October 2 Discussion Summary

The takeaways from the discussion with Daniel are below, prefixed with the part of the process they pertain to. I've ordered them by this stage.

- (**Problem Statement**) Daniel mentioned that the focus of the project is to extract the triplets, and that the validity of the edges is a different problem. My understanding of the way he phrased it is that false positives are okay, but we want to avoid false negatives (i.e., we're happy to use a pipeline that also yields garbage like, "wet bathing suits cure cancer" if it means it also gives us the rare drug-disease pairs).
- (**Data Collection**) With regard to learning vector representations of words for classification, we could potentially use PubMed, a comprehensive collection of medical papers. Although PubMed only allows us to grab the abstracts (https://www.ncbi.nlm.nih.gov/pmc/tools/ftp/), we could get a subset of the papers through PubMed Central (https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/).
- (**Data Collection**) PubMed has about 28M papers, while PubMed Central has about 2M papers. Each file (PDF) is roughly 1MB. If we downloaded the entire data, this should be roughly 2TB. We can also download ~100MB chunks of data, but Daniel believes even the 2TB is only a moderate amount for the domain. We would need department resources to process (download, extract text, and run model) on this data.
- (**Modeling**) We could potentially simply build a list/dictionary of drug and disease names. If this is possible, it would obviate the need to learn this. Standard NLP techniques such as POS tagging and stemming would be useful here (we do not want to ignore the setence, "*I had skin lesions on my legs ...*" simply because we only had "lesion" in our dictionary).
- (**Modeling**) As an alternative to using a neural network, we could potentially cluster the word vectors, and see what terms cluster together. Do drugs cluster separately from diseases? Or does the word embedding algorithm put them together? What word embedding technique and clustering algorithm would work best (in particular, I'm interested in k-means vs. spectral clustering). Potentially, if the clustering algorithm gave well-separated clusters, we could use an SVM with an appropriate kernel to label each word.
- (**Modeling**) Daniel pointed out how a lot of medical terms are adjectives together with nouns. A paper I found later (https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7747810) uses a similar concept. Their results show that a convolutional neural network architecture does not perform better than traditional methods (and they cite several works that note the superiority of Conditional Random Fields over SVMs). Their paper does not link to their code, but it ought to be easy to implement (and I'll read about CRFs in the meantime). Of course, this says nothing about how a recurrent structure such as an LSTM would perform, but my point was that CRFs are another viable option, and their results seem to indicate that word vectors are better features to pass to the CRF.