Evaluation

We evaluated the changes that we applied to the **filter recommendations** and the **search results**.

Filter recommendations:

Originally, we were using MetaMap to suggest related terms that we can offer to the user as filters. However, an issue with the MetaMap terms is that there is no indication of how related terms are with others, and how common these terms are. Because of this, we could not identify which filters would be the relevant to the user. To help with this issue, we wanted to create more personalized filter recommendations. We created a search history which collected the users search terms every time they click on the search button.

Then we used a co-occurrence network where each term is a node, and these nodes are connected with all of their related terms according to MetaMap. We used the degrees of the nodes to indicate how relevant these related terms are.

We evaluated this system by using three series of search terms and found that this method of filter recommendation gave more relevant results. Although there were still some uncommon terms, or terms that we could not find a direct relationship between, such terms were less common for the new filter recommendation system than they were for the previous system.

One issue with this co-occurrence network is that there are some terms that have many related terms. Since more terms means more edges to draw, this can cause the network to take over a minute to load. However, this does not affect the time to perform searches and apply filters – it only applies when the user clicks the "Network" button.

Search results:

To improve quality of search results, we also used a ranking system which organizes the search results based on the number of replies they have. We use the number of replies to measure the amount of discussion a post has, assuming that posts with more discussion are more likely to contain information that the user is looking for. When the user clicks the search button it performs the queries in Solr and then orders the posts by replies before displaying them to the user.