# Progress Made

Below is the progress made on ArXiv data set and model. All code relating to this step is in [this branch](https://github.com/sanjibg01/CS410-project-team-trailblazers/tree/md_preprocess_abstracts) of our Github repository. The steps involved included:

* Read in the JSON
* Preprocessed the text:
  + Tokenized
  + Removed numbers
  + Lemmatized
  + Removed stop words
  + Limited to nouns (trying this for the topic modeling)
* Created features for modeling from the preprocessed ArXiv data
  + Created a doc-term matrix and transformed with TF-IDF
  + Currently using single tokens, but may n-grams as well depending on topic modeling results
* Created some very initial topic models, with the main goal of understanding the library

I used the Python library spaCy for the text processing. This library was easy to learn how to use and very helpful, particularly for the part-of-speech tagging that let me easily limit to nouns.

I then created the document-term matrix and applied the TF-IDF transform using sklearn’s feature\_extraction library. I am creating the topic models with sklearn’s implementation of Latent Dirichlet Allocation.

Below is the progress made on the Coursera lecture data set. Code related to this step is in [this branch](https://github.com/sanjibg01/CS410-project-team-trailblazers/tree/paulzuradzki) of our Github repository.

* Script that downloads Course transcripts (using coursera-dl Python library), parses them into JSON files and writes to Cloud storage so that all team members can access from a centralized place.
* Preprocessed the text using metapy:
  + Tokenized
  + Lower case
  + Length filter
  + Lemmatized
  + Removed stop words from lemur stopwords list and manual list unique to transcript data i.e. ‘[MUSIC]’, ‘[SOUND]’, ‘[INAUDIBLE]’
  + Created unigrams (trying for topic modeling)

# Remaining Tasks

* Read up on Latent Dirichlet Allocation to better understand the model and its parameters.
* Determine how to quantify the performance of the topic modeling, and use this to test parameters and pre-processing steps against one another to select the best model for the arXiv dataset. This will involve setting up a train/test splitting scheme.
* Create feature set for and conduct topic modeling on the coursera transcript dataset.
* Link the content between the data sources using the topic modeling results.

# Challenges/Issues

* For the arXiv dataset, I am currently working with a small subset of my data to build the pipeline and understand topic modeling more broadly. I’ll need to expand to the full dataset, which might lead to challenges with computation time and slow down progress.
* Because we are matching the results from a broad-topic data set (arXiv) to a more narrow-topic data set (Text Info Sys lectures), it’s possible that matches for some keywords/topics will be limited or that we’ll get the same matches back across many queries.