CS 410 – Fall 2020 – Tech Review – Zach Dick – zdick2

Technology Review: A Comparison of Topic Modeling Toolkits

1. Intro:

This technology review is going to be a comparison of the three following topic modeling toolkits: MeTA, Genism, and the Stanford Topic Modeling Toolbox. This will review the set-up process, ease of use, and versatility of each of these toolkits. The review process will be based on the documentation available for each toolkit. The structure of this article will provide a summary of each of the 3 toolkits separately followed by an analysis, which compares them all.

2. MeTA:

The MeTA toolkit has a relatively simple set up and can be used on Linux, Mac, and Windows Operating Systems. This project is implemented in C++ as a cmake project. Each OS has its own prerequisites, but in all cases the users need to clone the meta git repository, initialize the submodules, and build and test the project.

Before running the topic-modeling functions in this package, the user needs to create an index of the documents and write a *config.toml* file to configure the model. When these steps are completed, the user should run the *lda* executable, passing it the *config.toml* file, to create the topic model. Printing the results of the topic model includes a run of the *lda-topics* executable. The user will need to pass in the *config.toml* file, the path to the phi file, and the number of top words to display from each topic as parameters for *lda-topics*. To confidently use this package the user would likely need at least an intermediate level of programming knowledge. C++ programming is one of the harder languages to use correctly, but it is also a common and seasoned language amongst the programming community so there are a lot of resources available to be able to get things right.

MeTA has 3 different inference methods of the LDA model. These methods include collapsed variational bayes, Gibbs sampling, and approximate parallel Gibbs sampling. To specify which inference method to use, the user should assign either “cvb”, “gibbs”, or “pargibbs” to the *inference* variable in the config.toml file. Users of the MeTA toolkit have the option to implement their own topic modeling methods since they will have the source code on their local machine. Upgrading to the latest versions would be relatively easy given that you would pull in the latest version from github and rebuild the project.

3. Genism:

Genism is implemented in python. Therefore, the setup process requires python to be installed and for the user to run the python installation command for the genism library (e.g. pip install –upgrade genism). This can be used on Windows, Mac, or Linux.

To use this package, the user must be familiar with the Python programming language. Python is an easy language to use and is also immensely popular. It enables the programmer to not have to think about things like memory management and data types. This package has extensive examples on their website as well. This library is also compatible with *numpy* and *scipy* which improve its ease of use because these packages are quite popular amongst the Python programming community.

Genism has many options for VSM transformations including *TF-IDF, LSI (LSA), Random Projections, LDA,* and *HDP*. There is a developer guide and API Reference available to ensure simplicity for users who wish to add their own VSM based transformation algorithms.

4. Stanford:

Stanford is implemented in Scala, but they state that knowledge of Scala is not required to use this library. Before installing this toolbox, users need to have a text editor for creating TMT processing scripts, and an installation of JAVA greater than or equal to version 6SE. This can be used on Windows, Mac, or Linux. Once the prerequisites are met the user can then download and save the TMT executable to their computer. This is a large file so it will take a while to download. Once downloaded the user should double-click the file or run from the command line “java -jar tmt-0.4.0.jar” to open the toolbox.

Before starting, the data needs to be converted into a CSV file. That means that other tools will likely need to preprocess any text documents so that they can be converted to CSV format. Once converted, the tool’s Scala scripts need to be updated to further prepare the data. Scala is not as well-known as other programming languages, I personally have never used it before, so for me to update these scripts I would need to at least spend time learning the syntax of the language.

With this package the user can choose between LDA, Labeled LDA, and Partially Labeled LDA. The inference methods to choose from are CVB and Gibbs. Users also have the option to download the source code for this project for when they wish to make updates to it. They offer an API Reference to help with this.

5. Analysis:

If starting from scratch, the setup of each of these packages is comparable in difficulty. All three packages have thorough setup instructions available on their websites. The download process for the Stanford package would have to be the simplest since the user is not required to use the command prompt for any of the steps. This means that they are less likely to run into issues with other packages such as git, pip, or cmake (e.g. missing environment variables).

Both C++ and Python are well known and well-established programming languages. Scala is not as well known (for instance I have never used it). Python is a much easier language to learn and safely use than C++ due to the user not needing to worry about memory management. MeTA can process the data set with its own library’s tools. Genism can be used seamlessly with the tools that process the data for it. Stanford requires the data to be put into CSV format before it can begin processing it. Therefore, I would say that Genism is likely the easiest of these libraries to use.

Both Stanford and MeTA implement LDA variations exclusively. Genism offers other VSM based algorithms to the user, and have detailed instructions allowing users to implement other algorithms. MeTA is naturally set up for users to be able to implement their own algorithms due to the set-up process involving cloning the git repository of the project. Stanford has the option to download the source code and make updates, but the process is not as easily achieved as with the other libraries. Due to C++ being a lower-level language than the others it naturally allows more versatility in how a user can implement their own solutions, and these solutions will most likely be more efficient than implementing them in higher level languages such as python or Scala. Therefore, from a base functionality perspective Genism is the most versatile, but from the perspective of user implementation possibilities MeTA is the best option.

6. Conclusion:

In conclusion, selecting a toolbox to use always depends on the needs and skills of the user. This review compared three popular Topic Modeling toolkits, MeTA, Genism, and the Stanford Topic Modeling Toolbox. This review determined that Stanford had the easiest set-up process, Genism is the easiest to use and has the most algorithm options, and MeTA is the best option for users to implement their own algorithms.