Context aware Named Entity Recognition and Relation Extraction with Domain-specific language model

Youngrok Jang[1], Hosung Song[1], Junho Lee[2], Gyeonghun Kim[1], Yireun Kim[1], Stanley Jungkyu Choi[1], Honglak Lee[1] and Kyunghoon Bae[1]

[1]LG AI Research, 30, Magokjungang 10-ro, Gangseo-gu, Seoul 07796, Korea
[2]LG Display, 30, Magokjungang 10-ro, Gangseo-gu, Seoul 07796, Korea

Abstract

ChEMU 2022 tasks 1a and 1b aim to NER (Named Entity Recognition) and EE (Event Extraction) benchmarks. EE is RE (relation extraction) between trigger word and entity. We develop context-aware NER and RE models based on the domain-specific language model and achieve the state-of-the-art performance in ChEMU 2022, the public exact match f1 score of tasks 1a is 96.33, and task 1b is 92.82. For the domain-specific language model, we post-train the Bio-linkBert model with various corpora. We then select the best performing model from domain-specific benchmark datasets consisting of BLURB (Biomedical Language Understanding & Reasoning Benchmark) and ChEMU 2020. For the NER model, we choose a sequence tagging model that outperforms the span-based model in CHEMU 2022 task 1a. For the RE model, we train the model to classify the relation types or no relation between every pair of trigger words and entities in the snippet. Furthermore, we train both models using inputs that contain multiple sentences rather than a single sentence so that the model can utilize contextual information. For the ensemble, we train the best-performing model with 10-fold cross-validation and then predict the results with soft-voting. Finally, we apply rule-based post-processing to the prediction results.

Keywords

Language Model, Named Entity Recognition, Relation Extraction, Event Extraction

1. Introduction

Named Entity Recognition (NER)[1] and Relation Extraction (RE)[2] are well-known tasks in the field of information extraction research. Previous research has focused on diverse domain datasets, such as ACE05[1] from Newswire and online forums, and SciERC[3] from scientific papers. Both NER and RE models are based on either a general domain language model[4] or a domain-specific language model[5],[6],[7], depending on the dataset. And most of the works employ either a pipeline approach or a joint approach. A pipeline approach is training one model to extract entities and another model to classify relations between them. A joint approach is training the model for both tasks simultaneously.

CLEF 2022: Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy
jyrok3357@lgresearch.ai (Y. Jang); hosung.song@lgresearch.ai (H. Song); junho1126@lgdisplay.com (J. Lee); ghkayne.kim@lgresearch.ai (G. Kim); yireun.kim@lgresearch.ai (Y. Kim); stanleyjk.choi@lgresearch.ai (S. J. Choi); honglak@lgresearch.ai (H. Lee); k.bae@lgresearch.ai (K. Bae)


CEUR Workshop Proceedings (CEUR-WS.org)

[1]https://catalog.ldc.upenn.edu/LDC2006T06

ChEMU (Cheminformatics Elsevier Melbourne University) 2022 introduces 5 tasks to extract information from the snippet of chemical patents to support the drug discovery process. Among these tasks, we focus on NER task 1a and EE (event extraction) task 1b. Task 1a aims to extract chemical entities from the snippet. Task 1b aims to extract trigger words and relations between trigger words and entities from the snippet. Task 1b includes both NER and RE tasks. The extraction of trigger words is the NER task and the relation between trigger word and entity is the RE task. We develop context-aware NER and RE models based on the domain-specific language model and achieve state-of-the-art performance in ChEMU 2022, the exact match f1 score of task 1a is 96.33, and task 1b is 92.82.[2] In this paper, we explain our contributions to improving the performance of ChEMU 2022 : (1) domain-specific language model, (2) best performing NER and RE models, (3) context-aware model with input consisting of multiple sentences, (4) post-processing to the model prediction, (5) cross-validation and ensemble. Finally, we experiment and analyze our contributions in section 4.

For the domain-specific language model, we post-train the Bio-linkBert[7] model with various chemical corpora. We then select the best performing model from domain-specific benchmark datasets consisting of BLURB (Biomedical Language Understanding & Reasoning Benchmark)[8] and ChEMU 2020[9]. Among the pipeline approach and the joint approach, we choose the pipeline approach because PURE[10] reports it gets higher performance than the joint approach. For the NER model, we experiment with two popular approaches, the sequence tagging approach[4],[11] and the span-based approach[10], [12]. And finally, we choose the sequence tagging approach that shows higher performance in ChEMU 2022 task 1a. The NER model is trained to predict both entities in task 1a and trigger words in task 1b. For the RE model, we train the model to classify the relation types or no relation between every pair of trigger words and entities in the snippet. In task 1b, the RE model predicts the relation between entities and trigger words predicted by the NER model. Furthermore, we train both models using inputs that contain multiple sentences rather than a single sentence so that the model can utilize contextual information. For the ensemble, we train the best-performing model with 10-fold cross-validation and then predict the results with soft-voting. Finally, we apply rule-based post-processing to the prediction results.

2. Related Work

Named Entity Recognition (NER)[1] and Relation Extraction (RE)[2] are well-known tasks in the field of information extraction research. These tasks have lots of applications in various domains such as news, social media, biomedical and chemical domains. There are two areas of study. The first is to improve the language model, and the second is to improve the NER and RE models based on that model.

Recently the pre-trained language model such as BERT[4] and Roberta[13] have improved all NLP task performance. To improve the language model for NER and RE tasks, LUKE[14] and KeBioLM[15] use additional information such as named entity labels to pre-train the model. However, it is not easy to prepare a lot of label data in the chemical domain. Others approach

[2]ChEMU 2022 website shows public and private exact match f1 score and the score mentioned above is the public score. The meaning of public and private scores is explained in section 4

such as BioBert[5], PubmedBert[6] and Bio-linkBert[7] are pre-trained using domain-specific corpora. Similarly, we train a domain-specific language model with a chemical domain corpus and ultimately improve the performance of the ChEMU 2022 task.

For NER tasks, there are the sequence tagging approach such as BERT[4] and the span-based approach such as PURE[10] and PL-Marker[12]. For the sequence tagging approach, BERT uses a BIO scheme to encode each token into a tag and trains a model to classify each token into its tag. For the span-based approach, PURE and PL-Marker generate entity span candidates whose length is shorter than the maximum span length and then train a model to classify them as entity type or no-entity. We experiment with both approaches and then select the model with the best performance in the ChEMU 2022 task.

For NER and RE tasks, there are the pipeline approach [10],[12] and joint approach[16]. In a pipeline approach, the NER model predicts entities and then the RE model predicts the relation between them. On the other hand, in the joint approach, a single model learns the NER and RE tasks simultaneously. Because PURE[10] reports the pipeline approach performs better than the joint approach, we adopt the pipeline approach.

3. Method

3.1. Domain-specific language model

We use transformer encoder-based models and these models are published in huggingface[3], such as BERT, Roberta, BioBert, PubmedBert, Bio-megatron[17], and Bio-linkBert. Most of the publicly available pre-trained language models (PLM) are trained using general domain or biomedical domain knowledge. However, ChEMU 2022 data is composed of text based on chemical patents. Therefore, when fine-tuning publicly available pre-trained models, the gap between the chemical domain and other domains reduces the utilization of pre-trained knowledge. For example, if a word such as chemical compound is split into several tokens because the tokenizer is not trained with chemical domain texts, the language model may not capture its original meaning. And the understanding of the context is lowered due to a homonym problem in the different domains.

To overcome this problem, the one way is to learn a model from scratch using only the chemical domain texts, as PubmedBert and Bio-linkBert did in the biomedical domain, but it was difficult due to the lack of time. We try to solve the problems mentioned above by applying domain transfer to the set using the post-training method. In this paper, we post-train Bio-linkBert with various chemical corpora and then select the best performing model from domain-specific benchmark datasets consisting of BLURB (Biomedical Language Understanding & Reasoning Benchmark) and ChEMU 2020. Since we believe that the Pubmed dataset used by the pre-trained Bio-linkBERT has some chemical information, we want to put additional data information into the model without losing the already learned information. The same methodology as Mix-Review[18], *a rehearsal-based* continual learning approach, is applied for domain transfer.

[3]https://huggingface.co/

3.2. Named Entity Recognition

Using the domain-specific language model mentioned above, we experiment with the sequence tagging approach and the span-based approach. And then we compare which is better for the entity and trigger word recognition of ChEMU 2022 tasks 1a and 1b. In the case of the sequence tagging approach, a bio scheme is used, each token is encoded with the *BEGIN* and *IN* tag of a specific entity or *OUT* tag. And then the model is trained to classify each token into its corresponding tag. To classify tags, the output representation of each token is simply fed into a linear layer. We also experiment with CRF or Bi-LSTM+CRF layers in appendix A.1, but there is no performance improvement. In the case of the span-based approach, we consider token sequences shorter than the maximum span length[4] as entity span candidates and then train the model to classify them to corresponding entity type or no entity. However, there are chemical entities much longer than the maximum sequence length in ChEMU 2022 task 1a. Therefore, we use several heuristic approaches to add long entity span candidates. One simple way we used is to add a space-split sequence of tokens. A span representation for classifying an entity is a concatenation of the first token, last token representation and width embedding to capture entity length information.[5] After experiments, we finally decide to go with the sequence tagging approach that shows higher performance.

According to the error analysis part of the ChEMU2020[9], in some cases, contextual information from other sentences is necessary to extract trigger words. So we train the context-aware model with input including multiple sentences rather than a single sentence. It goes through several processing steps to generate the input data. First, we split the snippet into sentences with spacy[6] library. Second, by sliding the sentences from left to right, we generate inputs that contain as many sentences as possible without exceeding the maximum sequence length of the model.

3.3. Relation Extraction

In this paper, we use the PURE[10] approach to extract the relation between entity and trigger word extracted from the NER model. For every entity and trigger word pair in the snippet, we generate input data to train the model to classify as a specific relation type or no-relation. Note that this input includes relations that occur in a single sentence as well as relations that occur in cross sentences. The pre-processing step to generate this input is as follows. First, we split the snippet into sentences as we did in the pre-processing step. Second, we add the sentences in which the entity or the trigger word occurs to the input. Third, if there are intermediate sentences between added sentences, we add them as well. If the generated input is longer than the maximum sequence length, we skip it. Since this input consists of a trigger word and an entity that is far from it, there doesn't seem to be any relation between them. So skipping this input doesn't affect the performance. However, if the generated input is shorter than the maximum sequence length, we add as many left and right sentences as possible to the input to train a context-aware RE model. Finally, in order to include information about the entity

[4]The maximum span length used in the PURE paper is 8. We use the same value

[5]Since entity span candidates can be too long, so 9 width embeddings are used, embeddings for 1 to 8 tokens and embeddings for tokens longer than 8.

[6]https://spacy.io/usage/spacy-101

and trigger span, special tokens are inserted before and after the entity and trigger span. Each special token indicates the type of entity or trigger and whether it is inserted before or after the span. After input is fed to the model, output representations of special tokens before entity and before trigger are concatenated. And then, this concatenated representation is fed to the linear layer to classify relations.

3.4. Ensemble

Ensemble methods combine predictions from multiple models to improve performance. We train NER and RE models using 10-fold cross-validation on the merged training and development datasets. We apply the soft voting ensemble method to output results from 10 models.

3.5. Post-processing

We apply three post-processing methods to correct the results mispredicted by NER & RE models. The first method is to correct the entity misclassification of STARTING_MATERIAL as REAGENT_CATALYST. According to ChEMU 2020[9], misclassifying STARTING_MATERIAL as REAGENT_CATALYST is one of the most common errors in the NER task. We design the rules according to the definition of the entities or trigger words. By definition, the difference between STARTING_MATERIAL and REAGENT_CATALYST is that STARTING_MATERIAL is consumed during the chemical reaction, while REAGENT_CATALYST is not consumed and only increases the reaction rate. In other words, unlike REACTION_CATALYST, the molecular structure of STARTING_MATERIAL is similar to REACTION_PRODUCT. Therefore, we measure similarity between STARTING_MATERIAL or REAGENT_CATALYST and REACTION_PRODUCT in the snippet and then correct the entity type if it appears to be misclassified. [7]

The second method is to correct mispredicted trigger word or entity spans. Sometimes the model predicts different spans for the same word in different sentences. For example, the model predicts "taken up" as a trigger word span, but sometimes only "taken" without "up" in other sentences. Sometimes this can happen because the labels for the same word are different from each other in the dataset. We apply post-processing that modifies all "taken" to "taken up". In the same manner, several spans of the entities are post-processed.

The third method is to correct the relation misclassification of WORKUP as REACTION_STEP. If the RE model predicts that a trigger word is related to REACTION_PRODUCT, YIELD_PERCENT, YIELD_OTHER at one time, the trigger word should be REACTION_STEP rather than WORKUP. The rule should capture the sentence at the end of the snippet that describes the material synthesis in which the product is finally formed. Thus, we force the trigger word WORKUP to be replaced by REACTION_STEP in this case.

4. Experiments

We evaluate our domain-specific language model with the BLURB benchmark and ChEMU 2020 dataset. We then evaluate NER and RE models with ChEMU 2022 task 1a and 1b datasets. Since

[7] the details are described in appendix A.2

Table 1
Overall statistics of train and development datasets in ChEMU 2022 task1b.

Feature	Value
# Patent snippets	1500
# Entities	26857
# Trigger Words	11236
# Relations	23445

the ChEMU 2022 test dataset may also consist of unseen data, we want to choose a domain-specific language model that generally performs well for the unseen data. This is why we use the BLURB benchmark dataset together rather than just ChEMU 2020. BLURB benchmark dataset is based on the biomedical domain, which has some relevance to the chemical domain. Furthermore, it also includes chemical domain data, such as BC5-chem and ChemProt. The train dataset of ChEMU 2020 is the same as that of ChEMU 2022, but the development and test datasets of ChEMU 2020 are the same as the development dataset of ChEMU 2022. The test data set of ChEMU 2020 is public, while that of ChEMU 2022 is not. In order to get the score of the ChEMU 2022 test data set, the model must be uploaded to the ChEMU website. For convenience, we use the ChEMU 2020 data set to evaluate domain-specific language models, but the ChEMU 2022 data set to evaluate our NER and RE models.

4.1. Dataset

4.1.1. ChEMU 2022 dataset

ChEMU 2022 includes five tasks: named entity recognition (task 1a), event extraction (task 1b), anaphora resolution (task 1c), chemical reaction reference resolution (task 2a), and table semantic classification (task 2b). Among them, we focus on tasks 1a and 1b. Task 1a is to extract chemical entities and task 1b is to extract both the trigger words and the relations between trigger words and chemical entities. The dataset of task 1b is a superset of task 1a. Table 1 shows the overall statistics of the train and development datasets in ChEMU 2022 task 1b.

4.1.2. BLURB benchmark dataset

The BLURB benchmark dataset consists of six tasks as follows : named entity recognition, PICO (patient population, interventions, comparator, and outcomes), relation extraction, sentence similarity, document classification, and question answer. Among them, we use only NER and RE datasets, which are the target tasks of ChEMU 2022. Table 2 summarizes the dataset we use.

Table 2
The NER and RE datasets in BLURB benchmark.

Dataset	Task	Train	Dev	Test	Evaluation Metrics
BC5-chem	NER	5203	5547	5385	F1 entity-level
BC5-disease	NER	4182	4244	4424	F1 entity-level
NCBI-disease	NER	5134	787	960	F1 entity-level
BC2GM	NER	15197	3061	6325	F1 entity-level
JNLPBA	NER	46750	4551	8662	F1 entity-level
ChemProt	RE	18035	11268	15745	Micro F1
DDI	RE	22233	5559	5716	Micro F1
GAD	RE	4261	534	535	Micro F1

4.2. Implementation

4.2.1. Domain-specific language model

We post-train Bio-LinkBert with different corpus combinations and then select the best-performing model. Based on the architecture and weight of Bio-LinkBert large[8], we post-train Bio-LinkBert on a task of masked language modeling[13]. We experiment with three corpora: (1) Google patent: 23 GB of chemical domain patents we crawled using chemical keywords, (2) Journal: 22 GB of chemical journal abstracts and body text (3) Pubmed abstract[9]: used by training BioBert[5], 38 GB of biomedical domain data. For the Pubmed abstract corpus, we use 12 GB, which is 30% of the total data. This is because Bio-LinkBert has already been trained with Pubmed abstract and the post-training aims to learn new information without losing what has been learned. The corpus used for our best-performing model is the combination of Journal and Pubmed abstract. We train our model for 15,000 steps (approx. 2 epochs) with sequence length 512, batch size 2k, weight reduction 0.01, warm-up 3000 steps, and learning rate 5e-5. The training time is about 13 hours using DeepSpeed[10] with 16 Nvidia A100 40GB GPUs.

4.2.2. NER & RE models

We train a NER model to predict both entities and trigger words and a RE model to predict relations between them. At inference time, the RE model predicts the relation using the results predicted by the NER model. Although the code for ChEMU 2022 task 1a and task 1b is published, we implement all codes for pre-processing, post-processing, and modeling for NER and RE. As we will discuss in the 4.4 section, we achieve higher performance than the other participants in tasks 1a and 1b even using publicly available domain-specific language models such as

[8]https://huggingface.co/michiyasunaga/BioLinkBERT-large
[9]https://github.com/EleutherAI/the-pile
[10]https://github.com/microsoft/DeepSpeed

Table 3
The Exact match results of the ChEMU 2020 test set. "GP" indicates training with Google Patent corpus we crawl. "PM" refers PubMed abstract corpus and "J" refers to Journal data. "p", "r" and "f1" means precision, recall and f1 score, respectively. We train these models with the input data generated from each paragraph, but (doc) means model trained at the document level.

Model	Trigger			Entity		
	p	r	f1	p	r	f1
PubmedBert-base	96.1	94.9	95.5	95.6	94.3	95.0
Bio-linkBert-large	96.3	95	95.6	95.9	94.2	95.1
+GP	95.9	97.4	96.6	95.8	96.1	95.9
+GP+PM	96.3	97.2	**96.8**	95.6	96	95.8
+J	96	97.3	96.6	95.8	96.1	96
+J+PM	96.6	96	96.3	95.9	96.1	96
+J+PM (doc)	96.2	97.1	96.7	96.1	96.3	**96.2**

PubmedBert and Bio-linkBert on huggingface. We train the NER model for 20 epochs with a learning rate of 5e-5. At each epoch, we evaluate it on the development dataset and choose the best-performing model. In the same manner, we train the RE model for 10 epochs with a learning rate of 2e-5 and choose the best-performing model. The training time of the NER and RE model is about 20 minutes and 24 hours with 1 Nvidia A100 40GB GPU. Because the RE model is trained to classify all pairs of trigger words and entities in the snippet, it takes longer than the NER model only to classify each token.

For the ensemble, We train a model with 10-fold cross-validation. And these 10 trained models predict entities or relations by soft-voting. Finally, post-processing is applied to the prediction results of the ensemble model.

4.3. Evaluation Result

We post-train Bio-linkBert with various corpora to obtain the domain-specific language model that achieves high performance in ChEMU 2022 tasks 1a and 1b. The performance verification of this model is performed using the CHEMU 2020 dataset and the BLURB dataset.

Table 3 shows the entity and trigger extraction performance of post-trained models in ChEMU2020 task 1a. Table 4 shows the results of the NER and RE performance of the BLURB dataset. In Tables 3 and 4, the post-training with the journal and Pubmed abstract on Bio-linkBert large achieves the highest overall score, so we choose this model as our final model. We train these models with the input data generated from each paragraph. Training the model with input generated at the document level gives a slight performance improvement. However, in ChEMU 2022, the score eventually drops slightly, so it is not used.

Table 5 and Table 6 show the evaluation results of task 1a and 1b in ChEMU2022, respectively.

Table 4
The evaluation results of NER and RE tasks in BLURB, the f1 score of the test dataset.

Model	BC5 -chem	BC5 -disease	NCBI -disease	BC2GM	JNLPBA	Chem Prot	DDI	GAD	Average score
PubmedBert-base	92.95	85.35	87.57	84.36	79.13	77.02	82.74	81.89	83.87
Bio-linkBert-large	93.33	85.65	87.62	84.61	79.08	77.68	82.03	84.15	84.26
+GP	94.1	85.79	88.46	84.9	**79.97**	77.85	82.74	**85.62**	84.92
+GP+PM	93.66	85.93	88.06	84.67	79.38	79.91	82.82	85.42	84.98
+J	**94.13**	85.64	88.61	84.51	79.53	79.49	83.39	84.66	84.99
+J+PM	94.11	**86.65**	88.11	85.03	79.79	79.92	83.17	85.04	85.24
+J+PM (doc)	93.92	85.58	**89.32**	**85.06**	79.53	**79.97**	**84.79**	84.33	**85.31**

Table 5
ChEMU 2022 task 1a: named entity recognition evaluation results of the test dataset.

Model	Exact F1		Relaxed F1	
	public	private	public	private
Hokkaido University	93.20	94.12	94.58	95.35
ChEMU Baseline	93.20	93.67	95.28	95.72
Virginia Commonwealth University	77.80	76.86	87.19	87.45
Ours (single)	95.52	95.86	97.10	97.33
Ours (Ensemble)	96.26	96.73	97.55	**97.93**
Ours (Ensemble) + post processing	**96.33**	**96.80**	**97.59**	**97.93**

The public and private scores are calculated from 30% and 70% of the test data set, respectively. [11] Both exact match and relaxed match require predicted entity or trigger word type to match the label. For span, exact match requires that predicted span exactly matches gold span. However, the relaxed match only requires that the predicted span overlap the gold span. Both metrics use f1 score which is the harmonic mean of the precision and recall.

In Table 5, our single model achieves public and private exact match f1 scores improvement of +2.32 and +1.74 compared to Hokkaido University which achieves the highest score among other participants. Also, our final model with ensemble and post-processing achieves +3.13 and +2.68.

In Table 6, the evaluation method of task 1b in ChEMU 2022 is very similar to 1a, except that the relation type must match the label as well. To check whether the performance of the RE model is higher than that of other participants, we predicted the relation using the entity and trigger prediction results of the model[12] that achieved the lowest performance among the

[11]The Private score was published before the submission deadline, and the public score was published after that time.

[12]EM F1 public score = 93.65, private score = 94.3

Table 6
ChEMU 2022 task 1b: Event extraction evaluation results of the test dataset.

Model	Exact F1		Relaxed F1	
	public	private	public	private
Hokkaido University	87.00	88.68	89.63	90.28
ChEMU Baseline	88.42	89.25	90.36	91.04
Virginia Commonwealth University	74.08	74.73	78.93	79.46
Ours (single, worst ner)	90.46	90.51	92.34	92.07
Ours (single)	92.00	91.84	93.75	93.48
Ours (Ensemble)	92.23	91.99	94.03	**93.63**
Ours (Ensemble) + post processing	**92.82**	**92.15**	**94.24**	93.61

Table 7
Ablation over the pre-tained language models. PLMs publicly opened in huggingface and our best performing PLM are evaluated.

Model	task1a : NER		task1b: EE	
	public	private	public	private
PubmedBert base	94.73	95.55	**92**	**91.84**
Bio-linkBert base	95.2	94.91	91.53	91.66
Bio-linkBert large	95.05	95.34	91.02	91.28
Ours	**95.52**	**95.86**	91.86	91.78

NER models we submitted. Even in this case, the public and private exact match f1 scores are +2.04 and +1.26 higher than the ChEMU Baseline, which has the highest performance among participants. Therefore, the proposed RE model also affects the performance improvement. The highest score is obtained by training an ensemble RE model using the prediction results of the best performing NER and applying post-processing to the prediction results. In this case, the public and private exact match f1 scores are +4.4 and +2.9 higher than the ChEMU Baseline.

4.4. Analysis

This section explains how PLM and data pre-processing methods affect the performance of ChEMU 2022 task 1a and 1b.

Table 7 shows the exact match f1 score of the publicly available PLMs and our language model, which is post-trained on Bio-linkBert large with journal and pubmed abstract data and the best performing model is used by measuring the Blurb and CHEMU20 performance at every 500 steps within 2 epochs. . In task 1a, our language model outperforms all other PLMs. However, although it is not common, the performance of Bio-linkBert large is lower than Bio-linkBert base. Therefore, post-training Bio-linkBert base as we did for Bio-linkBert large may improve the performance. In task 1b, our model outperforms the Bio-linkBert large model, but PubmedBert base gets the highest score. As a result, we use our language model for task 1a and PubmedBert base for task 1b.

Table 8 shows the comparison of the exact match f1 score in ChEMU 2022 according to the

Table 8
Ablation over pre-processing methods, how to generate input data. For each task 1a and 1b, the best performance PLM is used respectively.

Pre-processing	task1a : NER		task1b : EE	
	public	private	public	private
line	93.89	94.6	91.89	91.63
snippet	**95.52**	**95.86**	**92**	**91.84**

Table 9
Ablation over post-processing methods. Similarity refers to post-processing using molecular similarity between REACTION_PRODUCT and STARTING_MATERIAL or REACTION_PRODUCT and REAGENT_CATALYST. Entity span and trigger span indicates post-processing of entities and trigger span mismatch ,respectively. Strict relation refers post-processing to forbid WORKUP relate to REACTION_PRODUCT, YIELD_PERCENT and YIELD_OTHER at once. Post-processed means applying all post-processing methods.

Model	public		private	
	Exact F1	Relaxed F1	Exact F1	Relaxed F1
NER base	96.26	97.55	96.73	**97.93**
NER similarity	96.29	**97.59**	96.73	**97.93**
NER entity span	96.29	97.55	**96.80**	**97.93**
NER post-processed	**96.33**	**97.59**	**96.80**	**97.93**
EE base	92.23	94.03	91.99	**93.63**
EE similarity	92.27	94.07	91.99	**93.63**
EE trigger span	92.56	94.03	**92.15**	**93.63**
EE entity span	92.27	94.03	92.01	**93.63**
EE strict relation	92.40	94.19	91.98	93.61
EE post-processed	**92.82**	**94.24**	**92.15**	93.61

pre-processing methods that generates the input data to train the model. A single snippet txt file of ChEMU 2022 data consists of multiple lines. We apply pre-processing methods mentioned in section 3.2 and 3.3 with two different ways. The first is to pre-process the data line by line and the second is to pre-process the entire snippet. If the input is generated only on each line, the model cannot predict the result by referencing the context information given in the other lines. Furthermore, for task 1b, the first deals with relations that occur on a single line, while the second also includes relations that occur in multiple lines. Therefore, the second outperforms the first.

Table 9 shows the ablation over three post-processing methods: (1) similarity: to correct the entity misclassification of STARTING_MATERIAL as REAGENT_CATALYST, (2) trigger &

entity span: to correct mispredicted trigger word or entity spans. (3) strict relation: to correct the relation misclassification of WORKUP as REACTION_STEP in some cases. In the NER model, the similarity method improves public exact and relaxed f1 by +0.03 and +0.04, respectively. The entity span method improves public exact f1 by +0.03. As a result, the score after applying all post-processing methods shows +0.07 and +0.04 improvement in public exact and relaxed f1, respectively. In EE model public score, the most effective method is trigger span method which improves public exact f1 by +0.33. Following method is strict relation, improves public exact f1 by +0.17 and relaxed f1 by +0.16. This rule drops the private score a bit, but it is still effective in the public score. Similarity improves public exact f1 by +0.04 and relaxed f1 by +0.04 and entity span method improves public exact f1 by +0.04. Finally all of post-processing methods applied EE model shows +0.59, +0.21 in public exact, relaxed f1 score higher than EE base model.

5. Conclusion

In this paper, we present the domain-specific language model and context-aware NER and RE models for ChEMU 2022. For the best performing domain-specific language model, We post-train Bio-linkBert with various corpora. Based on this language model, we present the NER model using the sequence tagging method and the RE model using the PURE approach. Pre-processing methods where the input contains multiple lines of sentences help the model to be context-aware, which ultimately improves performance. Finally, we train the ensemble model and apply some rules as post-processing.

We achieve state-of-the-art performance on ChEMU 2022 tasks 1a and 1b and analyze contributions to performance. However, there is still room for improvement. First, because our language model has a maximum sequence length of only 512 tokens, the model can not predict entities and relations referencing the entire snippet. Second, some chemical entities consist of too many tokens, which can degrade the performance of the model. These will be our future works to develop improved language models.

References

[1] E. F. Tjong Kim Sang, F. De Meulder, Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition, in: Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, 2003, pp. 142–147. URL: https://aclanthology.org/W03-0419.

[2] D. Zelenko, C. Aone, A. Richardella, Kernel methods for relation extraction, in: Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002), Association for Computational Linguistics, 2002, pp. 71–78. URL: https://aclanthology.org/W02-1010. doi:10.3115/1118693.1118703.

[3] Y. Luan, L. He, M. Ostendorf, H. Hajishirzi, 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, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 3219–3232. URL: https://aclanthology.org/D18-1360. doi:10.18653/v1/D18-1360.

[4] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, 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), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.

[5] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, BioBERT: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics (2019). URL: https://doi.org/10.1093/bioinformatics/btz682. doi:10.1093/bioinformatics/btz682.

[6] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, 2020. arXiv:arXiv:2007.15779.

[7] M. Yasunaga, J. Leskovec, P. Liang, Linkbert: Pretraining language models with document links, in: Association for Computational Linguistics (ACL), 2022.

[8] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, ACM Transactions on Computing for Healthcare 3 (2022) 1–23. URL: https://doi.org/10.1145%2F3458754. doi:10.1145/3458754.

[9] J. He, D. Q. Nguyen, S. A. Akhondi, C. Druckenbrodt, C. Thorne, R. Hoessel, Z. Afzal, Z. Zhai, B. Fang, H. Yoshikawa, A. Albahem, L. Cavedon, T. Cohn, T. Baldwin, K. Verspoor, Chemu 2020: Natural language processing methods are effective for information extraction from chemical patents, Frontiers in Research Metrics and Analytics 6 (2021). URL: https://www.frontiersin.org/article/10.3389/frma.2021.654438. doi:10.3389/frma.2021.654438.

[10] Z. Zhong, D. Chen, A frustratingly easy approach for entity and relation extraction, in: North American Association for Computational Linguistics (NAACL), 2021.

[11] J. Wang, Y. Ren, Z. Zhang, Y. Zhang, Melaxtech: A report for CLEF 2020 - chemu task of chemical reaction extraction from patent, in: L. Cappellato, C. Eickhoff, N. Ferro, A. Névéol (Eds.), Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, Thessaloniki, Greece, September 22-25, 2020, volume 2696 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2020. URL: http://ceur-ws.org/Vol-2696/paper_238.pdf.

[12] D. Ye, Y. Lin, P. Li, M. Sun, Packed levitated marker for entity and relation extraction, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 4904–4917. URL: https://aclanthology.org/2022.acl-long.337. doi:10.18653/v1/2022.acl-long.337.

[13] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692 (2019).

[14] I. Yamada, A. Asai, H. Shindo, H. Takeda, Y. Matsumoto, LUKE: Deep contextualized entity representations with entity-aware self-attention, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, 2020. URL: https://aclanthology.org/2020.emnlp-main.523. doi:10.18653/v1/2020.emnlp-main.523.

[15] Z. Yuan, Y. Liu, C. Tan, S. Huang, F. Huang, Improving biomedical pretrained language

models with knowledge, in: Proceedings of the 20th Workshop on Biomedical Language Processing, Association for Computational Linguistics, Online, 2021, pp. 180–190. URL: https://aclanthology.org/2021.bionlp-1.20. doi:10.18653/v1/2021.bionlp-1.20.

[16] S. T.Y.S.S, P. Chakraborty, S. Dutta, D. K. Sanyal, P. P. Das, Joint entity and relation extraction from scientific documents: Role of linguistic information and entity types, in: EEKE@JCDL, 2021.

[17] H.-C. Shin, Y. Zhang, E. Bakhturina, R. Puri, M. Patwary, M. Shoeybi, R. Mani, Biomegatron: Larger biomedical domain language model, 2020. URL: https://arxiv.org/abs/2010.06060. doi:10.48550/ARXIV.2010.06060.

[18] T. He, J. Liu, K. Cho, M. Ott, B. Liu, J. Glass, F. Peng, Analyzing the forgetting problem in pretrain-finetuning of open-domain dialogue response models, in: EACL, 2021.

[19] S. Kim, P. A. Thiessen, E. E. Bolton, J. Chen, G. Fu, A. Gindulyte, L. Han, J. He, S. He, B. A. Shoemaker, et al., Pubchem substance and compound databases, Nucleic acids research 44 (2016) D1202–D1213.

[20] T. Tanimoto, Ibm internal report, 17th november, IBM Company: Armonk, NY, USA (1957).

[21] A. Tversky, Features of similarity., Psychological review 84 (1977) 327.

Table 10
Ablation over NER approaches and classification layers in ChEMU 2020

Approach	PLM	Exact F1	
		Entity	Trigger word
Span-based	PumbedBert base	86.9	96.6
Sequence tagging (CRF)	PumbedBert base	**96**	96
Sequence tagging (Dense)	PumbedBert base	95.9	**96.5**

A. Appendix

A.1. Additional Ablation Studies for the NER Model

Table 9 shows the ablation studies over the NER approach and classification layer in ChEMU 2020. As mentioned in section 3, sequence tagging approach outperforms span-based approach. However, the performance difference between the dense layer and CRF layer is not significant.

A.2. Post-processing to correct the entity misclassification of STARTING_MATERIAL as REAGENT_CATALYST

As we mentioned in 3.5, we measure the similarity between STARTING_MATERIAL or REAGENT_CATALYST and REACTION_PRODUCT in the snippet and then correct the entity type if it appears to be misclassified. We use Pubchem[19] Python package pubchempy[13] to parse chemical entity from text, and then the similarity is measured using Python package RDKit[14] with Tanimoto coefficient[20] and Tversky index[21].

[13]https://github.com/mcs07/PubChemPy
[14]The RDKit: Open-Source Cheminformatics Software, version 2022.03.2. http://www.rdkit.org