End-to-End NLP Knowledge Graph Construction

Ishani Mondal[*]
Microsoft Research
Bengaluru, India
ishani340@gmail.com

Yufang Hou
IBM Research Europe
Dublin, Ireland
yhou@ie.ibm.com

Charles Jochim
IBM Research Europe
Dublin, Ireland

Abstract

This paper studies the end-to-end construction of an NLP Knowledge Graph (KG) from scientific papers. We focus on extracting four types of relations: *evaluatedOn* between *tasks* and *datasets*, *evaluatedBy* between *tasks* and *evaluation metrics*, as well as *coreferent* and *related* relations between the same type of entities. For instance, "*F1 score*" is coreferent with "*F-measure*". We introduce novel methods for each of these relation types and apply our final framework (*SciNLP-KG*) to 30,000 NLP papers from ACL Anthology to build a large-scale KG, which can facilitate automatically constructing scientific leaderboards for the NLP community. The results of our experiments indicate that the resulting KG contains high-quality information.

1 Introduction

As interest in the NLP research community grows, the number of NLP tasks, datasets, and metrics for evaluation also grows, making it increasingly difficult for researchers to keep track of the plethora of new resources. In order to tackle this problem, recently there have been a few manual efforts to summarize the state-of-the-art on selected subfields of NLP in the form of leaderboards that extract tasks, datasets, metrics and results from papers, such as *NLP-progress*[1] or *Paperswithcode*[2]. However, these manual efforts are not sustainable over time for all NLP tasks.

Meanwhile, although there are various studies focusing on extracting entities and relations from scientific literature (Augenstein and Søgaard, 2017; Luan et al., 2018; Gábor et al., 2018; Hou et al., 2019; Jain et al., 2020), there is less work on constructing KGs that contain rich information about

tasks, datasets, and metrics. Such a KG would be highly beneficial for the researchers to understand the underlying related literature for a particular task, or to perform comparable experiments.

In this paper, we propose an end-to-end approach to construct a KG from the NLP papers. Our KG contains three types of entities (**tasks**, **datasets**, and **metrics**) and four types of relations connecting them. Figure 1 (bottom left) depicts these relations in our large-scale graph built from 30,000 NLP papers from the ACL Anthology. For instance, "sentiment analysis (task)" is **evaluatedOn** "IMDb dataset", "opinion analysis (task)" is **evaluatedBy** "Precision (metric)", "F1-score (metric)" is **coreferent** with "F1-measure (metric)", and "YELP review dataset" is **related** to "IMDb dataset".

We develop a framework (*SciNLP-KG*, Section 5) to extract these relations based on NLP papers that are tagged for Task (T), Dataset (D), and Metric (M) entities using a TDM entity tagger. Our framework primarily consists of three learning-based models. First, motivated by Hou et al.'s work on tagging NLP papers with valid TDM triples based on a small manually created gold TDM taxonomy, we design a hybrid NLI (Natural Language Inference)-based relation extraction model to extract *evaluatedOn* and *evaluatedBy* relations. Our model can extract these two relations at the document level even if the related entities do not appear in the same sentence. Second, for the *coreferent* relation, we use a mention-pair model to identify the same entities within and across documents. We use a few heuristics to generate training instances, such as that authors often use abbreviations to refer to the common terms (i.e., *NER – Named Entity Recognition*). Third, we propose another model, *term2vec*, which is trained on pseudo-sentences that contain tagged TDM entities from different documents. We use the resulting embeddings to extract the *related* relation between similar type of

---

[*] Work done during internship at IBM Research.
[1] https://github.com/sebastianruder/NLP-progress
[2] https://paperswithcode.com

1885

entities. For instance, "*semantic role labeling*" is related to "*argument identification*" and "*GENIA Corpus*" is related to "*NCBI Corpus*".

To evaluate our end-to-end *SciNLP-KG* framework, we manually construct a small-scale NLP KG based on our proposed schema (Section 4.2), which contains 85 nodes and 625 links. Experiments show that our system achieves reasonable results for all relation types on this small-scale graph with all possible meaningful links manually annotated. We further apply our framework on 30,000 NLP papers from ACL Anthology to build a large-scale NLP KG containing 5,374 nodes and 15,762 relations. We evaluate the quality and coverage of the KG by manually assessing random samples and comparing it with *Paperswithcode*. We found that our KG contains high-quality information.

Overall, the contributions of our work are three-fold. First, we propose and design a new schema that represents knowledge about *tasks* (T), *datasets* (D) and *metrics* (M) in the NLP domain. Second, we develop a novel framework (*SciNLP-KG*) for constructing an NLP KG from the scientific literature in an end-to-end manner. Finally, we *automatically* build a large-scale NLP KG that contains high-quality information about the Task-Dataset-Metric (TDM) entities. However, our method is generalized in a way that it could be extended to the domains of computer vision or bioinformatics. Our code and datasets are made publicly available at `https://github.com/Ishani-Mondal/SciKG` to fuel further research.

2   Related Work

There is a wealth of research in the NLP community on extracting information from scientific literature. Earlier work identified citation contexts and then classified them (Teufel et al., 2006; Abu-Jbara et al., 2013; Jurgens et al., 2018). Other lines of research include unsupervised approaches for extracting paper concepts (Gupta and Manning, 2011; Tsai et al., 2013) and keyphrases (Kim et al., 2010; Hasan and Ng, 2014). Here we are most interested in entity and relation extraction in scientific papers (Tateisi et al., 2014; Augenstein and Søgaard, 2017; Gábor et al., 2018). In SemEval 2017 Task 10, Augenstein et al. (2017) focused on extracting three types of entities (*Process, Task, and Method*) and two relation types (*hyponym-of* and *synonym-of*). The dataset from this task has been used to explore various neural models for IE on scientific

literature (Luan et al., 2017; Ammar et al., 2017). Luan et al. (2018) released another dataset which contains annotations for five types of entities and eight types of relations on 500 scientific abstracts. Ammar et al. (2018) built a literature graph for different domains and they focused on identifying entities and linking them to the existing knowledge bases. Hou et al. (2019) developed a system to identify Task-Dataset-Metric triples in NLP papers based on a small set of manually constructed gold TDM triples. Recently, Jain et al. (2020) introduced *SciREX*, a document level IE dataset that encompasses multiple IE tasks, including salient entity identification and document level N-ary relation identification from scientific articles. Unlike the above mentioned work, we concentrate on building a TDM KG by leveraging entity recognition and relation extraction within and across different sentences/documents. Our constructed KG can facilitate large-scale NLP leaderboard construction and help the researchers to gain insights from NLP literature.

3   NLP Knowledge Graph Schema

In this section, we propose an NLP KG schema which contains three types of entities (i.e., *task*, *dataset*, and *metric*) and four types of relations between them. An entity mention is a single or multi-word nominal phrase that represents a task (e.g., *named entity recognition*), dataset (e.g., *IMDB*), or evaluation metric (e.g., *F1-score*) entity. We mainly focus on these three types of entities because they are the core concepts of the NLP community. In addition, they are relatively stable across different papers compared to other types of entities, such as method, result, or experiment.[3] Similarly, we focus on the following four types of relations between TDM entities that represent the collectively shared view among the NLP researchers:

1. evaluatedOn: This relation implies that a task T is often evaluated on a dataset D (e.g., *sentiment analysis* $\rightarrow$ *IMDB*).

2. evaluatedBy: This relation implies that NLP

---

[3]By referring to the stability of the entities, we meant to say that TDM entities are more likely to reflect the collectively shared views in the NLP domain, i.e., researchers will use the same name to refer to the same entity in different papers (e.g., sentiment classification). In contrast, method/result/experiment in research papers are non-standardized in nature and very difficult to normalize. For instance, a main method "LSTM-CNN-sentiment" in one paper might be referred to in another paper as "LC-based text classification".

researchers usually evaluate a task T using metric M (e.g., *Named Entity Recognition → F1*).

3. coreferent: This relation is used to capture the fact that an entity may be referred to differently in the same or different papers, such as *Named Entity Recognition – NER*, *NCBI dataset – NCBI corpus*, or *F1 score – F1 measure*. The *coreferent* relation can help us to canonicalize TDM entities in the constructed KG.

4. related: Similar to word relatedness, this relation captures all types of associations between the same type of entities.[4] For instance, "*semantic role labelling*" is related to *argument identification* because the latter is a sub-task of the former. Also "*GENIA Corpus*" is related to "*NCBI Corpus*" because both datasets are used to develop NER models in the biomedical domain. In practice, there is not a clear way to define all possible relations between TDM entities, and the *related* relation provides a practical and efficient way to navigate KG.

4 Dataset Construction

We create two new datasets for testing NLP KG construction, both of which are derived from the TDM corpus from Hou et al. (2021) (see Section 4.1). The first dataset is a manually constructed small-scale NLP KG according to the schema described in Section 3. The second dataset is constructed for training a model to extract *evaluatedOn* and *evaluatedBy* relations.

4.1 TDM Tagged Corpus

Our target entities are abstract objects of type **Task, Metric, and Dataset (TDM)** which are specific instantiations of the entities in document. We use the recently released state-of-the-art TDM tagger (Hou et al., 2021), trained on the Flair framework (Akbik et al., 2019) based on the cased BERT-base embeddings (Devlin et al., 2019), to obtain the mentions of Task, Dataset and Metrics. This tagger is trained on a corpus of 1,500 sentences taken from the full text of NLP papers, which have been annotated by domain experts for TDM concepts. Finally, it has been applied on 30,000 NLP papers from the ACL Anthology. We refer to the resulting dataset as *TDM-NLP-Papers* for the rest of the paper and

---

[4]In principle, the *related* relation can be applied to different type of TDM entities. Here we only consider it between the same type of entities because the *evaluatedOn* and *evaluatedBy* relations already capture the most prominent relations across TDM entities.

Nodes		Relations		
Type	**Count**	**Type**	**Count**	κ
Task	43	evaluatedOn	81	0.78
Dataset	23	evaluatedBy	249	0.73
Metric	19	coreferent	38	0.84
		related	257	0.94
Total	**85**	**Total**	**625**	

Table 1: Statistics of *smallNLP-KG*. κ indicates inter-annotator agreement for each relation type.

it will be used as input for our proposed framework to construct an NLP KG.

4.2 smallNLP-KG

In order to evaluate our approach to construct an NLP KG more efficiently during model development, we have conducted an annotation study to build a small-scale gold NLP KG following the schema given in Section 3.

Specifically, the first author sampled a small amount of TDM entities from the tagged dataset *TDM-NLP-Papers*. The chosen entities contain well-established tasks (e.g., *dependency parsing* or *named entity recognition*) and their corresponding datasets as well as evaluation metrics. In order to better facilitate the type of *coreferent* edge in the KG, entities that are abbreviated or have a small edit distance to the existing entities are also added to the list (e.g., *NER*).

Two domain experts then independently annotate all possible relations described in Section 3 between any two entities. If necessary, they read the corresponding literature to make a decision. Finally, the annotators reconcile the differences in their annotations and produce the final canonical annotation. The final gold graph contains 85 entities (nodes) and 625 relation instances. Table 1 lists statistics for each entity type and each relation type. We also calculate the inter-annotator agreement per relation type using Cohen's κ (see κ column in Table 1). Overall, we achieve high inter-annotator agreement for all the relation types.

Apart from these relations, we also explored other relations between similar types of entities, such as *hypernym* (Hearst, 1992) and *part-of* relations. In a pilot annotation study, we found that finer-grained relation types between TDM entities are more difficult to annotate, e.g., the relation between "semantic role labelling" and "argument identification". Ultimately, in order to help users to navigate through the built KG, we decide to use the
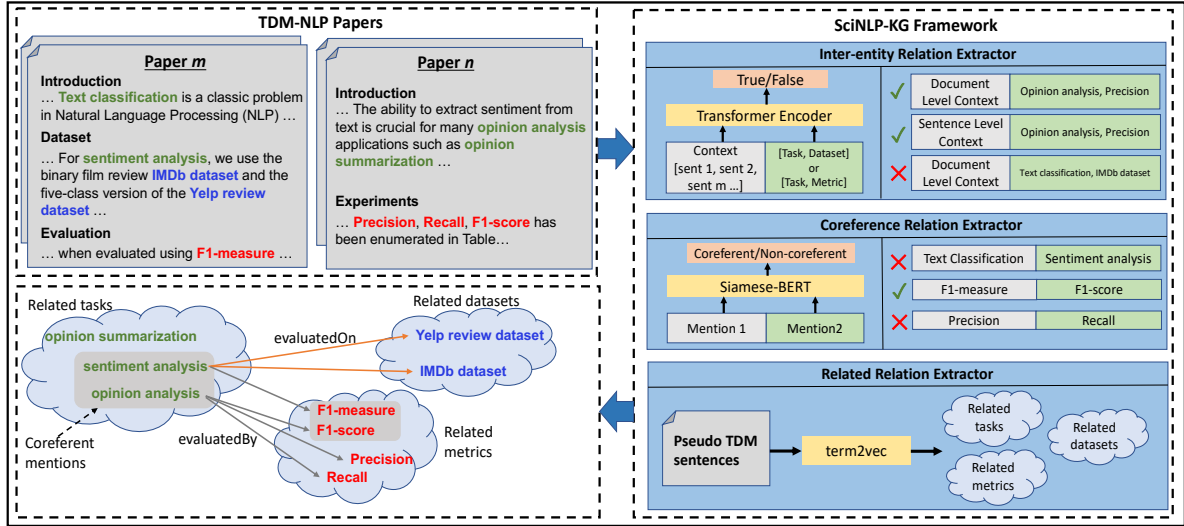
Figure 1: Overview of the SciNLP-KG Framework

"related" relation to capture all types of associated relations between the same type of entities (e.g., when a user clicks "coreference resolution", the system can recommend related tasks).

4.3 TD/TM Training Dataset

To facilitate the extraction of the *evaluatedBy* and *evaluatedOn* relations, we create a corpus (*TD-TM-Rel*) containing 600 sentences randomly chosen from the tagged *TDM-NLP-Papers* corpus. Each sentence has at least two different types of tagged TDM entities. Two domain experts then annotate the valid *evaluatedBy* and *evaluatedOn* relations for each sentence. The inter-annotator agreements on the *evaluatedOn* and *evaluatedBy* relations are 0.96 and 0.91 (Cohen's κ), respectively. Below is an example of TD/TM entities appearing in the same sentence but not expressing *evalutedOn/evaluatedBy* relations: *"As a testbed for this task, we introduce the **SentiHood** dataset extracted from a **question answering** platform where urban neighbourhoods are discussed by users"*. This sentence does not express a valid *evaluatedOn* relation between the "question answering" task and the "SentiHood" dataset.

5 SciNLP-KG Framework

In this section, we describe our proposed framework, *SciNLP-KG*, to construct an NLP KG from unstructured text as shown in Figure 1. Our framework consists of three models to extract four types of relations as described in Section 3. Instead of proposing an end-to-end model (which could suffer from component-wise propagation errors (Wadden

et al., 2019; Jain et al., 2020)), we incrementally develop separate components based on the properties of the target relations and the availability of training datasets, and finally aggregate results of each component to create the final KG.

5.1 Inter-Entity Relation Extractor

The *evaluatedOn* and *evaluatedBy* relations between task and dataset/metric entities depend on the document-level context. It is often the case that the entities involved in these two relations do not occur in the same sentence. On the other hand, a document containing a task mention t and a dataset mention d does not necessarily imply that there exists a positive *evaluatedOn* relation between them.

Sentence-Level NLI Model (S-NLI). We train a S-NLI model based on our corpus *TD-TM-Rel* (Section 4.3). Here *context* is a sentence, and *TD/TM hypothesis* is one of the combinations of TD/TM entities tagged in the sentence. During training, *[CLS]context[SEP]TD/TM hypothesis[SEP]* is given as input to the BERT-based model, followed by sigmoid activation on [CLS] token to predict if the context expresses a valid *evaluatedOn/evaluatedBy* relation or not. The model is trained using binary cross-entropy loss. Note that among all combinations, only the ones that correspond to the annotated *evaluatedOn/evaluatedBy* relations are valid TD/TM hypotheses. At inference time, given a task t and a dataset d (or metric m), we first extract all sentences from the tagged corpus, *TDM-NLP-Papers*, which contain both t

and d (or m). We then try to predict whether an *evaluatedOn* (or *evaluatedBy*) relation can be inferred from any of these sentences using the trained model.

Document-Level NLI model (D-NLI). Instead of looking only at the sentence level, we also adopt an approach in which we form the *TD/TM hypothesis* space at the document level. For example, if a tagged NLP paper has n task entities (t_1, t_2,.... t_n) and m dataset entities (d_1, d_2,.... d_m), we generate $n * m$ combinations of TD tuples as our testing TD hypotheses. For each testing TD hypothesis, we construct *context* by concatenating sentences that contain at least one corresponding entity from the tagged NLP paper. Different from the S-NLI model, the D-NLI model can capture the *evaluatedOn/evaluatedBy* relations across sentence boundaries. We use the recently released *SciREX* corpus (Jain et al., 2020) to train our D-NLI model. *SciREX* contains document-level annotations of $<$ *Task, Dataset, Metric, Method* $>$ tuples as well as the individual entities for 438 NLP papers. Most binary relations such as *Task–Dataset* (*evaluatedOn*) and *Task–Metric* (*evaluatedBy*) occur across sentences. Specifically, for each valid TD/TM relation annotated at the document level, we treat them as a valid TD/TM hypothesis. We form the *context* by concatenating all sentences from the paper that contain the annotations for at least one element in this relation. Similarly, we construct negative training instances using other TD/TM combinations by assuming that 4-ary tuple annotations in *SciREX* contain all valid TD/TM pairs for a specific paper.

Hybrid NLI Model (H-NLI). Although the D-NLI model can capture *evaluatedOn/evaluatedBy* relations across sentences, we assume that the S-NLI model is more accurate because it is easier for the model to learn patterns from shorter contexts. To combine the strengths of both models, we propose a hybrid NLI model. More specifically, given a task t and dataset d (or metric m) and the corresponding NLP papers containing these entities, we first apply our S-NLI model to all sentences containing both entities to decide whether an *evaluatedOn* (or *evalautedBy*) relation is held between them. If such a sentence does not exist, we use D-NLI model to make the final prediction. By doing so, we cascade the outputs from both S-NLI and D-NLI models in a sieve-based fashion, with

the higher priority sieve being the S-NLI model. Thus, it combines the strength of capturing higher precision with S-NLI and better recall with D-NLI.

5.2 Coreferent Relation Extractor

Unlike coreference resolution on news corpora, which mainly depends on context, we notice that in our scenario, researchers often use abbreviations to refer to common terms (e.g., *NER–Named Entity Recognition*). Sometimes different researchers use slightly different variations to refer to the same entity (e.g., *F1 score–F1*). Motivated by these observations that the surface forms play a pivotal role for TDM coreferent relation extraction, we design a mention-pair model to capture the coreferent relations between the same type of entities.

We generate positive training instances for our mention-pair model using a few heuristics. Specifically, we apply a regular pattern to check whether a tagged entity is followed by its abbreviation in brackets in the tagged NLP papers, such as "*Named entity recognition (NER)*". We further extract pairs of entities which have a small edit distance, such as "*F1 score – F1*". Finally, we generate the same number of negative training instances by randomly pairing entities of the same type, which do not meet the criteria of the above heuristics.

We fine-tune our mention-pair model on a *BERT-Siamese Network* (Reimers and Gurevych, 2019). We use mean pooling over the output of two [CLS] tokens and use the Euclidean distance function in the penultimate layer, which is followed by a fully connected softmax layer with two labels (*coreferent* and *non-coreferent*). We form mention-pairs from within and across documents, so this component can be seen as a cross-document coreference resolution module.

5.3 Related Relation Extractor

We observe that in our *TDM-NLP-Papers* corpus, the co-occurrence patterns of TDM entities in different contexts (documents) encode rich semantic information about each individual entity. Motivated by *word2vec* (Mikolov et al., 2013), we propose an unsupervised *term2vec* model to capture related relations between TDM entities.

We hypothesize that the TDM entities in a single document are somehow related, even if they do not occur in the same sentence. We then generate a pseudo-sentence for each tagged paper. A pseudo-sentence per paper treats the whole paper as the

context which contains special "word"–TDM entities. Two such examples are: *"sent0: sentiment analysis, aspect-based sentiment analysis, semeval 2014 task 4 laptop, sentihood, text classification"*, and *"sent1: sentiment classification, semeval 2014 task 4 laptop, sentihood"*. After applying *term2vec* on these pseudo sentences, terms with similar contexts will be close to each other in the embedding space, which can help us to identify related relations between TDM entities.

Specifically, for each paper in *TDM-NLP-Papers*, we concatenate all tagged entities to form a pseudo sentence. We treat each entity (term) as a single word in this sentence and generate vector representations for each term using the Skip-gram model, which preserves the type of each term.

For a term j, the algorithm models the neighborhood of this term as shown in equation (1):

$$\arg \max_{\theta} \sum_{j \in V} \sum_{i \in K_V} \sum_{c_i \in N_i(j)} \log p(c_i|j;\theta) \quad (1)$$

Here, K_V denotes the type of nodes. $N_i(j)$ denotes the neighborhood of the term j with respect to the ith type of terms and $p(c_i|j;\theta)$ represents the conditional probability of having a context term c of type i given the term j. The objective of this algorithm is to predict the neighborhood of a given term of a particular type i to other terms of similar type. After obtaining the d-sized embeddings for each term, we use the unsupervised K-means clustering algorithm to determine the clusters for each term. These clusters, thus generated, are among the same type of entities and encode the *related* relations among these entities.

6 Experimental Setup on smallNLP-KG

We first evaluate our framework on *smallNLP-KG*. We report precision, recall, and F-score fore each relation type. Note that for evaluating *coreferent* and *related* clusters, we consider all pairs of entities in a cluster to be linked.

Dev/test split. The nodes and relations in *smallNLP-KG* are carefully divided in 10-90% dev-test split. To avoid any leakage problem as observed in KG completion tasks (Sun et al., 2020; Pezeshkpour et al., 2020), we exclude training instances from the NLI models and the mention-pair model that involve any of the testing entities. More specifically, first, we went through the whole

smallNLP-KG to make sure any TDM entities (including their coreferent mentions) in the test data do not appear in the dev set. Second, we exclude all these entities (including their coreferent mentions) from the training data for each relation extraction component. For the rule-based and unsupervised baselines of *coreferent* and *related* relations (Section 7.3 and Section 7.4), we tune our parameters on the dev set and report results for all relations on the test set.

Implementation details. Given all testing entities from *smallNLP-KG* as the input, we use our framework presented in the previous section to build the KG.

For the hybrid NLI model to extract inter-entity relations (Section 5.1), we fine-tune all NLI models for 3 epochs with a learning rate of $5e - 5$ and a batch size of 32. We also carry out experiments with different pre-trained contextual embeddings: *BERT-Base*, *BERT-Large*, *RoBERTa-Base*, as well as *scibert-scivocab-cased* (Beltagy et al., 2019) from the PyTorch-Transformers library. During the inference stage, let there be n_t tasks, n_d datasets and n_m metrics in the *smallNLP-KG* test corpus, so we can have total $(n_t \times n_d)$ and $(n_t \times n_m)$ combinations of possible *evaluatedOn* and *evaluatedBy* relations respectively. We apply our trained hybrid NLI model to test all combinations.

For the *term2vec* model to extract related relations (Section 5.3), we use the Skip-gram word2vec implementation from Gensim with a window size of 5, min count of 1, and an embedding dimension of 100. We run the model on all pseudo sentences generated from the whole *TDM-NLP-Papers* corpus. After obtaining term embeddings, we use K-Means clustering algorithm from Scikit-learn to generate clusters among the same type of entities based on nodes from *smallNLP-KG*. We set K equal to number of gold clusters in *smallNLP-KG* for entity type.

7 Results on smallNLP-KG

7.1 Comparison With the Existing Baselines

We compare our *SciNLP-KG* framework against a few baselines from previous work:

1. E2E Rel-Gold (Miwa and Bansal, 2016). The TDM entities present in *TDM-NLP-Papers* are being fed as input to this LSTM-based model, which uses word sequences and dependency tree structures to predict relations. We train the model

Methods	evaluatedOn			evaluatedBy		
	P	R	$F1$	P	R	$F1$
Sent-Level						
-BERT-S	0.77	0.23	0.35	0.68	0.25	0.36
-BERT-L	0.80	0.23	0.36	**0.71**	0.27	0.39
-SciBERT	0.81	0.22	0.35	0.70	0.22	0.33
-RoBERTa	**0.83**	0.24	0.37	0.70	0.27	0.39
Doc-Level						
-BERT-S	0.51	0.80	0.62	0.44	0.68	0.53
-BERT-L	0.52	0.80	0.63	0.45	0.71	0.55
-SciBERT	0.57	0.84	0.68	0.43	**0.73**	0.54
-RoBERTa	0.62	**0.86**	0.73	0.45	0.72	0.55
Hybrid						
-BERT-S	0.60	0.78	0.68	0.50	0.64	0.56
-BERT-L	0.63	0.80	0.70	0.51	0.65	0.57
-SciBERT	0.64	0.81	0.72	0.48	0.67	0.56
-RoBERTa	0.67	0.82	**0.74**	0.59	0.69	**0.64**

Table 2: Ablation study of NLI Models for extracting *evaluatedOn*, *evaluatedBy* relations on *smallNLP-KG*.

Methods	evaluatedOn			evaluatedBy		
	P	R	$F1$	P	R	$F1$
Baselines						
E2E Rel	0.58	0.62	0.60	0.45	0.54	0.49
E2E Rel-Gold	0.63	0.69	0.65	0.51	0.58	0.54
SciIE Rel	0.68	0.78	0.72	0.51	0.62	0.56
DOCTAET	0.62	**0.86**	0.73	0.45	**0.72**	0.55
S-NLI	**0.83**	0.24	0.37	**0.70**	0.27	0.39
D-NLI	0.62	**0.86**	0.73	0.45	**0.72**	0.55
H-NLI	0.67	0.82	**0.74**	0.59	0.69	**0.64**

Table 3: Results of *SciNLP-KG* for extracting *evaluatedOn* and *evaluatedBy* relations and comparison with the best-performing variant of existing baselines.

using our TD/TM training dataset and perform intra-sentence *evaluatedOn/EvaluatedBy* relation extraction using this model.

2. E2E Rel (Miwa and Bansal, 2016). This is another variant of the previous model, which predicts the TD and TM relations in an end-to-end fashion without using the TDM entities tagged in *TDM-NLP-Papers*. We train this model to tag the TDM entities and predict the relations in an end-to-end fashion instead of providing the gold mentions as input to the relation extractor.

3. DocTAET (Hou et al., 2019). This model is analogous to our D-NLI model which works in document-level. We use the NLI model proposed in this paper for predicting TD and TM binary relations individually for comparable experimental evaluation, without considering them as a 3-ary relation between <Task, Dataset, Metric>).

4. E2E Coref (Lee et al., 2017). For coreference resolution, we consider the nouns as we consider only three types of nominal entities. We make use of the end-to-end BiLSTM-CRF based architecture using ELMO embeddings optimized using the conditional likelihood of predicting the correct antecedent given a mention. In our experiments, we used the noun phrase coreference resolution only and discarded the pronoun-based coreference resolution component.

5. SciIE-Rel (Luan et al., 2018). This is a multi-task model to extract coreferent and inter-entity relations from the scientific abstracts. We retrain this

model on our training datasets and use it to predict *evaluatedOn/evaluatedBy* relations based on the predicted Task, Dataset, and Metric mentions from the *NLP-TDMS-Papers* corpus.

6. SciIE-Coref (Luan et al., 2018). This is a component of the above mentioned *SciIE-Rel* system. We use this component to extract the coreferent clusters of the same types of entities from the predicted Task, Dataset and Metric mentions of the *NLP-TDMS-Papers* corpus.

7.2 Results on Inter-Entity Relations

Table 3 shows the results of our hybrid NLI model together with the S-NLI and D-NLI models to extract *evaluatedOn* and *evaluatedBy* relations on the *smallNLP-KG* test set. The ablation analysis using different embeddings is tabulated in Table 2. It seems that the S-NLI model suffers from lower recalls, with 0.24 on the *evaluatedOn* relation and 0.27 on the *evaluatedBy* relation. Exploring the D-NLI context, we observe a drop of 0.21 and 0.25 in terms of precision on *evaluatedOn* and *evaluatedBy* relations respectively, which is mainly attributed to the relatively longer *context* in the NLI model. There is a significant rise in recall for both relations, with improvements of 0.86 and 0.72 on the *evaluatedOn* and *evaluatedBy* relations, respectively. Finally, our H-NLI model combines the strengths of both S-NLI and D-NLI models, with decent improvement in terms of precision compared to the D-NLI architecture, while preserving better recall than S-NLI model.

7.3 Results on *coreferent* Relations

We compare our mention-pair model (Section 5.2) to two coreference resolution systems (*E2E Coref* and *SciIE Coref* in Section 7.1). We also imple-

Methods	coferent			related		
	P	R	$F1$	P	R	$F1$
Baselines						
Rule-based	0.43	0.58	0.49	-	-	-
E2E Coref	0.61	0.67	0.64	-	-	-
SciIE Coref	0.65	0.70	0.69	-	-	-
PMI	-	-	-	0.61	0.61	0.61
SciNLP-KG	**0.77**	**0.79**	**0.77**	**0.67**	**0.68**	**0.67**

Table 4: Results of *SciNLP-KG* for *coferent* and *related* relation extraction compared to baselines.

ment a heuristic baseline that uses the Jaccard similarity of two entity mentions and predicts those with score greater than 0.75 as coferent.

Table 4 (left) lists the results of our mention-pair model for *coferent* relation extraction on the *smallNLP-KG* test set, with an overall Macro F1 of 0.77. Overall, we found that our system outperforms the other three baselines slightly. During qualitative analysis, we observe that our learning-based model eliminates some false positive links proposed by the rule-based approach, such as <*Rouge-1* **coferent** *Rough-2*> and also captures more true positive links such as <*sentiment mining* **coferent** *sentiment analysis*>.

7.4 Results on *related* Relations

We compare our *term2vec* model (Section 5.3) to a baseline model which use *pointwise mutual information* (PMI) (Church and Hanks, 1990) to measure the association score between two TDM entities of the same type.

Table 4 (right) lists the results of our *term2vec* model for extracting *related* relations on *smallNLP-KG*. We found that our *term2vec* model outperforms the PMI-based baseline on *task* and *metric* entities by a good margin. We also observed that there is a slight improvement on the dataset clusters. This happens because the datasets inside the same documents are related most of the time, whereas the same is not always true for tasks and metrics, in which the context window of their mentions plays an important role to capture whether they are *related*. During qualitative analysis, we observe some false positives generated from the PMI-based baseline such as <*f1-score* **related** *rouge*>. Interestingly, we find that our *term2vec* model captures both coferent relations as well as other related relations between the same type of entities, such as *hypernym* relations (e.g, <*parsing* **related** *dependency parsing*>, <*rouge* **related** *rouge-n*>).

Nodes		Relations	
Type	Count	Type	Count
Task	2,441	evaluatedOn	4,137
Dataset	2,314	evaluatedBy	4,019
Metric	619	coferent	2,074
		related	5,532
Total	**5,374**	**Total**	**15,762**

Table 5: Overall statistics of *largeNLP-KG*.

7.5 Comparison with the Existing Models

Our final Hybrid-NLI model performs better than the two recent relation extraction systems (*E2E Rel* and *SciIE Rel*), similarly our cross-document mention-pair coreference resolver outperforms two related baselines. Probing deeper, we observe that existing baselines struggle because: 1) they fail to resolve certain long-range relations due to the end-to-end setup and suffer from error propagation; and 2) they cannot handle inter-document coreference resolution, failing to generate some <Task, Dataset> pairs found when canonicalizing entities in cross-document discourse. Hou et al. (2019)'s method suffers from low precision due to the bottleneck of the D-NLI approach (see Table 3). Our hybrid NLI approach achieves better precision while still keeping a reasonable recall.

It is worth noting that in *smallNLP-KG* we annotated completed relations for every pair of TDM mentions. The annotations already take care of inference-based KG consistency (e.g., T1 *coferent* T2, T1 *evaluatedOn* D1, T2 *evaluatedOn* D1). In general, our framework handles each of the relations separately and achieves good performance compared to the baselines which use joint modelling approaches on the *smallNLP-KG* testing set.

8 Experiments on LargeNLP-KG

We apply our trained *SciNLP-KG* framework to the *TDM-NLP-Papers* corpus (Section 4.1) to build a large-scale KG (*largeNLP-KG*). The resulting KG contains 5,374 TDM entities and 15,762 relations (see Table 5). We use both human evaluation as well as automatic evaluation to assess the quality and coverage of *largeNLP-KG*.

Human Evaluation. We randomly sample 100 instances from the final large-scale KG for *evaluatedOn*, *evaluatedBy*, and *coferent* relations, which were then manually assessed by an NLP expert. Note that these relation instances do not appear in the training datasets of our *SciNLP-KG* sys-

Relations	#	$Prec$	$P@5$	$P@10$	$P@20$
evaluatedOn	100	0.84	-	-	-
evaluatedBy	100	0.77	-	-	-
coreferent	100	0.79	-	-	-
related	600	-	0.77	0.71	0.60

Table 6: Precision of the randomly selected samples from *largeNLP-KG*. # indicates sample size.

Method	evaluatedOn	evaluatedBy
Relaxed Match	0.49	0.58
+coreferent	0.56	0.63

Table 7: Coverage of *largeNLP-KG* compared with *Paperswithcode* on *evaluatedOn/evaluatedBy* relations.

tem. Table 6 reports the precision for each relation type. It is encouraging to see that our large-scale NLP KG obtains reasonably high precision (0.84, 0.77 and 0.79) on the *evaluatedOn*, *evaluatedBy* and *coreferent* relations, respectively. We found that most false positives are from the TDM tagger errors. For instance, *Stanford-CoreNLP* is tagged as a *dataset* entity.

In case of *related* relation clusters, we randomly sample 10 entities from each entity type (i.e., *Task, Dataset, Metric*) and choose the top 20 nearest neighbors of the same type based on the cosine-similarity between entities. An NLP expert assessed the correctness of these 600 relations (200 for each of the T-T, D-D, and M-M related relations) based on their common sense knowledge about NLP. We report $Macro-precision@K$, where $K= 5, 10, 20$ in Table 6 (last row). In general, for a given entity, our unsupervised *term2vec* model provides reasonable suggestions of the related entities.

Automatic Evaluation. In order to better understand the coverage of *largeNLP-KG*, we compared *evaluatedOn* and *evaluatedBy* relations from *largeNLP-KG* with the manually constructed NLP leaderboards in *Paperswithcode*.[5]

The recent version of *Paperswithcode* (Aug-2020) contains leaderboard information for 265 NLP tasks and the corresponding 100 datasets. We consider only the NLP-related TDM entities. Each leaderboard is a tuple of four elements (<*task, dataset, metric, score*>) and it encodes the valid relations between the task and the dataset/metric, which corresponds to our *evaluatedOn* and *evalu-*

---

[5]We do not evaluate our system on SciREX because we use annotations from SciREX to train our D-NLI model.

atedBy relations. In total, we obtain 450 TD pairs and 623 TM pairs from *Paperswithcode*.

We automatically check how many of these pairs are encoded in our large-scale NLP KG. Note that we do not compare TD/TM pairs that appear in the training dataset for our NLI models (Section 5.1). Specifically, we use an edit-distance matching algorithm to match TD/TM pairs between *largeNLP-KG* and *Paperswithcode* (*Relaxed Match*). We consider the edit-distance of our extracted TD and TM pairs with those in *Paperswithcode* and choose those with normalized edit-distance less than 0.3 as positive instances. For instance, the *Paperswithcode* TM pair "*sentiment mining–F1-score*" is equivalent to extracted TM pair "*sentiment analysis–F1-scores*".

Table 7 shows that with *Relaxed Match*, our *largeNLP-KG* contains 49% and 58% of TD and TM pairs from *Paperswithcode*, respectively. We further employ coreferent relations to generate more *evaluatedOn* and *evaluatedBy* relations, which helps to achieve a higher coverage of 56% and 63% on *Paperswithcode* on TD and TM pairs respectively. The mismatch part between our large KG and *Paperswithcode* is mostly due to the fact that *Paperswithcode* contains recent papers while our KG is built on papers from 1974-2019. For instance, our graph contains a task called "textual entailment", which is not included as a task entity in *Paperswithcode*.

9 Conclusions

In this paper, we propose *SciNLP-KG* framework to build a large-scale NLP KG from NLP papers in an end-to-end manner. An interesting direction of further research is the diachronic analysis of TDM entities. For instance, for the two tasks-"textual entailment" and "NLI", it seems that after the emergence of SNLI corpus paper (Bowman et al., 2015), the NLP community switched to use the new name to refer to same task. In practice, the automatically-built KG (*largeNLP-KG*) has the potential to assist the researchers to search related papers and develop comparable experiments. In future, we plan to build a web-based visualization tool to enable researchers to explore KG and related papers.

Acknowledgement

References

Amjad Abu-Jbara, Jefferson Ezra, and Dragomir Radev. 2013. Purpose and polarity of citation: Towards NLP-based bibliometrics. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 596–606, Atlanta, Georgia. Association for Computational Linguistics.

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, Minneapolis, Minnesota. Association for Computational Linguistics.

Waleed Ammar, Dirk Groeneveld, Chandra Bhagavatula, Iz Beltagy, Miles Crawford, Doug Downey, Jason Dunkelberger, Ahmed Elgohary, Sergey Feldman, Vu Ha, Rodney Kinney, Sebastian Kohlmeier, Kyle Lo, Tyler Murray, Hsu-Han Ooi, Matthew Peters, Joanna Power, Sam Skjonsberg, Lucy Wang, Chris Willhelm, Zheng Yuan, Madeleine van Zuylen, and Oren Etzioni. 2018. Construction of the literature graph in semantic scholar. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 84–91. Association for Computational Linguistics.

Waleed Ammar, Matthew Peters, Chandra Bhagavatula, and Russell Power. 2017. The ai2 system at semeval-2017 task 10 (scienceie): semi-supervised end-to-end entity and relation extraction. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 592–596. Association for Computational Linguistics.

Isabelle Augenstein, Mrinal Das, Sebastian Riedel, Lakshmi Vikraman, and Andrew McCallum. 2017. SemEval 2017 task 10: ScienceIE - extracting keyphrases and relations from scientific publications. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 546–555, Vancouver, Canada. Association for Computational Linguistics.

Isabelle Augenstein and Anders Søgaard. 2017. Multi-task learning of keyphrase boundary classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 341–346, Vancouver, Canada. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Kata Gábor, Davide Buscaldi, Anne-Kathrin Schumann, Behrang QasemiZadeh, Haïfa Zargayouna, and Thierry Charnois. 2018. Semeval-2018 task 7: Semantic relation extraction and classification in scientific papers. In *Proceedings of The 12th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT, New Orleans, Louisiana, June 5-6, 2018*, pages 679–688.

Kata Gábor, Davide Buscaldi, Anne-Kathrin Schumann, Behrang QasemiZadeh, Haïfa Zargayouna, and Thierry Charnois. 2018. SemEval-2018 task 7: Semantic relation extraction and classification in scientific papers. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 679–688, New Orleans, Louisiana. Association for Computational Linguistics.

Sonal Gupta and Christopher Manning. 2011. Analyzing the dynamics of research by extracting key aspects of scientific papers. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 1–9, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Kazi Saidul Hasan and Vincent Ng. 2014. Automatic keyphrase extraction: A survey of the state of the art. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1262–1273, Baltimore, Maryland. Association for Computational Linguistics.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The 15th International Conference on Computational Linguistics*.

Yufang Hou, Charles Jochim, Martin Gleize, Francesca Bonin, and Debasis Ganguly. 2019. Identification of tasks, datasets, evaluation metrics, and numeric scores for scientific leaderboards construction. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5203–5213.

Yufang Hou, Charles Jochim, Martin Gleize, Francesca Bonin, and Debasis Ganguly. 2021. TDMSci: A specialized corpus for scientific literature entity tagging of tasks datasets and metrics. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 707–714, Online. Association for Computational Linguistics.

Sarthak Jain, Madeleine van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy. 2020. SciREX: A challenge dataset for document-level information extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7506–7516, Online. Association for Computational Linguistics.

David Jurgens, Srijan Kumar, Raine Hoover, Dan McFarland, and Dan Jurafsky. 2018. Measuring the evolution of a scientific field through citation frames. *Transactions of the Association for Computational Linguistics*, 6:391–406.

Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. 2010. SemEval-2010 task 5 : Automatic keyphrase extraction from scientific articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 21–26, Uppsala, Sweden. Association for Computational Linguistics.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232. Association for Computational Linguistics.

Yi Luan, Mari Ostendorf, and Hannaneh Hajishirzi. 2017. Scientific information extraction with semi-supervised neural tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2641–2651, Copenhagen, Denmark. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositional-

ity. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using LSTMs on sequences and tree structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1105–1116, Berlin, Germany. Association for Computational Linguistics.

Pouya Pezeshkpour, Yifan Tian, and Sameer Singh. 2020. Revisiting evaluation of knowledge base completion models. In *Automated Knowledge Base Construction*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Zhiqing Sun, Shikhar Vashishth, Soumya Sanyal, Partha Talukdar, and Yiming Yang. 2020. A re-evaluation of knowledge graph completion methods. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5516–5522, Online. Association for Computational Linguistics.

Yuka Tateisi, Yo Shidahara, Yusuke Miyao, and Akiko Aizawa. 2014. Annotation of computer science papers for semantic relation extraction. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1423–1429, Reykjavik, Iceland. European Language Resources Association (ELRA).

Simone Teufel, Advaith Siddharthan, and Dan Tidhar. 2006. Automatic classification of citation function. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 103–110, Sydney, Australia. Association for Computational Linguistics.

Chen-Tse Tsai, Gourab Kundu, and Dan Roth. 2013. Concept-based analysis of scientific literature. In *Proceedings of the 22nd ACM international conference on information & knowledge management*, CIKM '13, pages 1733–1738, New York, NY, USA. ACM.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5784–5789, Hong Kong, China. Association for Computational Linguistics.