Arda Bedoyan

Technology Review

CS 410: Text Information Systems

Topic Modeling Toolkit Comparison

Topic modeling is an important and useful component of text mining and analysis. Topic mining allows the discovery of trends and insights for text documents. There are a variety of toolkits available for topic mining. I will be discussing three such toolkits; namely, the Stanford Topic Modeling Toolbox, MeTA, and Gensim.

The Stanford Topic Modeling Toolbox (TMT) was developed and released in 2009. TMT has the ability to train topic models using Latent Dirichlet Allocation (LDA), Labeled LDA, and Partially Labeled LDA. It allows for users to select the parameters and can generate an output compatible with Excel to be used for further analysis. To begin using TMT, a user must import a CSV or TSV file that contains text-heavy data to be analyzed. After loading in a file, users can extract and tokenize text. TMT allows for a user to select the column of text data to be analyzed, and multiple columns can even be combined into one. There are features that allow for the removal of stop words and for tokens to be stemmed. Very common words and very rare words can be removed as well. Then parameters can be chosen to train the dataset using LDA for instance. The generated output includes a description of the model, a topic distribution for each document in the dataset, and the probability of each term in a specific topic. Although TMT has a lot of useful features and seems straightforward to learn, it uses an old version of Scala and a linear algebra library that is no longer maintained. Therefore, trying to adapt TMT into code for a larger project may be very challenging. Additionally, TMT seems to only work with CSV and

TSV files, which makes it difficult to adapt to projects that include the analysis of text files. For instance, my final project is about analyzing the genres (i.e., topics) of movie scripts. I do not want to have to transfer the scripts into CSV files and separate lines or words into different rows for TMT to be able to be used. Because of its age and the input file limitations, TMT is a great tool to be aware of, but perhaps not one to actively use in projects.

The Modern Text Analysis (MeTA) toolkit is one that has many different features that can be used for text retrieval and mining tasks. The MeTA toolkit allows for generating topics in a general way using an LDA topic model. It allows for different inference methods, such as collapsed variational bayes, collapsed Gibbs sampling, and parallel Gibbs sampling. Once an inference method is chosen and parameters are set, such as the number of iterations and topics, then the toolkit can begin to run. The program will be run for either the maximum number of iterations or until the model has converged, whichever comes first. MeTA will report out the number of topics found and for each topic, the top words associated with it. The number of words per topic output depends on what the user specified before inference was run. Because MeTA works with python and I have used it already for homework assignments during this class, it might be one worth exploring when it comes time to my project. Some limitations I will need to work around will be to have the program label the topics as genres, instead of numbers 0, 1, 2, etc. If the topics can be labeled and the probability of each topic in each input file can be outputted, then this would work well for my project idea about analyzing genres based off of movie scripts.

The last topic modeling toolkit I looked at was Gensim. Gensim can be used to train semantic natural language processing models, represent text as vectors, and to find semantically related documents. It is fast and simple to install, and is compatible with any platform that

supports Python, such as Windows, Linux, and Mac OS. The basic concepts needed to understand and use Gensim are document, corpus, vector, and model. A document is some text, a corpus is a collection of documents, a vector is a mathematical representation of a document, and a model is an algorithm to transform vectors. Gensim allows for a variety of preprocessing steps, such as tokenization and stop word removal. Vectors are used to represent the documents through numerical representations. For example, a vector can represent how often each word in a dictionary occurs in a document. Once the corpus is vectorized, it can begin to be transformed using models, such as the tf-idf model. This model will return vectors where each word in a document is weighted based on its frequency in the entire corpus. One feature available with Gensim is the ability to find the similarity between documents. Given a query document, we can run Gensim and find a percentage of how similar that query is to the other documents in the corpus. This can help with similarity-based tasks and can be used in recommender systems, to recommend a text based on how similar it is to another.

Of the three topic modeling toolkits discussed, Gensim seems to be the most robust and provides the most training documentation to allow a beginner to learn how to perform topic modeling. Stanford TMT is old and is hard to adapt to newer projects; although, it seems simple to use and convenient for spreadsheet-based input files. MeTA's topic modeling approach was focused on finding topics and outputting the top words associated with those topics. However, it seems like further analysis would need to be done to define what the resulting topics are. Gensim offers a wide variety of models for topic modeling and has a lot of open source material and tutorials to understand all that it can do. Ultimately, MeTA or Gensim both seem like viable options for topic modeling projects and are top contenders for the toolkit I will use for my final project.

Works Cited

https://downloads.cs.stanford.edu/nlp/software/tmt/tmt-0.4/

https://meta-toolkit.org/topic-models-tutorial.html

https://radimrehurek.com/gensim/index.html

https://radimrehurek.com/gensim/auto_examples/core/run_core_concepts.html#sphx-glr-auto-examples-core-run-core-concepts-py