Mitigating the Metadata Mess: Autonomous Metadata Extraction Pipelines for Large-Scale Data Repositories

Matthew Chen
University of Illinois at Urbana-Champaign

Erica Hsu
Carnegie Mellon University

Tyler Skluzacek (Advisor)
University of Chicago

Kyle Chard (Advisor)
University of Chicago

Introduction

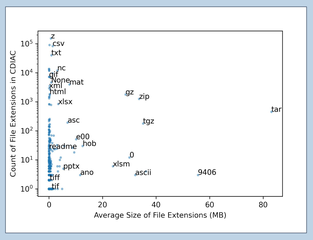
- Rapid accumulation of scientific data emphasizes the need for data to be findable, accessible, interoperable, and reusable (FAIR)
- Making data FAIR requires that data be associated with rich and useful metadata [1]
- Current metadata extraction systems such as Xtract [2] and Apache Tika naively associate files with their corresponding extractors based on MIME type/file extension; however, such approaches are prone to error and lead to wasted resources when applying incorrect metadata extractors
- We propose a scheduler which intelligently maximizes the amount of valuable metadata extracted given compute constraints by leveraging recent developments in file type identification (FTI)
- We evaluate the efficiency and effectiveness of our scheduler using various metadata metrics

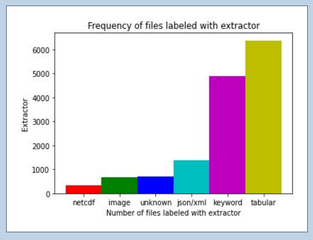
Dataset

Carbon Dioxide Information Analysis Center (CDIAC) data repository. This contains:

- over 500,000 files of diverse types, structures, and sizes
- Over 150 unique file extensions

We use a 20,000-file subset of this data





Metadata Extractors

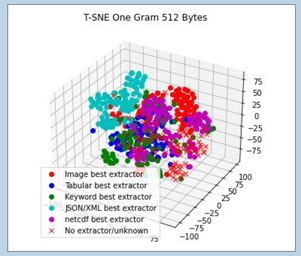
Extractor	Description
Tabular	Column headers and general aggregates (means/medians, etc.),
Image	RGB values and image type
Keyword	Frequency of most common words
NetCDF	Global attributes, descriptions of data (e.g., min, max, units, computational representation)
JSON/XML	Headers, encapsulation depth, class representations, data summaries

Data Type Predictions

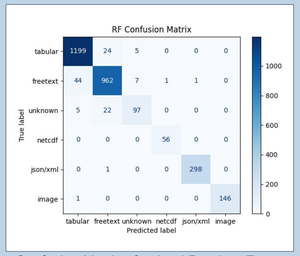
- Use predictive methods to determine file type from a small sample of file content
- Extract 512 byte heads from files for both visualization and classification
- T-SNE visualizations show visually distinguishable clusters
- We applied SVM, Logistic Regression and Random Forest models
- Used an 80-20 train/test split on byte heads
- All three models performed well with accuracy, precision, and recall above 90%

Metric	SVM	Logit	RF
Precision	92.67	90.41	95.13
Recall	93.58	91.13	96.06
Accuracy	93.27	90.66	95.67

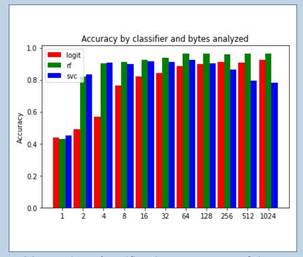
Comparing classification metrics for SVM, Logistic Regression, and RF



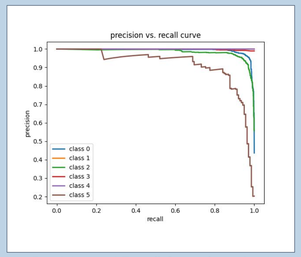
T-SNE Visualization of 512 Byte Heads



Confusion Matrix of trained Random Forest Model

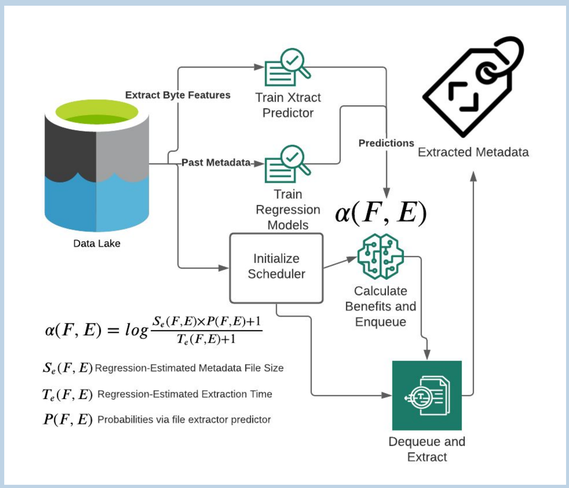


Measuring classification accuracy of the three models by byte head size



Precision and Recall Curve for trained Random Forest model

Probabilistic Extraction Pipeline



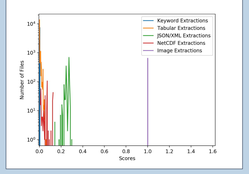
Scheduler Workflow

- The goal of the scheduler is to intelligently order the queue of file/extractor pairs to be processed such that the amount of metadata generated is maximized with minimal costs.
- We use regression models correlating file size vs. metadata file size and file size vs. extraction time and the file type predictor
- We apply an objective function $\propto$ with inputs of a file/extractor pair and outputs the expected benefit of computing the file/extractor pair as an ordering metric in the scheduler's queue
  - Incorporates values from regression models and file type predictor to calculate output
  - Higher values from the objective function suggest higher benefit and thus the file/extractor pair are at the front of the queue

Assessing Metadata Quality

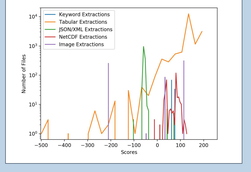
- We defined three metrics to capture *metadata quality*
- These metrics aim to capture how effective the metadata is at fulfilling its intended purpose
- Metadata is ultimately used to aid in the discovery and use of data

Completeness
Did we get all the metadata possible?



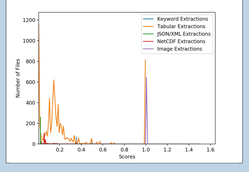
Distribution of completeness scores by extractor

Readability
Can a human read the metadata?



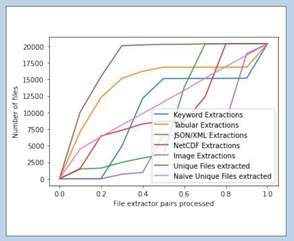
Distribution of readability scores by extractor

Entropy
How unique is the metadata?

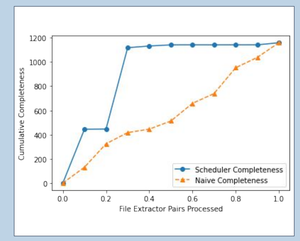


Distribution of entropy scores by extractor

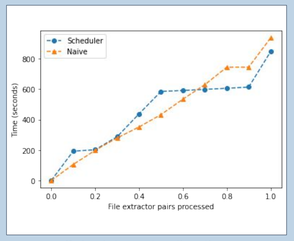
Evaluation



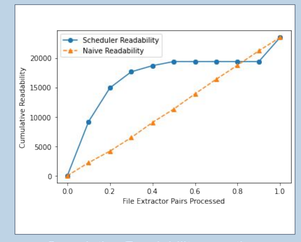
Prevalence of extractors in queue



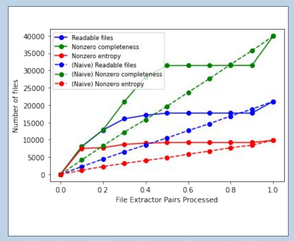
Cumulative Completeness over time



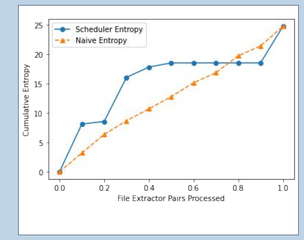
Scheduler extraction time



Cumulative Readability over time



Number of files with quality metadata



Cumulative Entropy over time

Conclusions

- As the percentage of files increases the marginal cumulative quality of the metadata decreases, suggesting that the scheduler successfully extracted the files with the most valuable data first
- Scheduler outperforms the naive approach
  - Sequentially applying each extractor to each file produces a linear increase in quality of metadata
  - Scheduler discovers an ordering of extractions that exponentially increases the amount of metadata quality generated early in the extraction queue
  - Because of the ordering, the scheduler approach runs slightly faster
- Scheduler tries to run one extractor on each file before running the rest
  - When computation is limited, i.e., if only 50% of file extractor pairs can be run, the scheduler emphasizes extracting the most important metadata from each file
  - Provides a great estimation for cataloging data

References

- [1] Jane Greenberg, Hollie C White, Sarah Carrier, and Ryan Scherle. A metadata best practice for a scientific data repository. Journal of Library Metadata, 9(3-4):194–212, 2009.
- [2] Tyler J Skluzacek, Ryan Wong, Zhuozhao Li, Ryan Chard, Kyle Chard, and Ian Foster. A serverless framework for distributed bulk metadata extraction. In Proceedings of the 30th International Symposium on High-Performance Parallel and Distributed Computing, pp 7–18, 2020.