




APPLYING NATURAL LANGUAGE PROCESSING TECHNIQUES TO INTERNATIONAL SAFEGUARDS

PROCEEDINGS OF THE INMM & ESARDA JOINT ANNUAL MEETING
MAY 23–26, 2023

 **Scott L. Stewart**
Oak Ridge National Laboratory*
Oak Ridge, TN, USA
stewartsl@ornl.gov

 **Carlos Soto**
Brookhaven National Laboratory
Upton, NY, USA

 **Alejandro Michel Zuniga**
Pacific Northwest National Laboratory
Richland, WA, USA

 **Nathan Martindale**
Oak Ridge National Laboratory
Oak Ridge, TN, USA

ABSTRACT

The International Atomic Energy Agency (IAEA) faces a significant, and growing, challenge in collecting and analyzing safeguards-relevant data. As the IAEA progressively continues to safeguard more facilities and material in the future, this data challenge will only grow. The IAEA collects and analyzes safeguards-relevant information primarily through data streams from open-source collection (text-based), in-field instrumentation (signals-based), surveillance (image or video based), and satellite imagery (imagery based). This process is currently mostly manual; however, intelligent automation will likely be required to process the growing volumes of safeguards-relevant information in the near future. This special session provided a demonstration of five natural language processing techniques that are relevant to text processing workflows in open-source collection. These techniques are currently in development at Brookhaven National Laboratory, Oak Ridge National Laboratory, and Pacific Northwest National Laboratory and include text classification with NukeLM, the Transformer eXplainability and eXploration tool, author disambiguation with S2AND, machine learning–based table extraction built as part of the Evaluated Nuclear Structure Data File (ENSDF) effort, and the Interactive Corpus Analysis Tool. NukeLM is a BERT-style transformer model that has been pretrained on 1.5 million abstracts from the US Department of Energy’s Office of Science and Technical Information database to provide more relevant document classification for the nuclear domain. The Transformer eXplainability and eXploration tool was created to provide users a tool to better understand the performance of language models being used for sequence classification tasks. The S2AND algorithm was developed by Allen AI and is particularly useful for disambiguating authors in a collection of publications. The ENSDF machine learning–based table extraction approach has been developed to automatically extract information from tables in non-machine-readable documents. Finally, the Interactive Corpus Analysis Tool was developed as a method to allow someone who is not an expert in machine learning to build a text processing workflow based on their subject matter expertise while still leveraging the field of machine learning.

Keywords data analytics · machine learning · data science · natural language processing · international safeguards

*Notice: This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan.

1 Introduction

The International Atomic Energy Agency (IAEA) faces a significant, and growing, challenge in collecting and analyzing safeguards-relevant data. As the IAEA progressively continues to safeguard more facilities and material in the future, this data challenge will only grow. The IAEA collects and analyzes safeguards-relevant information primarily through data streams from open-source collection (text-based), in-field instrumentation (signals-based), surveillance (image or video based), and satellite imagery (imagery based). This process is currently mostly manual; however, intelligent automation will likely be required to process the growing volumes of safeguards-relevant information in the near future.

As part of an ongoing conversation on this topic, the authors of this proceedings article organized a special session at the INMM and ESARDA Joint Annual Meeting in Vienna, Austria, in May 2023. During the special session, the authors demonstrated five different natural language processing technologies that could be applied to international safeguards. These technologies included NukeLM, the Transformer eXplainability and eXploration (TX²) software package, a set of machine learning- and natural language processing-powered models for automatic table extraction from non-machine-readable documents, the S2AND entity disambiguation tool, and the Interactive Corpus Analysis Tool (ICAT). Each of these methods will be discussed in greater detail in the following text.

2 NukeLM: A BERT Style Language Model for Nuclear and Energy Domains

NukeLM is a Pacific Northwest National Laboratory research effort that leverages the performant capabilities of BERT transformer models for domain-specific document classification (1). The language model was pretrained on 1.5 million abstracts from the US Department of Energy Office of Scientific and Technical Information (OSTI) database, and is fine-tuned for both binary classification as well as multiclass classification (2). NukeLM provides users the ability to properly triage manuscripts to better understand citation networks that publish in the nuclear space. Moreover, NukeLM can help uncover new areas of research in the nuclear or nuclear relevant domains.

Natural language processing is one field of machine learning that seeks to make use of unstructured, textual data for analytical purposes. Some typical tasks that a natural language processing algorithm might perform include semantic similarity, text generation, and entity extraction. NukeLM is one method that can be used in document classification, where the goal is to assign each document in a corpus (or set of a documents) a label.

Document classification has numerous applications, ranging from information retrieval and content filtering to sentiment analysis and topic modeling. For international safeguards, document classification could be used to help a subject matter expert quickly label documents that are relevant for further review and consideration.

Many machine learning techniques exist for document classification. Naive Bayes classifier, term frequency-inverse document frequency, and even word embeddings are some of the earlier methods used for document classification. The advancement of deep learning methods has led to the popularity of transformer-based models for textual data. These models have revolutionized document classification by capturing contextual information and achieving state-of-the-art performance on various natural language processing tasks.

Transformer models, such as BERT (Bidirectional Encoder Representations from Transformers) or the currently popular GPT (generative pretrained transformer), have demonstrated remarkable success in natural language processing (3; 4). By leveraging the power of self-attention mechanisms, transformer-based models can effectively capture long-range dependencies and contextual relationships between words in a document (5). They are pretrained on large-scale corpora to learn the intricacies of language and then fine-tuned on task-specific datasets, making them highly adaptable and capable of understanding complex textual patterns.

The features extracted from transformer models come from the contextualized representations of tokens in the document. These models generate high-dimensional embeddings that encode semantic information, enabling them to capture nuanced meanings and improve the accuracy of document classification. The attention mechanisms also allow the models to focus on relevant parts of the document while making predictions, enhancing their ability to identify important features for classification. The combination of transformer-based models with transfer learning has further amplified their effectiveness. Leveraging pretrained models and fine-tuning them on specific document classification tasks has led to excellent performance even with limited labeled data.

NukeLM is a modified version of the RoBERTa model that used both pretraining and fine-tuning to create a nuclear domain aware language model. RoBERTa is a variant of BERT that builds upon its architecture and training methodology, aiming to further improve performance on language understanding tasks (6). NukeLM is specifically fine-tuned for two different classification tasks. These tasks are

1. binary classification of abstracts to determine if they are related to the nuclear fuel cycle or not, and

2. multiclass classification of abstracts based on more than 50 categories that researchers assigned to the abstract when it was uploaded to OSTI.

The binary classification model can be used as a first step to sort abstracts. Although the multiclass classification model could be used to further categorize documents that are already classified as nuclear fuel cycle relevant into subcategories.

3 Transformer eXplainability and eXploration

Oak Ridge National Laboratory has developed Transformer eXplainability and eXploration (TX²), a software package to allow machine learning researchers to better understand the performance of transformer models used for sequence classification (7). The tool can take a trained transformer model and a dataset split into training and testing populations and produce an IPyWidget (8) dashboard with several visualizations to understand model performance with an emphasis on explainability and interpretability. The software has also been released as open source for community use (9).

3.1 Motivation

Complex machine learning models, such as transformer-based large language models, are often described as black boxes. We can characterize the data that a model is trained with and mathematically define how the model works, but characterizing in a meaningful way why a model outputs a particular prediction for a particular input is a challenging problem. This is one of the fundamental motivations behind explainability and interpretability. Especially in high consequence domains with human-in-the-loop systems, a level of founded trust and understanding has to exist between the human analyst and how the model is arriving at its conclusions, rather than blind acceptance. Evaluating the performance of transformer models based solely on quantitative metrics such as accuracy may not be sufficient for establishing this understanding. Tools that can be used for testing and interacting with a trained model and that provide mechanisms for exploring the embedding space might allow catching a model that would perform poorly in practice as well as inform what may need to be fixed in further fine-tuning.

3.2 Features

The TX² package is primarily intended to integrate into a workflow centered around Jupyter Notebooks or Jupyter Lab and currently assumes the use of PyTorch (10). The dashboard includes several visualization and data exploration features, including an interactive Uniform Manifold Approximation and Projection (UMAP) embedding graph to understand classification clusters (11), a word salience map that can be updated as researchers alter textual entries in near real time, a set of tools to understand word frequency and importance based on the clusters in the UMAP embedding graph, and a set of traditional confusion matrix analysis tools.

The salience map, shown in Figures 2a–2b, is computed by comparing the change in output classification probabilities for an instance when each word in the text is individually removed. This acts as a loose proxy for how important each word is to the prediction, allowing the user to see what the model is “focusing” on. A text box next to the salience map reflects the text of the current instance, and any changes the user makes to it both updates the salience and renders an arrow in the UMAP, demonstrating how the change moved the output point in the projected embedding space.

When viewing the 2D projections of the points, specifically analyzing the visual (2D) clusters can be useful. TX² runs a configurable clustering algorithm on these 2D data points and places cluster labels in the UMAP plot. Various information for these clusters is then precomputed for the remaining visualizations, including the frequency of the most common words per cluster and the words with the highest overall aggregated salience impact per cluster. Additionally, random sampling buttons are provided for each cluster.

Finally, views for overall model details such as microprecision/macroprecision/recall/f1 scores as well as a confusion matrix are provided on a separate tab in the dashboard.

TX² seeks to be a tool that can help the researcher explore and ask the following questions about their model:

- What words are the transformer relying on to classify points—are these reasonable signal words to be using or are they spurious or likely irrelevant?
- How well clustered by class are the points? Are there classes that appear to be ambiguous? If so, why?
- Where throughout the graph do specific words we know are important appear?
- When sampling through all of the incorrectly classified points, are there commonalities in the language that may be throwing it off?

- Which words are most likely to influence classification within a particular cluster?

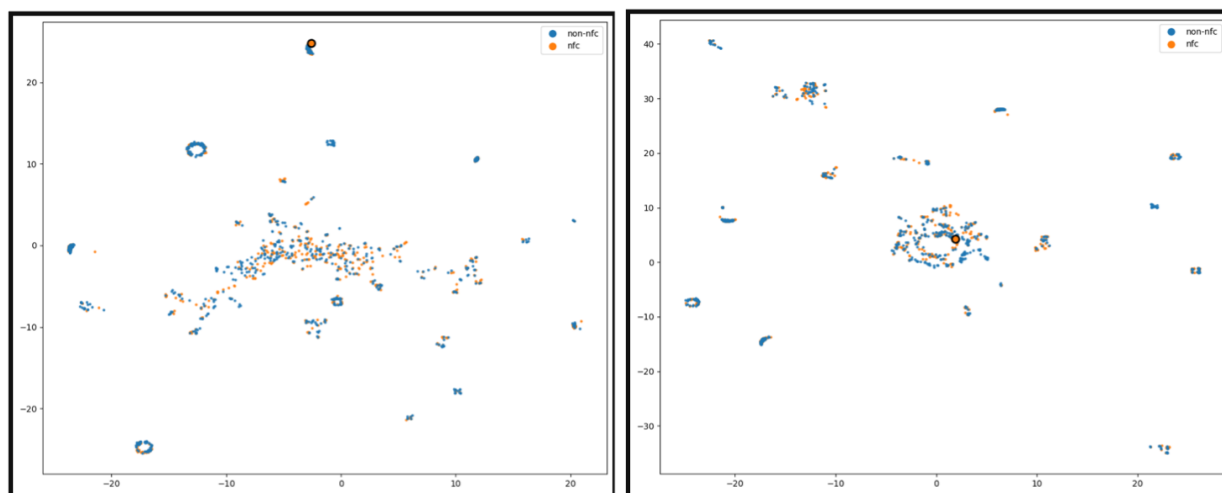
3.3 Example Application

For a demonstration problem, we used TX² to evaluate various transformer models trained on predicting which paper abstracts from OSTI are relevant to the nuclear fuel cycle. We did not have ground truth labels, but we used the first OSTI category as a proxy and considered an abstract to be related to the nuclear fuel cycle if it was in a list of relevant categories.

We trained three different models, all fine-tuned versions of a galactica-125m base model (12):

- The first model was trained purely on the target binary classification task.
- In the second model, we first trained on multiclass classification to try to directly predict the original category, and then fine-tuned on the target binary classification task.
- For the final model, we used 50% of the abstracts to do additional pretraining, and then fine-tuned on the binary classification task.

From the UMAP plots shown in Figures 1a and 1b, we saw that the two classes did not achieve good separation—they were fairly intermingled across all models. The individual clusters did get substantially farther apart from model 1 to model 3. TX² helped us find several data cleaning issues; for example, we discovered one cluster that was made up solely of abstracts with the text “no abstract prepared.”



(a) UMAP plot from model 1, with only binary classification fine-tuning (b) UMAP plot from model 3, additional pretraining on the data before fine-tuning on binary classification

Figure 1: 2D UMAP embedding plots from models displayed in TX².

The intent for TX² is to allow both quantitative and qualitative review. In our example, model 1 had almost 10% better accuracy than 2 and 3, but qualitatively the model seemed to do worse. For example, there are several instances where model 1 got the answer “correct,” but the salience maps were unfocused, where both models 2 and 3 (though incorrect) focused most heavily on words that seemed more relevant. The figures below demonstrate this—Figure 2a shows the salience map of model 1 on an instance, and there is little variance among the words, indicating a lack of signal. Figure 2b shows the salience map of the same instance for model 2, and there are a few heavily highlighted words that are much more relevant to nuclear fuel cycle–adjacent subjects.

This qualitative approach also helped identify a problem with our overall labeling system. There were many instances that models 2 and 3 were incorrect, predicting that a text was not related to nuclear fuel cycle even when our “ground truth” pertained to the nuclear fuel cycle. Looking at the actual texts, there were many that were clearly not actually related to nuclear fuel cycle but were still found in one of our nuclear fuel cycle–indicating categories (e.g., “nuclear physics and radiation physics”). This highlights a flaw in how we are providing labels to the model as it trains. Overall, TX² was useful for finding issues with the models and the underlying data.

The NiCrFe Alloy 600 is the pressurized water reactor (PWR) steam generator tube material used in essentially all US PWRs, in most foreign PWRs, and thus is the tube material with the greatest service experience. Numerous laboratory investigations and service experience have demonstrated that Alloy 600 has good corrosion resistance in the steam generator secondary side environments expected under normal operating conditions. Experience in the laboratory and in the field, however, has demonstrated that even low levels of contaminants may concentrate in tubesheet crevices, in deep tubesheet sludge piles, and in tube support structure crevices to levels that promote several types of corrosion (pitting, intergranular attack, stress corrosion cracking, wastage) of the tube material. Low levels of contaminants, such as chlorides, sulfates, and oxygen, in the secondary system result from condenser cooling water and air in-leakage and from impurity releases from condensate polishers. This discussion reviews the corrosion-related secondary side tube degradation processes that have affected steam generator tubes in recent years and also mechanical tube damage resulting from fatigue and wear of the tube material.

(a) Salience of words with only binary classification

The NiCrFe Alloy 600 is the pressurized water reactor (PWR) steam generator tube material used in essentially all US PWRs, in most foreign PWRs, and thus is the tube material with the greatest service experience. Numerous laboratory investigations and service experience have demonstrated that Alloy 600 has good corrosion resistance in the steam generator secondary side environments expected under normal operating conditions. Experience in the laboratory and in the field, however, has demonstrated that even low levels of contaminants may concentrate in tubesheet crevices, in deep tubesheet sludge piles, and in tube support structure crevices to levels that promote several types of corrosion (pitting, intergranular attack, stress corrosion cracking, wastage) of the tube material. Low levels of contaminants, such as chlorides, sulfates, and oxygen, in the secondary system result from condenser cooling water and air in-leakage and from impurity releases from condensate polishers. This discussion reviews the corrosion-related secondary side tube degradation processes that have affected steam generator tubes in recent years and also mechanical tube damage resulting from fatigue and wear of the tube material.

(b) Salience of words where fine-tuning using category information occurred before binary classification.

Figure 2: Salience Maps from the TX² application.

4 Machine Learning–based Table Extraction for ENSDF

Brookhaven National Laboratory is developing machine learning– and natural language processing–powered models for automatic table extraction from non-machine-readable documents. Tables are a common and high-density information resource, yet their contents often cannot be accessed or processed in an automated manner. This is particularly a problem for PDFs, which not machine-readable (a problem that is apparent to anyone who has attempted to copy and paste contents out of a PDF). Current tools for table extraction from PDFs are rule-based and require manual bounding box alignment, making them unscalable, and they nonetheless produce many extraction artifacts. Our approach is vision- and natural language processing–based and automatically detects tables, their contents, and their structure to enable automated extraction at scale. Brookhaven National Laboratory’s present application is isotope information extraction from the nuclear physics literature to accelerate expansion and evaluation of the XUNDL (Experimental Unevaluated Nuclear Data List) and ENSDF (Evaluated Nuclear Structure Data File) databases.

4.1 Problem and Motivating Application

A significant portion of the information content in documents and records of all types reside in tables. Similarly, many information extraction, curation, and analysis campaigns concerning collections of documents focus on tabular data. However, although these tables may appear an ideal information-dense resource for such tasks, in practice many challenges stand in the way of extracting their contents, particularly at scale. Perhaps the most pressing of these challenges is that many such tables are contained in documents whose format is not machine readable, particularly PDFs. Even when digitally sourced and produced, the contents in PDFs—including tables—cannot be readily accessed, retrieved, or extracted in a consistent and noise-free manner by software. Again, this is readily apparent to anyone who has ever tried to copy and paste a table out of a PDF (e.g., into a Word document or Excel sheet). The result of such an attempt is usually a mix of errors affecting the tables’ formatting, content, and structure. These errors are not overly problematic for manually copying one table or even a few of them. In such cases, the errors can be manually fixed in at most a few minutes. However, when many documents—hundreds to millions—must be processed, these errors present an insurmountable challenge. That is, table extraction from PDFs does not presently scale because it cannot be automated.

In this work, we apply a visual machine learning approach to the problem of table extraction from PDFs. This work is part of an ongoing modernization effort for the ENSDF managed by Brookhaven National Laboratory’s National Nuclear Data Center (13). As part of the nuclear data pipeline, this resource is maintained by expert evaluators who collect, evaluate, and disseminate nuclear physics data sourced from published experimental results. This work includes gathering tabular contents from many published PDFs, and in practice often means spending significant time fixing table extraction errors before being able to move on with their evaluation work. Thus, this machine learning–based work presents a valuable opportunity to accelerate this part of their workflow and may similarly serve to accelerate or even automate many other document processing efforts.

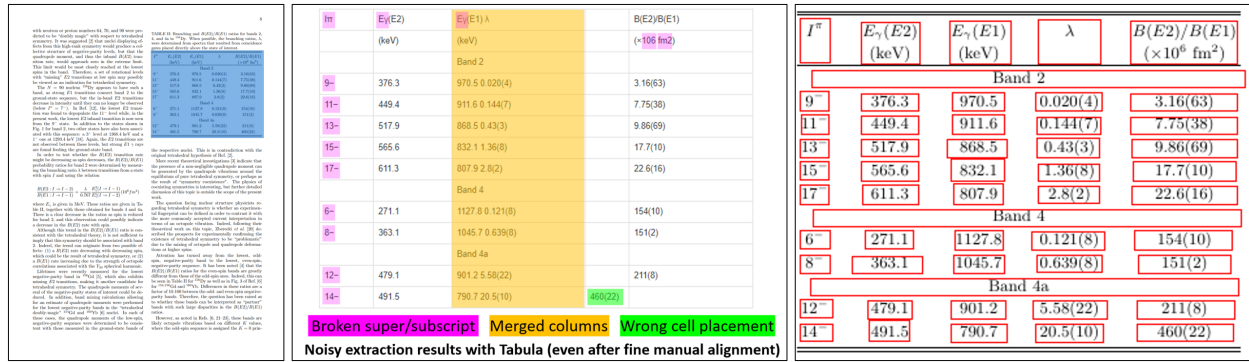


Figure 3: Sample document page containing a table of interest (left), representative extraction errors using current tools (middle), and sample results of our automatic table contents detection machine learning method under development (right).

4.2 Baseline Methods and Problems

Several tools exist for table extraction from PDFs, including open-source software like Xpdf, Poppler, and Tabula, as well as commercial products from Adobe, Abbyy, Microsoft, and others. These tools work by reading a PDF file’s underlying encoding, filtering page elements that are contained within a table’s bounding box, and applying a complex series of heuristic rules to reconstruct the table in an editable form (e.g., CSV, XML) from the inconsistent and disordered document encoding. However, as shown in Figure 3, this approach is still prone to many extraction errors, particularly for tables that are large, have complex formatting, or contain non-ASCII characters (e.g., Greek letters or super-/subscripts). Furthermore, these tools often require that the user first select (i.e., draw a box around) the table they wish to extract, limiting their utility for large-scale or automated document processing.

4.3 Methods and Status

Rather than processing a PDF’s underlying encoding, we treat the table extraction task as a visual problem and apply visual machine learning methods to first detect all tables in a document and then extract and order their contents. We evaluated many machine learning models for these tasks, with particular early emphasis on two models called CascadeTabNet and Local-Global Pyramid Mask Alignment, both of which use convolutional object detection pipelines and were originally trained on relatively small hand-labeled table-extraction datasets (hundreds to thousands of samples) (14; 15). We adapted and tuned these models to tables sampled from *Physical Review C* (annotated with the help of matching LaTeX source). With these models, we achieved strong performance on table detection (0.98 F1) and cell detection (0.96 F1) in our ENSDF-relevant document domain but poor structure recognition performance (73% TEDS, tree edit distance similarity). Accurate structure recognition is required to correctly order and format detected table contents, so this weak link was a critical shortcoming. We attempted to resolve this structure recognition quality issue by training a dedicated seq2seq Long Short-Term Memory (LSTM) model to recover the correct structure from the raw recognition model output. The correction model operated over XML tags and was able to somewhat improve structure performance to 86% TEDS, but this too was not sufficiently accurate for our purposes (we also trained a Transformer seq2seq model with similar results).

We ultimately changed approaches and models entirely and have adopted TATR (table transformer), which is built on the Detection Transformer (DETR) architecture and was pretrained on approximately 1 million machine-annotated tables (16; 17). TATR also allows mapping the structure recognition task as a specialized object detection problem via auxiliary annotations, and thus enables an improvement in structure quality to 95% GriTS (note, the performance metric—grid table similarity—is different because of a change in the problem domain) (18). Finally, we have a selection of optical character recognition models we may apply to read the table cell contents (e.g., Tesseract, EasyOCR, im2markup), all of which perform with 99% accuracy on digital documents.

With these methods in place, we have developed a web interface that ENSDF evaluators can use to quickly upload and view a PDF document, with all detected tables identified and selectable. Upon selection, evaluators receive an easy-to-copy HTML or CSV version of the extracted table contents. The web interface is connected to a back-end machine learning server that processes the PDFs with the trained models and returns extraction results. We are presently optimizing this machine learning server and making various fixes and improvements to the PDF processing pipeline and the user interface. Figure 4 shows the current status of the user interface.

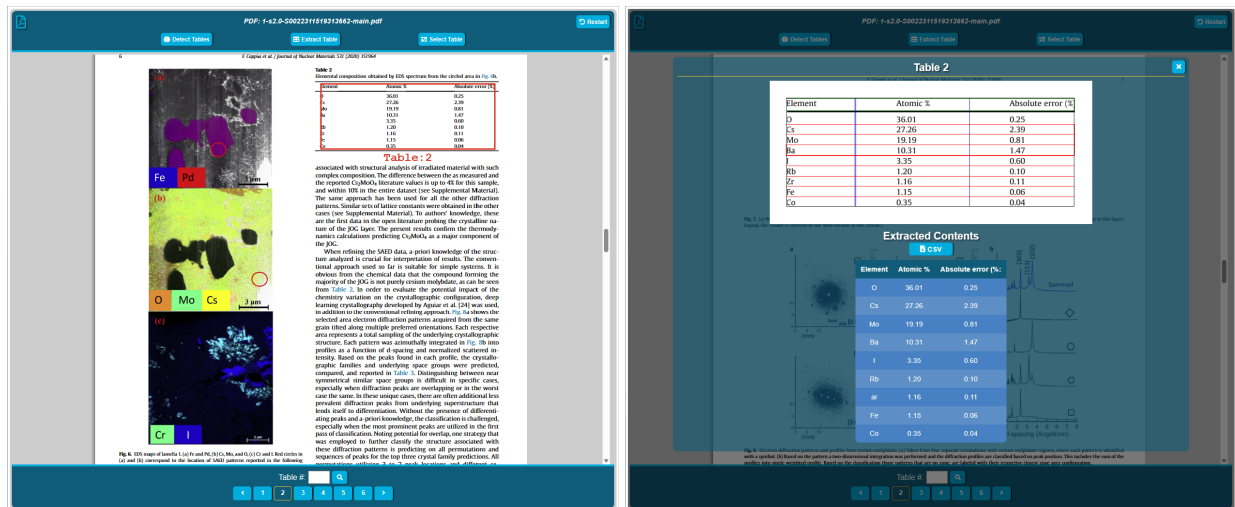


Figure 4: The interface under development automatically identifies all document tables (left) and extracts their content when selected (right).

5 Semantic Scholar’s Author Disambiguation Algorithm and Evaluation Suite (S2AND)

The S2AND algorithm from Allen AI provides users the ability to disambiguate authors from a collection of publications (19). The algorithm works by feeding publication features (e.g., publication venue, publication title/abstract, coauthorships, references) into a gradient boosted tree to run pair-wise classification on potentially coreferent (potentially the same person/author) authors. The algorithm provides unique keys in situations where ORCID’s are not provided, thereby providing easier verification in identifying correct authors (20).

The proliferation of digital content, especially in academic and online publishing platforms, has led to an exponential increase in the number of authors with identical or similar names. As one might imagine, the task of attributing works to the correct authors becomes increasingly intricate, often resulting in confusion and misrepresentation. When it comes to safeguards publications, or any journal relating to nuclear insights, resolving authorship ambiguity becomes essential for producing accurate citations, recommending relevant papers, and retrieving the most up-to-date journals on the latest updates. The presence of ORCID’s or similar unique identifiers aids in author disambiguation, but these types of identifiers are only used by a subset of authors and publications.

A common issue with many preexisting author disambiguation datasets is that they tend to cover idiosyncratic and biased slices of the literature (e.g., AMiner dataset consists of only Chinese names, whereas SCAD-zbMATH contains only mathematics papers). These datasets also may contain unique features that not present in other datasets. Such niche datasets may impair generalization performance as algorithms trained to perform well on one dataset may generalize poorly to others. The S2AND dataset, however, attempts to alleviate this issue by combining resources from a variety of papers, from math and medicine to social sciences and literature. Training on the union of all datasets, rather than any single data source, is an effective way to transfer to out-of-domain data and produces models that are more robust across all the existing datasets.

From a high-level perspective, S2AND works in three steps:

1. Group candidate records into disjoint, potentially coreferent blocks
2. Score the similarity of each pair of records within a block based on available features
3. Cluster the records based on the pairwise scores

The blocking technique is performed heuristically, based on author names. The goal is to group records such that each pair of records within a block potentially refers to the same author (potentially coreferent). The blocks are disjoint, meaning that no two blocks can contain the same record (author-paper combination). Although different datasets use different blocking functions, a typical choice is to put all records with the same last name and first initial into the same block.

Within each block, the similarity of each record pair is estimated using Gradient Boosted Trees (LightGBM) (21). If the score is high enough, we assume that the two records were written by the same author. Note, S2AND uses an ensemble of two classifiers:

1. A classifier trained on the full feature set
2. An identical classifier trained on the “nameless” feature set

This “nameless” feature set is identical in every respect to the full feature set but does not contain any features related to the author names (coauthor names are still included).

After scoring each pair within a block, S2AND implements hierarchical agglomerative clustering (22). Specifically, the classifier from the previous step is used to construct a distance matrix D , where $D_{i,j}$ is the probability that two records i and j are not by the same author. Each block is then partitioned into clusters with hierarchical agglomerative clustering over the matrix D .

The paper mentions how the clustering will depend on a linkage function that estimates the dissimilarity between two clusters (in terms of the pairwise distances between the individual elements of each cluster). Our experiments suggest that a straightforward average of all the pairwise distances performs best.

In total, there are 15 features used to estimate similarity between documents. Of these 15, three have the biggest influence on author disambiguation:

1. SPECTER embeddings
2. Affiliations
3. Name counts (last names)

Although these features appear vital in S2AND, this does not imply these three features alone would lead to adequate model training. We also compared feature variations (i.e., excluding certain features to evaluate performance) and found that most design alternatives hurt performance.

Despite its state-of-the-art performance, note that S2AND features, particularly SPECTER embeddings, are only intended for English-language records. Also, the hierarchical agglomerative clustering pipeline does not allow for the similarity of one pair of records to influence the similarity of another pair of records.

6 Interactive Corpus Analysis Tool

The Interactive Corpus Analysis Tool (ICAT) is an interactive machine learning dashboard for unlabeled text/natural language processing datasets that allows a user to iteratively and visually define features, explore and label instances of their dataset, and simultaneously train a logistic regression model. ICAT was created to allow subject matter experts in a specific domain to directly train their own models for unlabeled datasets visually, without needing to be a machine learning expert or needing to know how to code the models themselves. This approach allows users to directly leverage the power of machine learning, but critically, also involves the user in the development of the machine learning model. This tool has been released to open source and is available for community use (23).

6.1 Motivation

Conventionally, the process for an analyst or subject matter expert to modify a machine learning model is indirect – involving communicating with a machine learning expert any problems they have or cases the model does not appear to handle well. The machine learning expert may then change and re-train the model, and pass the updated version back to the analyst. This indirect feedback loop introduces a great deal of friction and requires the subject matter expert to be able to explicitly communicate the model’s shortcomings. An interactive machine learning tool has the potential to reduce this friction by decreasing the iterative loop, letting them explore the model themselves, and allowing the use of their tacit understanding of the field to improve the model rather than having to explicitly communicate it.

6.2 Features

The basic process of interacting with ICAT is through the AnchorViz (24) visualization. A randomly sampled collection of points is displayed in the ring, which the user can look through, and then create concept magnets or “anchors.” Every time they change the anchors (interactive featurizing) or provide a label for a data point (interactive labeling), the underlying logistic regression algorithm is retrained on all the labels, and the visualization is then updated with the new predictions (indicated by the point color) and their position based on the updated influence of the anchors, if an anchor

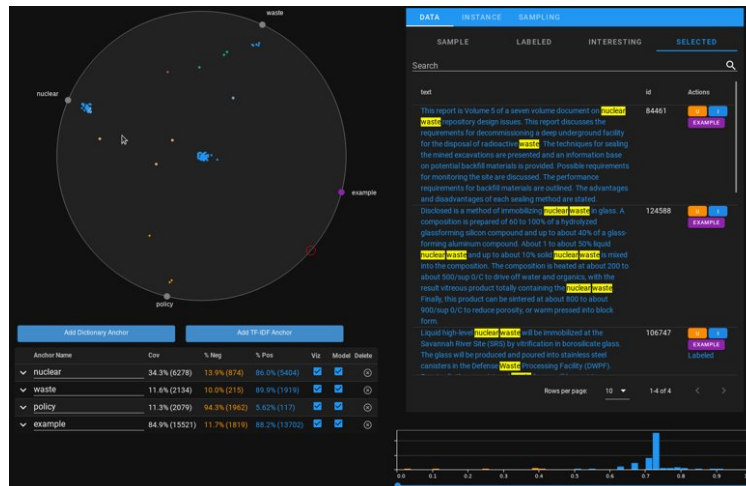


Figure 5: Screenshot of the ICAT user interface.

change was made. These modes of interactivity are what enable the human-in-the-loop approach, which is fundamental to interactive machine learning.

The AnchorViz ring itself as well as the other supporting visualizations on the dashboard allow the user to evaluate how well represented the data is with the current feature set and how well the model is doing.

- Inside of the AnchorViz ring, points that are in the middle and do not move with any of the anchors have no features influencing them, and so are not represented well. This indicates additional anchors need to be made or additional keywords added to existing anchors.
- The features panel shows the overall coverage of each anchor, as well as the breakdown of interesting versus uninteresting predictions, indicating if it is a discriminating feature or not.
- A histogram shows the distribution of output scores from the underlying model, which can indicate if there are many instances close to the decision boundary that should be labeled to clear up uncertainty.

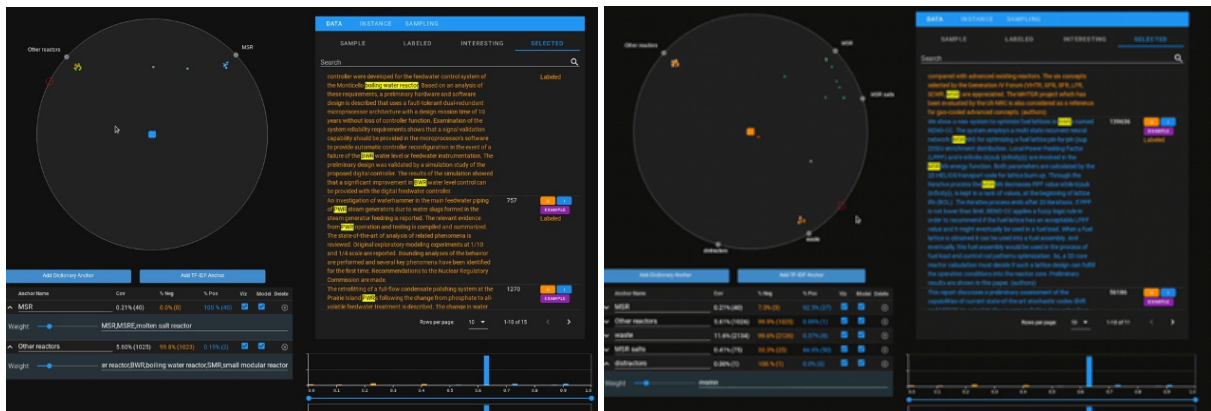
Concept magnets (or “anchors”) in AnchorViz are a function of a single data point that returns some strength in the range, representing how much the magnet influences the given data point. Examples of concept magnet functions could be cosine similarity between Term Frequency-Inverse Document Frequency (TF-IDF) vectors of the given text and some reference text, or a normalized count of keywords (i.e., a bag of words) that show up in the given text. However, they can be any arbitrary function, and as discussed in the original AnchorViz paper, can even include other machine learning models.

6.3 Interactions

As an example use case, ICAT can be used to parse through abstracts from OSTI. A large language model was used to prescreen abstracts for their relevance to the nuclear fuel cycle. From there, relevant abstracts were input into ICAT with the goal of selecting abstracts that were relevant to molten salt reactor research.

Potential strategies to perform this separation include:

- An anchor with MSR terms as well another with non-MSR/other reactor terms split many points in between the two anchors, allowing quick visualization of the crossover as seen Figure 6a.
- Lasso-selecting points is effective for quickly analyzing a small subset of points or a particular cluster. This allows finding keywords within that cluster, providing a few labels, or a combination of both.
- Looking through points unaffected by any anchor in the center can help generate ideas for new keywords for other anchors.
- Using the search functionality can help determine how many points will be affected by a new keyword anchor.
- The weight of an anchor can be adjusted to indicate to the model that that anchor is more important than others. This can change the classification of border points that might be influenced by multiple anchors.
- An anchor created for “distractors,” or words that look similar to something a relevant anchor should care about but explicitly are not, can help pull those away.



(a) Menu for defining anchors is shown on the right.

(b) Anchor coverage is shown here with the menu collapsed.

Figure 6: ICAT user interactions with the anchors.

Using these strategies, ICAT could be used by a subject matter expert to further refine classifications from a different model in a way that might match their normal use-case (SQL-like keyword filter queries). It could also be used by an subject matter expert as a labeling tool to provide additional labels for a full dataset.

7 Acknowledgements

The authors would like to thank the US Department of Energy, National Nuclear Security Administration, Office of International Nuclear Safeguards, Concepts & Approaches Subprogram for funding this combined demonstration. The authors would also like to acknowledge past funding contributions by the US Department of Energy, National Nuclear Security Administration, Office of Defense Nuclear Nonproliferation Research and Development (NA-22) and the US Department of Energy, Office of Science, Nuclear Physics Program, which enabled the creation of these specific technologies.

References

- [1] L. Burke, K. Pazdernik, D. Fortin, B. Wilson, R. Goychayev, and J. Mattingly, “NukeLM: Pre-trained and fine-tuned language models for the nuclear and energy domains,” *arXiv preprint arXiv:2105.12192*, 2021.
- [2] “U.S. Department of Energy Office of Scientific and Technical Information,” <https://www.osti.gov/>, accessed: 2023-07-07.
- [3] I. Tenney, D. Das, and E. Pavlick, “BERT rediscovers the classical NLP pipeline,” *arXiv preprint arXiv:1905.05950*, 2019.
- [4] L. Floridi and M. Chiriatti, “GPT-3: Its nature, scope, limits, and consequences,” *Minds and Machines*, vol. 30, pp. 681–694, 2020.
- [5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in Neural Information Processing Systems*, vol. 30, 2017.
- [6] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “RoBERTa: A robustly optimized BERT pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.
- [7] N. Martindale and S. L. Stewart, “TX²: Transformer eXplainability and eXploration,” *Journal of Open Source Software*, vol. 6, no. 68, p. 3652, 2021. [Online]. Available: <https://doi.org/10.21105/joss.03652>
- [8] “Interactive Widgets for the Jupyter Notebook,” <https://github.com/jupyter-widgets/ipywidgets>, accessed: 2023-07-07.
- [9] “Transformer eXplainability and eXploration Software,” <https://github.com/ORNL/tx2>, accessed: 2023-07-10.
- [10] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, “PyTorch: An imperative style, high-performance deep learning library,” *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [11] L. McInnes, J. Healy, N. Saul, and L. Großberger, “UMAP: Uniform manifold approximation and projection,” *Journal of Open Source Software*, vol. 3, no. 29, p. 861, 2018. [Online]. Available: <https://doi.org/10.21105/joss.00861>

- [12] R. Taylor, M. Kardas, G. Cucurull, T. Scialom, A. Hartshorn, E. Saravia, A. Poulton, V. Kerkez, and R. Stojnic, “GALACTICA: A large language model for science,” 2022.
- [13] A. Hayes, E. McCutchan, S. Yoo, A. Mattera, S. McCorkle, B. Shu, A. Sonzogni, C. Soto, S. Zhu, F. Kondev *et al.*, “Modernization and expansion of the evaluated nuclear structure data file database (ENSDF),” *Bulletin of the American Physical Society*, vol. 66, 2021.
- [14] D. Prasad, A. Gadpal, K. Kapadni, M. Visave, and K. Sultanpure, “CascadeTabNet: An approach for end to end table detection and structure recognition from image-based documents,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 572–573.
- [15] L. Qiao, Z. Li, Z. Cheng, P. Zhang, S. Pu, Y. Niu, W. Ren, W. Tan, and F. Wu, “Lgpma: Complicated table structure recognition with local and global pyramid mask alignment,” in *International Conference on Document Analysis and Recognition*. Springer, 2021, pp. 99–114.
- [16] B. Smock, R. Pesala, and R. Abraham, “PubTables-1M: Towards comprehensive table extraction from unstructured documents,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4634–4642.
- [17] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, “End-to-end object detection with transformers,” in *European Conference on Computer Vision*. Springer, 2020, pp. 213–229.
- [18] B. Smock, R. Pesala, and R. Abraham, “GriTS: Grid table similarity metric for table structure recognition,” *arXiv preprint arXiv:2203.12555*, 2022.
- [19] S. Subramanian, D. King, D. Downey, and S. Feldman, “S2AND: A Benchmark and Evaluation System for Author Name Disambiguation,” in *JCDL ’21: Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2021*, ser. JCDL ’21. New York, NY, USA: Association for Computing Machinery, 2021.
- [20] “Open Researcher and Contributor ID,” <https://orcid.org/>, accessed: 2023-07-10.
- [21] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, “Lightgbm: A highly efficient gradient boosting decision tree,” *Advances in neural information processing systems*, vol. 30, 2017.
- [22] D. Müllner, “Modern hierarchical, agglomerative clustering algorithms,” *arXiv preprint arXiv:1109.2378*, 2011.
- [23] “Interactive Corpus Analysis Tool Software,” <https://github.com/ORNL/icat>, accessed: 2023-07-10.
- [24] J. Suh, S. Ghorashi, G. Ramos, N.-C. Chen, S. Drucker, J. Verwey, and P. Simard, “AnchorViz: Facilitating Semantic Data Exploration and Concept Discovery for Interactive Machine Learning,” vol. 10, no. 1, pp. 7:1–7:38. [Online]. Available: <https://doi.org/10.1145/3241379>