Jiayao:

1. Upload the data of all abstracts.
2. Preprocessing
   1. Remove stop words
   2. Plural to singular
   3. Convert verb tense
   4. Do not convert adverb to adjective
   5. Any additional adjustment you feel necessary
3. Generate a word count table and rank by most to least frequent words.
   1. top a few hundred words may be enough.
4. Create bag-of-words representation for cold (both MSH and Pediatrics), ventricle (both MSH and Pediatrics), secure (Pediatrics), poor (Pediatrics).
   1. The output format should be:
      1. name the folder using the word, eg “cold\_Ped”, “ventricle\_MSH”.
      2. Within each folder, including the following files:
         * “train.txt”
         * “test.txt”
         * “vocab.pkl”
         * “train.labels”
         * “test.labels”
   2. Use the test script I sent you earlier to make sure the files can be loaded properly.

Linying:

1. Run LDA on the four words mentioned above
   1. For the two words appeared in MSH (“cold\_MSH”, “ventricle\_MSH”), evaluate accuracy on the test set.
   2. For the instances appeared in Pediatrics, sample a few from each cluster for Katherine to review. Report accuracy, and other observations about how things work/fail.

Katherine:

1. Project can be found on my github: <https://github.com/zhangly811/WordSenseDetection>

To clone this repository,

* 1. open your terminal, in your terminal:
  2. cd into the folder where you want this project to be, eg. “cd foldername”
  3. git clone https://github.com/zhangly811/WordSenseDetection.git

1. All data can be found at google drive (give me your gmail so I can add you)
2. Take a look of the abstracts, or the word count table and find a few more ambiguous words (prefer nouns ;))
   1. You can also take a look of the two labeled dataset WSH and NLM to see what words they found ambiguous, if this helps orient you.
3. Think about how to show the results during the demo. Sketch some ideas and we can discuss.