2018 IEEE International Conference on Cognitive Computing

A Neural Network-Powered Cognitive Method of Identifying Semantic Entities in Earth Science Papers

Xiaoyi Duan, Jia Zhang
Carnegie Mellon University
Mountain View, CA 94087
{xiaoyi.duan;jia.zhang}@sv.cmu.edu

Rahul Ramachandran, Patrick Gatlin, Manil Maskey
NASA/MSFC
Huntsville, AL 35811
{rahul.ramachandran;patrick.gatlin;manil.maskey}@nasa.gov

Jeffrey J. Miller, Kaylin Bugbee
University of Alabama in Huntsville
Huntsville, AL 35811
{jjm0022;kmb0100}@uah.edu

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

***Abstract*—In the current era of knowledge explosion, it is becoming increasingly critical to help researchers quickly grasp the core ideas and methods used in the sea of published articles. As a first step toward the aim, this paper proposes a novel approach that simulates the cognitive process of how human beings read Earth science articles, and automatically identifies semantic entities from the articles. Among others, one major objective is to identify the datasets studied in articles. Oftentimes, however, researchers do not explicitly cite the datasets used. Thus, we propose a profile-matching method strengthened by a neural network-based method to identify implicitly cited dataset entities based on the context. Our experiments have demonstrated the effectiveness of our approaches.**

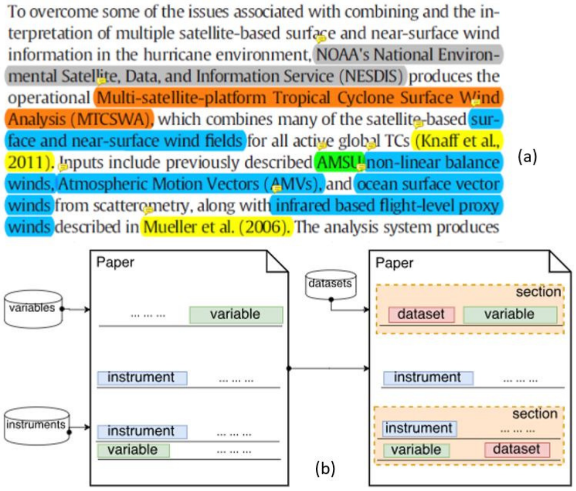
Keywords*—*Cognitive computing, automatic paper understanding, semantic entity identification.

I. Introduction

In order to stand on the shoulders of giants to conduct scientific exploration effectively and efficiently [1], it is critical for researchers to keep on reading publications to understand existing research methods and outcome. However, as the number of publications increases significantly in the current era of knowledge explosion, quickly reading and understanding all related papers remains a substantial challenge for most researchers. With the advancement of artificial intelligence, computer scientists have been attempting to train machines to help people both read and understand research papers. However, it is not a trivial task for several significant reasons:

1) Nearly all of the information contained within research papers is unstructured, which makes it difficult to parse papers.
2) There may not be an agreed upon method for describing certain things within a domain, for example phenomena and other science concepts.
3) It is impractical to manually create a big volume of labeled data to serve as ground truth for machine learning.

In this project, we aim to adopt cognitive computing techniques to tackle such challenges. We study how domain scientists read a paper and understand its major content, and then simulate how the human brain works in order to train machines. As a first step, we combine rule-based and neural network methods to simulate how researchers read a paper, which is a cognitive process to identify important entities from the article. Without losing generality, we focus on research papers in the Earth science domain.



To overcome some of the issues associated with combining and the interpretation of multiple satellite-based surface and near-surface wind information in the hurricane environment, NOAA's National Environmental Satellite, Data, and Information Service (NESDIS) produces the operational Multi-satellite-platform Tropical Cyclone Surface Wind Analysis (MTCSWA), which combines many of the satellite-based surface and near-surface wind fields for all active global TCs (Knaff et al., 2011). Inputs include previously described AMSU non-linear balance winds, Atmospheric Motion Vectors (AMVs), and ocean surface vector winds from scatterometry, along with infrared based flight-level proxy winds described in Mueller et al. (2006). The analysis system produces

Fig. 1: How Geo-scientists Identify Semantic Entities in a Paper

Through a series of workshops with Earth scientists, we have summarized what can generally be learned from an Earth science paper. At a high level, a paper typically describes how to conduct an analytical study over some datasets in order to draw conclusions to support or disprove a scientific hypothesis. While a dataset may possess observed or computed values of a number of variables (i.e., properties), one paper usually focuses on a subset of the variables. A reader thus intends to learn the data analytics process described in the paper. Such a cognitive process is illustrated in Fig. 1a, where a reader has browsed through a segment of a hurricane paper and identified some semantic entities related to the data analytics research. As shown in Fig. 1a, the reader highlighted an instrument

978-1-5386-7241-9/18/$31.00 ©2018 IEEE
DOI 10.1109/ICCC.2018.00009

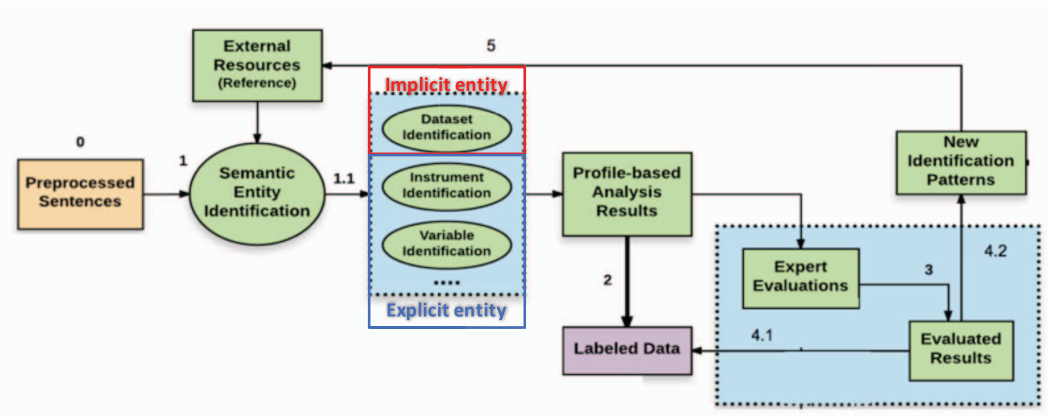


Fig. 2: Framework of Semantic Entity Identification

(NESDIS), a dataset (MTCSWA) and variables studied (several wind properties). Identification of semantic entities will be a major first step in helping researchers understand key contextual information about a paper. We hypothesize that machines can be trained to identify semantic entities within a research paper.

Our workshops also revealed another common identification obstacle. As shown in Fig. 1b on the left-hand side, some semantic entities can be easily identified (such as instruments and variables), while others (such as datasets used) are more difficult to be directly identified. As shown in Fig. 1b on the right-hand side, the citation of a dataset may not be explicit. Until recently, many datasets were not assigned unique digital object identifiers (DOIs). As a consequence, many geo-scientists are just beginning to cite dataset DOIs in their papers. For example, from the datasets indexed by NASA SEDAC[1] Distributed Active Archive Center (DAAC), we studied how the datasets are cited in journal papers. Among about 1,300 publications focusing on atmosphere research, we found that only 35 (less than 3%) articles actually cite the dataset DOIs. Additionally, since many dataset names are sometimes descriptively long, authors often do not use the full name to cite the datasets either. Interestingly, however, human readers often experience no difficulties in deducing which datasets are mentioned in a paper. Our study over a reader's decision making process of identifying the cited datasets has exposed a heuristic process. As shown in Fig. 1, if a dataset is implicitly cited, in a contextual section of the paper, authors usually mention the instrument that collects the datasets, the project that creates the dataset, as well as the specific variables of the dataset studied in the paper.

Based on our understanding of how human beings read a paper, we have developed a technique that trains a machine to simulate such a cognitive process of identifying semantic entities in a research article. Our contributions are three-fold:

1) We have formalized a cognitive process to identify semantic entities in research papers.
2) We have developed a neural network model to identify datasets implicitly cited given a context of surrounding entities.
3) We have implemented experiments to verify the effectiveness of our methods.

The remainder of the paper is organized as follows. Section II describes an overall framework and formally defines the problem of identifying semantic entities from academic publications. Section III presents the detailed profile-based and neural network-based methodologies. Section IV presents experiments and discussions. Section V discusses related work. Finally, Section VI draws conclusions.

II. Overall Framework

The blueprint of our overall methodology and its comprising steps are presented in Fig. 2. Two key components make our methodology unique and innovative. First, the Semantic Entity Recognition for Earth Science uses external resources (including metadata catalogs and controlled vocabularies such as a dictionary of instruments that are known in the Earth science field) as references to guide entity extraction and recognition (i.e., labeling) from unstructured text, in order to build a large training set to seed the subsequent auto-learning component. This process is both objective and can scale up as it requires minimum human interferences. Second, the machine learning-powered auto-learning goes beyond heuristics-based entity recognition and leverages state-of-the-art machine learning technique to incrementally learn and refine the rules and patterns through iterations. To make the overall methodology easier to understand, we have labeled each step using step numbers in Fig. 2.

A. *Stepwise Methodology*

Step 0 consists of preprocessing unstructured text. Preprocessing covers tasks such as paper format transformation, and identifying specific sections of a paper (such as abstract, introduction, methodology, and conclusions). A number of tools are adopted, such as the Apache Tika toolkit[2] for converting a paper from PDF format to HTML, and the pdffigures tool[3] for analyzing and extracting figures, tables and associated captions. "The Structure of a Scientific Paper" [1] is used as a guide to identify sections such as Introduction, Background,

[1] http://sedac.ciesin.columbia.edu/

[2] https://tika.apache.org/

[3] pdffigures.allenai.org

Data and Methods, and References. Dividing sections is useful as the probability of finding certain entities is larger in certain sections as compared to others. For example, the possibility of finding citation of a dataset is higher in section Data and Methods than in section Introduction.

Step 1 aims for semantic entity recognition. We first build multi-dimensional profiles for Earth science-oriented semantic entities using their existing metadata. For example, we construct profiles for each NASA dataset using the metadata records stored in the Common Data Repository (CMR) catalog. Such profiles are used to identify datasets in papers through match-making processes. Specific details will be described in Section III.

Step 2 aims for profile-based analysis, and the profile-based analysis results will then be labeled. Labels such as $variable$, $organization$, and $dataset$ are attached to corresponding sentences. Here is an example sentence labeled with three types of semantic entities that were identified:

PW data was obtained from National Centers for Environment Protection/National Center for Atmospheric Research (NCEP/NCAR) reanalysis data, {*variable*}, {*organization*}, {*dataset*}

Step 4 consists of an optional evaluation by a human subject matter expert. Having an expert-based validation step helps to confirm and enhance the quality of our training set, which will in turn improve the effectiveness of our model training process. The identified semantic entities and the surrounding sentences in the training set will be sent to experts for validation. Passing validation, the sentences will be properly labeled and added into the training set (Step 4.1). New patterns and rules may be identified from the validated results (Step 4.2). Incorrect sentences will be removed from the training set. Newly identified patterns and rules will be added to the pattern library (Step 5). The overall process will iterate until the training datasets remain stable.

B. Semantic Entity Extraction

The goal of this research is to automatically extract key semantic entities from research papers which are important for user to understand the papers. For example, after reading a paper, a geoscientist intends to find out the answers to questions such as: what $instruments$ or $platforms$ are used in the paper? and what $variables$ and $datasets$ are studied? In general, key entities can be divided into two types in terms of how they are mentioned or described in papers: explicit entity and implicit entity. An explicit entity is mentioned by the entity name explicitly, while an implicit entity is usually mentioned implicitly and described by sentences in close proximity to the entity. In the Earth science domain, $instrument$, $platform$ and $variable$ are typically explicit entities that are cited by certain names, but $datasets$ are usually an implicit entity. Our investigation reveals that Earth scientists typically do not cite $datasets$ directly nowadays. However, in the domain of medical science, most concepts are explicit entities because they are required to be precisely referenced.

Therefore, we have developed different methods to extract explicit entities and implicit entities, respectively, as shown in Fig. 2. For explicit entities, it is intuitive to extract them by using handcrafted rules and patterns. For implicit entities, extra information other than entity names is required to help us identify them. Furthermore, extracted explicit entities can be leveraged to help identify implicit entities. After studying how Earth scientists identify possible datasets mentioned in a paper, we combine rule-based and neural-network-based methods to identify possible datasets. The two methods are complementary with each other. Handcrafted rules are good at fine-grained linguistic analysis, while deep learning is effective at coarse-grained tasks. In other words, rules help us find ”precise” results, which can be used by deep-learning algorithms to train models. Afterwards, deep learning algorithms are adopted to find missing results.

Definition 2.1 (Semantic Entity Identification): The process of identifying semantic entities is formalized as to automatically identify key semantic entities $e_1, e_2, ..., e_n$ from contents of paper p. Such entities can be categorized into two classes, explicit entity Ex and implicit entity Im. In Earth Science domain, instrument I, variable V and platform P are explicit entities, $I, V, P \in Ex$. Dataset D is an implicit entity type, $D \in Im$. Therefore, entities $e_1, e_2, ..., e_n$ are represented as $i_1, ..., i_i \in I$, $v_1, ..., v_j \in V$, $p_1, ..., p_k \in P$, and $d_1, ..., d_m \in D$ where $i + j + k + m = n$.

III. Semantic Entity Identification Methodology

In this section, we present a technique to identify semantic entities in research papers. Without losing generality, we focus on research papers in the Earth science domain. We first define heuristic rules to extract explicit entities, e.g., *instruments* and *variables*. Then we present weighted-profile-matching method and neural-network-based method to identify implicit entities, specifically *datasets*.

Our technique basically simulates the cognitive process illustrated in Fig. 1b. For each paper (left-hand side ”document”), variables (green rectangles) and instruments (blue rectangles) are first extracted based upon their libraries. Afterwards, on the right-hand side in ”paper,” we can infer areas (orange dashed boxes) possibly describing datasets (red rectangles) around previously extracted variables and instruments.

A. Heuristic Extraction

Rule-based extraction is applied to identify explicit entities, i.e., *instruments*, *platforms* and *variables*. In the simplest case, instruments and platforms can be identified by entity names. In more complex cases, the same variable names may belong to multiple higher-level concepts. In order to address possible ambiguity, we have to consider both name and contextual information, i.e., class information, when identifying a variable. Here we introduce the extraction rules for instruments and variables only, while the rules for platforms are similar to those for instruments.

1) Instrument: The name of an instrument typically comprises a full name and an acronym. For example, ”GPS” stands for ”Global Positioning System.”

Definition 3.1 (Instrument): The name of an instrument is a tuple $t_i = (S, L)$, where S is the short name for the instrument, and L is its long name.

In the instrument identification process, both the long name and short name will be considered. When short names are found in a sentence, we further check if their long names appear in the full text of the paper. At its first appearance in a paper, an instrument name is mostly in the format of $S(L)$ or $L(S)$. Offline, we apply a known approach in biomedical text [2] to extract all long names and their acronyms from the content of papers in the corpus to enrich our instrument dictionary.

The rules of matching each instrument within a sentence are summarized as follows: (1) Use L-searching in the sentence: if L is found, then consider the instrument as a candidate; (2) Use S-searching in the sentence and also check if L appears in the full content of the paper: if S and L are both found, then consider the instrument as a candidate; (3) If L is included by another candidate's full name, discard the candidate with shorter name.

2) Variable: Every variable has a name and one or more class attributes (e.g., GCMD topic). The attributes are "topic" and "term," which classify the variable into a specific context. For example, "sea surface temperature" is the name of a variable, which is classified into topic "*OCEANS*" and term "*OCEAN TEMPERATURE*."

Definition 3.2 (variable): A variable is a tuple $t_v = (n, \{c\}1..*)$, where n is the variable name, and each c is a set of attribute keywords made up of a topic keyword TP and a term keyword T, i.e., $c = \{c|c \in (TP, T)\}$.

If one variable has multiple class attributes, for example, $t_v = (rainfall\ amount, \{(Precipitation, Precipitation\ Amount), (Precipitation, Rain)\})$, the variable named "*rainfall amount*" has two class attributes, which are topic "*Precipitation*," and two terms "*Precipitation Amount*" and "*Rain*."

The rules of matching each variable t_v within a sentence are summarized as follows. (1) Remove stop words and special characters in the sentence; (2) If n is less than four words, t_v can be a candidate only when n is completely matched in the sentence; (3) If n is more than four words, calculate LCS (longest common subsequence) between n and the sentence. If the ratio of LCS to sentence length (number of words) is greater than a predefined threshold (e.g., 0.7), t_v will be considered as a candidate; (4) If the context keywords $c(TP, T)$ of one of the candidate variables both appear in the full content of the paper and the paragraph containing the sentence (to be stricter), the variable mentioned in the sentence will be considered as a candidate.

After the explicit entities are extracted through our heuristic methods, we use them to train the Conditional Random Field (CRF) model [3] to refine the results.

B. Weighted-Profile-Matching Dataset Extraction

After extracting explicit entities, we utilize them to help identify implicit entities such as *datasets*. According to common writing practices among Earth scientists, datasets are typically mentioned surrounded by some explicit entities as illustrated in Fig. 1b. Hence, the first step is to identify the potential area (sentences) describing a dataset, and then determine which dataset is mentioned. We have developed a profile-oriented approach to realize this goal. The overview of our profile-matching method is shown in Fig. 3. Without losing generality, all NASA Common Metadata Repository (CMR) datasets are examined and profiled based upon their references to attributes including keywords, platforms, instruments, and variables. Such dataset profiles are stored in a database as domain knowledge. For a given paper, all areas (a sequence of sentences) that potentially cite a dataset are identified. For each such area, we build its profile based on its surrounding identified explicit entities (e.g., instruments, variables, locations, and semantic ontology keywords). The full content of the paper is taken into consideration as well. Thus, the profile of each potential dataset is compared with all CMR dataset profiles in the database in order to confirm its identity.

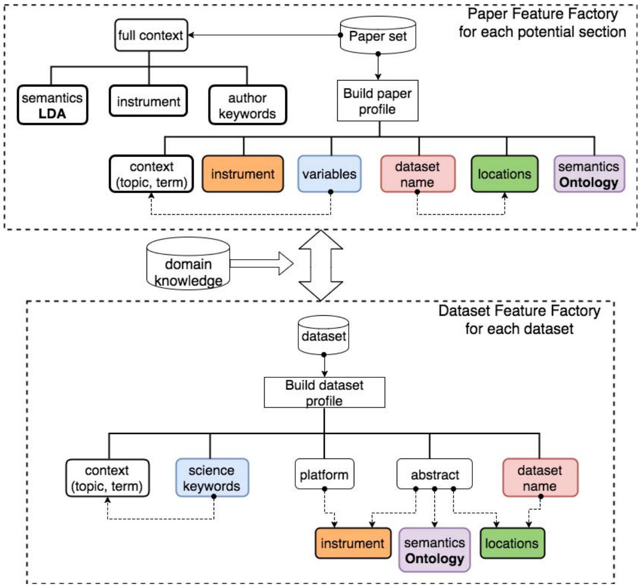


Fig. 3: Profile Matching Framework

The pseudo code of the algorithm is summarized in Alg. 1. In lines 1-3, we construct profiles for each CMR dataset. The attributes considered are extracted from the metadata of the datasets, including name, a paragraph of summary, instrument(s), variable(s), platform(s), and so on. In addition, the summary text is further examined to extract more information such as locations and instruments. Such profiling will result in an attribute vector for each of the CMR datasets, and can be preprocessed offline and stored for real-time query.

In line 6, we locate potential areas in each paper where explicit entities (i.e., instruments, platforms or variables) are mentioned. Line 8 intends to construct a profile vector for each potential area. In lines 10-13, the constructed profile of each potential area is compared with that of every CMR dataset. As shown in Fig. 3, the profile is made up of a feature factory, which can be further extended and enriched in the future. We build the feature factory through multiple features extracted from the paper set and dataset, respectively. Given different weights on features based on domain knowledge, a weighted score is calculated to measure each dataset candidate. The CMR dataset profile with the highest score is considered the most similar one to the specific potential area in the paper.

For different features in the feature factory, we apply different rules to identify them, and different comparison methods to calculate similarity. (1) Dataset name is matched by the LCS algorithm, which similarity is measured by the ratio of the length of LCS to that of the name. (2) There are hierarchical relations among locations where one location can be included by another parent location. For example, ”USA” is included in ”North America.” Every location is extracted to match either a leaf node or a parent node. Note that locations, instruments, and variables are extracted as sets. Therefore, a similarity score can be computed between 0 and 1 based upon the overlap of elements in the profile of the area and the dataset. (3) Semantic words are matched and ranked in all possible root words in the ontology. Without losing generality, in our research we adopt the Semantic Web for Earth and Environmental Terminology (SWEET, https://sweet.jpl.nasa.gov/). Each semantic word extracted will receive a computed score to the root, which results in one or more root score vectors. The similarity of two vectors for the same root is calculated using cosine similarity.

Algorithm 1: Dataset Profiling Matching

```
Result: Extracted datasets
Input  : paper p, weight vector $W \leftarrow (w_t, w_i, w_v)$,
         dataset collection $D$
Output: datasets studied in paper

foreach dataset d do
    $d_i, d_v, d_p \leftarrow$ extract instruments, platforms and
      variables from summary d
end
$max\_score\_dataset \leftarrow null$
$max\_score \leftarrow 0$
$potentialSections \leftarrow$ less than 10 consecutive
  sentences with moret than one type of entities among
  instruments, platforms and variables.
foreach sec in potentialSections do
    $sec_i, sec_p, sec_v \leftarrow$extract instruments, platforms and
      variables from $sec$
    foreach dataset d do
        $score_p \leftarrow$ match($sec_p$, $d_p$)
        $score_i \leftarrow$ match($sec_i$, $d_i$)
        $score_v \leftarrow$ match($sec_v$, $d_v$)
        $S = (score_t, score_i, score_p, score_v)$
        $Score = W \cdot S$
        if $Score > max\_score$ then
            $max\_score \leftarrow Score$
            $max\_score\_dataset \leftarrow d$
        end
    end
end
```

C. Neural Network-Powered Entity Extraction Framework

As explained in the previous section, different attributes in a constructed dataset profile vector weigh differently when computing similarity. However, it is hard to predefine the attribute weights in the profile-matching method. Therefore, we introduce a neural-network-based method to learn such weights and extract implicit entities incrementally. The Continuous Bag of Words (CBOW) is a technique used in Natural Language Processing deep learning [4], which uses context to predict a missing word. In the CBOW model, context is represented by multiple words surrounding a given target word. In our research, we propose a Neural Network Entity Extraction (NNEE) Framework that applies the CBOW model to predict/identify a *dataset* based on the context (surrounding explicit entities) in a paper. The framework is illustrated in Fig. 4, together with an example. Every entity is regarded as a word, and explicit entities in one identified potential area make up one sentence to train the NNEE model. For example, one paper contributes such a sentence ($i_0 d_{26}$) comprising a dataset $d_2 6$ surrounded by instrument i_0. After training, the NNEE model will be able to predict the missing dataset (d_{26}) given its surrounding instrument (i_0) and platform (p_2). The input layer is set to have as many neurons as there are entities extracted from papers' potential areas for training. The hidden layer size is set to the dimensionality of the resulting word vectors. The size of the output layer is the same as the input layer. Thus, assuming that the vocabulary for learning word vectors consists of V entities and N to be the dimension of word vectors, the input to hidden layer connections can be represented by a matrix of size VxN with each row representing an entity. In the same way, the connections from the hidden layer to the output layer can be described by a matrix of size NxV. In this case, each column of the output matrix represents a word from the given vocabulary. The input and output matrices are updated using backpropagation [5], as new contexts are discovered in recently published papers and added into the ground truths.

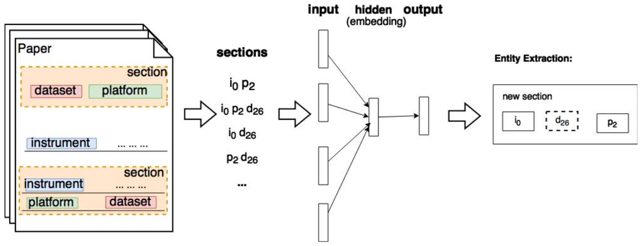


Fig. 4: CBOW-powered Framework

One example is a bag of entities ”i_1, i_2, p_1, v_2” which are identified as explicit entities in an area, representing two instruments (i_1 and i_2), one platform (p_1), and one variable (v_2) are mentioned in the area. Another example ”i_1, p_1, v_1” represents one instrument (i_1), one platform (p_1), and one variable (v_1) mentioned in another area. Given the context of explicit entities, our goal is to extract the most relevant dataset referenced in the context.

1) Neural Network Entity Extraction Algorithm: The cost function is defined as $L = \sum_{d \in D} \log p(d|c_d)$, where d is a dataset from the entity set of *dataset*, D. The goal is to maximize the log probability of the dataset given any context entities. Mathematically, the cost function is

$$\text{J} = -\log p(d|c_d) \tag{1}$$

where $c_d = e_{c-m}, ..., e_{c-1}, e_{c+1}, ..., e_{c+m}$, m is the size of window, and $e_{c-m}, ..., e_{c-1}, e_{c+1}, ..., e_{c+m}$ are extracted entities in the context using the heuristic method described in Section III-A. $p(d|c_d)$ is commonly defined as a softmax function, that is:

$$p(d|c_d) = \sigma({E'}_d^T \cdot c_d) = \frac{exp({E'}_d^T \cdot c_d)}{\sum_{e' \in E} exp({E'}_{e'}^T \cdot c_d)} \qquad (2)$$

where d is the target *dataset* entity, and c_d is the context of entities surrounding d. ${E'}_d^T$ is the row of embedding of entity d in embedding matrix E'. Considering a paper as shown in Fig. 4, the potential area is extracted by identifying more than one type of explicit entity in a range of sentences (described in Section III-B). Each candidate area is transformed as an input to the model. Specifically, if an area includes "i_1, i_2, p_1, v_2," the input vector is an average embedding of "i_1, i_2, p_1, v_2." They can be randomly initialized in the beginning. The output layer will obtain the possibilities for each *entity*, so that datasets can be identified and the most relevant one selected.

After each round of the training process, the model will update entity embeddings. Hence, the model can help enhance the embedding results. When a new paper is published, it can become a new input to aid in further training of the model.

2) Negative Sampling: To achieve efficient optimization, we apply the negative-sampling technique [6]. A small set of entities are sampled from the training data to calculate softmax. Given a negative sample size K, The objective function in Eq. 1 is updated as:

$$J = -\log \sigma(E_d \cdot c_d) - \sum_{i=1}^{K} \log \sigma(-\widetilde{E}_d \cdot c_d) \qquad (3)$$

where $\widetilde{E}_d$ is one of the negative samples from k samples. The model is updated by the stochastic gradient descent algorithm. The derivation of output and embeddings are as follows:

$$\begin{aligned} \frac{\partial J(E)}{\partial E_{u_t^k}} &= (\sigma(E_{u_t^k} \cdot E_v - \mathbb{I}_{c_d}[u_t^k]))E_v \\ \frac{\partial J(E)}{\partial E_v} &= \sum_{k=0}^{K}(\sigma(E_{u_t^k} \cdot E_v - \mathbb{I}_{c_d}[u_t^k]))E_{u_t^k} \end{aligned} \qquad (4)$$

where $\mathbb{I}_{c_d}[u_t^k]$ is an indicator function to indicate whether u_t^k is the context entity of c_d. When $k = 0$, $u_t^0 = c_d$. The pseudo code of our CBOW-powered entity extraction algorithm is summarized in Alg. 2.

```
Algorithm 2: NNEE Dataset Extraction
  Result: Extracted datasets
  Input  : Training set (entity $e \in \{I, P, V\}$), embedding
           dimension $r$, window size $k$
  Output: Entity embeddings $E \in R^{|N(e)| \times r}$
  Initialize E
  foreach t in training set do
      foreach entity e in t do
          $E_e^{new} = E_e^{old} - \eta \cdot \frac{\partial J(E)}{\partial E}$
          /* $\eta$ is learning rate, see Eq. 4
             */
      end
  end
```

IV. Experiments

In this section, we evaluate our proposed methods on real publication datasets in the Earth science domain.

A. Experimental Setup

The data set of publication dataset is downloaded from the website of Socioeconomic Data and Applications Center (SEDAC), which is a NASA data center hosted at Columbia University. The dataset is a collection of publications with manually identified citations over sedac data collection, which is publicly available[4]. A sedac data collection may contain multiple data sets. However, since the website only provides citations of data collections, we use sedac data collection as an equivalent to dataset in our entity identification experiments. A data collection is cited in a paper, meaning that it is mentioned or referenced in the paper.

Publication: Considering the domain expertise possessed by our research team, we concentrate on papers on atmosphere research (i.e., Science Direct[5], Wiley Online Library[6], and American Meteorological Society[7]). Since our technique requires access to full content, papers not publicly accessible were not considered. Some older papers only have PDF versions, which introduce a lot more parsing difficulties. Thus, they are excluded in this experiment. We thus crawled and parsed a total of 849 publications. In this testbed, about 90.81% papers cite one data collection, 9.19% papers cite two data collections, and only a few cite more than two data collections. As a result, we found 195 papers citing sedac data collections by collection names, which can be regarded as ground truth.

Sedac data collection: There are 41 sedac data collections in total.

Semantic entities: The semantic entities studied in our paper include instruments, platforms, datasets and variables. The numbers of the entities are summarized in Table I.

TABLE I: Source of Semantic Entities

Instrument	1,391
Platform	821
Variable	3,090
Data collection	41

B. Entity Extraction Experiments

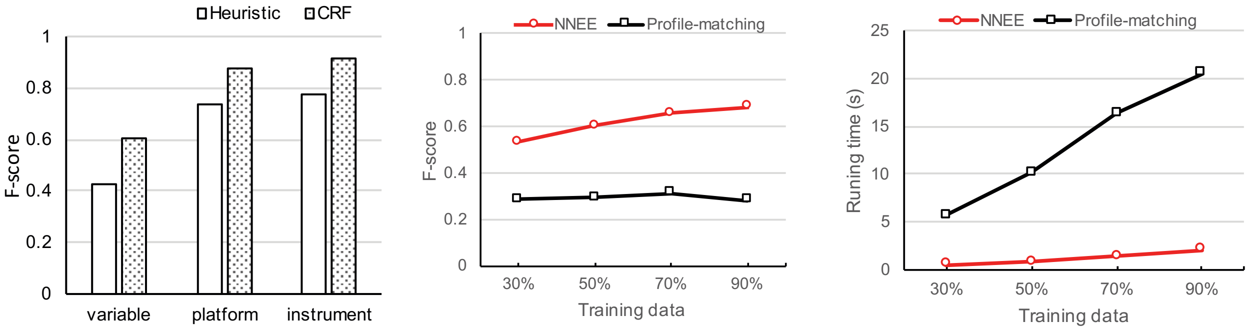
1) Explicit Entity Extraction: From the 849 papers in the test bed, we used the extracted entities through our heuristic methods to train the Conditional Random Field (CRF) model [3]. A randomly selected 14 new papers from the Dust area were identified for manual evaluation. A team of five domain scientists from NASA read those papers throughly and recorded their identified entities for each paper independently. To calculate the accuracy, precision is the number of correct entities divided by the number of all identified

[4] http://sedac.ciesin.columbia.edu/citations-db

[5] http://www.sciencedirect.com

[6] http://onlinelibrary.wiley.com/

[7] http://journals.ametsoc.org



(a) Explicit Entity Extraction

(b) F-score comparison

(c) Comparison of training time and query time

Fig. 5: Experiment Results

results, while recall is the number of correct entities divided by the number of entities that should have been identified. The calculation of F-score is: $F_1 = 2 \cdot \frac{\text{precision}\cdot\text{recall}}{\text{precision}+\text{recall}}$. The results of F-score comparing between the heuristic method and the machine learning method in identifying variables, platforms and instruments are summarized in Fig. 5a. From the results, both methods achieve high accuracy in identifying instruments and platforms than variables, because *platform* and *instrument* names are more formalized while *variables* names are less rigorously defined. Additionally, CRF increases 15% in identifying explicit entities comparing to the heuristic extractions in average. This experiment shows that the heuristic method can help us build large amount of training data, then machine learning can build a model to refine the results.

2) Implicit Entity Extraction: Based on the extracted explicit entities, the corresponding sections were exported as candidate sections to evaluate the profile-matching method and the neural-network method in extracting implicit entities, i.e., *sedac data collections*.

For the profile-matching method, all dataset profiles were preprocessed to extract entities and stored in a database. Profile-matching method selects the dataset with the highest matching score within the candidate section. For our neural-network-powered method, the exported sections with sedac data collections explicitly mentioned were used as ground truth (273 sections). In the training process, we set the count of ignoring word as 1, window size as 3, the size of each embedding as 20, and the number of negative samples as 5. We used 30%, 50%, 70% and 90% of ground truth data to train the NNEE model, and the same number of candidate sections to match in the preprocessed *dataset* profiles. The remaining data is for testing. After training the NNEE model, we used the model to output *datasets* by inputting a context of explicit entities from the test data. If the output *dataset* falls in the same cluster as the test data, it is considered as correct.

The first experiment evaluates the accuracy of the results by calculating the F-score. Fig. 5b shows that the F-score of our NNEE method is significantly higher than that of the profile matching-based method. The F-score for the profile-matching method remains unchanged as the training data size is incrementally increased, but the F-score for the NNEE method increases if more training data is added. If new data is added into the training data, the model was retrained and enhanced.

The second experiment compares the running time to train the NNEE model and profile matching method. Fig. 5c shows that the running time of using different sizes of training data in NNEE is less than five seconds and improves slowly. However, the query time of profile matching is almost linearly increasing as the input data is incrementally increased. This is due to the fact that the query time of profile matching is proportional to the size of input.

V. Related Work

Knowledge extraction is widely applied and studied in many areas, such as natural language processing, text mining and knowledge graph. The task is to extract information from unstructured and ambiguous text, such as sentences and documents, and then use the knowledge to build knowledge base systems. Google Knowledge Graph[8] (name changed to Knowledge Vault) [7] project integrates both structured data (e.g., Freebase[9], Wikipedia[10]) and unstructured data (e.g., the web) to answer user queries. DeepDive[11] [8] from Stanford provides an engineering procedure to process dark data into databases. GeoDeepDive [9] is an extension of DeepDive, focusing on processing dark data from geological articles. Microsoft Academic Graph (MAG)[12] [10] builds a heterogeneous graph aiming to support generic scientific research on scholarly big data. IBM Watson [11] aims to create a question answering (QA) computing system by learning domain knowledge from various sources. On building these knowledge bases, knowledge extraction relies on extensive human involvement by defining hand-crafted extraction rules or hand-labeled training data. According to Nickel et al. [12], existing knowledge base construction efforts can be divided into two categories, based on whether a fixed or open lexicon of entities are employed. In a schema-based approach, on the one hand, tuples (entities and relationships) are represented by globally unique identifiers, and all possible relationships are predefined in a fixed vocabulary. In a schema-free method, on the other hand, an Open Information Extraction (OpenIE) [13] technique is adopted so that tuples are represented via normalized but not disambiguated strings.

[8] https://www.google.com/intl/en_us/insidesearch/features/search/knowledge.html
[9] https://developers.google.com/freebase/
[10] https://en.wikipedia.org/wiki/Wikipedia:Database_download
[11] http://deepdive.stanford.edu/
[12] https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/

Entity extraction is the first and basic step in the knowledge extraction process, which is the focus of our paper. Extracting entities from unstructured textual data remains a challenging task due to the uncertainty of description in unstructured text. One solution with high precision is to specify rules and ontologies [14] manually. In regard to efficient semantic entity extraction methods, works can be viewed from three supervision aspects, i.e., supervised learning, semi-supervised learning and unsupervised learning. Supervised learning is the most common way in knowledge extraction. Named entity recognition [3] is one representative technique, which uses the heuristically-labeled training data to train a Conditional Random Field (CRF) [15] model. However, such methods demand a lot of labeled data, which is expensive and time consuming.

Unsupervised learning to extract knowledge is the cheapest way to induce entities and relations from corpus. For example, [14] introduces a system without requiring human input to extract entities and relations. However, the extracted results led to significant amount of noises and lack of semantics. For semi-supervised classification problem, [16] and [17] extend the semi-supervised expectation-maximization (EM) [18] algorithm for flat and hierarchical classification tasks, respectively. They can discover new classes from unlabeled data. For example, [19] improves the unsupervised learning system.

In contrast to existing general-purpose entity recognition methods, our approach focuses on domain-oriented entity extraction from research articles. Cognitive process is studied thus simulated to develop a heuristics-based and neural network-powered approach, which provides a solution to incrementally build a large training dataset with minimal manual intervention.

VI. Conclusions

In this research, we have developed techniques to automatically extract semantic entities from unstructured academic papers, simulating the cognitive process of how humans read articles. A layered framework is presented to extract implicit entities and explicit entities. Our experimental results show that such a model achieves higher accuracy than using rule-based or profile-matching method alone. Although our work focuses on the Earth science domain, our technique can be applicable to other domains where semantic entities may be implicitly referenced in papers.

We plan to continue our research work in the following three directions. First, we will study deep learning methods to further enhance the accuracy of implicit semantic entity extraction. Second, we will build a software portal to enable and facilitate researchers in providing evaluation results which will be used as incremental ground truths to better train our model. Third, we will try to extend our approach to other research domains such as astronomy.

Acknowledgement

This work is partially supported by National Aeronautics and Space Administration under grant NNX16AE15G, and National Science Foundation under grant ACI-1443069.

References

[1] F. Suppe, “The structure of a scientific paper,” *Philosophy of Science*, vol. 65, no. 3, pp. 381–405, 1998.

[2] A. S. Schwartz and M. A. Hearst, “A simple algorithm for identifying abbreviation definitions in biomedical text,” in *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, 2003, pp. 451–62.

[3] J. R. Finkel, T. Grenager, and C. D. Manning, “Incorporating non-local information into information extraction systems by gibbs sampling,” in *Proceedings of 43rd Annual Meeting of the Association for Computational Linguistics*, 2005, pp. 363–370.

[4] 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*, 2013, pp. 3111–3119.

[5] M. Riedmiller and H. Braun, “A direct adaptive method for faster backpropagation learning: The rprop algorithm,” in *Proceedings of IEEE International Conference on Neural Networks*, 1993, pp. 586–591.

[6] Y. Goldberg and O. Levy, “word2vec explained: deriving mikolov et al.’s negative-sampling word-embedding method,” *arXiv preprint arXiv:1402.3722*, 2014.

[7] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, and W. Zhang, “Knowledge vault: a web-scale approach to probabilistic knowledge fusion,” in *Proceedings of The 20th International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 601–610.

[8] F. Niu, C. Zhang, C. Ré, and J. W. Shavlik, “Deepdive: Web-scale knowledge-base construction using statistical learning and inference,” in *Proceedings of VLDS*, ser. CEUR Workshop, vol. 884, Istanbul, Turkey, 2012, pp. 25–28.

[9] C. Zhang, V. Govindaraju, J. Borchardt, T. Foltz, C. Ré, and S. Peters, “Geodeepdive: statistical inference using familiar data-processing languages,” in *Proceedings of the International Conference on Management of Data*, 2013, pp. 993–996.

[10] *An Overview of Microsoft Academic Service (MAS) and Applications*, 2015.

[11] R. High, “The era of cognitive systems: An inside look at ibm watson and how it works,” *IBM Corporation, Redbooks*, 2012.

[12] M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich, “A review of relational machine learning for knowledge graphs,” *Proceedings of the IEEE*, vol. 104, no. 1, pp. 11–33, 2016.

[13] O. Etzioni, M. Banko, S. Soderland, and D. S. Weld, “Open information extraction from the web,” *Communications of the ACM*, vol. 51, no. 12, pp. 68–74, 2008.

[14] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. H. Jr., and T. M. Mitchell, “Toward an architecture for never-ending language learning,” in *Proceedings of the 24th Conference on Artificial Intelligence*, 2010.

[15] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, “Conditional random fields: Probabilistic models for segmenting and labeling sequence data,” in *Proceedings of the 18th International Conference on Machine Learning*, 2001, pp. 282–289.

[16] B. Dalvi, W. W. Cohen, and J. Callan, “Exploratory learning,” in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2013, pp. 128–143.

[17] B. Dalvi, A. K. Mishra, and W. W. Cohen, “Hierarchical semi-supervised classification with incomplete class hierarchies,” in *Proceedings of the 9th ACM International Conference on Web Search and Data Mining*, 2016, pp. 193–202.

[18] T. K. Moon, “The expectation-maximization algorithm,” *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp. 47–60, 1996.

[19] S. Soderland, J. Gilmer, R. Bart, O. Etzioni, and D. S. Weld, “Open information extraction to KBP relations in 3 hours,” in *Proceedings of the 6th Text Analysis Conference*, 2013.