IST687 – Text Mining HW

Now that we are doing text mining, we will be creating our own termDocMatrix.

This was also done in class, when we analyzed the structure of the "I have a dream" speech – in terms of the use of positive and negative words. However, in that effort, we treated all positive words the same (ex. good is the same as great). This might not be appropriate – maybe we should count more positive (and negative) words more than other words. For example "I loved the movie" might be stronger than "I liked the movie".

There is a different word file that ranks each word on a scale of -5 to 5 (negative to positive). It is known as the AFINN word list.

Your task for this homework is to adapt the lab that we did in class, to compute the score for the MLK speech using the AFINN word list (as opposed to the positive and negative word lists).

1. First read in the AFINN word list. Note that each line is both a word and a score (between -5 and 5). You will need to split the line and create two vectors (one for words and one for scores).

2. Compute the overall score for the MLK speech using the AFINN word list (as opposed to the positive and negative word lists).

3. Then, just as in class, compute the sentiment score for each quarter (25%) of the speech to see how this sentiment analysis is the same or different than what was computing with just the positive and negative word files.

   Note that since you will be doing almost the exact same thing 4 times (once for each quarter of the speech), you should create a function to do most of the work, and call it 4 times.

4. Finally, plot the results (i.e, 4 numbers) via a bar chart.

Learning Goals for this activity:

A. Consider how a simple text mining technique can be applied to a variety of kinds of source data.
B. Provide practice in conditioning data to prepare for analysis.
C. Develop skill in the setup, execution, and interpretation of text mining analytics.
D. Increase familiarity with bringing external data sets into R.

Essential Guide for All IST687 Activities (appears at the end of all activity guides)

1. All IST687 activities work on what some people call a "constructivist learning" model. By developing a product on your own, testing it to find flaws, improving it, and comparing your solution to the solutions of other people, you can obtain a

deeper understanding of a problem, the tools that might solve that problem, and a range of solutions that those tools may facilitate. The constructivist model only works to the extent that the student/learner has the drive to explore a problem, be frustrated, fail, try again, possibly fail again, and finally push through to a satisfactory level of understanding.

2. Each IST687 activity builds on skills and knowledge developed in the previous activities, so your success across the span of the course depends at each stage on your investment in earlier stages. Take the time to experiment, play, try new things, practice, improve, and learn as much as possible. These investments will pay off later.
3. Using the expertise of others, the Internet, and other sources of information is not only acceptable - it is expected. You must ***always, always, always*** give credit to your sources. For example, if you find a chunk of code from r-bloggers.com that helps you with developing a solution, by all means borrow that chunk of code, but make sure to use a comment in your code to document the source of the borrowed code chunk. The discussion boards in the learning management system have been setup to encourage appropriate sharing of knowledge and wisdom among peers. Feel free to ask a question or pose a solution on these boards.
4. Building on the previous point, when submitting code as your solution to the activity, the comments matter at least as much, if not more than the code itself. A good rule of thumb is that every line of code should have a comment, and every meaningful block of code should be preceded by a comment block that is just about as long as the code itself. As noted above, you can use comments to give proper credit to your sources and you can use comments to identify your submission as your own.
5. Sometimes the building process reveals unexpected results that are themselves very informative in learning. When you completed the exercise above, what did you find that was unexpected? What did you do about trying to understand what had happened? Did you do further exploration? What did that further exploration reveal?
6. Here's a new bonus item: Frustration is actually a powerful source of learning, if you can push through to the "other side" (i.e., you can ultimately work around the source of the frustration). The challenge layer in this exercise is an important tool in this regard: Combining the skills from previous lessons with new skills and applying them to a difficult and novel problem will almost inevitably lead to glitches in the process of constructing your artifact (in this case the R code to solve the challenge). Embrace that frustration and see if you can get through it to deeper learning.