2 out of 3 available models for the assignment (you are welcome to also work with the third if you so choose). The three models are: plain LDA, M4, and Block HMM. The last two are the models we discussed in Week 1 Lecture 2. You will find the output from the three models in the folders named after the models. I have set up the output so that it is in a common formal across models. The goal of your analysis will be to answer the following question: Which of the two models you selected produced the most reasonable output? It is up to you to decide how to make the argument.

Deliverable: What you will turn in is your write up where you will describe between 2 and 4 experiments you did with the output of each model that you designed to evaluate how reasonable the output from the model was. In order to do the analyses you will have to write some scripts in R, Python, or some other programming or scripting language you feel comfortable with. These scripts will be run on the model output files you'll find in the model output folders. You should turn in your scripts with your write up. You will package up these files as a single zip file with your name as the filename and turn them in to the Turnin link in the Assignment1a folder on blackboard.

Here are some example experiments/analyses you could run (I encourage you to think creatively about what you could investigate using the files you are given):

1. Use the topic distribution per turn as features to predict one or more of the columns of annotations. You can evaluate the result in terms of kappa, accuracy, and f-measure in addition to doing an error analysis to identify which types of turns were easier and harder.
2. Examine the ranked words per topic to see which model produced word rankings that appear coherent and distinct.
3. Examine differences in word rankings per topic between different conditions, sessions/classrooms, roles (i.e., tutor vs student) to see whether the distinction makes sense based on what you would expect.

Technical Details:

1. All of the topic models were built with 5 set as the number of topics
2. The plain LDA model was built in R using the tm and topicmodels packages. The input file is just the column of text from the AnnotatedData.xls file.
3. The M4 and Block HMM models were built using Michael Paul's downloadable code, which you can get and run if you are interested. His code includes instructions for compiling and running the code as well as formatting the input. You can easily construct the input format using columns line, parent, turnlabel, and content from the AnnotatedData.xls file.
4. The Output.txt file in the model output folders is in the format of Michael Paul's assignments file. Each line is a line from the corpus, and each word is labeled with the topic number assigned to it.
5. The csv files in the model output folders give other more specific information about the models:
    a. DocumentTopicWeights: gives a probability for each topic for each turn
    b. DocWordToTopicNum: gives the assignment of each word in the text to the topic number (1-5). 0 indicates that the word was not present.
    c. DocWordtoWordCounts: gives the number of times each word occurred in each document
    d. TopicWordWeights gives a word weight for each word in each topic. But sorting on the topic, you can get the word rankings for that topic.
6. The corpus and coding scheme for the annotations are described in the chapter found in BioChatDataChapter.pdf.