## Task 1.1

For this task I basically tried to replicate the same analysis that was given in the example for this task. So, I was looking for the 10 most probable words in 10 topics using the LDA topic model generation method. I am learning to use the R statistical analysis package so I decided to do the analysis as well as the visualization in R. Here the high level steps that I took to complete this analysis in R (see the R script in Appendix I)

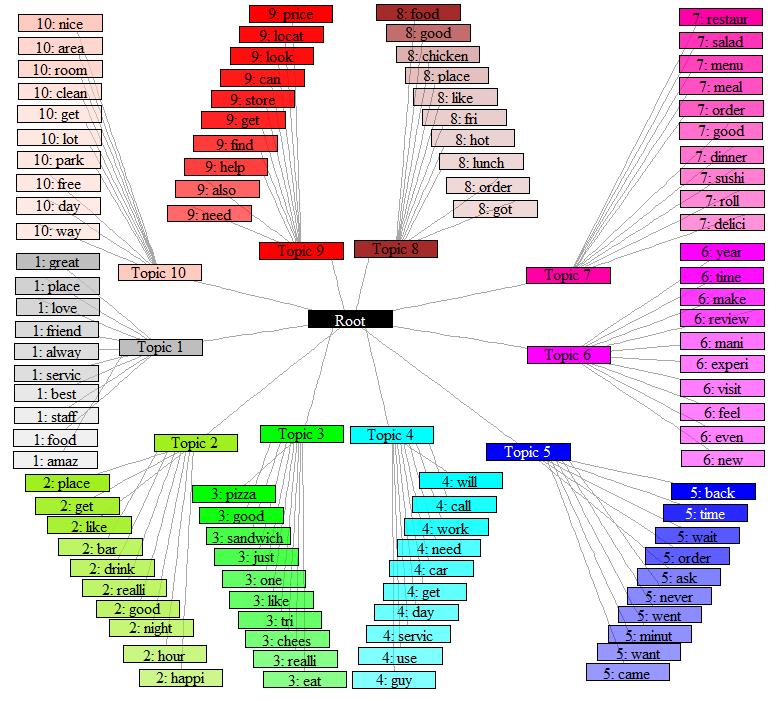
*Calculate the topic model term posterior probabilities*

1. Read in the JSON file into a variable called “review”
2. Use a binomial distribution with success = 0.5 to randomly sample reviews from “review”
3. Create an R corpus from the first 5000 reviews (mostly because if this got too much bigger the analysis would take way too long on my computer)
4. Scrub/clean up the text
   1. Remove numbers
   2. Turn all characters to lower case
   3. Remove stop words
   4. Remove punctuation
   5. Take out any white space
   6. Use word stemming to combine words with similar roots
5. Turn the remaining corpus into a “DocumentTermMatrix” (this is a variable class in the tm package in R) that lists documents in rows and the words/terms in the columns
6. Use LDA model to generate a list of 10 topics
7. Retrieve the posterior probabilities from the LDA model for each of the 10 topics found
8. Create a table of the topics, words and their probabilities to be saved to the hard drive. This is mostly because I wanted to keep this and not rerun the code above again if I could avoid it because it took soooo loooong ☺.

*Create the visualization*

1. Create a links table where I link a root node to the 10 topics
2. Create a nodes table where I list all of the nodes to be plotted
3. Add to the links and nodes tables by appending the links between the topics and their top 10 words and adding all of those top 10 words to the nodes list
4. Create a color scale that allows the most probable term in each topic to be completely opaque (posterior probability of 0 would be completely transparent). Random colors are also assigned to each of the 10 topics.
5. Use “igraph” package in R to create a network diagram with this data
6. Create an initial layout of the network using the Fruchterman-Reingold algorithm
7. Use tkplot function in R to manually manipulate the locations of the nodes in the plot to make it look better
8. Save this layout to file just in case (because the manual manipulation in step 15 also took a long time and I would have hated to lose this)
9. Plot the network to the screen (defining node shape and size, and color which includes the transparency)

Here is how the visualization for task 1.1 turned out:



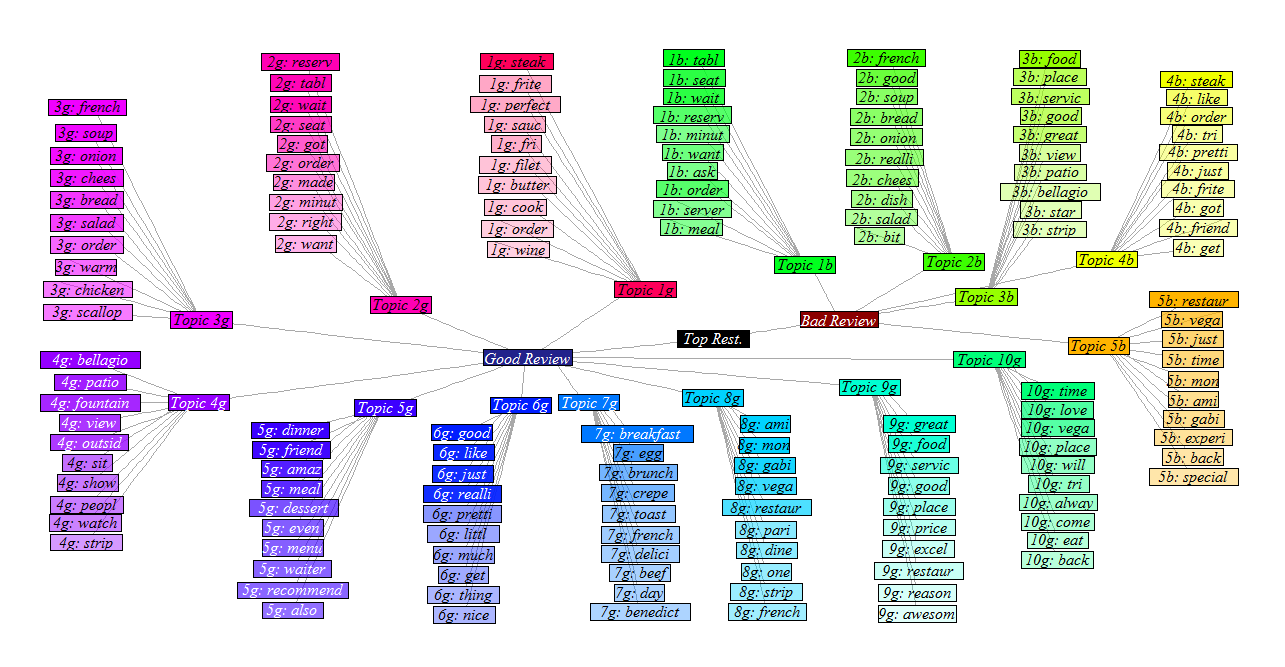
When you look at the visualization for task 1.1, what appears to be jumping is out is the different kinds of restaurants that people review. For example, topic 3 looks like it’s about pizza places, while topic 2 is about bars, or someplace you might go for happy hour, and topic 7 could be sushi places… Not all of the topics follow this pattern (e.g. topic 1 seems like the generic good review for a restaurant with “great”, “love”, “service[e]”, and “amaz[ing]”)

## Task 1.2

For task 1.2, I decided to subdivide the data little differently than the suggested example. I found the restaurant that had the highest number of reviews (by the way this is a restaurant at the Bellagio hotel in Las Vegas that overlooks the fountain) and then split the reviews into good (median rating over higher) and bad (below median reviews) chunks. Since there were so many more good reviews than bad ones, I decided to split the good reviews into 10 topics and bad reviews into 5 topics. I created the topics using LDA much the same way I did for task 1.1. The process was almost identical to the steps taken in task 1.1, so I’ll just list the places where things differed here (see full R script in Appendix II):

* Used R to find the most reviewed restaurant
* Found the median rating for that restaurant
* Subset the reviews into 2 chunks for this restaurant (good and bad reviews)
* For the visualization, I got a little more sophisticated on task 1.2 and included rectangles widths that changed based on the length of the text

This is what the visualization ended up looking like:



You can see in this visualization that the topics in the good review section are much more specific to this restaurant. Topic 4g talks about the location of the restaurant (“bellagio”, “patio”, “fountain”, “view”, “outsid[e]”…) and I guess that the restaurant has pretty good steak because topic 1g is HEAVILY weighted on “steak”. On the bad review side, it appears that people have some concerns about the wait to get a seat at this place (topic 1b: “tabl[e]”… “wait”… “minut[e]”…). It also seems that not everyone likes the steak as much since topic 4b is definitely about the steak as well. Overall, I think this is a very interesting visualization. If I were a restaurant owner I would love to be able to see something like this to help me understand what my customers think of my restaurant.