**Scalable Curation of Email with Open-Source Tools: Review, Appraisal, and Triage of Mail (RATOM)**

**Tutorial Submission (2-page Abstract)**

Christopher (Cal) Lee and Kam Woods

Email has served, and continues to serve, as the communication and recordkeeping substrate of numerous contemporary phenomena across all sectors of society. A small but essential portion of the email has continuing value and warrants long-term preservation. Many collecting institutions have acquired email accounts and collections. Unfortunately, the pipeline from email acquisition into digital preservation environments is still relatively immature. One of the fundamental challenges has been that formats for packaging email accounts (particularly PST) can be brittle, complex and dependent on proprietary software. Another fundamental challenge is that information within email can be subject to various sensitivities (e.g., protected health information, financial information) that collecting institutions must address. Without reliable and scalable tools to address these two challenges, email risks being locked into relatively hidden collections. Quite the opposite of date for all!

The creation of Archival Information Packages (AIP) depends on a variety of functions including, but not limited to: extracting email content from proprietary packages; locating messages with retention value; identifying instances of sensitive and personally identifying information; and accurately tagging features of email messages, including those corresponding to real-world entities such as persons, places, organizations, and events.

The Review, Appraisal, and Triage of Mail (RATOM) project presented in this tutorial began as a two-year effort to develop and test software and workflows to support the review and processing of email in collecting institutions. The RATOM project developed software to scan email archive files (including PST, OST, and mbox) and record and export content, metadata, and derived features such as entities identified using natural language processing (NLP) into a simple SQLite database that can be queried as part of a larger set of digital curation workflows.

The RATOM tools are designed to minimize the effort required to run computationally complex email analysis tasks. For example, RATOM provides a single-command tool that makes it simple to replace the default NLP model used to identify entities with a model for a different language, a custom model, or a multi-language model. While some existing LAM access systems incorporate NLP to describe the contents of collections, this technology is often tightly coupled to the platform being used, or is applied strictly to file types that tend to share common structures and metadata.

Email contains encoded text, markup, and attachments, but importantly also structured metadata in the header that can be used to cue identification of persons and organizations and describe their relationships. Identifying entities, relationships, and other features of interest by processing open text from heterogeneous collections of files (such as those extracted from a disk image) is inherently “noisier,” as the extracted text will often contain patterns of features (such as persons, places, and organizations) common to a wide range of devices and production environments (e.g. documentation of system files). By exposing header metadata features in database entries where they are explicitly linked to entities identified in open text, the RATOM tools provide a mechanism by which cross-format search procedures can be easily implemented.

In this tutorial, participants will learn to work with the core RATOM tools, including the email processing library and associated command-line utilities. Using publicly available corpora including PST files from the Enron collection, participants will explore the different options provided by libratom and its utilities to extract content and metadata from email backup files, scan content for entities of interest, and query the SQLite database it produces as output. The tutorial will also include an introduction to selecting and working with pre-trained spaCy language models, and provide participants with a clear understanding of which models are appropriate for which tasks and use cases.

No programming experience or prior experience with command-line tools is required. Participants may bring a laptop with Windows 10 or 11, macOS 11 or 12, or modern Linux distribution (Ubuntu 22.04LTS or later) to fully participate. Instructions for installing the software in advance may be found at <https://github.com/libratom/libratom/blob/master/README.md>. However, it is not mandatory for participants to run the software on their own machines. The tutorial will be conducted in part using remotely hosted interactive notebooks and a web-based SQLite database browsing utility.