Members: Bill Guo (billguo2)

Topic: Generating Hierarchical Topic Taxonomies over Text

We've discussed topic modeling and its uses in class. However, the topic modeling approaches covered, such as LDA, do not capture topic hierarchies and subtopics. For example, in a corpus of documents, LDA might capture the topic "gardening" but not the more specific topic "succulents." Or it might identify both, but not that "succulents" is a subtopic nested under "gardening."

This is an interesting problem because it comes closer to reflecting a human understanding of the world, and how different concepts are related to each. It is also a practical task. For example, I work in advertising technology, and the concept of categorizing keywords into a topic taxonomy to improve targeting and ad relevance is critical. We built a taxonomy manually – we even hire people whose whole jobs is to do this – so it would be interesting to see how to automate this process.

I can gather a dataset of advertising keywords from my workplace for testing purposes. Some of this data was already labeled by hand with a topic hierarchy, so I can evaluate my outputs with it. Specifically, I am interested in:

- Implementing hierarchical topic model methods. Two I found are hierarchical PLSA and hierarchical LDA. This will entail reading and implementing the relevant papers.
- Adding support for different text representations, including bag-of-words and tf-idf representations as discussed in class, as well as word embeddings.
    - I also want to explore fine-tuning pretrained embeddings with a given corpus
- Building a hierarchical modeling algorithm via successive application of flat topic modeling methods like normal PLSA or LDA, and seeing how effective that is.
- Packaging the above (code for creating numeric representations of text data, training/tuning word embeddings, and running hierarchical topic models) into a Python package.
- Evaluating performance on my dataset.

Language: Python

Workload:

- Gathering dataset – 2 hr
    - I will not be sharing the dataset, but will report aggregate performance metrics, and possibly provide some representative examples
- Writing text vectorizers – 3 hr
    - Should be very straightforward; budgeting some extra time for looking into word embeddings and doing transfer learning
- Researching and writing hierarchical topic model implementation – 20+ hr
    - I think this is likely a very conservative estimate of how long it will take me to digest and code a paper. I aim to only implement one of hPLSA/hLDA, but if things go more smoothly than I predict, maybe both
- Building hierarchical modeling algorithm via successive application of flat topic modeling methods – 5 hr