Named Entity Recognition Under Domain Shift via Metric Learning for Life Sciences

Hongyi Liu[1], **Qingyun Wang**[2], **Payam Karisani**[2], **Heng Ji**[2]
[1] Shanghai Jiao Tong University, [2] University of Illinois at Urbana-Champaign
liu.hong.yi@sjtu.edu.cn,
{qingyun4,karisani,hengji}@illinois.edu

Abstract

Named entity recognition is a key component of Information Extraction (IE), particularly in scientific domains such as biomedicine and chemistry, where large language models (LLMs), e.g., ChatGPT, fall short. We investigate the applicability of transfer learning for enhancing a named entity recognition model trained in the biomedical domain (the source domain) to be used in the chemical domain (the target domain). A common practice for training such a model in a few-shot learning setting is to pretrain the model on the labeled source data, and then, to finetune it on a hand-full of labeled target examples. In our experiments, we observed that such a model is prone to mislabeling the source entities, which can often appear in the text, as the target entities. To alleviate this problem, we propose a model to transfer the knowledge from the source domain to the target domain, but, at the same time, to project the source entities and target entities into separate regions of the feature space. This diminishes the risk of mislabeling the source entities as the target entities. Our model consists of two stages: 1) entity grouping in the source domain, which incorporates knowledge from annotated events to establish relations between entities, and 2) entity discrimination in the target domain, which relies on pseudo labeling and contrastive learning to enhance discrimination between the entities in the two domains. We conduct our extensive experiments across three source and three target datasets, demonstrating that our method outperforms the baselines by up to 5% absolute value[1].

1 Introduction

Named entity recognition is a crucial step in IE tasks. Existing models have achieved remarkable performance in the general domain (Lin et al., 2020; Wang et al., 2021b; Zhang et al., 2023; Shen et al., 2023b). However, in the scientific domains, e.g., medical or chemical domains, these models usually struggle due to the extremely large quantity of concepts, the wide presence of multi-token entities, and the ambiguity in detecting entity boundaries.

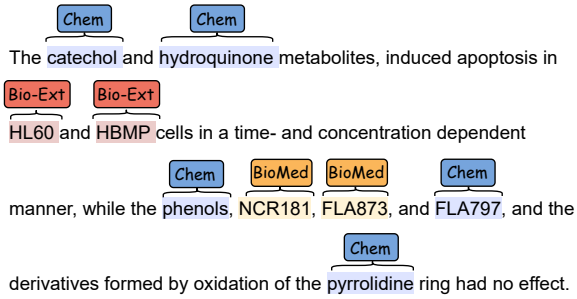

Figure 1: A test example in the chemical domain. The words marked with blue indicators are chemical entities, and the words marked with red and orange indicators are biomedical entities. The entities in red are mislabeled by a few-shot model as chemical entities.

Large language models (LLMs) show an impressive performance on various NLP tasks such as question answering or text summarization (OpenAI, 2022). Models such as ChatGPT (OpenAI, 2022) can achieve outstanding results given just a few training examples (Wang et al., 2022). However, Kandpal et al. (2023) recently report that the performance of these models is proportional to the number of relevant documents present in their pretraining corpus. Thus, one can expect that their performance fluctuates across domains. This is particularly expected to occur across certain scientific subjects, where the data may be scarce. Given the already existing challenges of the named entity recognition task in the scientific domain—mentioned earlier—this factor can further exacerbate the problem. For instance, in our early few-shot learning experiments, we observed that the results of ChatGPT in the chemical domain named entity recognition task are significantly worse than

[1]Code, data, and resources are publicly available for research purposes: https://github.com/Lhtie/Bio-Domain-Transfer.

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 1–21

those in the general domain.[2]

In the present work, we employ transfer learning (Pan and Yang, 2010) to alleviate this problem. Transfer learning methods exploit the label data from one domain (called the source domain) to minimize the prediction error in another domain (namely the target domain). We particularly focus on a realistic setting, where given a large set of labeled data from the source domain and a small set of labeled data from the target domain, the goal is to develop a model for the target domain[3]. There are more resources in the biomedical domain than in the chemical domain due to funding priorities and BioNLP workshops. Therefore, as a case study, we take the biomedical domain as the source and the chemical domain as the target.

Figure 1 shows the challenges a named entity recognition model can face in the chemical domain. The model is trained in a few-shot learning setting. Thus, it is trained on labeled biomedical data, and then, further finetuned on a few labeled examples from the chemical domain. The task is to recognize only the chemical entities—ignoring other types of entities. Blue indicators represent chemical entities, while red and orange indicators denote biomedical entities. The model categorizes the entities marked with blue and red as chemical entities. The first observation is that, to an inexpert human, it is difficult to perform the task because the entities are highly domain-specific. The second observation is that the model wrongfully labels some entities from the source domain as the target entities. Therefore, engineers developing such a model face a dilemma: while not using the source data dramatically deteriorates performance,[4] simply pretraining with the source data increases the false positive rate. The third, and perhaps the most important, observation is that the examples from the source and target domains can co-occur in the same document. This problem setting contradicts the regular transfer learning setting, where the examples from the source and the target domains are fully disjoint (Ben-David et al., 2010). The latter property poses difficulties in the applicability of the traditional transfer learning methods in this setting.

Our core idea is to train a named entity recognition model such that it is able to project the representations of the source and target entities into separate regions of the feature space. Such a model can potentially transfer knowledge from the source domain to the target domain by constructing a shared feature space between the two domains. Furthermore, it reduces the similarity between the representations of the entities in the two domains, and consequently, it can potentially minimize the number of source entities that are mislabeled as target entities. To achieve this, our model consists of a pretraining stage on the source data, and a finetuning stage on the target data.

In the pretraining stage, we propose two methods to enrich the feature space with auxiliary data. The auxiliary data is extracted from the event mentions that the entities participate in. Additionally, during this stage, we propose to employ the multi-similarity loss term (Wang et al., 2019), which enables us to partition (or group) the source entities. Our empirical analysis shows that constructing such a feature space during the pretraining stage facilitates our projection step during the finetuning stage. In the finetuning stage, we detect the potentially false positive entities by pseudo-labeling them. Then, we aim to construct a feature space that projects the pseudo-labeled entities and the target entities into separate regions. We achieve this by employing the multi-similarity loss again. We evaluate our method across twelve use cases and show it outperforms the baselines in most experiments, with improvements of up to 5% in absolute value. We also empirically analyze our method and show that each proposed technique is individually effective. Our contributions are threefold:

- We propose a new pretraining algorithm for the named entity recognition task in the transfer learning setting. Our algorithm involves two steps: first, extracting auxiliary information about the source entities through the event mentions they participate in; and second, proposing an entity grouping technique using the multi-similarity loss. Our methods have proven effective for the named entity recognition task in the target domain. Our study is carried out in the scientific domain, particularly from the biomedical data as the source domain to the chemical data as the target domain. This is a crucial and challenging real-world problem. All of our claims, here and later, are only about this task.
- We propose a finetuning algorithm, which aims to project the target entities and the

[2]We report this complementary experiment in Appendix A.

[3]In the literature, this setup is often called the semi-supervised transfer learning setting (Saito et al., 2019).

[4]We empirically support this in the analysis section.

entities that may be potentially mislabeled into separate regions of the feature space. It comprises two steps: first, detecting the potentially out-of-domain entities by pseudo-labeling them; and second, obtaining dissimilar representations for the two sets of entities using the multi-similarity loss.

- We conduct extensive experiments across twelve cases, showing that our method significantly outperforms the baselines and shedding light on various aspects of our model.

2 Background

2.1 Named Entity Recognition

We view the named entity recognition task as a sequence labeling problem. We denote the training data by $D = \{(\mathbf{X}_i, \mathbf{Y}_i)\}_{i=1}^{n}$, where n is the number of input passages (or texts). To train a classifier f with parameter θ, we minimize the loss as follows:

$$\mathcal{L}_{NER} = \mathbb{E}_{(\mathbf{X}_i, \mathbf{Y}_i) \sim D} \left[CE \left(f(\mathbf{Y}_i | \mathbf{X}_i; \theta), \mathbf{Y}_i \right) \right], \quad (1)$$

where CE is the cross-entropy loss.

2.2 Transfer Learning

We are given examples from the source domain $\mathbb{S}$ and the target domain $\mathbb{T}$, where the training set size in the target domain is much smaller than the source domain, i.e., $|\mathbf{X}^{\mathbb{T}}| << |\mathbf{X}^{\mathbb{S}}|$. Given this data, we aim to develop a model for the target domain and minimize its prediction error.

We focus on the named entity recognition task, and take the biomedical domain as the source and the chemical domain as the target. Note that the data distributions in the two domains are different. Therefore, a model solely trained on the source data is usually not as competitive as one trained on the target data. The baseline solution in this setting, *direct transfer*, is to pretrain the model on the labeled source data and finetune it on the labeled target data.

2.3 Multi-Similarity Loss

We enhance our named entity recognition model in the source domain by capturing the similarities between entity pairs. To achieve this, we employ an objective term called the multi-similarity (MS) contrastive loss proposed by Wang et al. (2019) for metric learning. MS can incorporate self-similarity, relative positive similarity, and relative negative similarity. Self-similarity depends on the properties of the data point itself, such as hardness, while negative and positive similarities are measured with respect to an anchor data point.

In the following sections, we use the final encoder hidden states of input tokens as entity representations. If an entity consists of multiple tokens, we take the average of their representations. Additionally, we denote the relative similarity between entity pairs as $S_{\cdot}$, using cosine similarity.

The multi-similarity loss is calculated in two stages. First, given the *i-th* entity denoted by x_i and its label denoted by y_i, we aim to extract the most difficult positive and negative entities. This is achieved by thresholding over the relative similarity scores as follows:

$$\begin{aligned} \mathcal{P}_i &= \{x_j | S_{ij}^{+} < \max_{y_k \neq y_i} S_{ik} + \epsilon\}, \\ \mathcal{N}_i &= \{x_j | S_{ij}^{-} > \min_{y_k = y_i} S_{ik} - \epsilon\}, \end{aligned} \quad (2)$$

where $\mathcal{P}_i$ is the set of positive, $\mathcal{N}_i$ is the set of negative, and ϵ is a margin penalty.

Second, we calculate a soft weight score for the extracted pairs to reflect their importance:

$$\begin{aligned} w_{ij}^{+} &= \frac{e^{-\alpha(S_{ij} - \gamma)}}{1 + \sum_{k \in \mathcal{P}_i} e^{-\alpha(S_{ik} - \gamma)}}, \\ w_{ij}^{-} &= \frac{e^{\beta(S_{ij} - \gamma)}}{1 + \sum_{k \in \mathcal{N}_i} e^{\beta(S_{ik} - \gamma)}}, \end{aligned} \quad (3)$$

where α, β, and γ are hyperparameters. We observe that in each set, the weights are the ratio of the self-similarity scores to the sum of all the relative scores in the set.

The final multi-similarity loss is:

$$\begin{aligned} \mathcal{L}_{MS} = \frac{1}{n_e} \sum_{i}^{n_e} \Bigg\{ & \frac{1}{\alpha} \log\Big[1 + \sum_{k \in \mathcal{P}_i} e^{-\alpha(S_{ik} - \gamma)}\Big] \\ & + \frac{1}{\beta} \log\Big[1 + \sum_{k \in \mathcal{N}_i} e^{\beta(S_{ik} - \gamma)}\Big] \Bigg\}, \end{aligned} \quad (4)$$

where n_e is the total number of the entities. Note that the values of $S_{\cdot}$ already contain the weights computed in Equation 3.

3 Proposed Method

Figure 2 illustrates our framework. It consists of two stages, pretraining on the source data, and then, finetuning on the target data. In the source domain, we employ external knowledge to construct

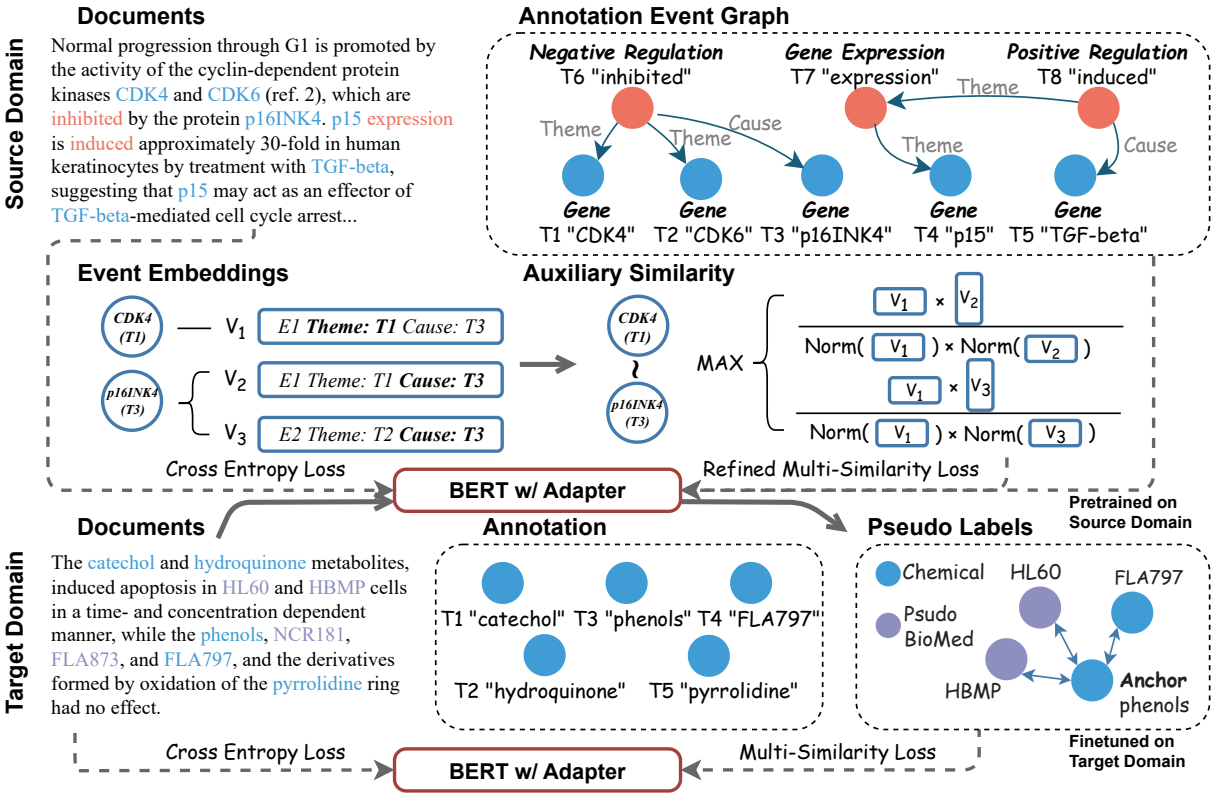

Figure 2: Overview of proposed entity grouping and entity discrimination frameworks. Entity grouping on the source domain is shown in the upper part. Based on event annotations, a set of event embeddings is constructed under two paradigms. Afterward, pairwise auxiliary similarity scores are calculated according to argument embeddings. Extended multi-similarity loss concerning four types of similarities, combined with cross-entropy loss, are jointly learned during pretraining. Entity discrimination on the target domain is shown in the lower part. Pseudo labels are formed by the named entity recognition model pretrained in the source domain, and in contrast to annotated labels, a multi-similarity loss is injected into finetuning.

an event feature space based on their arguments, and to calculate the auxiliary similarity scores between entities. Then, the similarity scores are used in the multi-similarity loss to shape the entity feature space. In the target domain, we aim to enhance the model's ability to distinguish the target entities from the entities that are likely to be mislabeled, such as the entities that potentially belong to the source domain. To achieve this, we propose an algorithm to extract pseudo-labels and employ the multi-similarity loss for the second time.

3.1 Source Domain Pretraining

Using external knowledge, we enrich entity representations for source domain pretraining. Since the source domain consists of a set of various subdomains (or sub-topics), discovering these subdomains in the source domain facilitates the subsequent process of domain transfer (Hoffman et al., 2012). Specifically, by detecting these sub-domains and grouping the data, we enable the contrastive loss in the next step to consider each one individually and transform them accordingly. This approach avoids the oversimplification of treating the entire source domain as a single cluster. Below, we propose two separate approaches to obtain the auxiliary embedding vectors, both exploiting event mentions that the entities appear in. Given the auxiliary vectors, we describe our method for calculating the auxiliary similarity scores. Finally, we provide an overview of the pretraining loss function, which incorporates the entity similarity scores and the auxiliary similarity scores.

Concatenation-based Event Embedding. Our first approach relies on an off-the-shelf token encoder pretrained on biomedical data, called SapBERT (Liu et al., 2021a). Given an event mention, we encode its arguments using SapBERT, and then concatenate the resulting vectors to obtain the event representation. Note that in some cases, an argument may be a nested event, or an event may have a varying number of arguments. In those cases, we

use vector averaging to compress the representations or padding to fill in the extra argument slots[5].

Sentence-Encoder based Event Embedding. In our second approach, we use templates generated by a LLM. We begin by extracting all event types from the source domain. We then submit each type and its arguments to the LLM, using prompts to construct a template. A few examples of such templates are reported in Table 1, and a larger set of templates along our prompt instruction can be found in Appendix C.2. Then, for the event mentions in the source data, we complete their corresponding templates by replacing the placeholders with the actual arguments. The resulting passages are sent through an off-the-shelf sentence encoder to obtain the final representation vectors. In our experiments, we use ChatGPT (OpenAI, 2022) as the LLM, and use the S-PubMed-BERT (Deka et al., 2022) as the sentence encoder. In § 4, we individually evaluate each one of the methods for extracting the auxiliary embedding vectors.

Auxiliary Similarity. Given $E(x_i)$ and $E(x_j)$ as the sets of auxiliary vectors for the entities x_i and x_j respectively, we define the similarity between the two entities as the maximum inter-similarity between all the vector pairs across the two sets. More specifically, we formulate it as follows:

$$\kappa_{ij} = \max_{\mu \in E(x_i), \nu \in E(x_j)} \frac{\mu^T \cdot \nu}{|\mu| \cdot |\nu|}. \tag{5}$$

The value of κ_{ij} captures the relatedness between the contexts that the two entities appear in. If $E(x_\bullet)$ is empty, then we set $\kappa_{i\bullet}$=0.

Contrastive Grouping. We adapt the multi-similarity loss (Wang et al., 2019) to consider the primary similarity scores between entities, which is the cosine similarity between the encoder outputs for the entities, as well as the auxiliary similarity scores described earlier in this section. Our core idea is to assign a higher weight to the more informative pairs. In the case of positive pairs, this translates into assigning a higher weight to the instances that have a smaller primary similarity and a higher auxiliary similarity. In the case of negative pairs, it is the reverse, i.e., assigning a higher weight to the pairs with higher primary similarity and a lower auxiliary similarity.

[5]The details can be found in Appendix C.1.

The intuition behind these design choices is as follows. In the case of positive pairs, a low primary similarity and a high auxiliary similarity potentially mean that the encoder is unable to properly project the entities, but there is a strong external signal that the pair must have similar representations. In the case of negative pairs, a high primary similarity and a low auxiliary similarity potentially mean that the model needs to revise the parameters to take into account the external signal.

To implement these ideas, we exploit the soft weights discussed in Equations 3 to derive the weights for the positive pairs as follows:

$$\begin{aligned} \hat{w}^+_{ij} &= \frac{1}{e^{-I^+_{ij}} + \sum_{k \in \mathcal{P}_i} e^{-J^+_{ik} + J^+_{ij}}}, \\ I^+_{ij} &= \alpha(\gamma - S_{ij}) + \rho\kappa_{ij}, \\ J^+_{ij} &= \alpha S_{ij} - \rho\kappa_{ij}. \end{aligned} \tag{6}$$

The value of $\hat{w}^+_{ij}$ is the second formulation of the soft weights introduced by Wang et al. (2019). We see that instead of only relying on the values of $S_\bullet$ to define $I^+_\bullet$, we incorporate the auxiliary similarity scores $\kappa_\bullet$ via the hyperparameter ρ.

Similarly, we re-define the soft weights for the negative pairs as follows:

$$\begin{aligned} \hat{w}^-_{ij} &= \frac{1}{e^{I^-_{ij}} + \sum_{k \in \mathcal{N}_i} e^{J^-_{ik} - J^-_{ij}}}, \\ I^-_{ij} &= \beta(\gamma - S_{ij}) + \tau\kappa_{ij}, \\ J^-_{ij} &= \beta S_{ij} - \tau\kappa_{ij}, \end{aligned} \tag{7}$$

where τ is a hyperparameter to balance the contributions of $S_\bullet$ and $\kappa_\bullet$.

Given the re-defined soft weights, the refined multi-similarity objective (RMS) can be re-written as follows:

$$\begin{aligned} \mathcal{L}_{RMS} = \frac{1}{n_e} \sum_{i}^{n_e} \Bigg\{ & \frac{1}{\alpha} \log\Big[1 + \sum_{k \in \mathcal{P}_i} e^{-\alpha(S_{ik} - \gamma) + \rho\kappa_{ik}}\Big] \\ & + \frac{1}{\beta} \log\Big[1 + \sum_{k \in \mathcal{N}_i} e^{\beta(S_{ik} - \gamma) - \tau\kappa_{ik}}\Big] \Bigg\}, \end{aligned} \tag{8}$$

where, as mentioned earlier, ρ and τ are to maintain a balance between the primary and the auxiliary representations. Note that we use the information extracted from events to construct the auxiliary representations. However, additional sources of information can be considered if it is present.

Type	Template
Phosphorylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> is modified by the addition of a phosphate group at a particular site <Site>, facilitated by another molecule <Cause>.
Acetylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> undergoes the addition of an acetyl group at a particular site <Site>, catalyzed by another molecule <Cause>.
Pathway	Indicated by the given trigger <Trigger>, involving one or more molecules <Participant> that collaborate to accomplish a specific biological function or response.

Table 1: Examples of templates for sentence-encoder based embeddings. Angle brackets <·> in templates are placeholders to be replaced by actual entities as corresponding arguments.

The pretraining objective function consists of the supervised named entity recognition term and the unsupervised RMS term, as follows:

$$\mathcal{L} = \mathcal{L}_{NER} + \lambda_{\mathbb{S}} \mathcal{L}_{RMS}, \tag{9}$$

where $\lambda_{\mathbb{S}}$ is a penalty term.

3.2 Target Domain Finetuning

The source and target domains in our problem setting share the same context. In the same documents (or even sentences) that the entities from one domain appear, the entities from the other domain may be used, too. This makes the recognition task particularly challenging in the target domain for two reasons: the training data in this domain is scarce, and the presence of entities from the source domain can potentially lead to a high false positive rate. Our core idea is that, while finetuning the model on the target data, we train the encoder such that it projects the target entities and the entities that potentially belong to the source domain into separate regions of the feature space. To implement this idea, we employ pseudo labeling along the multi-similarity loss–introduced in §2.3.

Pseudo Labeling. Given a passage with annotated target entities in the target training data, we use the model introduced in §3.1 to automatically detect the entities that may belong to the source domain. These entities act as pseudo labels in our algorithm. Note that while there may be passages that do not contain such entities, in general, due to the nature of the two domains that we are studying (i.e., biomedical and chemical domains), this is an expected observation. In the results section, we will also empirically support our argument.

Contrastive Discrimination. In the next step, we enable the model to discriminate between the target and pseudo-labeled source entities. For this purpose, we use the multi-similarity loss. We use multi-similarity loss in Eq. 4 to calculate contrastive objective by defining the labels as follows:

$$y_i = \begin{cases} 0, & x_i \in \mathcal{Q} \\ 1, & x_i \notin \mathcal{Q} \end{cases} \tag{10}$$

where x_i represents the entity and $\mathcal{Q}$ is the set of entities with pseudo labels.

The final target domain fine-tuning objective is:

$$\mathcal{L} = \mathcal{L}_{NER} + \lambda_{\mathbb{T}} \mathcal{L}_{MS}, \tag{11}$$

where $\lambda_{\mathbb{T}}$ is a penalty terms.

4 Experiments

4.1 Experimental Setup

Dataset	# Train/ Valid/ Test	# Train/ Valid/ Test (All)
PC	-	260 / 89 / 175
ID	-	150 / 46 / 117
CG	-	300 / 100 / 200
CHEMDNER	81 / 72 / 2478	2916 / 2907 / 2478
BC5CDR	86 / 88 / 500	500 / 500 / 500
DrugProt	93 / 88 / 600	597 / 597 / 600

Table 2: Overview of the datasets. The top three are the biomedical source datasets, and the bottom three are the chemical target datasets. The target datasets were down-sampled randomly to be used in the few-shot setting.

Datasets. As the source datasets, we use three benchmarks: Pathway Curation (PC), Cancer Genetics (CG), and Infectious Diseases (ID). The first two datasets were released by BioNLP Shared Task 2013 (Nédellec et al., 2013), and the third one was released by BioNLP Shared Task 2011 (Pyysalo et al., 2011). All three datasets have the same format. We aggregate them to create a fourth dataset called the *biomedical multi-task* dataset. As the target datasets, we use three benchmarks: CHEMDNER (Krallinger et al., 2015), BC5CDR (Kim et al., 2015), and DrugPort (Miranda et al., 2021).

Target Tasks	*CHEMDNER*			*BC5CDR*			*DrugProt*		
Metrics	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
Target Only	42.73	51.69	46.77	72.44	85.86	78.51	63.80	67.42	65.49
Direct Transfer	44.11	48.60	46.18	71.92	86.54	78.55	66.36	70.13	68.17
EG(concat)	43.63±1.7	51.06±0.1	47.03±1.0	72.10±1.3	87.12±0.5	78.89±0.6	68.49±0.9	66.54±0.9	67.49±0.2
EG(sentEnc)	43.09±3.7	51.60±3.3	46.91±2.9	73.96±1.1	86.56±0.8	79.76±0.3	66.02±0.5	69.97±0.8	67.93±0.3
ED	**45.71**±2.3	50.13±1.2	47.78±1.3	75.01±2.0	**87.80**±0.5	80.88±0.9	**70.15**±0.3	**71.79**±0.3	**70.96**±0.1
EG(concat)+ED	43.83±0.6	**52.17**±0.9	47.64±0.7	73.70±0.8	85.50±0.9	79.15±0.1	68.01±0.2	69.06±1.3	68.52±0.6
EG(sentEnc)+ED	45.28±0.4	51.68±1.8	**48.26**±0.9	**76.06**±1.8	86.58±1.3	**80.95**±0.5	67.16±0.8	69.04±1.0	68.08±0.9

Table 3: Evaluation results precision, recall, and F1(%) scores on three target tasks with Biomedical Multi-task as source task. All the reported scores are averaged over 3 different random seeds. We include two baselines, along with our methods EG (Entity Grouping), ED (Entity Discrimination), and their combination.

Each dataset contains extra annotations unrelated to its domain. For instance, the CG dataset has annotations not relevant to biomedicine. We preprocess all the datasets by removing such annotations. For few-shot experiments, we down-sample the training and validation sets of target datasets to sizes randomly chosen between 70 and 100. Table 2 summarizes the dataset statistics.

Baselines. We compare our method with two baselines. *Target Only*: a model finetuned on the labeled target data. *Direct Transfer*: a model pretrained on the labeled source data and then finetuned on the labeled target data.

Implementation Details. We use BERT (bert-base-uncased) (Devlin et al., 2019) as the backbone for all the models. To train the model, we update the parameters of the adapter layers (Houlsby et al., 2019) and freeze the rest, due to limited computational resources. We iteratively select each source and target dataset pair as the training and evaluation benchmarks. All the experiments are repeated three times. Following Nakayama (2018), we report average macro Precision, Recall, and F1 scores[6].

4.2 Main Results

Table 3 reports the performance of our model compared to the *target Only* and the *Direct Transfer* models, when the dataset *biomedical multi-task* is used as the source data. All the other use cases are reported in Appendix E. Our final model EG(•)+ED outperforms the baselines in the majority of the cases. We also report the performance of each component of our method in the table—i.e., our entity grouping (EG) and entity discrimination (ED) techniques. We observe that, on average, the method further improves when they are both used in the pipeline. One exception is the DrugProt dataset, which we discuss in the next section.

[6]Training and tuning details can be found in Appendix D.

Target Tasks	*CHEMD*	*BC5CDR*	*DrugProt*
Direct Transfer	46.18	78.55	68.17
Pse-Augment	47.13±0.8	76.43±0.2	68.20±0.8
Pse-Classifier	46.93±1.4	78.80±0.3	69.42±0.1
Ours	**47.78**±1.3	**80.88**±0.9	**70.96**±0.1

Table 4: F1(%) scores of our method compared to two alternative methods for using pseudo-labels. All the reported scores are averaged over 3 different random seeds. Additional experiment details are in Appendix E.3.

4.3 Empirical Analysis

Pseudo Label Usage. Table 4 reports a comparison between our model and two alternative methods. *Pse-Augment*, where the detected pseudo-labels are marked and augmented with the target entities and a classifier trained to label unseen target entities. *Pse-Classifier*, where a classifier is trained to detect pseudo-labels and to filter them out, before being potentially mislabeled. This experiment aims to reveal the efficacy of the multi-similarity (MS) loss for discriminating between the source and the pseudo-labeled entities. The results indicate that our ED method that leverages MS loss is an effective way to use pseudo labels. For *Pse-Augment*, adding source domain entity labels to the target task leads to the negative transfer (NT) problem (Zhang et al., 2020). For *Pse-Classifier*, the pipeline suffers from error propagation, where errors caused by the classifier can severely affect the performance of target entity predictions.

How is the Representation Enhanced? To investigate the effect of our proposed methods, we project entity representations (i.e., averaged hidden states of entity tokens in input texts) into a two-

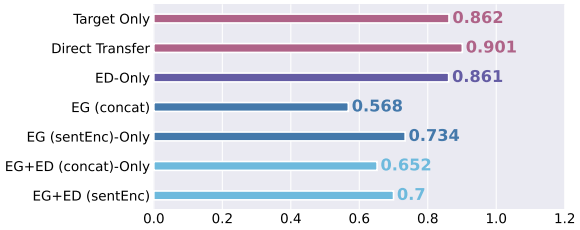

Figure 3: Davies-Bouldin index criterion of clusters. For baseline and ED-concerned settings, pseudo entities are included and viewed in the same cluster as *Disease*.

dimensional space using t-SNE (van der Maaten and Hinton, 2008)[7]. For better comparison, we adopt the Davies-Bouldin index (DB) (Davies and Bouldin, 1979) as the criterion. The lower the DB, the better the clustering of the data points.

With our ED method, the model effectively learns to disperse the representations of chemical entities and pseudo-labeled entities. Therefore, it becomes easier for our model to assign a negative label to a source domain entity by measuring similarities between its representations and the target domain entity representations. Furthermore, our EG method also plays a vital role in the formation of feature space of the target domain. The projection results in clearer clustering of chemical entities, while disease entities are dispersed from them, making it easier to extract chemical entities. For a more precise comparison, Figure 3 reports the DB index. Our methods achieve lower DB compared to the baselines, which indicates that the improved representations of the target domain are learned with our methods.

Method	Precision	Recall	F1
Target Only	41.32±1.9	45.52±2.0	43.31±1.9
Direct Transfer	42.16±1.3	48.51±1.4	45.08±0.6
EG(sentEnc)	45.49±1.0	49.03±1.9	47.16±0.8
ED	45.35±3.2	47.88±3.5	46.58±3.3
EG(sentEnc)+ED	44.94±0.4	49.37±2.0	47.03±0.8

Table 5: Averaged F1 scores (%) over 3 different random seeds for CHEMDNER trained on full BERT model.

Role of Adapters To clarify that the use of adapters does not interfere with the conclusion of our proposed methods, we additionally finetune the full BERT model on the CHEMDNER dataset, and the results are reported in Table 5. The performance of the full model is similar to the performance of adapters or even slightly worse than our adapter models (47.03 vs 48.26). Our methods remain effective when fine-tuned with the full model, demonstrating that the results in the paper are reliable and sufficient.

[7]We show the visualization of target task BC5CDR in Appendix F.

Domain	#Train/Valid/Test	#Train/Valid/Test (All)
Science	28 / 66 / 543	200 / 450/ 543
AI	13 / 46 / 431	100 / 350 / 431

Table 6: Overview of CrossNER dataset.

Target Only	Direct Transfer	EG(sentEnc)	ED
0.59	16.74	19.12	17.18

Table 7: Averaged F1 scores (%) over three different random seeds for the CrossNER dataset, transferring from the Science domain to the AI domain.

Compatibility Across Other Domain Pairs In the above experiments, we focus on transfer learning between the biomedical domain and the chemical domain. To show the generalization ability of our proposed framework on other domain pairs, we conduct the experiments on two additional domains based on CrossNER (Liu et al., 2021b), transferring from the Science domain to the AI domain. These two domains are highly related and share similar named entities. We downsample the train and validation data to roughly 10% of the full dataset for the target domain. Detailed statistics are shown in Table 6. The F1 scores averaged over three runs are reported in Table 7. It shows that our methods have strong generalization ability.

5 Related Work

Biomedical and Chemical Entity Extraction. Entity extraction is a primary step in facilitating scientific discovery (Wang et al., 2021a). Previous biomedical entity extraction methods can be categorized into several classes, including domain adaptive pretraining (Labrak et al., 2023), boundary denoising diffusion model (Shen et al., 2023a), question answering-based classification (Arora and Park, 2023), Cocke-Younger-Kasami (CYK) algorithm (Corro, 2023), in-context learning (Chen et al., 2023), synthetic data (Khandelwal et al., 2022; Chen et al., 2022; Hiebel et al., 2023), and prototype learning (Cao et al., 2023).

Although there's a shared corpus between the biomedical and chemical domains, entity extrac-

tion in the chemical domain remains underexplored. The chemical entity extraction task is usually viewed as an auxiliary task for biomedical named entity recognition (Phan et al., 2021; Kocaman and Talby, 2021; Luo et al., 2022; Lee et al., 2023). Similar to Nguyen et al. (2022) and Wang et al. (2024), our paper differs from previous papers by viewing the chemical domain as an independent subject. Previous methods try to address this task by distant-supervision (Wang et al., 2021c) or span-representation learning (Nguyen et al., 2023). On the contrary, given the shared corpus between the biomedical and chemical domains, we leverage the large labeled data in the biomedical domain through transfer learning.

Transfer Learning for Named Entity Recognition. Transfer learning is an effective method to address low-resource named entity recognition tasks (Lee et al., 2018; Cao et al., 2018) and has shown its effectiveness in the biomedical domain (Peng et al., 2019). Prior work has explored the role of continual pretraining on the target domain data (Gururangan et al., 2020; Liu et al., 2021b). However, Mahapatra et al. (2022) argues that continual pretraining is inefficient regarding computational resources. In contrast to domain-adaptive pretraining, we aim to improve the representation of entities by projecting the source and target entities into separate regions of feature space. Inspired by the success of incorporating external knowledge for biomedical information extraction (Zhang et al., 2021; Banerjee et al., 2021), we use biomedical events to augment the representations. To separate the potentially false positive examples in the target domain, we introduce pseudo-labels. Previous papers have adopted pseudo-labels in cross-lingual named entity recognition (Zhou et al., 2023; Ma et al., 2023). However, they aim to align the entities in the source and target language instead of separating source entities from target entities. Compared to our method, Zhou et al. (2023) has a different contrastive objective, which aims to separate different entity types, rather than the entities from the two domains.

6 Conclusions

We proposed a named entity recognition task for transferring knowledge from the biomedical domain to the chemical domain. Our core idea is to train a shared feature space between the two domains to facilitate the knowledge transfer, and, at the same time, to project the source and target entities into separate regions of the feature space to reduce the false negative rate. We achieve this in a few steps. We begin by enriching the source feature space with information about events, then train a named entity recognition model to cluster similar entities into groups. We then use the trained model to label the entities that may belong to the source domain, and use these entities in a multi similarity loss function to achieve our goal. Our experiments across three source and three target datasets signify the effectiveness of our method.

7 Limitations

Our method partly relies on external knowledge. Therefore, the quality of external knowledge significantly influences the effect of our method. Especially when human annotations are unavailable, the performance of automatic annotators, typically neural networks, is an important factor to consider.

In this paper, we propose a framework that incorporates external knowledge during training. For instance, the compression function in §3.1 and templates in Section C.2 can be altered. Besides, such designs require prior knowledge of the source domain.

Our method leverages a contrastive learning strategy. However, the training algorithm doesn't fully utilize GPU resources, leading to training inefficiencies.

Acknowledgements

This work is supported by U.S. DARPA ITM FA8650-23-C-7316, by the Molecule Maker Lab Institute: an AI research institute program supported by NSF under award No. 2019897, by DOE Center for Advanced Bioenergy and Bioproducts Innovation U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research under Award Number DESC0018420, by U.S. the AI Research Institutes program by National Science Foundation and the Institute of Education Sciences, Department of Education through Award No. 2229873 - AI Institute for Transforming Education for Children with Speech and Language Processing Challenges, and by AI Agriculture: the Agriculture and Food Research Initiative (AFRI) grant no. 2020-67021- 32799/project accession no.1024178 from the USDA National Institute of Food and Agriculture. The views and conclusions contained herein are those of the authors and should

not be interpreted as necessarily representing the official policies, either expressed or implied of, the National Science Foundation, the U.S. Department of Energy, and the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

Jatin Arora and Youngja Park. 2023. Split-NER: Named entity recognition via two question-answering-based classifications. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 416–426, Toronto, Canada. Association for Computational Linguistics.

Pratyay Banerjee, Kuntal Kumar Pal, Murthy Devarakonda, and Chitta Baral. 2021. Biomedical named entity recognition via knowledge guidance and question answering. *ACM Trans. Comput. Healthcare*, 2(4).

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Machine learning*, 79(1-2):151–175.

Jiarun Cao, Niels Peek, Andrew Renehan, and Sophia Ananiadou. 2023. Gaussian distributed prototypical network for few-shot genomic variant detection. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 26–36, Toronto, Canada. Association for Computational Linguistics.

Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2018. Adversarial transfer learning for Chinese named entity recognition with self-attention mechanism. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 182–192, Brussels, Belgium. Association for Computational Linguistics.

Jiawei Chen, Yaojie Lu, Hongyu Lin, Jie Lou, Wei Jia, Dai Dai, Hua Wu, Boxi Cao, Xianpei Han, and Le Sun. 2023. Learning in-context learning for named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13661–13675, Toronto, Canada. Association for Computational Linguistics.

Shuguang Chen, Leonardo Neves, and Thamar Solorio. 2022. Style transfer as data augmentation: A case study on named entity recognition. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1827–1841, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Caio Corro. 2023. A dynamic programming algorithm for span-based nested named-entity recognition in $o(n^2)$. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10712–10724, Toronto, Canada. Association for Computational Linguistics.

David L. Davies and Donald W. Bouldin. 1979. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227.

Pritam Deka, Anna Jurek-Loughrey, and P Deepak. 2022. Improved methods to aid unsupervised evidence-based fact checking for online health news. *Journal of Data Intelligence*, 3(4):474–504.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark. Association for Computational Linguistics.

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.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Nicolas Hiebel, Olivier Ferret, Karen Fort, and Aurélie Névéol. 2023. Can synthetic text help clinical named entity recognition? a study of electronic health records in French. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2320–2338, Dubrovnik, Croatia. Association for Computational Linguistics.

Judy Hoffman, Brian Kulis, Trevor Darrell, and Kate Saenko. 2012. Discovering latent domains for multisource domain adaptation. In *Computer Vision – ECCV 2012*, pages 702–715, Berlin, Heidelberg. Springer Berlin Heidelberg.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. *Computation and Language Repository*, arXiv:1902.00751.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2023. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 15696–15707. PMLR.

Anshita Khandelwal, Alok Kar, Veera Raghavendra Chikka, and Kamalakar Karlapalem. 2022. Biomedical NER using novel schema and distant supervision. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 155–160, Dublin, Ireland. Association for Computational Linguistics.

Sun Kim, Rezarta Islamaj Dogan, Andrew Chatr-Aryamontri, Mike Tyers, W John Wilbur, and Donald C Comeau. 2015. Overview of biocreative v bioc track. In *Proceedings of the Fifth BioCreative Challenge Evaluation Workshop, Sevilla, Spain*, pages 1–9.

Veysel Kocaman and David Talby. 2021. Biomedical named entity recognition at scale. In *Pattern Recognition. ICPR International Workshops and Challenges*, pages 635–646, Cham. Springer International Publishing.

Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M Lowe, et al. 2015. The chemdner corpus of chemicals and drugs and its annotation principles. *Journal of cheminformatics*, 7(1):1–17.

Yanis Labrak, Adrien Bazoge, Richard Dufour, Mickael Rouvier, Emmanuel Morin, Béatrice Daille, and Pierre-Antoine Gourraud. 2023. DrBERT: A robust pre-trained model in French for biomedical and clinical domains. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16207–16221, Toronto, Canada. Association for Computational Linguistics.

Md Tahmid Rahman Laskar, M Saiful Bari, Mizanur Rahman, Md Amran Hossen Bhuiyan, Shafiq Joty, and Jimmy Huang. 2023. A systematic study and comprehensive evaluation of ChatGPT on benchmark datasets. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 431–469, Toronto, Canada. Association for Computational Linguistics.

Dong-Ho Lee, Ravi Kiran Selvam, Sheikh Muhammad Sarwar, Bill Yuchen Lin, Fred Morstatter, Jay Pujara, Elizabeth Boschee, James Allan, and Xiang Ren. 2023. AutoTriggER: Label-efficient and robust named entity recognition with auxiliary trigger extraction. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3011–3025, Dubrovnik, Croatia. Association for Computational Linguistics.

Ji Young Lee, Franck Dernoncourt, and Peter Szolovits. 2018. Transfer learning for named-entity recognition with neural networks. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online. Association for Computational Linguistics.

Fangyu Liu, Ehsan Shareghi, Zaiqiao Meng, Marco Basaldella, and Nigel Collier. 2021a. Self-alignment pretraining for biomedical entity representations. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4228–4238, Online. Association for Computational Linguistics.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2021b. Crossner: Evaluating cross-domain named entity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13452–13460.

Ling Luo, Po-Ting Lai, Chih-Hsuan Wei, Cecilia N Arighi, and Zhiyong Lu. 2022. BioRED: a rich biomedical relation extraction dataset. *Briefings in Bioinformatics*, 23(5):bbac282.

Tingting Ma, Qianhui Wu, Huiqiang Jiang, Börje Karlsson, Tiejun Zhao, and Chin-Yew Lin. 2023. CoLaDa: A collaborative label denoising framework for cross-lingual named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5995–6009, Toronto, Canada. Association for Computational Linguistics.

Aniruddha Mahapatra, Sharmila Reddy Nangi, Aparna Garimella, and Anandhavelu N. 2022. Entity extraction in low resource domains with selective pre-training of large language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 942–951, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Antonio Miranda, Farrokh Mehryary, Jouni Luoma, Sampo Pyysalo, Alfonso Valencia, and Martin Krallinger. 2021. Overview of drugprot biocreative vii track: quality evaluation and large scale text mining of drug-gene/protein relations. In *Proceedings of the seventh BioCreative challenge evaluation workshop*, pages 11–21.

Hiroki Nakayama. 2018. seqeval: A python framework for sequence labeling evaluation. Software available from https://github.com/chakki-works/seqeval.

Claire Nédellec, Robert Bossy, Jin-Dong Kim, Jung-jae Kim, Tomoko Ohta, Sampo Pyysalo, and Pierre

Zweigenbaum. 2013. Overview of BioNLP shared task 2013. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 1–7, Sofia, Bulgaria. Association for Computational Linguistics.

Ngoc Dang Nguyen, Lan Du, Wray Buntine, Changyou Chen, and Richard Beare. 2022. Hardness-guided domain adaptation to recognise biomedical named entities under low-resource scenarios. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4063–4071, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Nhung T. H. Nguyen, Makoto Miwa, and Sophia Ananiadou. 2023. Span-based named entity recognition by generating and compressing information. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1984–1996, Dubrovnik, Croatia. Association for Computational Linguistics.

OpenAI. 2022. Introducing chatgpt.

Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.*, 22(10):1345–1359.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020. AdapterHub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 46–54, Online. Association for Computational Linguistics.

Long N Phan, James T Anibal, Hieu Tran, Shaurya Chanana, Erol Bahadroglu, Alec Peltekian, and Grégoire Altan-Bonnet. 2021. Scifive: a text-to-text transformer model for biomedical literature. *Computation and Language Repository*, arXiv:2106.03598.

Sampo Pyysalo, Tomoko Ohta, Rafal Rak, Dan Sullivan, Chunhong Mao, Chunxia Wang, Bruno Sobral, Jun'ichi Tsujii, and Sophia Ananiadou. 2011. Overview of the infectious diseases (ID) task of BioNLP shared task 2011. In *Proceedings of BioNLP Shared Task 2011 Workshop*, pages 26–35, Portland, Oregon, USA. Association for Computational Linguistics.

Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. 2019. Semi-supervised domain adaptation via minimax entropy. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 8049–8057. IEEE.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023a. DiffusionNER: Boundary diffusion for named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3875–3890, Toronto, Canada. Association for Computational Linguistics.

Yongliang Shen, Zeqi Tan, Shuhui Wu, Wenqi Zhang, Rongsheng Zhang, Yadong Xi, Weiming Lu, and Yueting Zhuang. 2023b. PromptNER: Prompt locating and typing for named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12492–12507, Toronto, Canada. Association for Computational Linguistics.

Hai-Long Trieu, Thy Thy Tran, Khoa N A Duong, Anh Nguyen, Makoto Miwa, and Sophia Ananiadou. 2020. DeepEventMine: end-to-end neural nested event extraction from biomedical texts. *Bioinformatics*, 36(19):4910–4917.

Laurens van der Maaten and Geoffrey E. Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

Qingyun Wang, Manling Li, Xuan Wang, Nikolaus Parulian, Guangxing Han, Jiawei Ma, Jingxuan Tu, Ying Lin, Ranran Haoran Zhang, Weili Liu, Aabhas Chauhan, Yingjun Guan, Bangzheng Li, Ruisong Li, Xiangchen Song, Yi Fung, Heng Ji, Jiawei Han, Shih-Fu Chang, James Pustejovsky, Jasmine Rah, David Liem, Ahmed ELsayed, Martha Palmer, Clare Voss, Cynthia Schneider, and Boyan Onyshkevych. 2021a. COVID-19 literature knowledge graph construction and drug repurposing report generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations*, pages 66–77, Online. Association for Computational Linguistics.

Qingyun Wang, Zixuan Zhang, Hongxiang Li, Xuan Liu, Jiawei Han, Huimin Zhao, and Heng Ji. 2024. Chem-FINESE: Validating fine-grained few-shot entity extraction through text reconstruction. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1–16, St. Julian's, Malta. Association for Computational Linguistics.

Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2021b. Automated concatenation of embeddings for structured prediction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2643–2660, Online. Association for Computational Linguistics.

Xuan Wang, Vivian Hu, Xiangchen Song, Shweta Garg, Jinfeng Xiao, and Jiawei Han. 2021c. ChemNER: Fine-grained chemistry named entity recognition

with ontology-guided distant supervision. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5227–5240, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. 2019. Multi-similarity loss with general pair weighting for deep metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5022–5030.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Sheng Zhang, Hao Cheng, Jianfeng Gao, and Hoifung Poon. 2023. Optimizing bi-encoder for named entity recognition via contrastive learning. In *The Eleventh International Conference on Learning Representations*.

Wen Zhang, Lingfei Deng, and Dongrui Wu. 2020. Overcoming negative transfer: A survey. *ArXiv*, abs/2009.00909.

Zixuan Zhang, Nikolaus Parulian, Heng Ji, Ahmed Elsayed, Skatje Myers, and Martha Palmer. 2021. Fine-grained information extraction from biomedical literature based on knowledge-enriched Abstract Meaning Representation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6261–6270, Online. Association for Computational Linguistics.

Ran Zhou, Xin Li, Lidong Bing, Erik Cambria, and Chunyan Miao. 2023. Improving self-training for cross-lingual named entity recognition with contrastive and prototype learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4018–4031, Toronto, Canada. Association for Computational Linguistics.

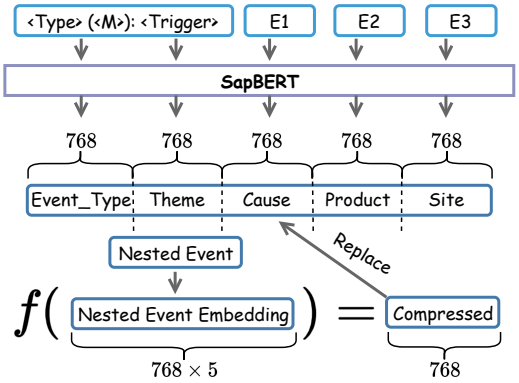

Figure 4: Components of concatenation based event embeddings. Arguments of events, along with event type, are encoded by an off-the-shelf model and concatenated afterwards. For nested events as arguments, we fill in compressed event embeddings recursively.

A Details of ChatGPT evaluation on CHEMDNER

To evaluate ChatGPT's few-shot ability of named entity recognition in the chemical domain, we elaborately select three named entity recognition examples in the CHEMDNER dataset that include all types of entities. An instructional description of the CHEMDNER task and examples constitute the prompt used to instruct the prediction. Configuration of prompts is shown in Table 8. The precision, recall, and macro-F1 scores on the CHEMDNER test set are 11.09%, 35.37%, and 16.88% respectively. Laskar et al. (2023) reports the ChatGPT's named entity recognition performance on general domain named entity recognition task-WNUT 17 dataset (Derczynski et al., 2017) with precision: 18.03%, recall: 56.16%, and F1: 27.03%. This highlights ChatGPT's weaker performance in solving chemical domain named entity recognition tasks.

B Notation Table

We present definitions for all notations we used in Table 9.

C Details of Event Embedding Construction.

C.1 Concatenation based Event Embedding

An overview of concatenation based event embedding strategy is shown in Figure 4. Each type of annotated event contains a trigger and various arguments, including theme, cause, product, and site. We prepare raw texts of each argument and encode

Role	Prompt
System	You are an expert of chemical named entity recognition tasks
User	Description: In this task, you are given a small paragraph of a PubMed article, and your task is to identify all the named entities (particular chemical related entity) from the given input and also provide type of the each entity according to structure-associated chemical entity mention classes (ABBREVIATION, IDENTIFIER, FORMULA, SYSTEMATIC, MULTIPLE, TRIVIAL, FAMILY). Specifically, the paragraph are given with seperate tokens and you need to list all the chemical named entities in order and also tag their types. Generate the output in this format: entity1 <type_of_entity1>, entity2 <type_of_entity2>. Examples: Input: In situ C-C bond cleavage of vicinal diol following by the lactolisation resulted from separated treatment of Arjunolic acid (1) , 24-hydroxytormentic acid (2) and 3-O-β-D-glucopyranosylsitosterol (3) with sodium periodate and silica gel in dried THF according to the strategic position of hydroxyl functions in the molecule . Output: C-C <FORMULA>, vicinal diol <FAMILY>, Arjunolic acid <TRIVIAL>, 24-hydroxytormentic acid <SYSTEMATIC>, 3-O-β-D-glucopyranosylsitosterol <SYSTEMATIC>, sodium periodate <SYSTEMATIC>, silica gel <TRIVIAL>, THF <ABBREVIATION>, hydroxyl <SYSTEMATIC> Input: Structural studies using LC/MS/MS analysis and (1) H NMR spectroscopy showed the formation of a glycosidic bond between the primary hydroxyl group of RVX-208 and glucuronic acid . Output: (1) H <FORMULA>, primary hydroxyl <FAMILY>, RVX-208 <IDENTIFIER>, glucuronic acid <TRIVIAL> Input: The lystabactins are composed of serine (Ser) , asparagine (Asn) , two formylated/hydroxylated ornithines (FOHOrn) , dihydroxy benzoic acid (Dhb) , and a very unusual nonproteinogenic amino acid , 4,8-diamino-3-hydroxyoctanoic acid (LySta) . Output: lystabactins <FAMILY>, serine <TRIVIAL>, Ser <FORMULA>, asparagine <TRIVIAL>, Asn <FORMULA>, formylated/hydroxylated ornithines <MULTIPLE>, FOHOrn <ABBREVIATION>, dihydroxy benzoic acid <SYSTEMATIC>, Dhb <ABBREVIATION>, 4,8-diamino-3-hydroxyoctanoic acid <SYSTEMATIC>, LySta <ABBREVIATION> Please continue: Input: %s Output:

Table 8: Prompts of ChatGPT evaluation. %s is to be replaced by test context.

Notation	Definition
$\mathbf{X}$	Set of input documents
$\mathbf{Y}$	Set of named entity tags
n_e	Total number of entities
x_i	The i-th entity
y_i	Label of the i-th entity
f	Classifier model
$\mathbb{S}$	Source domain
$\mathbb{T}$	Target domain
S_*	Relative similarity
$E(*)$	Set of embedding vectors
κ_*	Auxiliary similarity
$\mathcal{P}_i$	Set of selected positive entities
$\mathcal{N}_i$	Set of selected negative entities
$\mathcal{Q}$	Set of entities with pseudo labels
I^+, J^+	Auxiliary term for positive pairs
I^-, J^-	Auxiliary term for negative pairs
w^+	Soft weight of positive pairs
w^-	Soft weight of negative pairs
$\hat{w}^+_{ij}$	Adjusted soft weight of positive pairs
$\hat{w}^-_{ij}$	Adjusted soft weight of negative pairs
$\mathcal{L}_{NER}$	Classification loss for NER task
$\mathcal{L}_{MS}$	Multi-similarity loss
$\mathcal{L}_{RMS}$	Refined multi-similarity loss
θ	Classifier parameter
ϵ	Margin penalty
$\alpha, \beta, \gamma, \rho, \tau$	Hyperparameters
$\lambda_{\mathbb{S}}, \lambda_{\mathbb{T}}$	Balance term for losses

Table 9: Notation Table

them into argument embeddings. To illustrate the formation of raw texts, let's consider an example. Imagine a gene named "IL-1ra" which is associated with two event annotations, "M1, Negation, E9" and "E9, Binding:forms a complex, Theme:IL-1ra, Theme2:Type I IL-1R". Raw text for "event_type" comprises the event name, "M" label, and trigger. The example's raw text should be "Binding (Negation): forms a complex". Raw text typically is the corresponding entity itself for the rest of the arguments. However, for the focusing entity, "IL-1ra" for instance, raw text is specified as "self", deriving "IL-1ra (self)" in this case. There are several corner cases to tackle with:

Nested Event. As mentioned in §3.1, for the nested event, we first recursively compose the event embedding for it and compress it to the same length as partial embedding. To achieve this, let's consider the nested event embedding as e. We then implement the compression function by averaging several successive elements:

$$f(e) = \left[\frac{1}{5}\sum_{k=0}^{4} e_k, \frac{1}{5}\sum_{k=5}^{9} e_k, ..., \frac{1}{5}\sum_{k=5i}^{5i+4} e_k, ... \right]^T, \tag{12}$$

Full embedding has 768×5 dimensions, and we average every 5 element and concatenate the values into a 768-dimension embedding.

Padding It is necessary that some arguments do not apply to some events or miss in annotation. A padding scheme is necessary for missing arguments, and we choose to fill in random partial embedding with the same length sampled from Gaussian distribution with the same mean and covariance as all other encoded partial embeddings.

C.2 Sentence-Encoder based Event Embedding

The prompt we use to instruct ChatGPT to generate explanatory templates for events is: *give a one-sentence definition of biomedical event type XXX with arguments XXX, XXX...*. For instance, we generate a template for the Phosphorylation event with *give a one-sentence definition of biomedical event type Phosphorylation with arguments Trigger, Theme:Molecule, Cause:Molecule, Site:Simple chemical*. The full list of the used templates is shown in Table 14, 15, 16, 17, and 18.

D Experiment Details

We select *bert-base-uncased* version of BERT model as our backbone model, and we only train 0.817% of parameters (894,528) of the entire model using transformer-adapter utils (Pfeiffer et al., 2020). For source task pretraining, we use a batch size of 64; while for target task fine-tuning, we use a batch size of 16, considering the relatively small training set. We use AdamW optimizer and an initial learning rate of $1e-4$ for pretraining and finetuning. To fully train our model, we first train 80 epochs and then stop when f1 scores on the held-out validation set fail to update the best score for at least 20 epochs in a row. For hyperparameters, we tune balance factor $\lambda_{\mathbb{S}}$ from scale $\{0.10, 0.15, 0.20, 0.25, 0.30\}$ and tune $\lambda_{\mathbb{T}}$ from scale $\{0.6, 0.8, 1.0, 1.2, 1.4\}$. Due to the limitation of computational resources, we select ϵ and γ as 0.1 and 0.5 respectively, and α, β, ρ, τ as 4.0, 3.0, 8.0, 6.0 respectively considering the ratio of number of positive and negative pairs in contrastive learning.

Our models are trained on 4 Nvidia RTX 2080Ti GPUs in a data parallel fashion. Source task pretraining with contrastive learning takes around 5 hours, while target task finetuning with contrastive learning takes around 30 minutes.

E Full Experiment Results

Full evaluation results are reported in Table 13.

Target Tasks	*CHEMD*	*BC5CDR*	*DrugProt*
Direct Transfer	46.18	78.55	**68.17**
EG(MS)	46.08	78.87	68.00
EG(concat)	**47.03**	78.89	67.49
EG(sentEnc)	46.91	**79.76**	67.93

Table 10: F1(%) scores on three target tasks. Performance of our EG method using vanilla MS loss without external knowledge is reported as *EG(MS)*. All the reported scores are averaged over 3 different random seeds.

E.1 Generalization Ability

Event Annotator	Target Tasks	*CHEMD*	*BC5CDR*	*DrugProt*
Gold-std	Concat	47.03	78.89	67.49
	SentEnc	46.91	79.76	67.93
Auto-sys	Concat	46.06	79.49	68.31
	SentEnc	46.11	79.39	68.04

Table 11: F1(%) scores of our proposed EG methods based on human/machine annotated events. We highlight better scores between Gold-std and Auto-sys annotators under each setting with underlines. All the reported scores are averaged over 3 different random seeds.

To alleviate the reliance on gold-standard event annotations, which may be hard to obtain, we generate the event annotations using DeepEventMine (Trieu et al., 2020). Table 11 reports the performance of the EG methods. We see that the performance is comparable to that of the gold-standard annotations. We also observe that in the DrugProt dataset, the performance with automatic annotations is better than that of the gold-standard annotations, suggesting that the human annotations are low quality and noisy.

E.2 External Knowledge

Table 10 reports the performance of our EG method without external knowledge (i.e., event annotations), where simple MS loss replaces our RMS loss. The performance over three target tasks mirrors the *Direct Transfer* setting, suggesting that the vanilla MS objective has minimal impact and the main improvement stems from the auxiliary data extracted from the event mentions.

E.3 Pseudo Label Usage

Pse-Augment We augment annotations of target tasks with pseudo entities within the target corpus labeled as "Out of Distribution (OOD)" entities. Then, the model is trained with a single cross-entropy loss.

Pse-Classifer We first use a pretrained model as a classifier separating pseudo entities and gold-standard annotated entities. We then predict with the former directly finetuned model and filter out entities labeled "OOD" by the classifier.

Dataset	Precision	Recall	F1
Few-shot	42.73	51.69	46.77
Oracle	**73.56**	**80.03**	**76.66**

Table 12: Comparison of the target-only results for different training set sizes in CHEMDNER.

E.4 Size of Target Set

Since the training documents for the target domain are downsampled to roughly 10% of the original corpus, we compare our downsampled training results (few-shot) with those trained on the entire target set (Oracle). We finetune the BERT model with adapters on the full target dataset CHEMDNER in Table 12. The large gap indicates that the limited training data indeed severely hinders the model's learning of the task and justifies the need for transferring knowledge from a high-resource source domain.

F Visualization

The t-SNE projection visualization of BC5CDR test entity representations is shown in Figure 5.

G Scientific Artifacts

We list the license used in this paper: PathwayCuration (CC BY-SA 3.0), InfectiousDisease (CC BY-SA 3.0), CancerGenetics (CC BY-SA 3.0), CHEMDNER (CC BY 4.0), BC5CDR (CC BY 4.0), DrugProt (CC BY 4.0), Huggingface Transformers (Apache License 2.0), SapBERT (apache-2.0), S-PubMedBert-MS-MARCO-SCIFACT (apache-2.0), OpenAI (Terms of use[8]). We follow the intended use of all the mentioned existing artifacts in this paper.

[8] openai.com/policies/terms-of-use

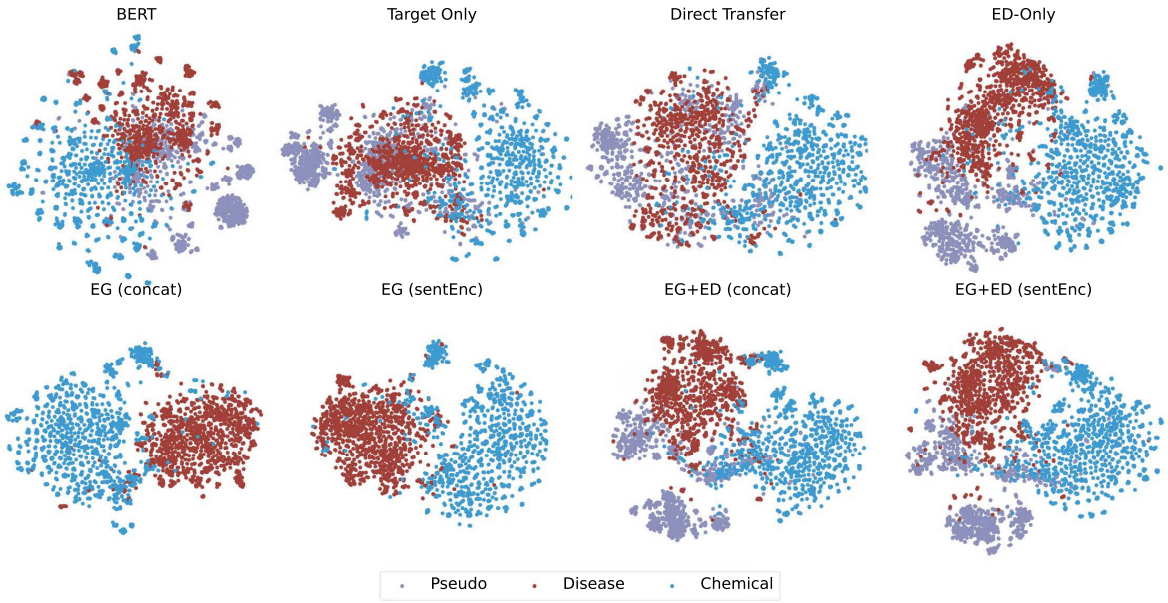

Figure 5: t-SNE visualization of entities in the test corpus of BC5CDR. *Pseudo* is labeled by model pretrained on source task, *Disease* and *Chemical* are gold-standard annotations. *BERT* represents vanilla BERT model without pretraining or finetuning, and all the settings are same as Main Results.

Evaluate Datasets	***CHEMDNER***			***BC5CDR***			***DrugProt***		
Metrics	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
Few-shot (BERT)	42.73	51.69	46.77	72.44	85.86	78.51	63.80	67.42	65.49
Pathway Curation									
Direct Transfer	42.51	49.70	45.82	66.79	**88.79**	76.22	60.94	**68.70**	64.52
EG(concat)	44.75	51.78	48.00	71.49	87.04	78.45	64.45	66.39	65.31
EG(sentEnc)	42.80	50.43	46.25	70.91	86.97	78.09	64.36	67.81	66.03
ED	**46.97**	**52.30**	**49.48**	74.71	83.96	79.06	**67.05**	67.13	67.08
EG(concat)+ED	43.44	51.54	47.11	72.34	86.09	78.61	66.59	66.13	66.32
EG(sentEnc)+ED	43.34	51.73	47.16	**75.66**	86.14	**80.55**	66.15	68.32	**67.21**
Infectious Diseases									
Direct Transfer	41.57	48.30	44.65	**74.92**	82.51	78.53	61.96	67.81	64.75
EG(concat)	45.26	50.65	47.77	72.28	**87.41**	79.12	63.76	65.25	64.43
EG(sentEnc)	**47.33**	50.41	**48.80**	74.58	86.37	**80.04**	64.83	63.60	64.20
ED	41.07	46.93	43.78	74.60	85.85	79.80	64.25	**69.75**	66.63
EG(concat)+ED	43.37	50.26	46.43	73.83	85.48	79.23	62.31	66.87	64.47
EG(sentEnc)+ED	41.50	**52.19**	46.18	74.70	85.95	79.93	**65.06**	69.11	**67.02**
Cancer Genetics									
Direct Transfer	45.22	51.80	48.27	72.37	86.56	78.82	62.94	69.61	66.07
EG(concat)	40.59	**53.19**	46.01	71.44	**87.22**	78.53	65.31	66.67	65.98
EG(sentEnc)	41.54	52.36	46.33	72.68	86.09	78.82	**66.99**	65.71	66.30
ED	**47.06**	50.58	**48.75**	**75.08**	86.96	**80.57**	66.26	**73.52**	**69.68**
EG(concat)+ED	45.16	51.72	48.16	73.37	85.06	78.76	65.60	67.67	66.64
EG(sentEnc)+ED	41.89	50.17	45.61	73.81	86.03	79.45	65.59	68.47	66.99

Table 13: Full evaluation results. Pathway Curation, Infectious Diseases and Cancer Genetics are set to be the source domain respectively. Experiment settings are the same as main results reported in Table 3. All the reported scores are averaged over 3 different random seeds.

Type	Template
Conversion	A specific trigger <Trigger> causes the transformation of a molecule <Theme> into another molecule <Product>.
Phosphorylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> is modified by the addition of a phosphate group at a particular site <Site>, facilitated by another molecule <Cause>.
Dephosphorylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> has a phosphate group removed from a particular site <Site>, facilitated by another molecule <Cause>.
Acetylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> undergoes the addition of an acetyl group at a particular site <Site>, catalyzed by another molecule <Cause>.
Deacetylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> has an acetyl group removed from a particular site <Site>, facilitated by another molecule <Cause>.
Ubiquitination	Indicated by the given trigger <Trigger>, a specific molecule <Theme> is modified by the attachment of one or more ubiquitin molecules at a particular site, facilitated by another molecule <Cause>, often involving a simple chemical group <Site> as the site of attachment.
Deubiquitination	Indicated by the given trigger <Trigger>, a specific molecule <Theme> has ubiquitin molecules removed from a particular site, facilitated by another molecule <Cause> involving a simple chemical group <Site> as the site of removal.
Hydroxylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> undergoes the addition of a hydroxyl group at a particular site, catalyzed by another molecule <Cause> involving a simple chemical group <Site> as the site of attachment.
Dehydroxylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> has a hydroxyl group removed from a particular site, facilitated by another molecule <Cause> involving a simple chemical group <Site> as the site of removal.
Methylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> undergoes the addition of a methyl group at a particular site, facilitated by another molecule <Cause> involving a simple chemical group <Site> as the site of attachment.
Demethylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> has a methyl group removed from a particular site, facilitated by another molecule <Cause> involving a simple chemical group <Site> as the site of removal.
Localization	Indicated by the given trigger <Trigger>, a specific molecule <Theme> is directed to or away from a particular cellular component <AtLoc><FromLoc><ToLoc> or subcellular location within the cell.
Transport	Indicated by the given trigger <Trigger>, a specific molecule <Theme> is moved or conveyed to or away from a particular cellular component <FromLoc><ToLoc> or subcellular location within the cell.
Gene Expression	Indicated by the given trigger <Trigger>, genetic information from a gene <Theme> is used to produce a functional gene product, such as RNA or protein.
Transcription	Indicated by the given trigger <Trigger>, genetic information from a gene <Theme> is transcribed into RNA, usually messenger RNA (mRNA), by RNA polymerase.
Translation	Indicated by the given trigger <Trigger>, the genetic information carried by mRNA <Theme> is used to synthesize a protein by ribosomes in the cell.
Degradation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> undergoes breakdown or degradation into smaller components.
Binding	Indicated by the given trigger <Trigger>, a specific molecule <Theme> interacts and forms a complex with another molecule(s) resulting in the product of a molecular complex <Product>.

Table 14: Templates for PathwayCuration Dataset.

Type	Template
Binding	Indicated by the given trigger <Trigger>, a specific molecule <Theme> interacts and forms a complex with another molecule(s) resulting in the product of a molecular complex <Product>.
Dissociation	Indicated by the given trigger <Trigger>, a specific complex <Theme> breaks apart, resulting in the release of individual molecules <Product> as products.
Regulation	Indicated by the given trigger <Trigger>, an entity <Theme> is controlled or influenced by another entity <Cause> to achieve a specific biological effect or outcome.
Positive Regulation	Indicated by the given trigger <Trigger>, an entity <Theme> is promoted or enhanced by another entity <Cause> to achieve a specific biological effect or outcome.
Activation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> is stimulated or facilitated by another entity <Cause> to increase its activity, function, or biological effect.
Negative Regulation	Indicated by the given trigger <Trigger>, an entity <Theme> is inhibited or suppressed by another entity <Cause> to achieve a specific biological effect or outcome.
Inactivation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> is deactivated or rendered inactive by another entity <Cause>, leading to a reduction or cessation of its biological function.
Pathway	Indicated by the given trigger <Trigger>, involving one or more molecules <Participant> that collaborate to accomplish a specific biological function or response.

Table 15: Continuation of templates for PathwayCuration Dataset.

Type	Template
Gene Expression	Indicated by the given trigger <Trigger>, a specific protein or a group of genes <Theme> are involved in the transcription and translation of genetic information to produce functional gene products, such as RNA or protein.
Transcription	Indicated by the given trigger <Trigger>, a specific protein or a group of genes <Theme> are involved in the synthesis of RNA from DNA template by RNA polymerase.
Protein Catabolism	Indicated by the given trigger <Trigger>, a specific protein <Theme> is broken down or degraded into smaller peptide fragments or amino acids.
Phosphorylation	Indicated by the given trigger <Trigger>, a specific protein <Theme> undergoes the addition of a phosphate group at a particular site <Site>, resulting in the modification of the protein's structure and function.
Localization	Indicated by the given trigger <Trigger>, a specific core entity <Theme> is directed to or away one location <AtLoc><ToLoc> within the cell or organism.
Binding	Indicated by the given trigger <Trigger>, a specific core entity <Theme> interacts and forms a connection with another entity <Site>, leading to the formation of a complex or association.
Regulation	Indicated by the given trigger <Trigger>, a specific core entity or event <Theme> is controlled or influenced by another core entity or event <Cause> through interactions at specific sites on molecules or entities <Site><CSite>, potentially resulting in modulation of biological processes.
Positive Regulation	Indicated by the given trigger <Trigger>, a specific core entity or event <Theme> is promoted or enhanced by another core entity or event <Cause> through interactions at specific sites on molecules or entities <Site><CSite>, potentially resulting in an increase in the intensity or rate of a biological process.
Negative Regulation	Indicated by the given trigger <Trigger>, a specific core entity or event <Theme> is inhibited or suppressed by another core entity or event <Cause> through interactions at specific sites on molecules or entities <Site><CSite>, potentially resulting in a decrease in the intensity or rate of a biological process.
Process	Indicated by the given trigger <Trigger>, involving a core entity that collaborates to accomplish a specific biological function or response.

Table 16: Templates for Infectious Diseases Dataset.

Type	Template
Development	Indicated by the given trigger <Trigger>, a specific anatomical or pathological entity <Theme> undergoes progressive changes or growth, leading to the formation of a more complex and specialized structure or condition over time.
Blood Vessel Development	Indicated by the given trigger <Trigger>, blood vessels <Theme> undergo progressive changes or growth at a specific location <AtLoc>, leading to the formation and maturation of the vascular network.
Growth	Indicated by the given trigger <Trigger>, a specific anatomical or pathological entity <Theme> undergoes an increase in size, quantity, or complexity over time.
Death	Indicated by the given trigger <Trigger>, a specific anatomical or pathological entity <Theme> ceases to exhibit signs of life and undergoes irreversible loss of vital functions.
Cell Death	Indicated by the given trigger <Trigger>, a specific cell <Theme> undergoes a series of events leading to its own demise, often through programmed cell death or other mechanisms.
Breakdown	Indicated by the given trigger <Trigger>, a specific anatomical or pathological structure <Theme> disintegrates, decomposes, or undergoes degradation over time.
Cell Proliferation	Indicated by the given trigger <Trigger>, a specific cell <Theme> undergoes rapid and controlled replication or division, leading to an increase in the number of daughter cells.
Cell Division	Indicated by the given trigger <Trigger>, a specific cell <Theme> divides into two or more daughter cells through mitosis or meiosis.
Cell Differentiation	Indicated by the given trigger <Trigger>, a specific cell <Theme> undergoes changes in gene expression and morphology to become specialized and acquire distinct functions at a specific anatomical or pathological location <AtLoc>.
Remodeling	Indicated by the given trigger <Trigger>, a specific tissue <Theme> undergoes structural changes, reorganization, and modification in response to various stimuli or during growth and development.
Reproduction	Indicated by the given trigger <Trigger>, a specific organism <Theme> produces offspring through sexual or asexual means, leading to the continuation of the species.
Mutation	indicated by the given trigger <Trigger>, a specific gene, genome, or protein <Theme> undergoes a heritable change in its genetic sequence or structure at a particular site <Site>, potentially leading to alterations in its function or expression within a specific anatomical or pathological context <AtLoc>.
Carcinogenesis	Indicated by the given trigger <Trigger>, a specific anatomical or pathological entity <Theme> undergoes a series of genetic and cellular changes at a specific anatomical or pathological location <AtLoc>, leading to the development of cancer.
Cell Transformation	Indicated by the given trigger <Trigger>, a specific cell <Theme> undergoes changes in its phenotype, function, or behavior at a specific anatomical or pathological location <AtLoc>, often associated with the acquisition of abnormal or cancerous characteristics.
Metastasis	Indicated by the given trigger <Trigger>, a specific anatomical or pathological entity <Theme> spreads and establishes secondary growths or lesions at a different anatomical or pathological location <ToLoc> from the primary tumor site.
Infection	Indicated by the given trigger <Trigger>, an organism <Participant> invades and establishes itself in a specific anatomical or pathological site <Theme>, leading to a disease condition.
Metabolism	Indicated by the given trigger <Trigger>, collective chemical reactions occur within an organism <Theme> involving the processing, transformation, and utilization of specific molecules to maintain cellular functions and energy requirements.
Synthesis	Indicated by the given trigger <Trigger>, a specific simple chemical <Theme> is produced or created through chemical reactions or biological processes.
Catabolism	Indicated by the given trigger <Trigger>, a specific molecule <Theme> undergoes chemical reactions or metabolic pathways to break down into simpler compounds, releasing energy in the process.
Amino Acid Catabolism	Indicated by the given trigger <Trigger>, a specific amino acid <Theme> is broken down through metabolic pathways, leading to the release of energy and the generation of byproducts like ammonia and carbon compounds.
Glycolysis	Indicated by the given trigger <Trigger>, a specific molecule <Theme> undergoes a series of enzymatic reactions, ultimately converting glucose into pyruvate and producing ATP and NADH as energy carriers.

Table 17: Templates for Cancer Genetics Dataset.

Type	Template
Glycosylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> undergoes the addition of carbohydrate molecules (glycans) to specific sites, typically on proteins or lipids, leading to the formation of glycoproteins or glycolipids with diverse biological functions.
Acetylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> undergoes the addition of an acetyl group at a particular site <Site>, catalyzed by another molecule <Cause>.
Deacetylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> has an acetyl group removed from a particular site <Site>, facilitated by another molecule <Cause>.
Ubiquitination	Indicated by the given trigger <Trigger>, a specific molecule <Theme> is modified by the attachment of one or more ubiquitin molecules at a particular site, facilitated by another molecule <Cause>, often involving a simple chemical group <Site> as the site of attachment.
Deubiquitination	Indicated by the given trigger <Trigger>, a specific molecule <Theme> has ubiquitin molecules removed from a particular site, facilitated by another molecule <Cause> involving a simple chemical group <Site> as the site of removal.
Gene Expression	Indicated by the given trigger <Trigger>, a specific gene, genome, or protein <Theme> is activated, leading to the production of RNA or protein and subsequent biological functions.
Transcription	Indicated by the given trigger <Trigger>, genetic information from a specific gene, genome, or RNA molecule <Theme> is used as a template to produce complementary RNA (usually messenger RNA) by RNA polymerase.
Translation	Indicated by the given trigger <Trigger>, the genetic information carried by messenger RNA <Theme> is used to synthesize a protein by ribosomes in the cell.
Protein Processing	Indicated by the given trigger <Trigger>, the series of post-translational modifications, folding, and transportation of a specific gene, genome, or protein <Theme> in the cell to achieve its mature and functional form.
Phosphorylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> undergoes the addition of a phosphate group at a particular site <Site> within a protein domain or region, often regulating the molecule's activity or function.
Dephosphorylation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> has a phosphate group removed from a particular site <Site> within a protein domain or region, often resulting in the modulation or termination of its activity or function.
DNA Methylation	Indicated by the given trigger <Trigger>, a specific gene or genome <Theme> has a methyl group added to a particular site <Site> within a protein domain or region, often resulting in the regulation of gene expression and epigenetic modifications.
DNA Demethylation	Indicated by the given trigger <Trigger>, a specific gene or genome <Theme> has a methyl group removed from a particular site <Site> within a protein domain or region, often resulting in the modulation of gene expression and epigenetic modifications.
Pathway	Indicated by the given trigger <Trigger>, involving a specific molecule <Participant> that collaborates to accomplish a specific biological function or response.
Binding	Indicated by the given trigger <Trigger>, a specific molecule <Theme> forms a physical association or interaction with another molecule or region <Site> on a protein or DNA, potentially leading to functional changes or regulatory effects.
Dissociation	Indicated by the given trigger <Trigger>, a specific molecule <Theme> separates or detaches from another molecule or region <Site> on a protein or DNA, leading to the termination or disruption of their interaction or complex formation.
Localization	Indicated by the given trigger <Trigger>, a specific molecule <Theme> is directed to or away from a particular cellular component <AtLoc><FromLoc><ToLoc> or subcellular location within the cell.
Regulation	Indicated by the given trigger <Trigger>, an entity <Theme> is controlled or influenced by another entity <Cause> to achieve a specific biological effect or outcome.
Positive Regulation	Indicated by the given trigger <Trigger>, an entity <Theme> is promoted or enhanced by another entity <Cause> to achieve a specific biological effect or outcome.
Negative Regulation	Indicated by the given trigger <Trigger>, an entity <Theme> is inhibited or suppressed by another entity <Cause> to achieve a specific biological effect or outcome.
Planned Process	Indicated by the given trigger <Trigger>, a specific entity <Theme> is involved or manipulated using an instrument <Instrument> for a predetermined outcome or purpose.

Table 18: Continuation of templates for Cancer Genetics Dataset.