An Analysis of Relation Extraction within Sentences from Wet Lab Protocols

Xianjun Yang
Department of Computer Science
University of California, Santa Barbara
Santa Barbara, USA
xianjunyang@ucsb.edu

Xinlu Zhang
Department of Computer Science
University of California, Santa Barbara
Santa Barbara, USA
xinluzhang@ucsb.edu

Julia Zuo
Materials Department
University of California, Santa Barbara
Santa Barbara, USA
jlzuo@ucsb.edu

Stephen Wilson
Materials Department
University of California, Santa Barbara
Santa Barbara, USA
stephendwilson@ucsb.edu

Linda Petzold
Department of Computer Science
University of California, Santa Barbara
Santa Barbara, USA
petzold@ucsb.edu

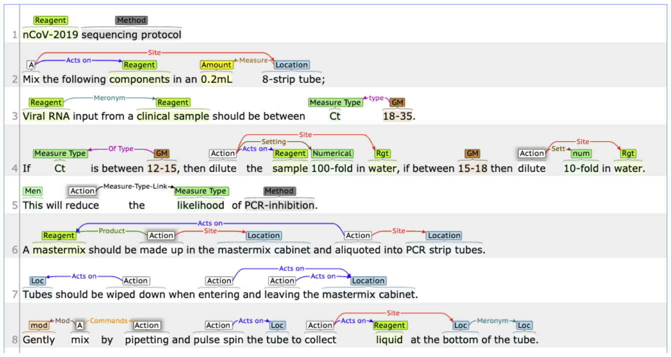
2021 IEEE International Conference on Big Data (Big Data) | 978-1-6654-3902-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/BigData52589.2021.9671781

***Abstract*—Wet lab protocols (WLPs) are sets of instructions written in domain-specific natural language for step-by-step biological experimental processes. There have been efforts to annotate WLPs for shallow semantic parsing to enable reproducible procedures, text mining, and automatic conversion into a machine-readable format. However, current methods have not fully exploited the relation extraction sub-task on the protocol corpus. Neural approaches have the potential to deal with the various noise and in-domain jargon in the texts. To explore the viability of neural methods for this task, we perform a thorough analysis of both graph and nongraph neural approaches. We find that both graph neural networks with generated parameters (GP-GNNs) and Context-Aware models show advantages in relation extraction and are well suited to our goal. Specifically, the GP-GNNs and Context-Aware models demonstrate similar performance on all three WLPs datasets when the full training set is used, both outperforming the previous best results significantly. This can be explained by the observation that considering multiple relations in a sentence enhances the predictive ability. In addition, our extensive experiments demonstrate that the Context-Aware approach in particular can achieve good results even with a limited amount of training data, providing new insights for low-resource scenarios.**

***Index Terms*—Text mining; Relation extraction; Wet lab protocols; Graph neural networks; Context-Aware.**

I. Introduction

An instructional language, known for being repetitive and semi-structured, usually follows a certain form of specialized vocabulary, syntax, and semantic relationships used within a particular domain [1], [2]. Lab instructions describe the natural language of scientific experiments, e.g. wet lab protocols(WLPs) contain instructions for biology laboratory operations [3], such as the example in Fig. 1. These procedures are not only research contributions themselves, but their interpretation also is critical for facilitating the conveyance of reproducible research [4], automatic conversion into a machine-readable format [5], [6], and large-scale data mining from the literature [7], [8].



1 nCoV-2019 sequencing protocol

2 Mix the following components in an 0.2mL 8-strip tube;

3 Viral RNA input from a clinical sample should be between Ct 18-35.

4 If Ct is between 12-15, then dilute the sample 100-fold in water, if between 15-18 then dilute 10-fold in water.

5 This will reduce the likelihood of PCR-inhibition.

6 A mastermix should be made up in the mastermix cabinet and aliquoted into PCR strip tubes.

7 Tubes should be wiped down when entering and leaving the mastermix cabinet.

8 Gently mix by pipetting and pulse spin the tube to collect liquid at the bottom of the tube.

Fig. 1. An annotated wet lab protocol example, taken from WNUT-2020.

Machine learning has been widely used in the design of automatic drug discovery in the biomedical domain [9] and in automation [4], [10], [11]. However, the scarcity of useable data from experiments has been a problem due to both collection and interpretation difficulties [12]. To date, the majority of the knowledge of the detailed procedures for carrying out specific biological experiments has been recorded only in scientific documents such as scholarly papers, electronic lab notebooks, lab protocols, etc. Therefore, recent research has begun to apply Natural Language Processing (NLP) techniques for extracting structured information of instructions or procedures from unstructured texts [3], [13]–[15]. Named entity recognition (NER) and relation extraction (RE) are the fundamental tasks towards successful information extraction from those noisy, domain-specific human-constructed instructional languages. In the past few years, efforts have been made to collect WLPs and label them as action graphs by domain experts for NER and RE tasks, namely the WLP [3], WNUT-2020 [16] and WLP-MSTG [15] corpora.

This work is supported by UC Santa Barbara NSF Quantum Foundry funded via the Q-AMASE-i program under NSF award DMR-1906325.

978-1-6654-3902-2/21/$31.00 ©2021 IEEE

There are several limitations of the currently used methodologies for the RE task on WLPs. First, prior baselines require feature engineering, which depends on human knowledge for constructing the features, rather than an end-to-end approach. For example, [3] utilize a maximum entropy classifier [17], [18] using diverse lexical, syntactic and semantic features derived from the text. Second, the existing methods are usually built on neural approaches with a Bidirectional Encoder Representations from Transformers (BERT) [19] layer for word embedding. At the same time, many other effective RE models have not been exploited for this task. For example, [20] argue that considering other relations in context is beneficial for predicting a target relation for sentence-level relation extraction. Reference [21] propose a novel graph neural network with generated parameters (GP-GNNs), which yields significant improvements for multi-hop reasoning on natural language text.

In this paper, we conduct comprehensive experiments by incorporating the aforementioned GP-GNNs and Context-Aware methods [20], [21] with biomedical domain-specific word embeddings [22], [23] and achieve significant improvements on all WLPs corpora. For this multi-class classification problem, the standard weighted-average Precision (P), Recall (R) and micro-average F1 score are calculated by Scikit-learn [24] for all of the evaluations.

The key contributions of this paper include:

- We survey previously existing techniques for intra-sentence RE and adopt the GP-GNNs [21] and Context-Aware [20] models to analyze RE from WLPs, comparing their corresponding performance with previous results. In practice, we see improvements of 3.68, 1.84, and 20.87 points using the Context-Aware model and 3.59, 2.00, and 20.67 using the GP-GNNs model in terms of F1 score against the prior best results on the WLP, WNUT-2020 and WLP-MSTG datasets, respectively.
- We perform comprehensive experiments to validate the effectiveness of our recommended approaches and provide label breakdown results. Compared with all methods currently in use, we find that both GP-GNNs and Context-Aware methods improve the classification result at almost all label levels significantly, and the improvements by both methods are actually similar when the full training set is used. This improvement can be explained by the fact that both methods consider multiple relationships in a sentence compared to prior analyses that extract relations independently.
- We vary the data size used for training and demonstrate that the Context-Aware approach works much better than GP-GNNs especially when the amount of training data is relatively small, making it quite viable for low-resource scenarios. We argue that this is because the Context-Aware method exhaustively aggregates all entity pairs in the same sentence for target relation prediction, while GP-GNNs only consider entity pairs involved with the multi-hop reasoning process.

II. Related work

Wet Lab Protocols: To the best of our knowledge, [3] contributed the first large enough annotated WLPs with semantic actions and arguments to the research community, namely the WLP corpus. They also provided a baseline approach by assuming the presence of pre-defined gold entities and training a maximum entropy model [17], [18] using five group features derived from the text. The WNUT-2020 corpus [16] was published with further re-annotation on the WLP corpus, adding 20,613 entities and 10,824 relations and simultaneously removing inconsistent annotations. In addition, 100 randomly sampled general protocols and 11 manually selected covid-related protocols from ProtocolIO[1] were collected and annotated as the WNUT-2020 test-20 set. Then [25] designed neural networks with three layers: 1) BERT layer, 2) entity recognition layer and 3) relation recognition Layer, and achieved the best results on both the WLP and WNUT-2020 corpora. Finally, [15] re-annotated the WLP corpus by including the 6 additional global cross-Action Phrase Temporal and Causal (cAP-TaC) relationships within intra- and inter-sentences and proposed a latent structure model for jointly learning entities and relations within and across multiple sentences.

Relation extraction: Relation extraction refers to determining the relation type between two target entities that appear in the same text. Many models have been proposed for the sentential relation extraction task. For example, [26], [27] demonstrated the capability of convolutional neural networks (CNN) to capture the pairwise relations between entities in text. Reference [28] demonstrated the effectiveness of Long Short-Term Memory (LSTM) [29] in extracting the relationships between entities. However, all of the aforementioned models fail to consider various relations in the same sentence and use only the target entities for the relation prediction. To overcome this weakness, [20] use an attention mechanism to take context relations into consideration for predicting the target relation. Moreover, various graph neural networks have also been designed for RE. For example, a relational graph neural network was introduced by [30] for the knowledge base completion task. Reference [31] improved the relation extraction by using graph neural networks to encode dependency trees. Reference [32] extracted relations via a graph-based neural model, in which all possible paths between entities are modeled as a graph. In addition, [33] proposed a general framework for information extraction using dynamic span graphs for jointly learning entities and relations and etc. This multi-task method took predicted entities by this model simultaneously as input for RE, whereas annotated gold entities are given as inputs for our task. Finally, [21] proposed a novel graph neural network with generated parameters (GP-GNNs) and showed that it yielded significant improvements for multi-hop reasoning on text. In contrast to the framework in [33], GP-GNNs make predictions on pre-annotated entities, which is consistent with our scenario. Overall, the Context-Aware method serves as the best nongraph approach while

[1] https://www.protocols.io/

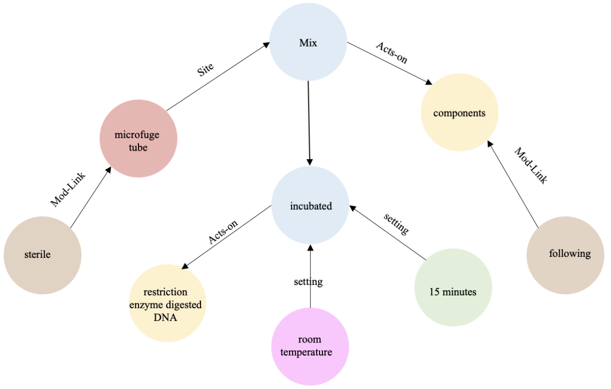


Fig. 2. Annotated example: An action graph derived from the annotations taken from WLP.

the GP-GNNs model is the preferred graph approach for our task. Thus in this paper, we use Context-Aware and GP-GNNs approaches to model relations among WLPs.

III. Methods

In this section we define the WLPs relation extraction task and introduce the `GP-GNNs` and `Context-Aware` frameworks.

A. Task Definition

Given a sequence of text $x = [x_1, x_2, \cdots, x_l]$ with m chemically-related entities $[e_1, e_2, \cdots, e_m]$, where e_i consists of one or a sequence of tokens, and n relation types $[r_1, r_2 ... r_n]$. The aim of this task is to predict the relation type r_{e_i,e_j} of each entity pair (e_i, e_j).

B. GP-GNNs

Problem Formulation A fully connected action graph $G = (\mathcal{V}, \mathcal{E})$ is built for relation construction among labeled entities from III-A. $\mathcal{V}$ denotes the set of entities, and $\mathcal{E}$ denotes the set of edges, such that $(e_i, e_j) \in \mathcal{E}$, $e_i, e_j \in \mathcal{V}$, describes the relationship between each two entities.

We follow the methodology as proposed by the original authors [21] to construct the model, which is composed of three modules 1) encoding module, 2) propagation module, and 3) classification module.

Encoding Module The encoding module is a function that maps sequences to the corresponding edges of transition matrices.

$$E(x_t^{i,j}) = [\mathbf{x}_t; \mathbf{p}_t^{i,j}], \tag{1}$$

$$\mathcal{A}_{i,j}^{(n)} = f(E(x_0^{i,j}), E(x_1^{i,j}), \dots, E(x_{l-1}^{i,j}); \theta_e^n), \tag{2}$$

where $E(\cdot)$ denotes the embedding function, a concatenation of the word embedding $\mathbf{x}_t$ and the relative position embedding $\mathbf{p}_t^{i,j}$ of the entity pair i and j relative to a word x_t, and $f(\cdot)$ denotes the encoder function, to encode the sequence data and output transition matrices $\mathcal{A}$. n denotes the index of layer.

Propagation Module The propagation module creates representations of nodes with multiple layers. The representation of the next layer is calculated given the previous one by

$$\mathbf{h}_i^{(n+1)} = \sum_{e_j \in \mathcal{N}(e_i)} \sigma(\mathcal{A}_{i,j}^{(n)} \mathbf{h}_j^{(n)}), \tag{3}$$

where $\mathcal{N}(e_i)$ corresponds to all of the neighbours of a node e_i in a graph $\mathcal{G}$, and σ is a nonlinear function.

Classification Module The classification module is utilized for relation prediction by feeding into node representations of the target entity pair (e_i, e_j),

$$\mathbf{r}_{e_i,e_j} = [[\mathbf{h}_{e_i}^{(1)} \odot \mathbf{h}_{e_j}^{(1)}]^\top; [\mathbf{h}_{e_i}^{(2)} \odot \mathbf{h}_{e_j}^{(2)}]^\top; \dots; [\mathbf{h}_{e_i}^{(K)} \odot \mathbf{h}_{e_j}^{(K)}]^\top], \tag{4}$$

where $\odot$ is element-wise multiplication, K denotes the number of layers. It outputs a probability distribution of relations given by

$$\mathbb{P}(r_{e_i,e_j} \mid h, t, s) = softmax(MLP(\mathbf{r}_{e_i,e_j})). \tag{5}$$

The GP-GNNs are trained with a cross entropy loss after K propagation layers in the propagation module,

$$\mathcal{L} = \sum_{s \in S} \sum_{i \neq j} log \mathbb{P}(r_{e_i,e_j} \mid i, j, s). \tag{6}$$

C. Context-Aware RE

Problem Formulation Given the sequence $\mathbf{x}$ and entities in III-A, the entity marker $\mathbf{e} = [...e_1...e_2...]$ represents the token in the sentence as either belonging to the first entity e1, the second entity e2 or neither of those.

We summarize the approach [20] as three modules 1) encoder module, 2) attention module and 3) classification module.

Relation Encoder Module The relation encoder is built by LSTM and the encoder generates a vector representation $\mathbf{o}_s$ of a relation between two entities in a sentence,

$$\mathbf{o}_s = LSTM(\mathbf{x}; \mathbf{e}; \mathbf{W}; \mathbf{M}), \tag{7}$$

where the $\mathbf{W}$ and $\mathbf{M}$ are the word embedding and entity marker embedding, respectively.

Context Attention Module An attention layer is used to consider the effect of other context relations in the same sentence for predicting the target relation

$$\mathbf{o}_c = \sum_{i=0}^{m} a_i \mathbf{o}_i, \tag{8}$$

$$a_i = \frac{exp(g(\mathbf{o}_i, \mathbf{o}_s))}{\sum_{j=0}^{m} exp(g(\mathbf{o}_j, \mathbf{o}_s))}, \tag{9}$$

where $\mathbf{o}_i$ is generated by the encoder in Eq. 7, m is the number of all possible pairs of entities and g_i computes an attention score for a context relation with respect to the target relation: $g(\mathbf{o}_i, \mathbf{o}_s) = \mathbf{o}_i \mathbf{Q} \mathbf{o}_s$, and the weight matrix $\mathbf{Q}$ is learned during training.

TABLE I
FEATURES OF THE WLP, WNUT-2020, AND WLP-MSTG ORIGINAL DATASETS.

Item	WLP	WNUT-2020	WLP-MSTG
Synthesis procedures	622	725	615
Sentences	13679	17658	14172
Avg. sentence length	12.99	13.45	15.63
Avg. sentences/Doc	21.99	24.36	23.04
Entities	60721	103387	64937
Entity types	18	18	18
Relations	42425	69139	70824
Relation types	13	14	16
Tokens	177770	237547	221531

TABLE II
HYPERPARAMETERS

Hyperparameters	Value
Learning rate	0.0008
Batch size	15
Dropout rate	0.1
Weight decay	0
Epochs	50
Grad_clip	0.15
Nonlinear activation σ	ReLU
Optimizer	Adam
Layer 1 embedding size	8
Layer 2 embedding size	12

Classification Module The $Softmax$ function outputs the probability by concatenating $\mathbf{o}_s$ and $\mathbf{o}_c$

$$\mathbb{P}(r_{e_i,e_j} \mid \mathbf{o}_s, \mathbf{o}_c) = softmax([\mathbf{o}_s, \mathbf{o}_c]). \qquad (10)$$

and a cross entropy loss is used for model training.

IV. EXPERIMENTS

We conduct experiments on three WLPs datasets introduced below, to evaluate model performance on relation extraction given human-annotated gold entities.

A. Datasets

WLP: The WLP[2] corpus presents entity, relation, and event annotations from 622 wet lab protocols, consisting of natural language instructions for carrying out chemistry or biology experiments. 9 objected-based, 5 measure-based and 4 other types are created as entity tags. 7 action relations (e.g. Acts-on, Creates, and Site, etc.) and 6 binary relations (e.g. Coreference, Measure, and Mod Link, etc.) are used to describe the relationships among annotated entities.

WNUT-2020: WNUT-2020[3] is the updated dataset for entity and relation recognition over wet lab protocols based on the previous WLP corpus. Through this re-annotation, the previously missing 20,613 entities along with the 10,824 relations were added to the WNUT-2020 corpus, and the inconsistent annotations were also removed from the previous dataset. An additional test set (Test-20), consisting of 100 randomly sampled general protocols and 11 manually selected Covid-related protocols from ProtocolIO, was also added.

WLP-MSTG: WLP-MSTG[4] is the latest dataset for entity and relation recognition over wet lab protocols derived from the WLP Corpus. The WLP-MSTG dataset focuses not only on intra-sentence relations but also on inter-relations from multiple sentences. Here the Inter-Action Phrase (iAP) relations refer to 13 local intra-sentence semantic relations. Besides, 6 additional global cross-Action Phrase Temporal and Causal ($cAP - TaC$) relationships were added to this corpus as described in [15]. Additionally, the annotators exclude entities and relations annotated for spurious descriptive sentences that do not prescribe any actions, to ensure that the graph is fully connected [15].

The statistics and more details of the three WLPs corpora can be seen from Table I.

B. Data Preprocessing

Each plain text document and its corresponding standoff annotation file are first tokenized by the standoff2conll tool provided by [16]. After the tokenization, each sentence is mapped with the corresponding vertex set (named entity type, position and token information), and the edge set (relations type, left and right position information). The vast majority of sentences contain fifteen to thirty words, but the maximum length can be up to 512 in some extreme cases. To save computing time and space, we manually truncated the maximum sentence length to 124 words. Then more than 99% of the original relations are kept after discarding the relations in the extremely long cases. Since here we only focus on intra-sentence relations, cross-sentence relations are not included in our experiments. We follow the same train-development-test split as previous works [13], [15], [25].

C. Word Representation

There are two 200-dimensional biological Word2Vec embeddings: BioWordVec [22] and PubMed-w2v [23], trained on biological paper abstracts, full text and clinical notes. In practice we observe slightly better prediction accuracy using BioWordVec than PubMed-w2v, thus we report only our results based on BioWordVec.

D. Hyperparameters

We select the best combination of hyperparameters from the development set by random search. The Adam optimizer [34] is used for all models. Other parameters are selected within a range of values, e.g. the learning rates are selected in {0.005, 0.001, 0.0008, 0.0005, 0.0001, 0.00005}, and the dropout rate is selected in {0.15, 0.25, 0.35, 0.5}. Tab. II shows the best hyperparameter settings, which are used across most of the experiments. The embedding sizes are set for GP-GNNs only, while other parameters also appear in the Context-Aware model. The models are implemented in PyTorch[5], and

[2] https://github.com/chaitanya2334/WLP-Dataset

[3] https://github.com/jeniyat/WNUT_2020_NER

[4] https://github.com/chaitanya2334/wlp-mstg-dataset

[5] https://pytorch.org

TABLE III
OVERALL EVALUATION RESULTS.

Datasets	WLP			WNUT-2020			WLP-MSTG		
Models	P	R	F1	P	R	F1	P	R	F1(%)
Kulkarni 2018 [3]	80.98	77.04	78.96	-[a]	-	-	-	-	-
Swarup 2020 [13]	82.29	81.02	81.65	-	-	-	-	-	-
Mgshohra 2020 [25](Ensemble)	88.75	84.86	86.75	80.86	80.07	80.46	-	-	-
Kulkarni 2021 [15]	-	-	-	80.40	79.30	79.90	67.90	68.20	68.00[b]
GP-GNNs	90.18	90.34	90.34	83.27	**82.46**	**82.46**	88.43	88.87	88.87
Context-Aware	**90.41**	**90.43**	**90.43**	**83.71**	82.30	82.30	**89.70**	**89.69**	**89.69**
Human agreement score[c]	-	-	66.25	-	-	75.00	-	-	78.23

[a] '-' denotes not reported in the corresponding methods, and the same hereinafter.
[b] denotes iAP result on multi-task predictions.
[c] this score is not necessarily F1 score in the original paper.

an NVIDIA GeForce RTX 2080 SUPER 16 GB GPU is used for all training, development, and test sets. We implement the GP-GNNs and Context-Aware models based on the authors' public repository[6]. In practice, three random seeds are chosen for all experiments and we report the results based on the median performance.

V. RESULTS AND ANALYSIS

In this section, we report the main results by carrying out a set of experiments to demonstrate the model performance. In addition, we compare our results with the previous baselines and previous SOTA results. We use three wet lab protocols, namely WLP [3], WNUT-2020 [3] and WLP-MSTG [15], introduced in Section V-A, to evaluate the model performance.

A. Baselines

Here, we summarize the models used in prior results.

Kulkarni 2018 [3]: This baseline approach, proposed in prior work on the WLP corpus, uses a maximum entropy model, with five types of features: 1) Word features, 2) Entity type features, 3) Overlapping features, 4) Chunk features and 5) Dependency features. The combinations of these features are tested to validate the efficiency of this approach. The best performance is achieved when using all features.

Swarup 2020 [13]: This is an instance-based edge-factored approach, based on the observation that specific scientific fields usually share quite formulaic writing. Thus the various types of relations can be expressed in only a small amount of grammatically similar labeled text. The strategy is to infer the relations of the test sentence by copying relations from similar sentences in the training set obtained from nearest neighbors. As for non-similar sentences, the authors adopt a global set of parameters for prediction by a feed-forward network.

WNUT-2020 [16]: In the WNUT-2020 shared task-1, the baseline relation extraction system employed a feature-based logistic regression model. The best performing team [25] utilizes PubMedBERT [35] for token embedding to obtain the entity pair representation and then classify the relation type with a softmax function. This neural exhaustive approach achieves the SOTA results on both WLP and WNUT-2020 corpora.

WLP-MSTG [15]: This work designs a latent structure model for jointly learning entities and relations within and across multiple sentences. The latent structure model comprises four parts: 1) span representation, 2) transcoder block, 3) scoring functions and 4) latent structures. The authors evaluate the intra- and inter-sentence relation extraction task on their model. Here we focus only on the former one. And we only compare our results with their predictions based on multi-task learning since no predictions on gold entities were reported in their original paper.

B. Overall results

Our overall results achieved the highest F1 score for the relation extraction task on all three datasets, outperforming the previous best results by 21.69 points on the WLP-MSTG dataset. Compared with prior best results on the WLP and WNUT-2020 corpora, the single Context-Aware and GP-GNNs model yields an improvement of 3.68 and 2.00 points against prior best ensemble methods, respectively. These results show that GP-GNNs and Context-Aware models have significant advantages over previous methods on relation extraction from WLPs. Table III presents a comparison.

TABLE IV
PER LABEL F1 PERFORMANCE ON THE WLP TEST SET.

WLP	Percent(%)	Swarup [13]	Context-Aware
$Acts-on$	29.95	86.51	**91.98**
$Measure\_Type\_Link$	0.45	-	64.86
Or	1.21	-	75.62
$Using$	8.77	72.34	**81.61**
$Setting$	16.08	-	96.73
$Measure$	16.80	-	95.40
Of_Type	0.18	-	64.29
$Meronym$	3.64	53.66	**88.26**
$Site$	11.30	-	84.59
$Mod-Link$	9.37	88.72	**92.67**
$Creates$	1.10	23.44	**47.95**
$Coreference-Link$	0.24	-	72.73
$Count$	0.91	82.76	**89.03**
$Overall$	100	81.65	**90.43**

[6] https://github.com/thunlp/gp-gnn

TABLE V
PER LABEL F1 PERFORMANCE ON THE WNUT-2020 TEST-18 SET.

WNUT-2020 test-18	Percent(%)	Mgsohrab [25]	GP-GNNs
$Coreference\_Link$	0.70	55.71	**58.54**
$Measure$	16.03	91.93	**95.74**
$Site$	10.39	84.61	**85.76**
$Meronym$	3.79	69.74	**84.62**
$Measure\_Type\_Link$	1.13	**88.26**	85.60
$Product$	0.89	27.34	**37.78**
$Commands$	0.19	05.88	**29.79**
Mod_Link	13.59	92.18	**92.40**
$Count$	0.90	85.57	**90.05**
$Acts\_On$	28.59	90.63	**92.28**
$Using$	7.89	76.26	**77.16**
$Setting$	14.20	91.80	**96.50**
Of_Type	0.18	**63.16**	51.06
Or	1.53	65.66	**75.15**
$Overall$	100	87.53	**90.01**

C. Label breakdown

We provide the label breakdown results below for a comprehensive understanding of how our recommended models perform across the relation types on different datasets between our results and prior results.

WLP Tab. IV shows our results on the WLP test set. Since the per label result is not reported in other prior methods except for [13], we compare our result only with that of [13], and demonstrate noticeable improvements on all reported label levels.

Here the single Context-Aware model outperforms the instance level approach [13] by 8.78 points. Compared with the previous best ensemble result [25], our result also achieves an improvement of 3.68. We note that the labels in the WLP-2018 dataset are distributed unevenly. For example, the percentage of $Acts-on$ is much larger than the percentage of other labels, while the percentage of $Measure\_Type\_Link$, Of_Type, $Coreference\_Link$, $Count$ is all less than 1%. We achieve the best result on label $Setting$ and $Measure$, with 96.73 and 95.40, respectively. Finally, even given only a few annotated examples for $Creates$(1.10%), our recommended approach achieves double the F1 score compared with [13], again showing the strong ability of relational prediction.

TABLE VI
PER LABEL F1 PERFORMANCE ON THE WNUT-2020 TEST-20 SET.

WNUT-2020 test-20	Percent(%)	P(%)	R(%)	F1(%)
$Coreference\_Link$	1.75	43.16	61.92	50.87
$Measure$	14.18	92.51	95.84	94.14
$Site$	10.07	88.04	71.09	78.66
$Meronym$	3.53	72.66	79.55	79.95
$Measure\_Type\_Link$	1.75	86.45	70.88	77.89
$Product$	0.24	02.04	02.78	02.35
$Commands$	0.08	11.54	25.00	15.79
Mod_Link	21.21	87.69	89.10	88.39
$Count$	1.18	70.29	69.89	70.09
$Acts\_On$	25.61	85.45	77.57	81.32
$Using$	7.11	73.55	68.08	70.71
$Setting$	11.21	76.24	95.25	84.69
Of_Type	1.19	75.00	67.80	71.22
Or	0.89	51.61	72.18	60.19
Overall	100	**83.27**	**82.46**	**82.46**
Mgsohrab [25](Ensemble)	100	80.86	80.07	80.46
Kulkarni [15](Single)	100	80.40	79.30	79.90

WNUT-2020 Tab. V and Table. VI demonstrate our label breakdown results compared with previous best results [25] on the WNUT-2020 test-18 and test-20 set, respectively. It is important to point out that we merged the original training and development sets as the training set, and treated the original test-18 as the development set, leaving test-20 as our test set to be consistent with the previous SOTA results [25]. Since the test set is composed of general protocols and Covid-related protocols [16], which are different from wet lab protocols, the prediction is more challenging. Here the GP-GNNs model achieves the top performance among all results. Typically, compared with the previous SOTA result [25] on the test-18 set, our reported F1 score demonstrates significant improvement at nearly all label levels, as shown in Tab. V. Furthermore, we also provide the label breakdown result at Tab. VI, and the previous SOTA result did not provide label breakdown result. We leave the explorations of ensemble prediction from different models for future work.

WLP-MSTG We present the Context-Aware model results on the WLP-MSTG test set in Tab. VII. Considering our focus on intra-sentence RE only, we use all of the intra-sentence relations derived from the annotated WLP-MSTG dataset. The original report [15] gives the intra-sentence RE results as iAP and $cAP - TaC$ relations separately and we compare our overall result with their highest iAP result, yielding a 21.69 points improvement in terms of F1 score. It is notable to point that the original report [15] was based on multi-task predictions, which means their relation predictions were based on predicted entities rather than gold entities. However, we can only compare in this way since they did not report predictions on gold entities and their codes have not been released either. But they provided predictions on gold entities on WNUT-2020 test-20 set as can be seen in the last row of Tab. VI, achieving lower result than our used methods.

We have further provided the label breakdown results, showing that the Context-Aware model can accurately deal with most label types. Particularly, the Context-Aware model achieves a score of 84.38, 37.31 compared with the previous best score [15] of 62.3, 29.1 on the $Enables$ and $Overlaps$ label. We have also shown that the Context-Aware model is capable of inferring labels that seldom appear in the dataset, such as $Count$ (0.93%, 93.28 points) and $Measure-Type-Link$ (0.59%, 71.26 points). However, the prediction accuracy of label $Overlaps$ is still very low. We suspect that this is due to 1) the few labeled cases, 2) the high level of ambiguity when annotating this label. We argue that this conclusion is also important for guiding the annotators to put more effort into this kind of label when building a new corpus in the future.

D. Limited training size scenarios

To investigate the effectiveness of our recommended approaches when the data size is limited, for all three WLPs cor-

TABLE VII
PER LABEL PERFORMANCE ON THE WLP-MSTG TEST SET.

WLP-MSTG	Percent(%)	P(%)	R(%)	F1(%)
Coreference − Link	0.48	79.49	45.59	57.94
Measure	13.70	96.74	95.13	95.93
Site	10.44	86.58	85.82	86.20
Meronym	4.24	87.38	90.00	88.67
Measure − Type − Link	0.59	65.96	77.50	71.26
Product	3.55	76.00	75.85	75.92
Mod − Link	9.92	88.64	95.06	91.74
Count	0.93	97.52	89.39	93.28
Acts − on	28.05	91.18	92.45	91.81
Using	6.36	76.12	77.74	76.92
Setting	14.98	96.70	97.02	96.86
Of − Type	0.24	88.00	64.71	74.58
Or	1.70	85.50	72.46	78.44
Prevents	0.02	06.25	33.33	10.53
Overlaps	1.03	76.60	24.66	37.31
Enables	3.77	84.14	84.62	84.38
Overall	100	**89.70**	**89.69**	**89.69**
Kulkarni [15] (iAP)	100	67.9	68.2	68.0
Kulkarni [15] (cAP-TaC)	100	52.6	30.8	56.9

TABLE VIII
LIMITED TRAINING SIZE SCENARIOS

	Data(%)	3	6	9	12	15
WLP	GP[a]	54.87	73.11	75.27	79.38	80.49
	CA[b]	**61.77**	**75.19**	**78.20**	**80.11**	**82.95**
WNUT-2020	GP	47.84	63.17	67.52	71.45	76.28
	CA	**53.92**	**66.70**	**70.50**	**72.21**	**76.79**
WLP-MSTG	GP	47.00	61.20	69.55	75.16	77.40
	CA	**57.54**	**68.36**	**70.23**	**77.11**	**78.04**

[a] GP denotes GP-GNNs model.
[b] CA denotes Context-Aware model.

pora, we changed the amount of training data and performed relation prediction on the full test sets. As shown in Table VIII, both of the approaches achieve a competitive F1 score while using just 15% of the original training data, which once again demonstrates the strong relation classification ability of the GP-GNNs and Context-Aware models.

When changing the percentage of data used for model training from 3% to 15%, the Context-Aware model always performs much better than the GP-GNNs model. We suspect that this is because GP-GNNs consider only relations that participate in the multi-hop reasoning process, while the Context-Aware approach exhaustively utilizes all relations in the sentence for target relation prediction. As a result, the Context-Aware approach can better use other relations in low context scenarios, thus achieving higher performance.

Based on our study, we recommend using the Context-Aware model for scenarios where training data is limited. This is particularly important for relation classification in low-resource scenarios, where the training data are expensive and time-consuming to collect. For example, there are few annotated examples of materials synthesis procedures in the materials science domain, and the size of one such corpus [36] is only around one-third of the WLP corpus. Thus we expect that our finding might also help for low-resource situations in other domains. We leave this for future work.

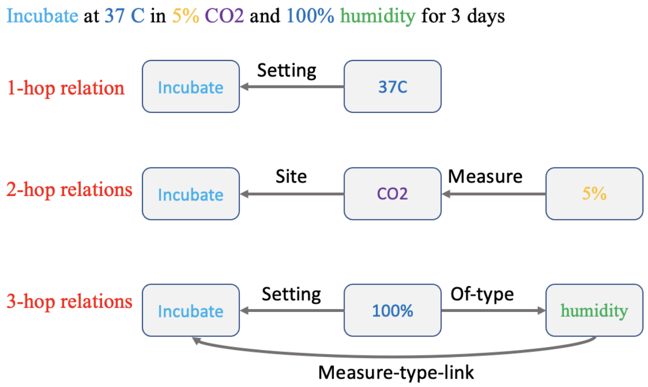


Fig. 3. Annotated examples of 1-hop, 2-hop, and 3-hop relations. Example is taken from the WLP-2020 dataset

E. Model Mechanism

As discussed in Section I, the main drawback of existing approaches is the failure of inferring the relations by utilizing other relations within the same sentence. For example, the previous best method [25] simply calculates the pair representation by picking two entity representations and then classifies it to predict the corresponding relation type. However, other information among multiple entity pairs and their relationships within a sentence are ignored.

Theoretically, a K-layer GP-GNNs has the ability to infer K-hop relations [21]. In Fig. 3 we give a demonstration of multi-hop relations with k = 1, 2, 3 and Tab. IX also presents the relation type distribution in the original train and test sets. The 1-hop and 2-hop relations dominate the relation types across all three corpora, while $K \geq 3$ is ignored as its percentage is less than 0.5%. This unbalanced distribution phenomena is also observed in [37] that relations beyond 3-hops are sparse and uninteresting. Therefore we choose the 2-layer GP-GNNs model in all the experiments, which can best utilize the 2-hop reasoning ability. On the other hand, the Context-Aware approach models the co-occurrence of various relations to enhance the relation prediction ability. The main difference is that the Context-Aware method considers each possible pair of entities for the target relation prediction, while the GP-GNNs model only uses the context relations that participate in the multi-hop reasoning process and then propagates information from layer to layer. Generally, the Context-Aware method demonstrates similar relational reasoning ability compared with the 2-layer GP-GNNs on all three WLPs corpora when trained on the full training set, which is consistent with findings in [21]. However, when trained on limited data, the Context-Aware approach is proven to perform better due to incorporating more context relations as discussed in section V-D.

F. Dataset Filtering

In the wet lab protocols, we find that many same sentences are labeled with inconsistent annotations. To prevent machine

TABLE IX
PERCENTAGE OF K-HOP RELATIONS (K=1, 2) IN THE ORIGINAL DATASETS.

K-hop relation	WLP	WNUT-2020	WLP-MSTG
Train set, K=1	72%	74%	53%
Train set, K=2	28%	26%	47%
Test set, K=1	67%	70%	52%
Test set, K=2	33%	30%	48%

TABLE X
DATASET FILTERING

Dataset	WLP	WNUT-2020	WLP-MSTG
Filtered	89.76	82.22	88.09
Original	**90.43**	**82.46**	**89.69**

learning confusion and data leakage, for duplicate sentences we randomly keep one of them left in the training set and eliminate the others. This process reduces the training set size by around 25.88%, 26.51% and 21.27% on the WLP, WNUT-2020 and WLP-MSTG corpus, respectively. Then the filtered training sets are used for model training, while the test sets remain the same in order to compare with previous results. As a result, in contrast to the previous findings [25] that the performance is slightly improved by the filtering process, here we observe a subtly drop over results trained on the full training set, as evaluated by the F1 score and shown in Table X. Possibly, the duplicate sentences in the test set with inconsistent labels may have hampered the reasoning ability of the neural models.

VI. CONCLUSION

We survey previously proposed relation extraction approaches, typically GP-GNNs and Context-Aware models, and adopt them into relation classification from the wet lab protocols. The Context-Aware method leverages all possible context relations to achieve stronger prediction ability, while the GP-GNNs approach propagates information from layer to layer to incorporate all participated relations in the reasoning process for improving performance. We find both two models exhibit similar performance on the full WLPs corpus, all outperforming the previous results significantly. This conclusion is also validated by a huge improvement on the label breakdown result at almost all label levels. Besides, the Context-Aware approach also exhibits powerful RE ability even when the training data size is much small, which is potentially meaningful for low-resource scenarios. Finally, in contrast to previous findings, we do not observe any improvement after filtering duplicate sentences in the train set likely due to reduced generalization ability.

However, this work focuses only on the RE task given annotated entities, while the entities are not prior knowledge in most cases. We leave the exploration of designing a joint model for predicting named entities and relations together for future work.

ACKNOWLEDGMENT

We gratefully acknowledge support from the UC Santa Barbara NSF Quantum Foundry funded via the Q-AMASE-i program under NSF award DMR-1906325. Any opinion or conclusions expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] R. Grishman, "Adaptive information extraction and sublanguage analysis," in *Proc. of IJCAI 2001*, 2001, pp. 1–4.

[2] R. Grishman and R. Kittredge, *Analyzing language in restricted domains: sublanguage description and processing*. Psychology Press, 2014.

[3] C. Kulkarni, W. Xu, A. Ritter, and R. Machiraju, "An annotated corpus for machine reading of instructions in wet lab protocols," *arXiv preprint arXiv:1805.00195*, 2018.

[4] R. D. King, J. Rowland, S. G. Oliver, M. Young, W. Aubrey, E. Byrne, M. Liakata, M. Markham, P. Pir, L. N. Soldatova *et al.*, "The automation of science," *Science*, vol. 324, no. 5923, pp. 85–89, 2009.

[5] L. N. Soldatova, D. Nadis, R. D. King, P. S. Basu, E. Haddi, V. Baumlé, N. J. Saunders, W. Marwan, and B. B. Rudkin, "Exact2: the semantics of biomedical protocols," *BMC bioinformatics*, vol. 15, no. 14, pp. 1–11, 2014.

[6] V. Vasilev, C. Liu, T. Haddock, S. Bhatia, A. Adler, F. Yaman, J. Beal, J. Babb, R. Weiss, D. Densmore *et al.*, "A software stack for specification and robotic execution of protocols for synthetic biological engineering," in *3rd international workshop on bio-design automation*, 2011.

[7] H. Shatkay and R. Feldman, "Mining the biomedical literature in the genomic era: an overview," *Journal of computational biology*, vol. 10, no. 6, pp. 821–855, 2003.

[8] A. P. Tafti, J. Badger, E. LaRose, E. Shirzadi, A. Mahnke, J. Mayer, Z. Ye, D. Page, and P. Peissig, "Adverse drug event discovery using biomedical literature: a big data neural network adventure," *JMIR medical informatics*, vol. 5, no. 4, p. e9170, 2017.

[9] G. Cano, J. Garcia-Rodriguez, A. Garcia-Garcia, H. Perez-Sanchez, J. A. Benediktsson, A. Thapa, and A. Barr, "Automatic selection of molecular descriptors using random forest: Application to drug discovery," *Expert Systems with Applications*, vol. 72, pp. 151–159, 2017.

[10] M. Bollini, S. Tellex, T. Thompson, N. Roy, and D. Rus, "Interpreting and executing recipes with a cooking robot," in *Experimental Robotics*. Springer, 2013, pp. 481–495.

[11] M. Bates, A. J. Berliner, J. Lachoff, P. R. Jaschke, and E. S. Groban, "Wet lab accelerator: a web-based application democratizing laboratory automation for synthetic biology," *ACS synthetic biology*, vol. 6, no. 1, pp. 167–171, 2017.

[12] E. Kim, K. Huang, A. Saunders, A. McCallum, G. Ceder, and E. Olivetti, "Materials synthesis insights from scientific literature via text extraction and machine learning," *Chemistry of Materials*, vol. 29, no. 21, pp. 9436–9444, 2017.

[13] D. Swarup, A. Bajaj, S. Mysore, T. O'Gorman, R. Das, and A. McCallum, "An instance level approach for shallow semantic parsing in scientific procedural text," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, 2020, pp. 3010–3017.

[14] A. Friedrich, H. Adel, F. Tomazic, J. Hingerl, R. Benteau, A. Marusczyk, and L. Lange, "The sofc-exp corpus and neural approaches to information extraction in the materials science domain," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 1255–1268.

[15] C. Kulkarni, J. Chan, E. Fosler-Lussier, and R. Machiraju, "Learning latent structures for cross action phrase relations in wet lab protocols," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 2021, pp. 6737–6750.

[16] J. Tabassum, S. Lee, W. Xu, and A. Ritter, "Wnut-2020 task 1 overview: Extracting entities and relations from wet lab protocols," *arXiv preprint arXiv:2010.14576*, 2020.

[17] A. E. Borthwick, *A maximum entropy approach to named entity recognition*. New York University, 1999.

[18] N. Kambhatla, “Combining lexical, syntactic, and semantic features with maximum entropy models for information extraction,” in *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, 2004, pp. 178–181.

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.

[20] D. Sorokin and I. Gurevych, “Context-aware representations for knowledge base relation extraction,” in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 1784–1789.

[21] H. Zhu, Y. Lin, Z. Liu, J. Fu, T.-S. Chua, and M. Sun, “Graph neural networks with generated parameters for relation extraction,” *arXiv preprint arXiv:1902.00756*, 2019.

[22] Y. Zhang, Q. Chen, Z. Yang, H. Lin, and Z. Lu, “Biowordvec, improving biomedical word embeddings with subword information and mesh,” *Scientific data*, vol. 6, no. 1, pp. 1–9, 2019.

[23] S. Moen and T. S. S. Ananiadou, “Distributional semantics resources for biomedical text processing,” *Proceedings of LBM*, pp. 39–44, 2013.

[24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, “Scikit-learn: Machine learning in python,” *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[25] M. G. Sohrab, A.-K. D. Nguyen, M. Miwa, and H. Takamura, “mgsohrab at wnut 2020 shared task-1: Neural exhaustive approach for entity and relation recognition over wet lab protocols,” in *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, 2020, pp. 290–298.

[26] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, “Relation classification via convolutional deep neural network,” in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 2014, pp. 2335–2344.

[27] D. Zeng, K. Liu, Y. Chen, and J. Zhao, “Distant supervision for relation extraction via piecewise convolutional neural networks,” in *Proceedings of the 2015 conference on empirical methods in natural language processing*, 2015, pp. 1753–1762.

[28] T. Muneeb, S. Sahu, and A. Anand, “Evaluating distributed word representations for capturing semantics of biomedical concepts,” in *Proceedings of BioNLP 15*, 2015, pp. 158–163.

[29] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[30] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, “Modeling relational data with graph convolutional networks,” in *European semantic web conference*. Springer, 2018, pp. 593–607.

[31] Y. Zhang, P. Qi, and C. D. Manning, “Graph convolution over pruned dependency trees improves relation extraction,” *arXiv preprint arXiv:1809.10185*, 2018.

[32] S. K. Sahu, F. Christopoulou, M. Miwa, and S. Ananiadou, “Inter-sentence relation extraction with document-level graph convolutional neural network,” *arXiv preprint arXiv:1906.04684*, 2019.

[33] Y. Luan, D. Wadden, L. He, A. Shah, M. Ostendorf, and H. Hajishirzi, “A general framework for information extraction using dynamic span graphs,” *arXiv preprint arXiv:1904.03296*, 2019.

[34] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[35] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, and H. Poon, “Domain-specific language model pretraining for biomedical natural language processing,” 2020.

[36] S. Mysore, Z. Jensen, E. Kim, K. Huang, H.-S. Chang, E. Strubell, J. Flanigan, A. McCallum, and E. Olivetti, “The materials science procedural text corpus: Annotating materials synthesis procedures with shallow semantic structures,” *arXiv preprint arXiv:1905.06939*, 2019.

[37] Y. Chen, D. Z. Wang, and S. Goldberg, “Scalekb: scalable learning and inference over large knowledge bases,” *The VLDB Journal*, vol. 25, no. 6, pp. 893–918, 2016.