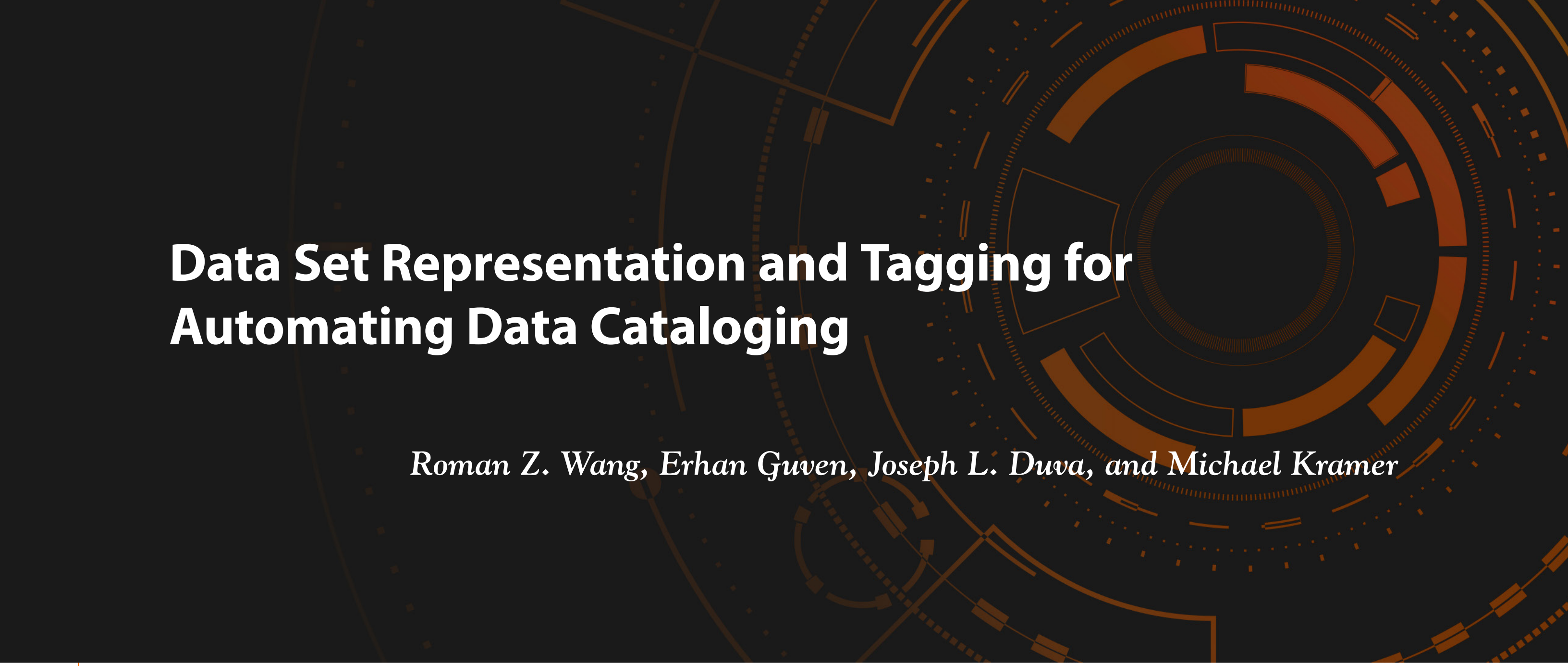


Data Set Representation and Tagging for Automating Data Cataloging

Roman Z. Wang, Erhan Guven, Joseph L. Duva, and Michael Kramer

ABSTRACT

In the last two decades, considerable increases in computing power and available data have led to an analytics and machine learning (ML) revolution. To make knowledge management less cumbersome for human operators, a team of researchers at the Johns Hopkins University Applied Physics Laboratory (APL) proposes an ML–based method to help automate knowledge management. This method discovers new data, represents it with descriptive metadata, automatically categorizes the metadata, auto-populates a data catalog with data sets, and evaluates the new data sets for data fusion options. We focus on a framework that can potentially leverage human–machine teaming to significantly reduce the human resource burden to develop and maintain an accurate accounting of existing data and capabilities within an organization. We explored numerous ML options to test our core hypothesis—that ML techniques can be employed to reliably determine the fundamental topic that an unknown data set represents, leading to increasingly granular data set recognition as more characterization and context information can be mined in the metadata extraction phase. Ultimately, we demonstrated that multiple classifier techniques exist that can predict data set topics with close to 90% accuracy, and some with 60%–80% accuracy, across multiple topics.

BACKGROUND

In the last two decades, considerable growth in computing power and available data have led to an analytics and machine learning (ML) revolution; yet, anywhere from 60% to 73% of all data within an enterprise remain unused.[1] In addition, only a minuscule fraction of the zettabytes of data on the web is being used for analytics. Not only is finding relevant data costly, so is cleaning the data—that is, detecting and correcting corrupt/inaccurate parts of the data. Surveys[2] and studies[3] estimate that data scientists spend 50% to 80% of their time gathering and preparing the data. Labeling metadata for organization and storage is another daunting task. Reducing these costs through machine assistance has enormous potential to free up time and resources to focus on analytics and decision-support automation for all fields and disciplines.

RELATED WORK

A data catalog is an organized inventory of metadata, or "data about data," such as a digital image's time and date of creation, file size, or means of creation. Modern data cataloging platforms include access controls, search, discovery, metadata curation, and collaboration.[4] Many data cataloging systems assume that all data sets have corresponding subject-matter experts who can readily label the data set with metadata. Existing data cataloging products have recently begun integrating ML to recommend actions based on prior user interaction and column matching learned from previous labels.[5] However, these products do not perform well with unseen data sets in the wild—meaning real-world data sets not presented to the algorithms during training—without the help of subject-matter experts, especially in situations with novel data demands.

A graph-analytics–based data set search method is introduced in a patent disclosure;[6] however, it relies on metadata labels being known beforehand.

One academic paper discusses a new algorithm that can recommend common data sets cited by research papers. However, this algorithm relies on prior research paper text and citations and cannot be immediately extended to instances that do not have paper–data set associations.[7]

Some supervised learning methods can classify table types such as relational/nonrelational, genuine/nongenuine, and table format. Unsupervised table representation learning approaches have also appeared in academic literature.[8–10] These approaches train a model on a combination of tasks that include row population, column population, table retrieval, masked token prediction, cell prediction, and cell type annotations using the Wikitable, WDC WebTable, and Common Crawl data sets.[11–15] Some of these approaches can also retrieve tables based on keyword input. However, retrieval differs from labeling in that retrieval tasks do not store associated labels for further analytics. Finally, one method can name tables based on table contents and page metadata.[16] This approach can generate a title but cannot rank and select the best topic among a set of topics. Overall, these methods are based on web tables often from single sources like Wikipedia, which are quite different from data sets found in the wild.

INTRODUCTION

To make knowledge management less cumbersome for human operators, a team of APL experts called the Rosetta Data Stone team proposes an ML-based method to help automate knowledge management. This method discovers and categorizes new data through a process of acquisition, representation, and model-driven recognition (Figure 1).

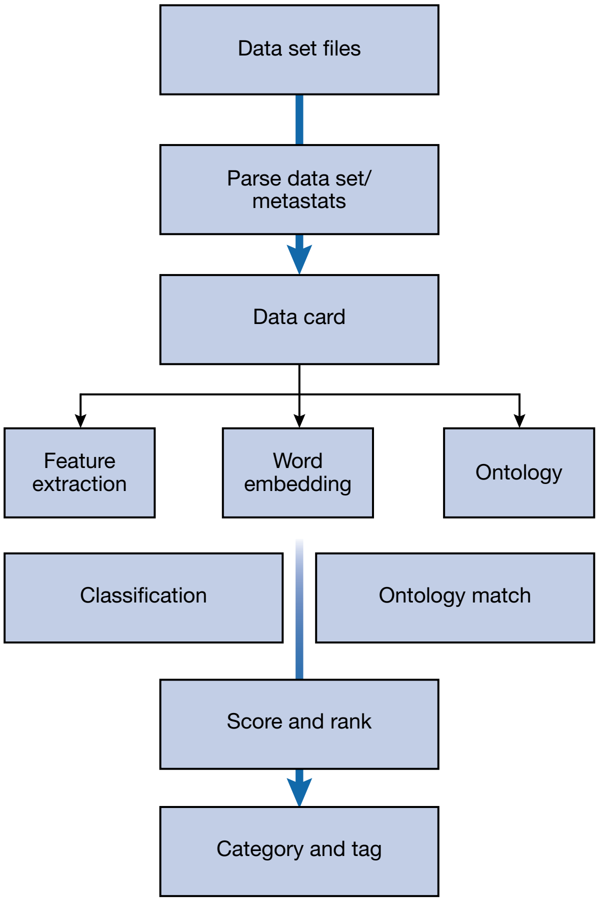


Figure 1. Overall pipeline predicting a category or creating a tag from a data set file or a set of data files. A corpus of data set files is used to create data cards, feature vectors, knowledge graph scores, classification, and similarity scores. Categories are used to mark relevant data set files for the ML models of interest. Tags are used to store the data set files in data set catalogs for faster and more relevant access.

Not only does our approach help address the cost issues associated with data management, but it also increases an organization's level of data readiness with the ability to acquire or identify sufficient usable data in preparation for unknown future tasks. For example, adverse events like the COVID-19 pandemic left many organizations scrambling to acquire and organize data. Our approach enables organizations to discover the resources in the wild, discover the resources they have at hand, and mobilize data for unexpected situations at a moment's notice. Furthermore, a fully matured version of this technology can organize and maintain a data bank from accessible data on the internet and an organization's intranet.

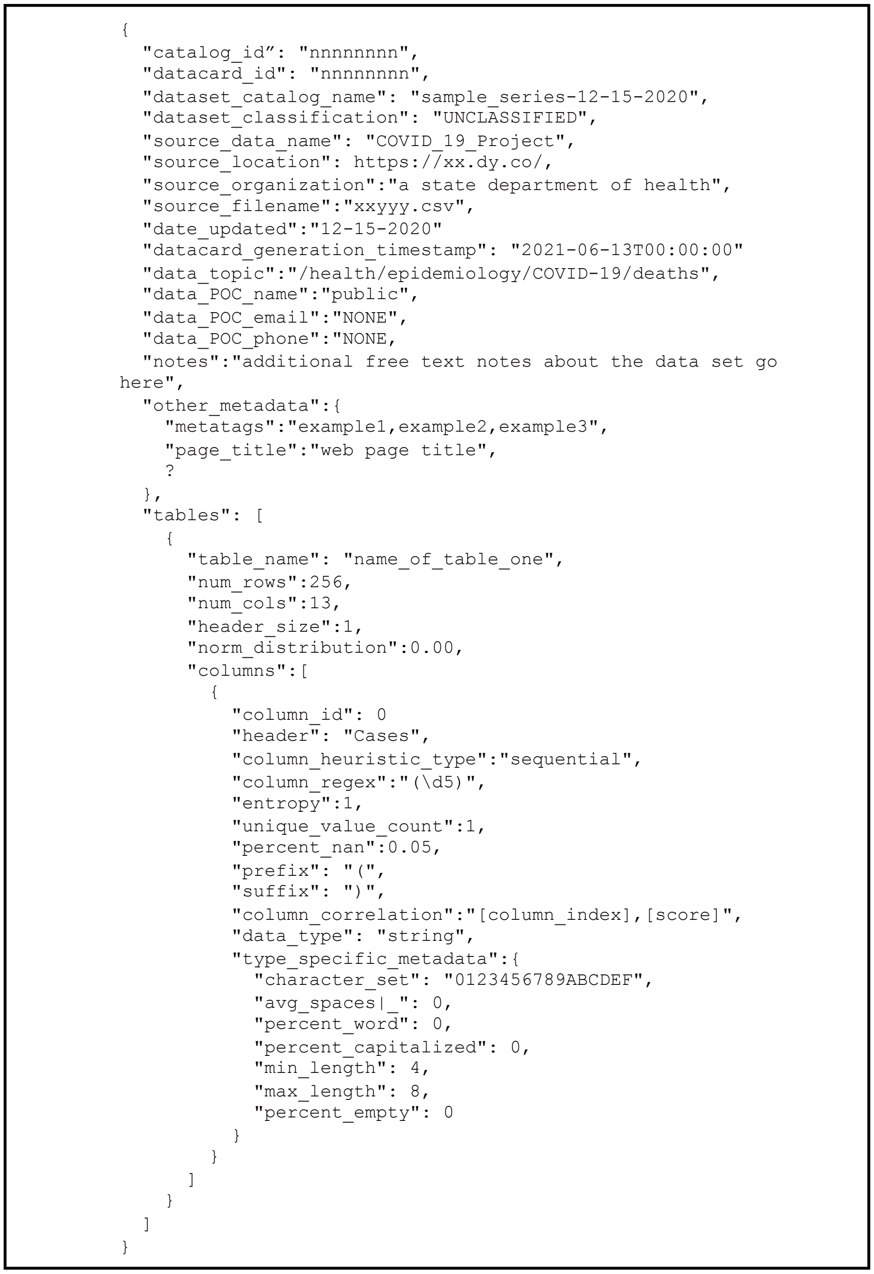
We aimed to tackle the problem of data set categorization and topic recognition. Our core hypothesis was that ML techniques can reliably determine a representative topic for a data set, leading to increasingly granular data set recognition as more characterization and context information can be mined in the metadata extraction phase. Much like how natural language processing (NLP) has successfully determined the subject or topic of written passages, our ML techniques can evaluate the subject matter of raw data sets.

METHODS

Data

The Rosetta Data Stone team initially faced the same data curation problem that we aimed to solve. We first had to find, collect, and extract the data set metadata for training and testing of our topic classifier models. At the start of the project, we only had access to the myriad of COVID-19 public data sets published and collected for daily tracking of the status of the pandemic at the local level. However, these data sets had little variation and required many hours of manual labeling. We quickly pivoted from COVID-19 data sets as our primary data source and developed additional collection agents for data.gov and Kaggle, both of which provided greater variation of topics, more text metadata, and preexisting labels. The COVID-19 data focused on low-level variations in COVID-19 topics, and the data.gov and Kaggle data sets spanned a broad set of topics.

We created a standardized representation of a data set's metadata, called a data card, to establish a universal data set metadata template. Each data card contains information such as the data owner, source location (or URL, for web data), date/time of collection, and file name. Metadata are stored in a key-value pair format to flexibly support the vast differences in metadata fields between data cards and addition of new data fields in the future. Additionally, the key-value pair format allows us to better capture hierarchical metadata such as table names (sheets in Excel), column names, and a set of computed attributes for the data in each column, shown in Figure 2.



```
{
  "catalog_id”: "nnnnnnnn",
  "datacard_id": "nnnnnnnn",
  "dataset_catalog_name": "sample_series-12-15-2020",
  "dataset_classification": "UNCLASSIFIED",
  "source_data_name": "COVID_19_Project",
  "source_location": https://xx.dy.co/,
  "source_organization":"a state department of health",
  "source_filename":"xxyyy.csv",
  "date_updated":"12-15-2020"
  "datacard_generation_timestamp": "2021-06-13T00:00:00"
  "data_topic":"/health/epidemiology/COVID-19/deaths",
  "data_POC_name":"public",
  "data_POC_email":"NONE",
  "data_POC_phone":"NONE,
  "notes":"additional free text notes about the data set go
here",
  "other_metadata":{
    "metatags":"example1,example2,example3",
    "page_title":"web page title",
    ?
  },
  "tables": [
    {
      "table_name": "name_of_table_one",
      "num_rows":256,
      "num_cols":13,
      "header_size":1,
      "norm_distribution":0.00,
      "columns":[
        {
          "column_id": 0
          "header": "Cases",
          "column_heuristic_type":"sequential",
          "column_regex":"(\d5)",
          "entropy":1,
          "unique_value_count":1,
          "percent_nan":0.05,
          "prefix": "(",
          "suffix": ")",
          "column_correlation":"[column_index],[score]",
          "data_type": "string",
          "type_specific_metadata":{
            "character_set": "0123456789ABCDEF",
            "avg_spaces|_": 0,
            "percent_word": 0,
            "percent_capitalized": 0,
            "min_length": 4,
            "max_length": 8,
            "percent_empty": 0
          }
        }
      ]
    }
  ]
}
```

Figure 2. Example of a data card. The data card presents metadata in a standardized way. In addition to the expected metadata fields like file name or creation date, our data card contains a hierarchical set of attributes to capture table names, column names, and column attributes—all important pieces of information for training the ML algorithms.

We created 119 data cards from COVID-19 data sets from public web sources, with data on cases, deaths, tests, vaccinations, and hospitalizations. We also created 11,013 data cards from data sets found on data.gov[17] and 1,460 data cards from data sets from kaggle.com,[18] each with its own set of topic classes. The COVID-19 data cards were combined with the coronavirus data cards into a single coronavirus topic class inside the data.gov data card set.

To automate data card creation, we developed custom Python data collection agents for COVID-19 data collection, data.gov, and kaggle.com. We provided the COVID-19 data collection agent with links to downloadable data files. In contrast, the data.gov and kaggle.com agents systematically crawled each website looking for data to download based on known website structure. In all cases, we configured the agents to download only comma-separated values (CSV) and Excel spreadsheet (XLSX) format files because of their popularity and ease of use. Upon collection, these agents then called a set of functions to analyze the data set to build a data card for that instance of the data set. The top eight categories were used for the combined data.gov and coronavirus data card set, and the top nine categories were used for the Kaggle data card set (Figure 3).

Models

We encoded characteristic metadata fields from the data cards into numerical matrix form for use in developing topic association models. The support vector machine, random forest, and knowledge graph matching on unsupervised word-vector encodings were the highest performing methods, but we also tested hierarchical clustering analysis (HCA), neural network (NN), and K-means clustering models.

We ran a set of experiments on each classifier and data card set. Data card set refers to a collection of data cards such as the combined data.gov and COVID-19 public data sets and the and Kaggle data sets. We predicted the best fit category for each data card in a data card set and computed the multicategory accuracies in each experiment.

Within these experiments, both one-hot encodings and word embeddings were chosen as testable techniques for vector encoding of the data cards (data signature approach). Word embeddings relied on a pretrained language model, GloVe,[19] to transform elements of the data cards into 300-dimensional vectors. The efficacy of these vectors in distinguishing between different topics was then validated using 2-D principal component analysis. We showed that clustering was possible across the entire corpus of data sets we had collected. These vectors were then used by the above classifier techniques to train the random forest (RF), support vector machine (SVM), and HCA topic classification models. Additionally, one-hot encodings were used to produce vectors for training RF, NN, and K-means clustering models for other experiments. Finally, five-fold cross validation was used to produce the model hyperparameters.

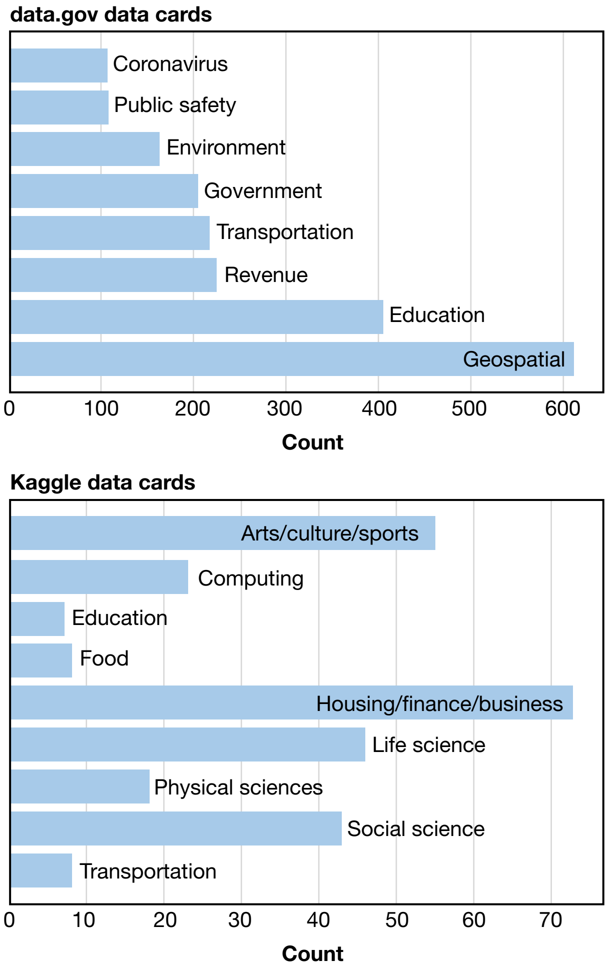


Figure 3. Data.gov and Kaggle data cards topic class distribution. Because the classes are highly imbalanced, the problem at hand is difficult.

Knowledge Graph Matching

A promising experiment relied on matching data card attributes with the Wikidata knowledge graph.[20] This approach is an unsupervised classification technique that uses pretrained language models (such as GloVe) to compute the shortest distances between metadata word-vectors and knowledge graph subtree vectors. To reduce computation time and hardware requirements, the team manually selected subtrees of the knowledge graph relevant to the topics found within the collected data sets. These trees were recursively defined as children of a root node with a subclass or subinstance relation. We found that extracted topics were not comprehensive, so we augmented the topic subtrees with entities based on human intuition and glossaries found on a respective topic.

The next step of the knowledge graph matching method was extracting a matrix representing the knowledge graph subtrees. Word embeddings were extracted for each word in the knowledge graph subtree using the GloVe word-embedding model. The word embeddings were used to form an n by m matrix, where n is the number of words in the knowledge graph topic subtree and m is the embedding dimension, 300.

We then extracted a matrix representing word embeddings of semantic entities found in the data card. Words were parsed from the column names, file name, URL, and online summary information; these were then used as semantic entities. A word embedding was calculated for each semantic entity and was used to form an m by k matrix, where k is the number of extracted semantic entities and m is the word-embedding dimension. Again, we used the GloVe language model to generate word vectors from the data cards.

The matrix multiplication of the two word-embedding matrices obtained was computed, representing each pairwise similarity score of semantic entities in the knowledge graph topic subtree and the semantic entities of the data card. We found the topic score by taking the average of the top 30 elements of the matrix. The topic with the highest score with respect to a data card was deemed the predicted topic class for the respective data card.

RESULTS

Hierarchical clustering and one-hot encoded supervised learning methods scored below 40% on all data and were not included in any of the final evaluations. RF and SVM yielded the highest accuracies for the data.gov data cards, achieving 82.0% and 86.3%, respectively (Figure 4). The knowledge graph matching method yielded 50.2% accuracy for the kaggle.com data cards, followed by 48.8% accuracy for the SVM model (Figure 5).

For the data.gov data cards, adding the knowledge graph subtree topic scores to the word-embedding data resulted a consistent 1% increase in the SVM model performance to 87.3% (best so far), and a consistent 2% increase in the RF model, demonstrating the benefit of hybrid approaches where statistical learning through word embeddings are improved by the rule-based learning from knowledge graphs.

Our experiments were run on two different data card sets or compilation of data sets from a single data repository. The data.gov data sets were more narrowly confined and had topic label associations that could be used for scoring the machine categorizations, and the Kaggle.com data sets were more diverse and representative of a data-in-the-wild ecosystem. Figures 4 and 5 are confusion matrices of our classifiers on the data.gov and kaggle.com data sets, respectively; these offer a more detailed breakdown of classification performance. A perfect confusion matrix would have a completely dark diagonal with completely white squares in other locations. For example, our two best classifiers consistently classified data cards in the coronavirus, revenue, education, and geospatial topic classes for the data.gov data set. Similarly, the knowledge graph matching method consistently classified data cards in the housing/finance/business and education categories for the kaggle.com data set. However, the other topic classes were much lighter on the diagonal and had more misclassifications distributed elsewhere.

The rows represent misclassifications for a particular topic classes. For example, as shown in Figure 4, food data cards were commonly misclassified as housing/finance/business, physical sciences, and social sciences data

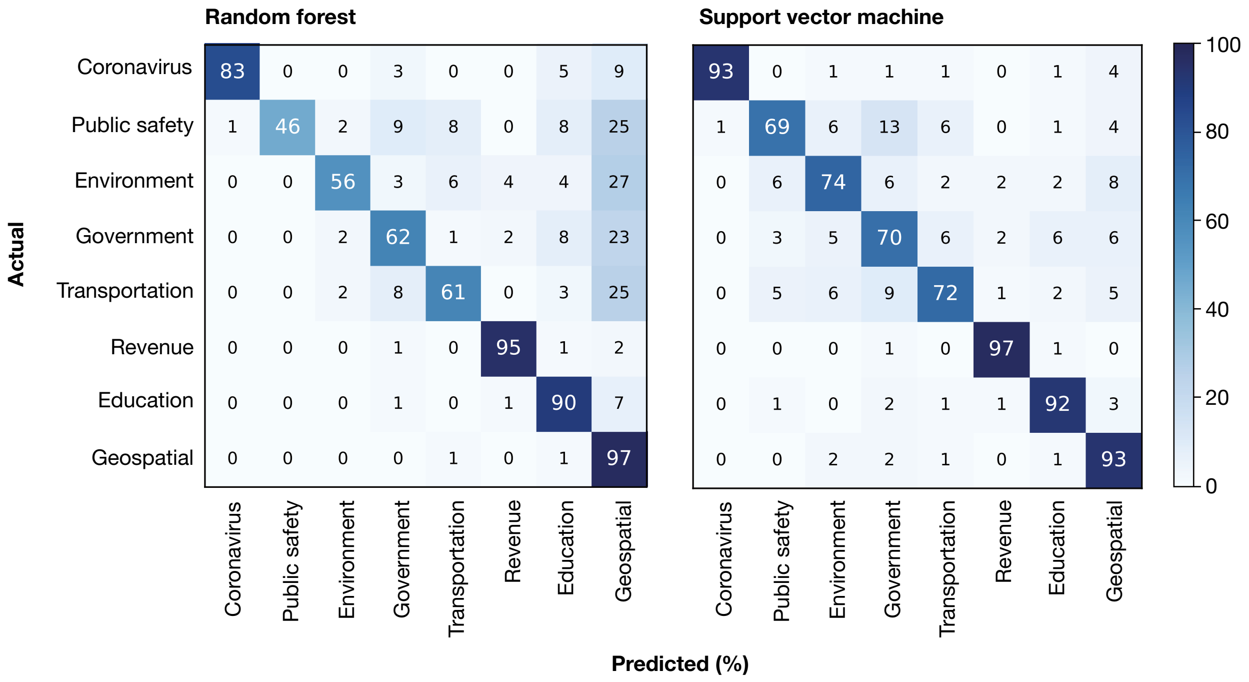


Figure 4. Performance of RF and SVM classifiers using the data signature engineered feature on data.gov data cards. The RF method achieves 82.0% accuracy and the SVM 86.3%. The rows represent misclassifications for a particular category/topic; columns represent actual category/topic predicted.

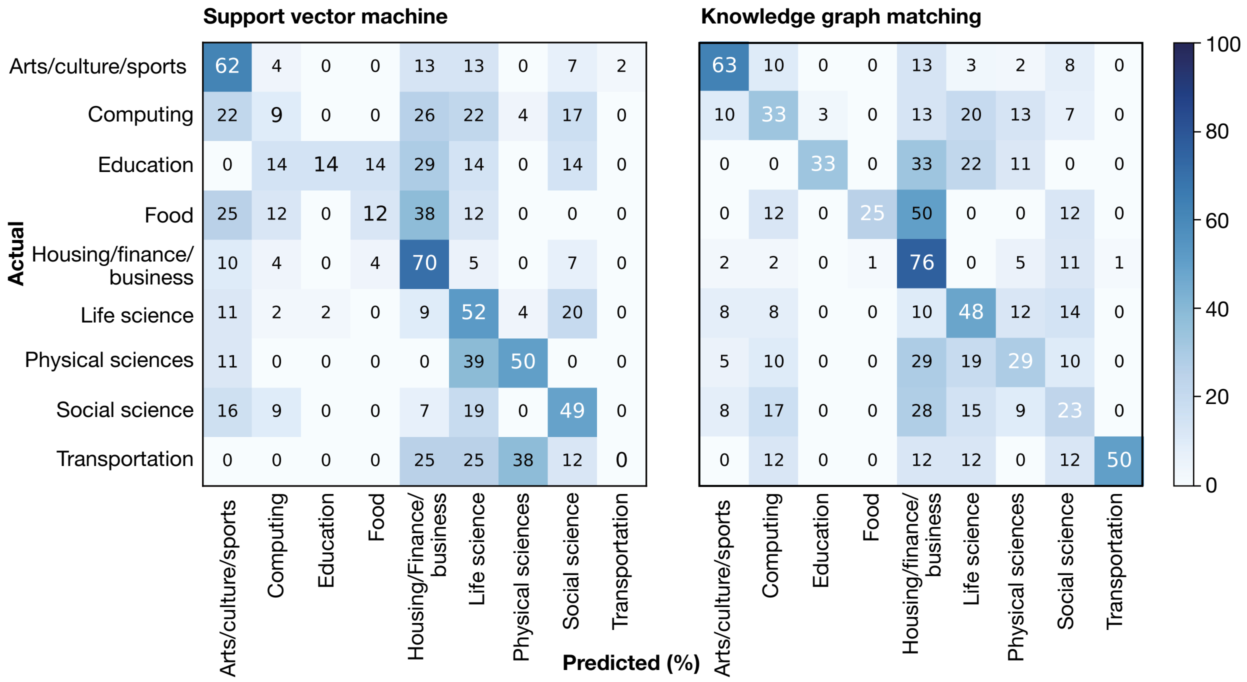


Figure 5. Performance of SVM and knowledge graph matching approaches on kaggle.com data cards. Average accuracies are 48.8% and 50.2%, respectively. The lower accuracy percentages relative to the data.gov data cards are because the kaggle.com data cards spanned a wider range of topics, with broader definitions. The data sets also often belonged to multiple topics, meaning the classification task may have encountered a higher inherent amount of noise.

cards. The columns represent the actual topic classes that we predicted. Although we had a high prediction accuracy for housing/finance/business data cards, the knowledge graph matching classifier misclassified many other data cards as housing/finance/business.

DISCUSSION

We explored a variety of classification techniques to determine the data set topic. Ranging from traditional statistical classifiers to the latest NLP techniques, we assessed basic performance metrics and determined which techniques are most promising. Of the techniques tested, performance was largely based on the method used to translate the data card into the data points used by the classifier algorithm. For example, classifiers that relied on one-hot encoding performed poorly, while those that leveraged many-dimensioned word-embedding vectors performed favorably.

The highest performing classifiers for the data.gov data card data set were the SVM and RF trained on the 300-dimensional GloVe word embeddings. The data.gov metadata set had large groups of data sets that were published by the same government agency for addressing specific issues. This method likely performed well because the learning algorithms were supervised, which enabled them to associate existing tightly grouped clusters of similar data sets to the corresponding topic class without misclassifying too many other data cards. The knowledge graph matching method relied heavily on extracted semantic entities. The data.gov data cards often lacked semantic entities in their file titles, URLs, and online summary text, likely causing poor performance for the knowledge graph matching method.

The best performing classifier for the kaggle.com data set was our knowledge graph matching approach. Kaggle.com data cards spanned a wider range of topics with broader class definitions. Data sets were created by a larger range of users for less specific problems and often belonged to multiple topics, which may suggest a higher inherent noise for the classification task. Kaggle.com data cards often had useful semantic entities in the file titles, URLs, and online data set summaries. As a result, the knowledge graph matching method performed marginally well in terms of average accuracy. Interestingly, the knowledge graph matching method performed significantly better in terms of precision on rare topic class data cards. Additional benefits of the knowledge graph matching method are the high interpretability, unsupervised nature, and potential integration with graph analytics. The method relies on language model embeddings of both the topic classes and data cards, and no labeled training examples were needed for learning. As a result, new candidate topic classes can be discovered through knowledge graph matching. Additionally, the set of candidate topics to classify can vary and only needs to be defined immediately before running the classifier. These benefits favor additional investigation into this method.

Overall, the current techniques rely heavily on general-purpose open-source knowledge graphs and word-embedding models. Open-source knowledge graphs and word-embedding models may not work well when used directly with classified data sets because they were not created with classified information and concepts in mind. However, fine-tuning general-purpose models and knowledge graphs with classified data can replicate or improve on our preliminary results. Conversely, augmenting proprietary and classified data with publicly and commercially available data could also yield performance benefits for classified mission areas.

FUTURE WORK

While the current work focused on validating our hypothesis of using ML techniques to classify the topic of an unknown data set, we also believe that no model will be perfect. An ensemble of models will provide greater overall accuracy, further reducing the burden on a human data curation team. The best performing methodologies, namely those applying NLP techniques to transform the metadata in our data cards into numerical vectors for training their models, are expected to be refined further and integrated into an ensemble. Future work will explore in-depth feature selection and fine-tune the classifiers using lessons learned from the current work.

While existing models are refined, more models will be developed in parallel for the ensemble in future work. Ensemble weighting approaches will begin basic assignments across models, with options to adjust individual weights algorithmically or with expert input after reviewing initial results. The ensemble should improve on the validation metrics of the individual models, allowing individual solutions to be compared directly with the combined solution.

While our current efforts focused on publicly available data sets and associated topics, we hope that future work will focus on mission-specific topics and mission data sets. Work is ongoing to solidify a viable concept of operations document among the different departments and sectors at APL. Using cumulatively trained models to deal with a variety of data across domains, we can use transfer learning in future deployments in operational environments.

In addition to the mission-focused domain in future work, we intend to demonstrate the utility of the Rosetta Data Stone concept by developing an end-to-end system that automates data curation from data set discovery to fully published data sets in a searchable data catalog. This demonstration will integrate other prototyped capabilities from this work that are still in progress, including data crawlers and graph analytics on data cards. The demonstration will also incorporate additional functional components: the fine-tuned topic class prediction models, classifier ensemble, human–machine teaming interface, and live data catalog.

Another capability to be implemented is expanding the data catalog into a complete knowledge base. This expansion would capture metadata about data usage, answering the who, what, when, where, and why about data sets and their use. This capability would provide additional usage statistics, such as which data sets are commonly requested together, to inform future data fusion prediction models.

Finally, these experiments were conducted with no feedback loop from the envisioned human–machine teaming that the final architecture would offer. Ideally, as deployed classification models generate topic predictions on new data sets, the human team would validate the results and make minor corrections, where necessary, before the system updates the catalog with the new data set information. This human feedback is critical to refining the word models, knowledge graphs, and subsequent deployed classification models for improved accuracy from the automated systems.

CONCLUSION

The Rosetta Data Stone effort successfully showed that there are viable ML methods for determining the topic that an unknown data set is describing. We showed that a more in-depth, standardized representation of a data set's metadata, which we call a data card, provided sufficient information about each data set to train classifiers that perform reasonably well. We also showed that our diversity of approaches was critical to addressing diverse data sets. We expanded data set metadata by building on common data set attributes used to catalog data sets for filtering and human retrieval. Attributes such as data set name, location, and classification—all high-level attributes—were augmented with attributes describing the data contained in the file, such as column names and entropy, data type, and column correlation for each column. Our goal was to use this augmented data card to train many classifiers on what data sets describing different topics "looked" like and to predict when new data sets resembled others of the same topic. This performance was achieved after transforming data cards into multidimensional word vectors using the publicly available GloVe word-embeddings model.

The current results are especially valuable in filtering out less viable approaches. However, more work is needed to study its relevance in classified contexts. In addition, accuracy can be improved with more data and by mining more contextual information on each data set. We look to complete our ensemble classifier for our end-to-end pipeline and investigate methods for using humans in the loop.

The vision for our proposal is to build the foundational capabilities needed in a data-driven ecosystem that will allow for variations in software analytic solutions, data storage solutions, computing platforms, and changing environments/requirements. The data-driven ecosystem must be agnostic to the mission, workflow, and technology to enable solutions to meet fluid data and demands. Without tremendous effort, these goals are unobtainable using today's commercial ecosystems, unless data are centrally stored within their environment—but even this leads to siloed and rigid capabilities and vendor lock. Rosetta Data Stone will provide an accessible framework that lowers the cost of entry for knowledge management and a roadmap for future adoption, including analytic research, applications, and refinement. By turning the associated risk profile on its head and breaking down barriers, this system can potentially disrupt the entire knowledge management ecosystem, leading to APL's next defining innovation.

ACKNOWLEDGMENTS: We thank Jim Maguire (APL), Alex Barghi (formerly APL, now Geico), Shelby Wilson (APL), Lai Lao (APL), Wayne Bethea (APL), I. K. Ashok Sivakumar (APL), and Louis Pang (formerly APL, now Enterprise Peak) for their research and development contributions. In addition, we thank APL's Propulsion Grant program for its valuable funding.

REFERENCES

[1] M. Gualtieri, "Hadoop is data's darling for a reason," Forrester.com, https://go.forrester.com/blogs/hadoop-is-datas-darling-for-a-reason/ (accessed Dec. 23, 2021).

[2] S. Lohr, "For Big-data scientists, 'janitor work' is key hurdle to insights," *New York Times*, Aug. 18, 2014, https://www.nytimes.com/2014/08/18/technology/for-big-data-scientists-hurdle-to-insights-is-janitor-work.html.

[3] J. D. Kelleher and T. Brendan, *Data Science*. Cambridge, MA: MIT Press, 2018, p. 79.

[4] H. Devane, "What is a data catalog? 5 features of a modern data catalog." Immuta.com. https://www.immuta.com/articles/what-is-data-catalog/ (accessed Dec. 27, 2021).

[5] M. Goetz, "The Forrester Wave™: Machine learning data catalogs, Q4 2020," Oct.14, 2020, https://www.forrester.com/report/the-forrester-wave-machine-learning-data-catalogs-q4-2020/RES157467?ref_search=0_1646340699750.

[6] A V. Russell and B. J. Quinn III, "Method and system for identifying and discovering relationships between disparate data sets from multiple sources," US Patent 10,997,244, issued May 4, 2021.

[7] B. Altaf, U. Akujuobi, L. Yu, and X. Zhang, "Data set recommendation via variational graph autoencoder," in *Proc. 2019 IEEE Int. Conf. Data Mining (ICDM)*, 2019, https://doi.org/10.1109/icdm.2019.00011.

[8] M. J. Cafarella, A. Y. Halevy, Y. Zhang, D. Zhe Wang, and E. Wu, "Uncovering the relational web," in *Proc. 11th Int. Workshop Web and Databases (WebDB 2008)*, 2008, https://www.cs.columbia.edu/~ewu/files/papers/relweb-webdb08.pdf.

[9] E. Crestan and P. Pantel, "Web-scale table census and classification," in *Proc. 4th ACM Int. Conf. Web Search and Data Mining*, pp. 545–554. 2011, https://doi.org/10.1145/1935826.1935904.

[10] J. Eberius, K. Braunschweig, M. Hentsch, M. Thiele, A. Ahmadov, and W. Lehner, "Building the Dresden Web Table Corpus: A classification approach," in *Proc. 2015 IEEE/ACM 2nd Int. Symp. Big Data Comput. (BDC)*, pp. 41–50, 2015, https://doi.org/10.1109/BDC.2015.30.

[11] L. Zhang, S. Zhang, and K. Balog, "Table2vec," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, pp. 1029–1032, 2019, https://doi.org/10.1145/3331184.3331333.

[12] X. Deng, H. Sun, A. Lees, Y. Wu, and C. Yu, "TURL: Table understanding through representation learning," in *Proc. VLDB Endowment 14*, 3rd ed., pp. 307–319, 2020, http://www.vldb.org/pvldb/vol14/p307-deng.pdf.

[13] H. Iida, D. Thai, V. Manjunatha, and M. Iyyer, "TABBIE: Pretrained representations of tabular data," in *Proc. 2021 Conf. North Amer. Chap. Assoc. for Comput. Linguistics: Human Lang. Tech.*, pp. 3446–3456, 2021, https://doi.org/10.18653/v1/2021.naacl-main.270.

[14] Z. Wang, H. Dong, R. Jia, J. Li, Z. Fu, S. Han, and D. Zhang, "TUTA: Tree-based transformers for generally structured table pre-training," in *Proc. 27th ACM SIGKDD Conf. Knowl. Discovery & Data Mining*, pp. 1780–1790, 2021, https://doi.org/10.1145/3447548.3467434.

[15] Y. Wang and J. Hu, "A machine learning based approach for table detection on the web," in *Proc. 11th Int. Conf. World Wide Web - WWW '02*, pp. 242–250, 2002, https://doi.org/10.1145/511446.511478.

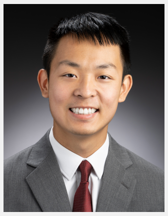
[16] B. Hancock, H. Lee, and C. Yu, "Generating titles for web tables," in *World Wide Web Conf.*, pp. 638–647, 2019, https://doi.org/10.1145/3308558.3313399.

[17] Data.gov. Repository of US government open data managed by the US Government Services Agency. https://www.data.gov/ (accessed Dec. 27, 2021).

[18] Kaggle.com. "Datasets." https://www.kaggle.com/datasets.

[19] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *Proc. 2014 Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, pp. 1532–1543, 2014, https://doi.org/10.3115/v1/d14-1162.

[20] MediaWiki contributors."Wikibase/DataModel." MediaWiki, https://www.mediawiki.org/w/index.php?title=Wikibase/DataModel&oldid=5116182 (accessed Apr. 1, 2022).



Roman Z. Wang, Asymmetric Operations Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Roman Z. Wang is a computer scientist in APL's Asymmetric Operations Sector. He is also a recent graduate of the Discovery Program. He earned his BS in computer science and mathematics from the University of Virginia and is pursuing his MS in computer science at Columbia University. His research interests include artificial intelligence, natural language processing, and target tracking. His email address is roman.wang@jhuapl.edu.

Erhan Guven, Asymmetric Operations Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Erhan Guven is a data scientist in APL's Asymmetric Operations Sector. He holds a BS and MS in electrical and electronics from Middle East Technical University and an MS and PhD in computer science, both from George Washington University. His current research includes GPGPU applications and deep learning and its application to image, speech, text, and disease data. He is also active in cybersecurity research, graph analytics, and generalized clustering techniques such as latent Dirichlet allocation. His email address is erhan.guven@jhuapl.edu.

Joseph L. Duva, Asymmetric Operations Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Joseph L. Duva is a project manager in APL's Asymmetric Operations Sector. He holds a BS in computer science from University of Maryland and an MS in computer science from Johns Hopkins University. He has over 20 years of professional IT experience ranging from data analytics, system administration, programming, management, and the creation of IT innovation and data-driven organizations. He has experience working in both the small business private sector, Department of Defense (DOD) contracting, and as a DOD civilian employee and supervisor. His email address is joseph.duva@jhuapl.edu.

Michael Kramer, Asymmetric Operations Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Dr. Michael Kramer is a project manager in APL's Asymmetric Operations Sector. He earned his BS in physics from Yeshiva College and a PhD in physics from the City University of New York. He has over 30 years of experience in leading software research and development from design and architecture to solutions, providing engineering services in the fields of big data, cloud computing, data science, cognitive computing, computer security, telecommunications, and computer software research and development. He is an inventor on more than 30 patents and a leader in intellectual property development initiatives.