Ontology-Oriented Automatic Tagging of Scientific Articles

Alexander Spangher, Jia Zhang
Carnegie Mellon University
Mountain View, CA 94087
{alexander.spangher;jia.zhang}@sv.cmu.edu

Rahul Ramachandran
NASA/MSFC
Huntsville, AL 35811
rahul.ramachandran@nasa.gov

Derek Koehl, Jeffrey J. Miller, Andrea Word
University of Alabama in Huntsville
Hunstville, AL 35811
{dk0044;jjm0022;worda}@uah.edu

Tsengdar J. Lee
Scientific Mission Directorate, NASA Headquarters
Washington, D.C. 20546
tsengdar.j.lee@nasa.gov

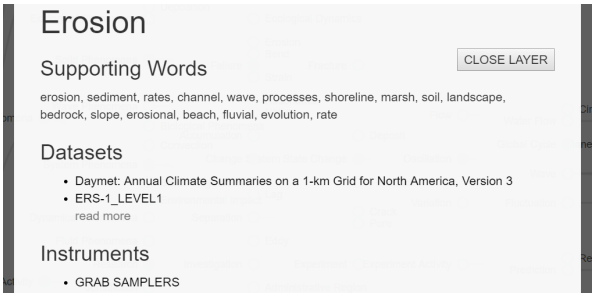
***Abstract*— In modern scientific research, interdisciplinary collaboration has become increasingly promising and on demand. However, it remains challenging for researchers to quickly understand the publications beyond their fields. This work introduces a novel technique to help scientists approach corpora across domains. We train machines to extract core information from papers into an annotated, succinct version that we call *micro-papers*, following how humans approach learning new fields. An ontology-oriented tagging approach is introduced to annotate scientific articles with an initial round of tags, using the concepts encompassed in domain ontology. A neural network-based entity extraction method is exploited to identify domain-specific semantic entities (e.g., datasets and instruments) in articles. External information related to such entities is incorporated into each article, as background knowledge. A Bayesian hierarchical topic model is adopted to refine the tagging phase and to summarize the corpora. Finally, an intuitive format is developed to represent information to help researchers explore corpora more efficiently and remain apprised to state-of-the-art research. The entire process is unsupervised and does not require domain expertise. Our experiments over 8,000 atmospheric research articles demonstrate significant improvements from such an iterative, combined approach.**

I. Introduction

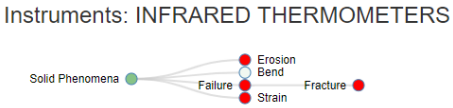
Understanding and comparing large bodies of work is increasingly a necessity for information-heavy disciplines. Journalists performing fact-checks, government officials administering programs and scientists conducting research all rely on widely available corpora of information. In modern scientific research, in particular, researchers are typically tasked with understanding and utilizing insights from large volumes of articles and participating in large-scale interdisciplinary collaborations. They are often expected to understand and interpret the history and cutting edge of several fields simultaneously. Combining these challenges with the rapid pace of scientific publication, even experienced scientists can become quickly overwhelmed.

The National Aeronautics and Space Administration (NASA) aims to tackle this challenge by constructing an enterprise knowledge graph based on published literature. In its early stage, *our primary goal is to make it easy for researchers and students to browse existing literature and easily compare work across domains.* In this paper,

(a) Domain-ontology subtree, rooted at *Solid Phenomena*, colored by the number of scientific articles associated with each node.



(b) "Knowledge Card" generated for each node in the ontology. In this example, *Erosion*.



(c) Red coloring shows fields that have used the instrument "infrared thermometers" in articles.

Fig. 1: **Knowledge Card:** In our interface, researchers are presented a corpus-mapped ontology (a). Exploring nodes, users can see a Knowledge Card (b), showing a summary of the node and entities that have been used to study this field. Users can also see commonalities with other fields (c).

we report how we apply cognitive computing to analyze the semantics of a corpora, and to summarize articles in the context of hierarchically-organized knowledge domains (as illustrated in Fig. 1). *Topic modeling*, an approach of modeling documents based on word-frequency and other observed components, has been used to organize corpora for similar ends. Since Latent Dirichlet Allocation (LDA) [2], researchers have started to apply machine learning to characterize documents by topics and their distributions, thus obtaining a more intuitive overview of a corpus. However,

LDA cannot result in specific topics. On the other hand, labeled topic models rely on massive amount of user tags, which is usually unrealistic to obtain in the scientific field.

Therefore, in this work, we explore how to automatically assign domain-specific ontology concepts, as tags, to Earth science papers. As a proof of concept, we adopt the NASA Semantic Web for Earth and Environmental Terminology (SWEET) [14], which is an ontology suite with about 6,000 concepts covering Earth system science. Our work utilizes concepts from computational linguistics including: *keyword assignment*, *entity extraction* and *topic modeling*. We synergistically combine concepts in these fields to augment each other in an unsurpervised manner and to answer questions such as: "How much prior work has been done in a field?" "What is a field about?" "What tools (e.g., datasets, instruments, methods) does a particular field utilize?"

We first introduce and compare a holistic set of word-matching approaches to assign an initial round of tags to Earth science articles. Meanwhile, domain-specific semantic entities are extracted from articles including the datasets under study, the instruments that gathered the datasets, and the variables of the datasets studied in the articles[1] [3]. External definitions for these semantic entities are appended to body text and combined with concept tags, and then enter into a *labeled hierarchical topic model* [12]. The trained model then refines the initial tagging phase and summarizes the corpus. Our quantitative experiments demonstrate the improvements in tagging accuracy of this iterative, combined approach. Based on the tagging results, we build an intuitive visualization to help researchers explore corpora more efficiently (as shown in Fig. 1(a)(c)). A "knowledge card" is generated for each concept, serving as a succinct micro-article summarizing the state-of-the-art research related to the concept (as shown in Fig. 1(b)).

The major contribution of this paper is that we have developed a framework centered on an unsupervised, iterative process to incrementally tag Earth science articles using domain ontology. Note that our technique is not limited to Earth science and can be applied to across domains.

The remainder of the paper is organized as follows. Section II discusses related work. Section III depicts an overview of our proposed framework, while Sections IV present detailed techniques and methodology. Section V discusses experiments and results. Section VI draws a conclusion.

II. Related Work

Keyword tagging methodologies broadly fall into two categories: *keyword assignment* and *keyword extraction*. Assignment refers to tagging documents from preset lists, while extraction refers to automatically generating lists of keywords from body text. Research utilizing graph-based approaches [1] and natural language processing [7] has shown improvements in accuracy for keyword extraction tasks. Despite the importance of keyword assignment, it is relatively less studied. Advances using rule-based approaches along with pre-specified thesauruses [16] and k-nearest neighbor approaches [17] show promise. In this work, we studied ontology-oriented keyword assignment in an unsupervised setting (i.e., documents are not pre-labeled).

Topic models typically count on a latent structure, or "generative model" to describe observable characteristics of a document, like *keyword tags*. Learning distributions over latent variables in these model can give researchers insight into corpora. Early models produce topic distributions that are difficult to interpret and do not handle correlations between topics [2]. More recent research has introduced labeled models [13] and non-parametric models [5]. Labeled models use tagged documents to restrict the topics in each document, producing directed topics associated with interpretable labels. Non-parametric models learn the number of topics present in a corpus and capture correlations among topics. The Labeled Hierarchical model [12] combines these approaches, which was leveraged in our work to refine concept tagging.

Work in entity extraction has progressed: recent approaches using neural networks [8] and long Markov-chain models [10] have increased the accuracy with which practitioners can identify basic entities like "Person" or "Organization" in a corpora. Some researchers focused on extracting nuanced entities, like "Dataset" or "Instrument" from scientific papers. These approaches use Conditional Random Fields (CRFs) over descriptions of entities and construct supervised learning problems to extract these entities [3]. In our work, we leveraged external information related to extracted semantic entities to enrich the articles under study.

III. Method Overview

In this section, we will outline our approach to summarizing a scientific corpora. As shown in Fig. 2, it is an iterative process containing three phases. In phase one, we use existing ontology concepts to apply initial tags to articles (see Section IV-B and Section V) and semantic entities are extracted from papers (see Section IV-C). We apply additional rules and heuristics to purify these extractions, and incorporate entity definitions into articles. In phase two, tagged articles, together with the ontology, train a hierarchical topic model (see Section IV-D). This model learns topics from documents tagged with labels from an ontology. In phase three, the learned hierarchical topic model will be used to update and refine the paper tagging as well as the ontology, and the iteration will continue and will terminate when no further change is detected. External resources are used to support the model training.

IV. Data and Methodology

In this section, we start by outlining the datasets that we used to form the basis of our research, and then we describe our methodologies.

A. Datasets

The datasets include a corpus of scientific articles published by Wiley, and a human-generated ontology that we use, the NASA SWEET Ontology.

[1]Cognitive science research has revealed that when humans conduct a reading, background knowledge is implicitly integrated with the literal word meanings of the text to construct a coherent model of the whole situation to assist in reading comprehension [6]. This finding implies that background knowledge is necessary for researchers to understand an article.

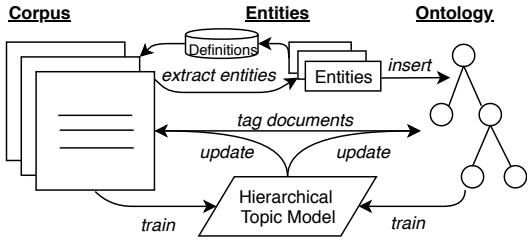


Fig. 2: Diagram capturing how we extract information from our corpus and use the topic model to update our representation. The pipeline is further described in Section IV-E.

1) Atmospheric Science corpus

Given the domain expertise of our research team, we chose to focus on literature published in the Earth Science. We concentrate this research on a corpus of 8,000 articles in the field of atmospheric research, published by the John& Sons, Inc. [2] from time period $11/1/2004 - 9/16/2018$. The articles are each labeled with a set of keywords, but such keywords are hand-assigned by authors and do not follow a schema or ontology. Additionally, the articles consist of textual data divided into sections.

2) SWEET Ontology

We use an ontology developed by NASA's Earth Sciences divisions. The NASA Semantic Web for Earth and Environmental Terminology (SWEET)[3] is an ontology generated by domain experts in 2006 and updated since. The ontology consists simply of word-phrases. The phrases are not defined and there is no additional information given within the SWEET dataset. The ontology is organized around eight principle nodes: $\{Substance, Realm, LandRegion, Process, Phenomena, HumanActivity, Representation, Property\}$.

3) External Information

We incorporate external knowledge from a number of sources: descriptions of datasets extracted from papers[4], a list of Greek and Latin morphemes scraped from Wikipedia[5], American Meteorological Society's Glossary[6], Oxford English Dictionary[7], and the Wiktionary dictionary[8]. Such dictionaries provide further information to support the modeling of the ontology and the content of the articles.

B. Automatic Concepts Assigning

Our principal aim is to accurately assign papers in our corpus to nodes (i.e., concepts) in the SWEET ontology. Two obstacles exist. First, some SWEET ontology concepts do not have definitions with them. Second, SWEET concepts are often complex, phrasal scientific words, for example: "orbital configuration" or "evaporative available potential energy." The former concern can be addressed by leveraging external dictionaries to provide descriptions to the SWEET concepts. To address the latter concern, our strategy is to match based on string-level similarities between the vocabulary in our corpus and SWEET concepts. First, we associate words in each article with nodes in the SWEET ontology. Then, we choose repetition thresholds above which we consider a SWEET concept as part of the article.

The first step, the word-association step, aims to build a similarity function between words and SWEET concepts. A *word* is considered matching the *concept* if this similarity function scores above a chosen threshold. In our study, we start from a baseline solution and then explore various ways to enhance performance. They are summarized in Table I:

1) Baseline

We consider a naive approach based on character counts. We calculate the Jaro similarity[9] between each word in our vocabulary and each SWEET concept.

$$sim_1(word, concept) = Jaro(word, concept) \tag{2}$$

Jaro similarity is considered as it normalizes for differing string length. We consider all matches with $sim_1(word, concept) > .9$ to be a candidate[10], and in the cases where multiple SWEET concepts map to a single word, we match to the one with the highest similarity.

Once we have defined similarity functions between words and concepts, we search for instances of these words in a document's pretagged keywords and body text. If the document contains one pretagged keyword that matches a SWEET concept, or seven or more body words that match a SWEET concept, we assign that SWEET concept to the document. We pick these thresholds to maximize scores of heldout articles (as described in Section V).

2) Word structure

A more nuanced approach is to consider the structure and etymology of the words. We use a predefined list of morphemesand perform exact substring matches searching for specified Greek and Latin morphemes. Within lists of matched words, we then calculate similarities. Our goal is to look at character similarities of the words as in **Baseline**, but to allow different prefixes and suffixes to supersede character-level similarities:

$$sim_2(word, concept) = \begin{cases} Jaro(word, concept), & \text{if } pref(word) = pref(concept) \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

We perform a many-to-many mapping from SWEET concepts to vocabulary words using sets of Greek and Latin morphemes, as shown in Table II. We then perform character-level similarity matches amongst these sublists to match each vocabulary word with it's closest SWEET concept, using $sim_2(word, concept) > .85$ as a cutoff. If a document contains five or more of this vocabulary word or one keyword,

[2]https://onlinelibrary.wiley.com/

[3]https://sweet.jpl.nasa.gov

[4]https://earthdata.nasa.gov/about/gcmd

[5]https://en.wikipedia.org/wiki/List˙of˙Greek˙and˙Latin˙roots˙in˙English

[6]http://glossary.ametsoc.org/wiki/

[7]https://developer.oxforddictionaries.com

[8]https://en.wiktionary.org/wiki/

[9]Jaro similarity is defined as:

$$Jaro(word, concept) = \begin{cases} 0, & \text{if } m = 0 \\ \frac{1}{3}\left(\frac{m}{|word|} + \frac{m}{|concept|} + \frac{m-t}{m}\right), & \text{otherwise} \end{cases} \tag{1}$$

Where m is the number of character matches and t is half the number of character transpositions. Characters are only considered matching if they are not farther than $\lfloor \frac{\max(|word|, |concept|)}{2} \rfloor - 1$.

[10]We set this threshold experimentally via cross-validation to maximize the output of our experiment, described in Section V.

1. Baseline (Base)	**2. Word Structure** (WS)	**3. Concept Identification** (CI)	**4. Dictionary Mapping** (DM)
1. Compute Jaro distance between text and SWEET concepts.	1. Match text and SWEET concepts by prefix. 2. Compute Jaro distance within matched groups.	1. Extract conceptual noun-phrases from body text. 2. Compute Jaro distance between concepts and SWEET concepts	1. Extract definitions for SWEET concepts. 2. Compute cosine distance between body-text and definition TF-IDF vectors.

TABLE I: Summary of Keyword assignment methodologies.

we label it with the corresponding SWEET concept. This approach to tagging is appropriate for scientific writing, where words often have Latin or Greek etymologies.

3) Concept identification

Another approach aims at identifying scientific phrases in our body text, as shown in Table IV. To do this, we process the abstracts of each paper using Stanford CoreNLP[11] to parse dependencies in each sentence and extract part-of-speech trees [9]. In our preliminary study, we apply a custom ruleset built that identify basic scientific relationships, focusing on subject→verb→object relationship, similar to [4]. More comprehensive relationship identification will be investigated in future research.

This approach has two main components. First, we identify entities in the text of a science article using the parts of speech output and universal dependencies output of Stanford's CoreNLP tool and a handcrafted ruleset. This heuristic parses scientific text to identify phrases that are directly relevant to the factual relationships expressed in the scientific article. Second, we perform simple matching on SWEET concepts using Jaro Distance.

4) Dictionary-based matching

We design a fourth approach from a different lens. We utilize three different online glossary sources to pull definitions of SWEET concepts. We start by scraping the American Meteorological Society's glossary, where we retrieved $1,558$ of $3,360$ SWEET concepts. Afterwards, we use the Oxford Dictionary API to retrieve 813 of the remaining $1,802$ labels. Finally, we scrape Wiktionary to retrieve 165 of the remaining 989 SWEET concepts. Overall, we retrieved definitions for $2,536$ concepts, or 75.4% of the terms.

With the definitions we have retrieved, we perform document matching between SWEET definitions and scientific articles using $TF-IDF$ character n-grams.We match SWEET definitions and documents with cosine similarity $\geq .2$.

These methods give us an approach for matching SWEET concepts with documents. We will explain later how we iterate and improve this assignment.

C. Concept Augmentation via Extracted Entities

We further augment the body text of an article by identifying semantic entities within the paper and incorporating external text describing these entities. For instance, a paper might mention a well-known dataset but not describe it fully. Experienced readers, though, would be able to recall related information about the dataset and integrate such background knowledge [6] for the comprehension of the paper.

[11]http://nlp.stanford.edu:8080/parser/.

To identify semantic entities, we extend previous work [3], with steps taken to improve generalization. We identified a collection of semantic entity types that researchers are especially interested in: *dataset*, *instrument*, *variables*, and *organization*. Our task is performed using a two-step process. Entities are divided into sets of explicit entities and implicit entities. Explicit entities are entities uniquely identified in a paper, like *instruments* and *variables*. Implicit entities are entities that are often named by shorthand but discernible by context, like *dataset*. In the first step, we match explicit entities using long-form names and known abbreviations. In the second step, we use these explicit entities to search for implicit entities by training a logistic classification model similar to Word2Vec's skipgram model [11]. We craft rules to filter out specific entities containing keywords: "award," "grant," and "sponsor."

We then gather additional text information provided by the Earth science domain describing each entity, which exist for a subset of entities. There are 647 papers in our corpus from which we extract such entities. For each of these documents, we append these entity descriptions and utilize the tagging approaches described in Section IV-B to extract additional SWEET concepts from this enriched text. We maintain the same thresholds as before, except only require a concept to appear three times in entity text.

D. Concept Refinement via Labeled Hierarchical Topic Model

As discussed earlier, scientists studying a phenomenon often need a more general summary of the ways the topic has been studied. Thus, we also sought to summarize fields by topic.

A *topic* is a distribution over the vocabulary of a corpus. Documents in the corpus are represented as mixtures of topics, and documents containing a topic will be more likely to use some words than others. For example, a document about the **Environment**, for instance, would be more likely to use words like "tree" or "cloud" than a document about **Finance** [15].

We establish two criteria for selecting an appropriate topic model. First, the topic model has to be informed by "labeled" documents, or documents tagged with concepts. Second, the labeled model has to learn topics hierarchically. In other words, topics have to also incorporate signals from the children and parents of that tag node, each of which are tagged to their own set of documents.

We chose the model *Label2Hierarchy* described in [12] as it captures both of these points well and, in addition, allows for errors in labeling. The generative model for *Label2Hierarchy* is given in Table IV. We utilize a sampling

Morpheme	SWEET Label Matches	Body Text Matches
arch-, arche-, archi-	[archipelago, archean, archetype, archives, archive, archived...	[archive, archiving]
aster-, astr-	[asterisk, asteroids, asterisks, asteroid, asteroidal, aster]	[asteroid, asteroseismology]
asthen-	[asthenosphere, asthenospheric]	[asthenosphere]
carbon-	[carbons, carbonate, carbonyl, carbonates, carbonaceous...	[carbon footprint, carbonate alkalinity, carbonate compensati...
cardin-	[cardinal]	[cardinal scale]

TABLE II: Morpheme Matches: SWEET Labels and vocabulary words that share Latin and Greek morphemes.

Body Text	SWEET Concept
geomagnetic activity index	geomagnetic index
radiative transfer	radiative transfer
tree ring	tree ring
sea breezes	sea breeze
bed load	bed load
spectral band	spectral band

TABLE III: Concept Matches: Examples of word-phrases matched to SWEET concepts.

Label2Hierarchy:

1) Sample an ontology, O, from a set of concept-tags, C.
2) For each concept-tag $c \in [1, C]$ in O:
   a) If c is the root, draw background topic $\theta_c \sim Dir(\beta u)$
   b) Otherwise, draw topic $\theta_c \sim Dir(\beta \theta_{\phi(c)})$ where $\phi(c)$ is node c's parent.
3) For each document $d \in [1, D]$ labeled with concept-tags c_d
   a) Define concept-tag sets C_d^0 and C_d^1 using T and c_d
   b) Draw $\theta_d^0 \sim Dir(C_d^0 \alpha)$ and $\theta_d^1 \sim Dir(C_d^1 \alpha)$
   c) Draw a stochastic switching variable $\pi_d \sim Beta(\gamma_0, \gamma_1)$
   d) For each word index in the document $i \in [1, W_d]$
      i) Draw set indicator $x_{d,n} \sim Bern(\pi_d)$
      ii) Draw concept-tag indicator $z_{d,i} \sim Mult(\theta_d^{x_{d,i}})$
      iii) Draw word $w_{d,i} \sim Mult(\theta_{z_{d,i}})$

TABLE IV: Generative model for *Labelshierarchy*.

approach to learn parameters in this generative model.

Step 1 draws an ontology, O (a spanning tree) from all concept tags in the corpus, C. This can be preset, as in our case. Ontology O can be updated at successive iterations. For instance, if **Mixture** and **Dust** tags are each associated with documents that use many of the same words, yet are in different lineages in the given ontology, this model can reorder the connections. Currently we do not use this updating feature, this is an active direction of future inquiry.

Additionally, steps 3a, 3c and 3d,i define C_d^1, C_d^0, π_d and $x_{d,i}$. This collection of variables allows the model to correct keywords. C_d^1 is the set of concepts pretagged to a document (and their parents), while C_d^0 is the complement. π_d is the probability that a document's true tags are contained in C_d^1. $x_{d,i}$ is a binary variable indicating, for each word in a document, whether it belongs to C_d^1 or C_d^0. This gives flexibility to the preassigned labels and allows the model to suggest corrections. For example, if a document is *not* tagged with **Mixture**, but uses many of the same words as a document that *is*, it's likely that that document *should be*. This is mediated by $z_{d,i}$.

To incorporate these corrections, we run the model and count $x_{d,i}$ and $z_{d,i}$ assigned to each document. We include tags that *should have* been assigned to a document and exclude tags that *were* assigned by computing $p(z_{d,.}) > .01$ as a cutoff. This allows us to correct errors to our tagging process and improves performance (Section V). This can be done iteratively: once corrections are made, we can relabel the documents and rerun the topic model. This will change the C_d^1 and C_d^0 for each document, yielding further corrections. For the present work we perform one correction.

E. Methodology Steps

Our processing pipeline occurs in three steps, broadly mirroring the methodological steps outlined in Fig. 2:

Step 1: Document-Level Information Extraction.

1) **Step 1a: Apply SWEET Concepts to Documents**. We first run the word-matching described in Section IV-B to tag documents to each concept in the SWEET ontology.
2) **Step 1b: Entity Extraction.** In parallel, we extract entities from each document. We use the methodology described in Section IV-C. These extracted entities are not updated after this step.

Step 2: Topic Modeling. Using the output from **Step 1a**, we run the hierarchical labeled topic model described in Section IV-D. The output for this step is a topic for each SWEET concept, and a probability distribution over SWEET concepts for each document.

Step 3: Update Concepts. As described in Section IV-D, we use the output of the topic model to update the concept labeling. We take the probability distribution over SWEET concepts per document as outputted by **Step 2** and set a threshold to correct concepts from **Step 1a**. We only run this update once. This can iterative, whereby upon completing **Step 3**, we return to **Step 2** and retrain the topic model.

Step 4: Join Data and Render. Following successful completion of the previous steps we join this information together to output an ontology such that each node in the ontology (i.e. SWEET concept) is mapped to a set of documents, entities and vocabulary words. We render this visually as shown in Section I.

F. Knowledge Cards

We build a knowledge card for each concept in our corpus, which maps scientific papers from a broad corpora into specific domains outlined in an existing ontology. Our platform, built using D3.js, facilitates the following interactions:

1) **On main page** (Figure 1a), users see the entire ontology, color-coded by degree of study. Exploring node pulls up a "Knowledge Card".
2) **On Knowledge Card** (Figure 1b), users see topics ("Supporting Word") and extracted entities ("Datasets", "Instruments", etc.), giving a brief primer on work done in the domain.
3) Clicking on each entity takes users back to the ontology, where they observe different knowledge domains that

have made use of a variable (Figure 1c).

Taken together, this visualization helps users explore the nodes in the ontology with facility. For each phenomenon, users can quickly see the topical context of that node as well as datasets, instruments, and variables that have been used to study that phenomenon in the past. Then, they can see other phenomena that utilize the same variables. In future work, we hope to give results demonstrating the usefulness of this assistant for scientists and engineers.

V. Experiments and Discussions

A. Experiments

Recall that a primary research question in our work is how to correctly tag scientific articles with concepts in an ontology (SWEET phenomena), given limited information for each concept. In Section IV-B we outlined four different methods for performing this tagging, and in Section IV-D we presented an approach for updating these tags. Here we experimentally validate which one of these approaches helps us tag documents optimally.

We set up a simple quantitative evaluation technique to compare these tag-sets with other observable information. Our evaluation is premised on the hypothesis that: *document pairs that study similar phenomena will be more likely to (a) have the same SWEET tags and (b) cite each other, than document pairs studying different phenomena*. We built a symmetrical citation graph:

$$C_{i,j} = \begin{cases} 1, & \text{if document } i \text{ cites document } j \text{ or v.v.} \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

In our corpus, there are $N = 8,600$ document pairs where $C_{i,j} = 1$. We sample an equivalent number of negative samples (i.e. where $C_{i,j} = 0$). For each of our tagging approaches, we calculated cosine similarity[12] between each document's tag vector:

$$similarity(i,j) = cosine(V_{i,t}, V_{j,t})$$

where:

$$V_{i,t} = \begin{cases} k_t, & \text{if documents } i \text{ is given SWEET label } t \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

Input k is calculated to give rarer tags a higher weight:

$$k_t = \frac{1}{\sum_{d \in corpus} d \text{ tagged with } t} \quad (6)$$

We note that k_t is potentially different for each tagging method, as each method uses a different set of tags for each article. We considered the following 12 tagging methods shown in Table V, plus a random label initialization to isolate the effects of the topic model.

In Fig. 3, we show the quantitative results of our experiment. Across methodologies, we see a significant difference between tag-similarities on citation-linked papers. Furthermore, we observe a significant increase in performance from topic-model corrected tags above initialized tags (3a).

[12]Cosine similarity is defined as:

$$cos(A,B) = \frac{A\dot{B}}{||A||||B||}$$

and is chosen because it normalizes for the frequency of tags. This is important as each tagging system tags with a different propensity.

Method	Topic-Model Correction	Entity Augmentation
(**1**) Baseline	(**2**) Base+TM	(**3**) Base w. Ent.
(**4**) Word Struct.	(**5**) WS+TM	(**6**) WS w. Ent.
(**7**) Concept Id.	(**8**) CI+TM	(**9**) CI w. Ent.
(**10**) Dict. Match	(**11**) DM+TM	(**12**) DM w. Ent.
(**13**) Random	(**14**) Rand.+TM	

TABLE V: List of Experimental Comparisons

Entity Augmentation also resulted in significant shifts (Fig. 3b). The number of article-pairs in our dataset that both cited each other and contained entities with associated text was small ($N = 64$). However, we note a statistically significant shift in performance for all methods except **Dictionary Matching** (Fisher-Exact $p < .01$).

Median lift[13] for all changes made is also measured. We see significant median lifts, in Table 3c for **Word Structure** and **Dictionary Matching** on a 10,000 bootstrapped sample. We also see significant median lifts for topic-model corrections (Table 3d).

Additionally, we run a topic-model correction trial with Random tags sampled over the observed label-count densities, shown in Fig. 3a. The median score after correction for this group is 0, indicating the importance of meaningful initial-assignments.

The first-pass assignment techniques assign between 6-17 tags to most articles[14], as shown in Fig. 5. More tags do not translate into higher performance: as shown, the CI method applies more tags than Baseline, but under-performs. The entity augmentation techniques assign 10-25 tags to most articles and topic model correction techniques assign 13-23 tags to most articles[15]. As shown in Fig. 4a, entity-augmentation adds 2-12 tags to most articles. Only Entity Augmentation with **Dictionary-based Matching** removes tags originally assigned, as it changes the TF-IDF vector used to match tags with articles. The topic model correction, shown in Fig. 4b adds 11-22 tags and removes 4-15 tags.

B. Discussions

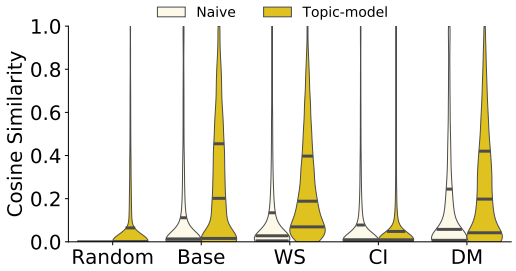
Effectively locating articles within pre-existing ontologies remains an open problem. We tested basic string-matching methodologies and compared with an augmentation scheme incorporating text from extracted entities. We used a hierarchical topic-model to "correct" initial tags.

We noted that the **Baseline** approach, $sim_1(word, node)$, fell short in cases where words were spelled similarly but had different meanings. For example, *circular* and *acicular* appeared similar when considering character counts, but the prefix *a* changed the meaning of the word. The **Word structure** approach, $sim_2(root, node)$ helps us correct for these differences. Both of these approaches, though, fail to characterize scientific phrases. The **Concept identification** goes a step beyond in this regard, although is an early application of ongoing work. We see promise in further development but note that the current methodology underperformed.

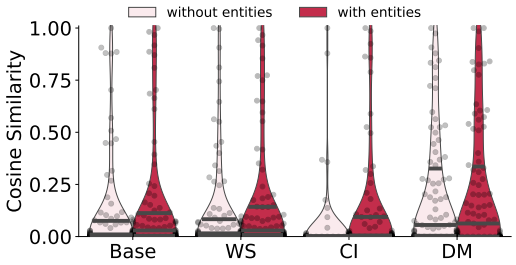
[13]Lift of variables a over b is defined here as $Lift(a,b) = a/b$.

[14]As calculated by the $25th$ and $75th$ percentiles of this group.

[15]Stakeholders can request a lower or higher count-density. Our output can be changed by adjusting thresholds, like the number of times matches appear in documents, or probabilities assigned by the topic model to tags.



(a) Scores across article-pairs, naive compared with topic-model corrections.



(b) Scores across article-pairs, naive compared with entity augmentation.

Base	WS	CI	DM
1.00	**2.09**	**0.72**	**4.31**

(c) Median lifts of methods in IV-B over Baseline from 10,000 bootstrapped sample. Significant results (at $> 2\sigma$) shown in **bold.**

	Base	WS	CI	DM
TM	**15.06**	**6.71**	**0.97**	**3.42**
Ent.	3.63	1.48	0.83	1.22

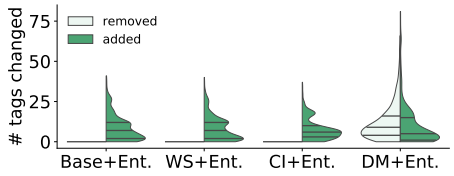
(d) Median Lift of augmentation methods over naive methods across 10,000 bootstrapped samples. Significant results ($> 2\sigma$) shown in **bold.**

Fig. 3: Tags applied before topic-model corrections (a), show Dictionary Matching (DM) outperforming, but topic-model corrections greatly increases scores across methods. Additional entity information (b) also increases scores (significant at $FisherExact < .01$). Black dots are single article-pairs. Lift is shown in (c) and (d).

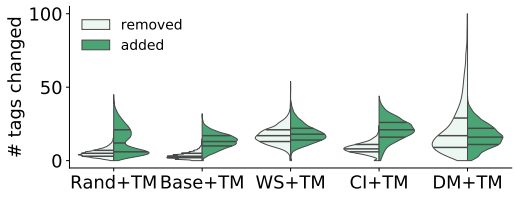
Finally, **Dictionary-Based** matching helps us avoid the more constrained word-based matches and incorporate external information. The Dictionary-Based matching performed the best, but the Word-Structure method is the most suitable when authoritative definitions are not accessible.

The performance increases yielded by applying the topic model correction were substantial, and seemed to have the additive effects: methods that already performed well, like **Dictionary-Based** matching, performed even better post-correction than methods that performed worse, like **Concept-Identification**. Taken together, these results suggests a powerful role for topic-modeling to add performance across tasks. Furthermore, this is the first known application of *Labels2Hierarchy*, which is a powerful model and well-suited to problems such as the one we addressed.

We note that we are measuring tagging *consistency*, not *correctness*. For example, a method that tagged **Forestry**



(a) Tag changes after entity augmentation, additions (white) and removals (green).



(b) Tag changes after running topic model, additions (white) and removals (green).

Fig. 4: Tag changes after applying topic-model correction (a) show both additions and removals of initialized tags. For Base, WS and CI, tags are only added after entity augmentation (b).

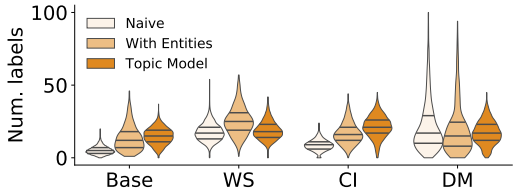


Fig. 5: The counts of tags applied to each paper varies more widely before topic-model corrections (white). After corrections(b), a stricter distribution of tags is enforced by the topic model (red).

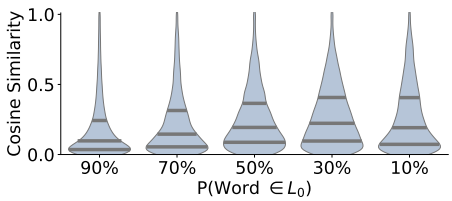
consistently to citation-linked space articles would score highly. Human evaluation could further explore this, and we hope to follow up with these trials in subsequent work.

C. Further Parameter Tuning

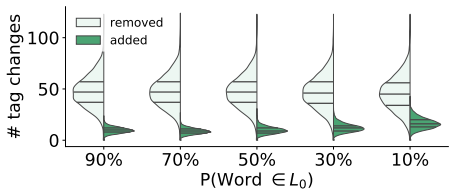
We further investigated the parameters of our topic model. We noted above that the topic model has a switching parameter, $x_{d,n}$ that controls when tags are corrected. We conducted a parameter sweep to control how much switching effects the outcome of our experiment. We focused on a single tagging-assignment methodology, **Word-Matching**, and tested its performance when tweaking this parameter in the model. As shown in Fig. 7j, the conservative 10% threshold, on which we ran our initial experiments, is not the most effective setting. It appears as though a higher switching probability, 30%, yields the best accuracy for our task.

Higher switching probabilities yielded higher numbers of tags per document, as shown in Fig. 6c. This is primarily driven by more added tags (Fig. 6b[16]). We are continuing to explore the implications of this observation.

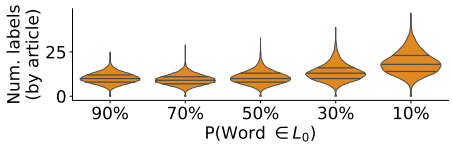
[16]Note that Fig. 4b and Fig. 6c show different number of labels for **Word-Matching** at 10% switching probability. This is because 6c is calculated based off of the input to the topic model, which by default, eliminated infrequently applied tags.



(a) Scores on WS method across a grid of parameters in the topic model.



(b) Tag changes across parameter grid.



(c) Number of tags across parameter grid.

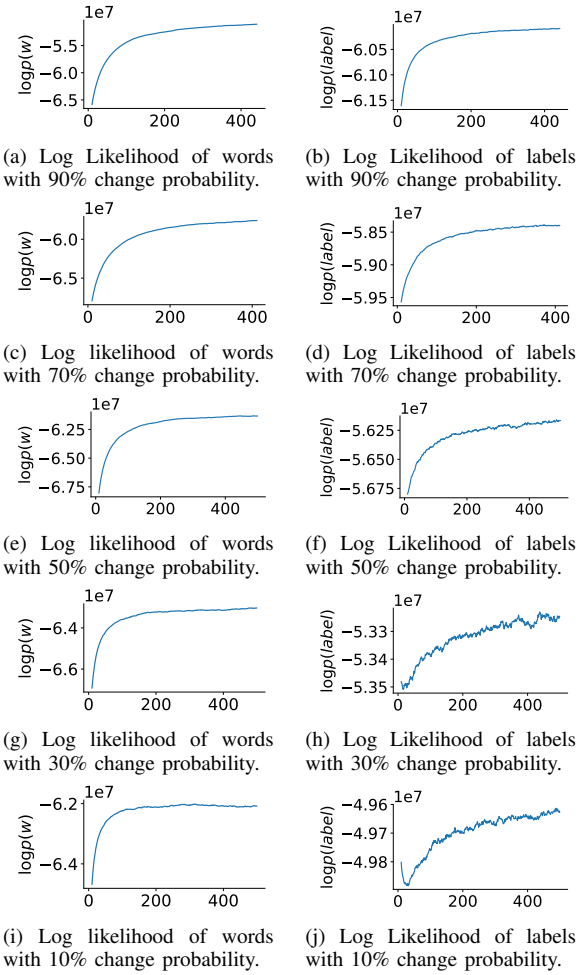
Fig. 6: Tag changes after applying topic-model correction (a) show both additions and removals. For Base, WS and CI, tags are only added after entity augmentation (b).

We also examined on the number of sample iterations we ran our topic model. We observed the changes in likelihood for the labels and the words assigned by the model at every iteration. We show in Fig. 7 that for all configurations, the likelihood of words in our corpus converges quickly across iterations. For high switching probabilities (like 90%), the labels appears to converge quickly as well. For lower switching probability, the model does not converged after 500 sampling iterations. This suggests room for increases in accuracy with further processing.

VI. Conclusions

In the Internet age, the widespread availability of information and the rapid production of corpora across domains promises to fuel rapid scientific development. We seek in this paper to demonstrate the power of novel advances in information retrieval powered by cognitive science, and how a combined approach can improve techniques in keyword assignment, topic modeling and entity extraction.

In this paper, we have explored foundations to our aim to deliver a functional *research assistant* for domain scientists and researchers. We have demonstrated the potential to accurately map documents into an ontology, and combine relevant information from these documents to characterize the ontology. In the future, we plan to extend our work in the following three directions. First, we plan to explore how to enable automatic ontology corrections with justifications. Second, we plan to study how to systematically extract factual relationships from articles. Third, we will quantitatively measure tagging correctness in human trials.



(a) Log Likelihood of words with 90% change probability.
(b) Log Likelihood of labels with 90% change probability.
(c) Log likelihood of words with 70% change probability.
(d) Log Likelihood of labels with 70% change probability.
(e) Log likelihood of words with 50% change probability.
(f) Log Likelihood of labels with 50% change probability.
(g) Log likelihood of words with 30% change probability.
(h) Log Likelihood of labels with 30% change probability.
(i) Log likelihood of words with 10% change probability.
(j) Log Likelihood of labels with 10% change probability.

Fig. 7: The likelihood of words (left column) and labels (right column) observed by the model over iterations.

References

[1] S. Beliga, A. Meštrović, and S. Martinčić-Ipšić. An overview of graph-based keyword extraction methods and approaches. *Journal of information and organizational sciences*, 39(1):1–20, 2015.

[2] D.M Blei, A.Y. Ng, and M.I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[3] X. Duan, J. Zhang, R. Ramachandran, P. Gatlin, M. Maskey, J.J. Miller, K. Bugbee, and T.J. Lee. A neural network-powered cognitive method of identifying semantic entities in earth science papers. In *Proceedings of IEEE International Conference on Cognitive Computing (ICCC)*, pages 9–16. IEEE, 2018.

[4] Anthony Fader, Stephen Soderland, and Oren Etzioni. Identifying relations for open information extraction. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1535–1545. Association for Computational Linguistics, 2011.

[5] T.L. Griffiths, M.I. Jordan, J.B. Tenenbaum, and D.M. Blei. Hierarchical topic models and the nested chinese restaurant process. In *Advances in Neural Information Processing Systems*, pages 17–24, 2004.

[6] E.D. Jr. Hirsch. Reading comprehension requires knowledge - of the words and the world. *American Educator*, 27(1):10–13,16–22,28–29,48, 2003.

[7] A. Hulth. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 216–223. Association for Computational Linguistics, 2003.

[8] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*, 2016.

[9] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and

D. McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, 2014.

[10] A. McCallum, D. Freitag, and F.C. Pereira. Maximum entropy markov models for information extraction and segmentation. In *Icml*, volume 17, pages 591–598, 2000.

[11] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013.

[12] Viet-An Nguyen, Jordan L Ying, Philip Resnik, and Jonathan Chang. Learning a concept hierarchy from multi-labeled documents. In *Advances in Neural Information Processing Systems*, pages 3671–3679, 2014.

[13] D. Ramage, D. Hall, R. Nallapati, and C.D. Manning. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing: Volume 1*, pages 248–256. Association for Computational Linguistics, 2009.

[14] Robert G Raskin and Michael J Pan. Knowledge representation in the semantic web for earth and environmental terminology (sweet). *Computers & geosciences*, 31(9):1119–1125, 2005.

[15] Alexander S. Building the next new york times recommendation engine. *The New York Times*, 2015.

[16] R. Steinberger. Cross-lingual keyword assignment. *Procesamiento del lenguaje natural, nº 27 (septiembre 2001); pp. 273-280*, 2001.

[17] C. Zhang and H. Xu. Using citation-knn for automatic keyword assignment. In *Proceedings of International Conference on Electronic Commerce and Business Intelligence*, pages 131–134. IEEE, 2009.