MUD 2014 Tutorial 2

Latent Dirichlet Allocation (LDA)
Abram Hindle, University of Alberta, Canada

Abstract

Topic analysis has become a hot topic in software engineering research, especially mining software repositories research. It is popular because often researchers have to sift through numerous documents and try to make sense of 1000s of bug reports, commits, or source files. Latent Dirichlet allocation (LDA) allows Software Engineering researchers to apply topic analysis to large sets of documents and come up with topics that cross-cut and relate documents. These topics are often viewed as similar to human conceptions of topics, but they are different, these topics are independent statistical signals observable within a collection of documents.

Topic analysis has been used to summarize behaviours and efforts within software repositories. It has been used to describe coupling between documents, imply document similarity, query and cluster documents. In many cases topic analysis is attractive because it is an unsupervised method that produces couplings and relationships between the multitude of documents, words, and topics.

In this LDA Tutorial participants will get hands-on experience with LDA and get some valuable LDA tools. We will discuss LDA, when one should use LDA and when one should not. Then we will go through the steps of collecting and extracting documents, cleaning them up for processing by an LDA tool, producing topics and then evaluating topics.

In particular we will extract issues from existing Github issue trackers and pre-process this data. We will strip stop-words, common words and extremely rare words and summarize each document as a bag of words. Then we will build a dictionary of the remaining terms and feed an LDA implementation, these words and topics.

The LDA implementation given the documents and parameters will analyze the documents and generate topics and relationships. Then we will extract these matrices that relate the topics, documents and words together. We will analyze, inspect, plot, and interpret these matrices.

Participants should pre-install Virtualbox (https://www.virtualbox.org) and Vagrant (www.vagrantup.com). A vagrant VM loaded with free software will be distributed to participants. This will enable participants to explore topic analysis on their own and at their leisure as well.

Resources for the tutorial will be found here: https://bitbucket.org/abram/lda-tutorial.
Biography

Abram Hindle is an assistant professor of computing science at the University of Alberta. His research focuses on problems relating to mining software repositories, improving software engineering-oriented information retrieval with contextual information, and the impact of software maintenance on software power use and software energy consumption. Abram received a PhD in computer science from the University of Waterloo. Contact him at abram.hindle@ualberta.ca.