# Relation Extraction with Instance-Adapted Predicate Descriptions

**Anonymous ACL submission**

## Abstract

Relation extraction (RE) is a standard information extraction task playing a major role in downstream applications such as knowledge discovery and question answering. Although decoder-only large language models are excelling in generative tasks, smaller encoder models are still the go to architecture for RE. In this paper, we revisit fine-tuning such smaller models using a novel dual-encoder architecture with a joint contrastive and cross-entropy loss. Unlike previous methods that employ a fixed linear layer for predicate representations, our approach uses a second encoder to compute instance-specific predicate representations by infusing them with real entity spans from corresponding input instances. We conducted experiments on three different RE datasets from both general and biomedical domains. Our approach achieved F1 score improvements ranging from 1% to 2% over state-of-the-art methods with a simple but elegant formulation. Ablation studies justify the importance of various components built into the proposed architecture.

## 1 Introduction

Relation extraction (RE) is a basic task in natural language processing (NLP), especially in applied domains such as biomedicine and healthcare where relations among biomedical entities drive disease and treatment processes. A relation typically connects a subject entity and an object entity via a predicate (or relation type) as in (*tamoxifen*, `treats`, *breast cancer*). The goal is to extract such relations from natural language inputs, with at times the added goal of normalizing the entity spans to standardized vocabularies. Having a database of relations pertinent to a domain of interest can enable knowledge discovery and question answering.

### 1.1 Relation extraction trends

Early RE efforts focused on rule-based systems, kernel methods, and shortest path algorithms (Riloff et al., 1993; Zelenko et al., 2002; Bunescu and Mooney, 2005). As the field evolved, methods shifted to purely supervised models with labeled data. An initial approach was to use n-gram features leveraging dependency paths between the subject and object entity spans (Kambhatla, 2004). Subsequently neural embeddings, convolutional (Nguyen and Grishman, 2015) and recurrent architectures (Miwa and Bansal, 2016) and their combinations (Vu et al., 2016) enhanced with attention mechanisms (Guo et al., 2019) became popular. Since transformers were invented, the BERT architecture and its variants became popular for named entity recognition (NER) and RE efforts (Lin et al., 2019; Joshi et al., 2020).

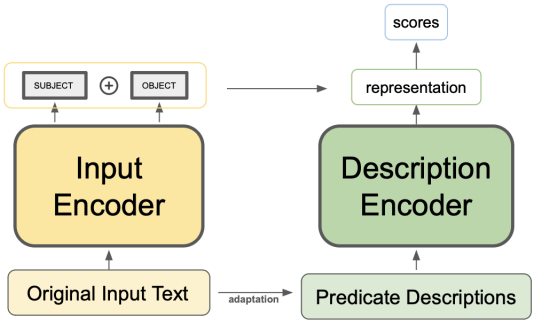


Figure 1: We employ a dual-encoder architecture with instance adapted predicate descriptions for relation extraction tasks.

There was a general consensus that joint end-to-end modeling (where entities and relations are extracted together in a single model) was better over pipeline based approaches (where an NER model and a separate RE model are stitched to form a pipeline). However, Zhong and Chen (Zhong and Chen, 2021) challenged that paradigm and showed that pipelines can still be superior with a clever marker based representation for entities. So pipelines are going through a revival and it is still worthwhile to build separate high quality mod-

els for NER and RE and join them in the end. Here, the RE component assumes the entities are already spotted. In this paper, we focus on this RE component that identifies relations between pre-spotted entities provided as part of the input.

We realize that decoder-only (autoregressive) large language models (LLMs) have become quite popular for general NLP tasks. While they clearly excel at generative tasks (e.g., summarization) and zero and few-shot situations for RE (Li et al., 2023), there is scarce evidence (if any) that they perform on par with encoder models when ample training data is available; their use has been mostly limited to data augmentation to enhance training dataset with synthetic examples while the eventual model to be trained is still a BERT variant or a encoder-decoder model such as T5 (Wadhwa et al., 2023).

### 1.2 High level idea of our method

Here we set out to improve relation classification abilities of encoder models. The default approach to RE once the entities are spotted is to derive entity (span) representations using the encoder and merging them in some way (typically, via concatenation) to derive softmax probability estimates for all predicates including the `NULL` (no relation) label. This corresponds to the left block of Figure 1 (note that softmax layer is not shown in the figure).

We propose to use predicate descriptions or definitions as an auxiliary signal. Most RE datasets/tasks have definitions of what a predicate is supposed to encode in relations that use it. For example, the US National Library of Medicine's semantic network[1] has the following official definition for the `TREATS` predicate: "Applies a remedy with the object of effecting a cure or managing a condition." This could be seen as a canonical way of describing a treatment relation although people could discuss it in myriad ways in natural language. The high level idea is to first derive an instance-adapted predicate description by *instantiating* the canonical predicate definition with entity spans from input text. Next, compare this description with the input text and pick the predicate whose instance-adapted description *matches* the most with the input. For example, consider the input sentence with italicized entity spans: "*Tamoxifen* is the most common endocrine therapy administered worldwide to women with hormone receptor-positive metastatic *breast cancer*." The `treats` predicate associated description for this instance is: "Applies a *tamoxifen* remedy with the object of effecting a cure or managing a *breast cancer* condition." It is straightforward to see this description semantically matches better with the input sentence compared with descriptions of other predicates (e.g., `CAUSES`). We carry this out using a dual encoder architecture as shown in Figure 1 where the left block encodes the input instance and the right block encodes the instance-specific predicate descriptions. In the rest of the paper, we formalize this intuition and evaluate the resulting method with three different datasets. We show F1 score improvements ranging from 1% to 2.1% compared to prior best methods. The datasets we used are all public and our code is attached for review and will be made available on GitHub if accepted.

## 2 Dual Encoder Architecture Details

Formally, for any input text containing mentions of entity spans say constituting set $E$, the goal is to determine a predicate $r \in R$ for each possible pair $(e_s, e_o) \in E \times E$, where $R$ includes the generally most frequent `NULL` predicate. As indicated in Section 1.2, we have two encoders, one for the original input text and one for the input adapted predicate descriptions. We first discuss the input text encoder.

### 2.1 Input representation

It is important to note we are representing the input text along with an entity pair $(e_s, e_o) \in E \times E$ to classify if they participate in a relation as asserted in the input. Since the eventual classification is dependent on the particular pair of entities, the representation is a linear projection of the entity embeddings from the first encoder model. Since we know the spans of $e_i$ and $e_j$, it is customary to encapsulate these spans with special tokens (Zhong and Chen, 2021; Ai and Kavuluru, 2023). Specifically subject $e_s$ is placed between entity marker tokens `[SUB:`$t_s$`]` and `[/SUB:`$t_s$`]` to denote the begin and end of the subject entity $e_s$ with entity type $t_s$. Likewise, object $e_o$ is placed between markers `[OBJ:`$t_o$`]` and `[/OBJ:`$t_o$`]`. The original input along with these demarcated spans is input to the encoder and the output embeddings of the start tokens of $e_s$ and $e_o$ are concatenated to represent the candidate pair. With $\mathcal{E}_T$ denoting the input encoder, the associated input representation is

$$\rho_T(e_s, e_o) = \mathbf{W_T}(\mathcal{E}_T\texttt{[SUB:}t_s\texttt{]} \,\|\, \mathcal{E}_T\texttt{[OBJ:}t_o\texttt{]}),$$

[1] https://lhncbc.nlm.nih.gov/semanticnetwork/

where the concatenated embedding is subjected to a linear transformation $\mathbf{W_T}$. The entity markers are crucial given it is important to capture the roles of subjects and objects and their types in determining viable predicates informed by contextual cues.

### 2.2 Predicate description representations

We recall that predicates in scientific areas have official descriptions of what they are expected to capture. For example, in BioRED dataset (Luo et al., 2022), the `POSITIVE CORRELATION` relation between `Chemical` and `Disease` entities is described as: "The drug A may induce the disease B, increase its risk, or the levels may correlate with disease risk." For the `NULL` predicate, we simply describe it as: "There are no relations between the drug A and disease B." When we find that the original definitions are overly simplistic or not sufficiently informative, we make necessary modifications to enhance clarity by prompting GPT-4. For example, in the SciERC dataset (Luan et al., 2018), the original definition provided for the predicate `PART-OF` reads: "B is a part of A." This definition, while broad, lacks sufficient elaboration and specificity needed for model training. To address this, we revise the definition to better capture the essence of the relationship, making it more informative and directly applicable for our purposes. Our revised definition, crafted to enhance clarity, is: "B is a component or segment that is integral to the structure or composition of A."

Instance adaption is accomplished by inserting the entity spans from the input into natural place holders for each predicate description. The main rationale for adaptation is to encode the entity spans in the context of the language used in the canonical definitions rather than simply using the definition without grounding in specific entities used. Since identifying subject/object placeholder slots in definitions is a one-time task for each predicate, this is done manually. Entity spans from the input are directly inserted into the chosen placeholder slots to create instance specific predicate descriptions, as shown for the `TREATS` predicate in Section 1.2. Though incorporating entity spans in the description texts grounds their representation, it primarily focuses on the "hard tokens", which may not capture the full essence of contextual nuances present in the input text. To address this potential limitation, we also incorporate the `[CLS]` representation from the first encoder $\mathcal{E}_T$ into the predicate description representation. Thus the instance-adapted representation for a specific $r \in R$ is

$$\rho^r_D(e_s, e_o) = \mathbf{W_D}(\mathcal{E}_T\texttt{[CLS]} \,\|\, \mathcal{E}^r_D\texttt{[SUB:}t_s\texttt{]} \,\|\, \mathcal{E}^r_D\texttt{[OBJ:}t_o\texttt{]}),$$

where $\mathcal{E}_D$ is description encoder, $\mathcal{E}^r_D$ is the representation derived for description of predicate $r$ grounded with entity spans from the input, and $\mathbf{W_D}$ is a linear transformation. By matching projection dimensions for $\mathbf{W_T}$ and $\mathbf{W_D}$, $\rho_T$ and $\rho^r_D$ lend themselves to similarity comparisons.

### 2.3 Contrastive and cross entropy objectives

Although two entities can be linked via multiple predicates in the real world, for a specific input textual instance, it is generally safe to assume only one predicate is at play. Using this multiclass (and not multilabel) assumption, we formulate a contrastive objective to push the representations of $\rho_T$ and $\rho^{r^+}_D$ ($r^+ \in R$, the correct predicate) closer to each other while pushing $\rho^{r^-}_D$ away from $\rho_T$ for all $r^- \in R \setminus \{r^+\}$, the incorrect predicates. We represent this closeness/farness via vector similarity $sim(\rho_T, \rho^r_D) = \cos(\rho_T, \rho^r_D)$. We chose the cosine distance which is naturally in the [0, 1] range as it was better than the dot product, which produced suboptimal performance due to scaling issues in initial experiments. (Note that we still use the normalized dot product formulation for cosine implementation instead of calling Python's math.cos().) With this setup, the contrastive loss function for a given input using the instance adapted predicate descriptions is

$$L_{ct}((e_s, e_o), r^+, r^-_1, \ldots, r^-_{|R|-1}) \quad (1)$$

$$= -\log \frac{e^{\text{sim}(\rho_T, \rho^{r^+}_D)}}{e^{\text{sim}(\rho_T, \rho^{r^+}_D)} + \sum_j e^{\text{sim}(\rho_T, \rho^{r^-_j}_D)}}$$

During our implementation, though contrastive training (from Equation (1)) was effective in pushing the input and positive predicate representations closer, we observed it does not learn robust relation representations of the input (the $\rho_T$s). To address this, we propose to add a new linear layer and simultaneously optimize the cross-entropy loss

$$L_{ce}(e_s, e_o) = -\sum_{r \in R} y_r \log(p_r) \quad (2)$$

where $y_r$ is a binary indicator of whether class label $r$ is the correct classification for the instance

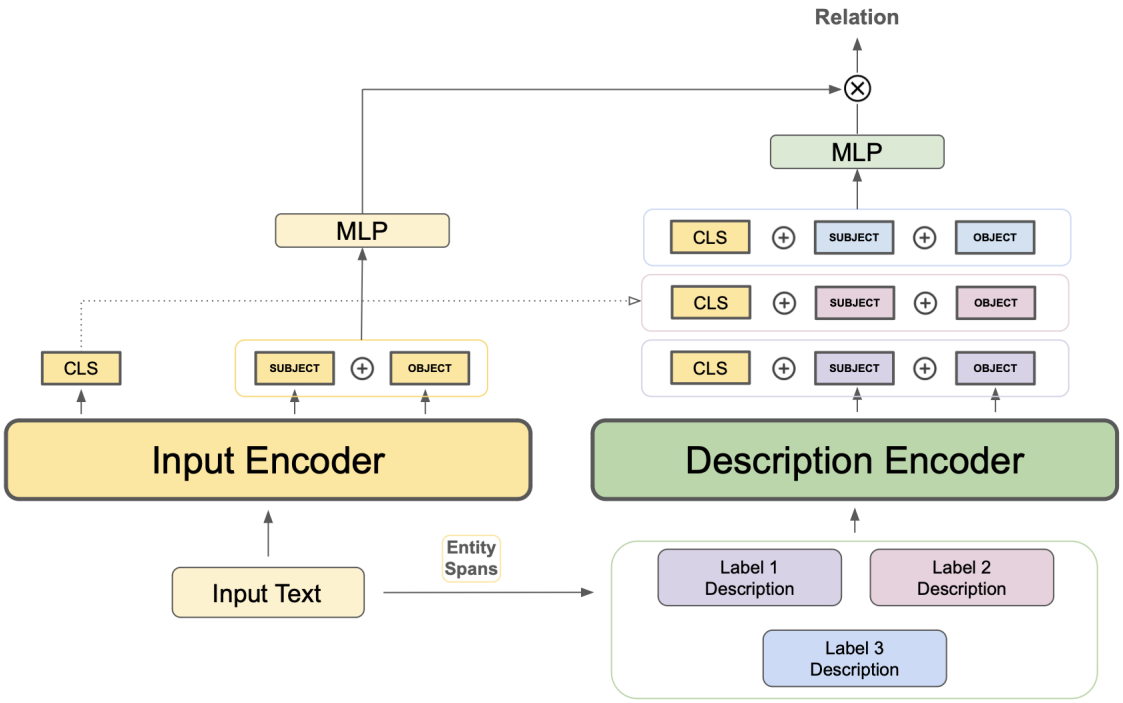


Figure 2: Our model architecture in detail with a 3-class example: we incorporate instance information by inserting entity spans from the input text and concatenate the `[CLS]` embedding or the input encoder to the description encoder representations. (Note that the linear and softmax layers used during training for equation (2) are not shown here.)

and $p_r$ is the predicted probability of the instance belonging to class $r$. We use the unified loss

$$L_u = \alpha L_{ce} + (1 - \alpha) L_{ct} \quad (3)$$

during training, where the $0 \leq \alpha \leq 1$ serves as a hyper-parameter that determines the influence of the contrastive loss component in the overall loss. Although training is done via Equation (3), our model exclusively relies on the contrastive scores to make predictions at inference time as

$$r_{pred} = \underset{r \in R}{\operatorname{argmax}}\, sim(\rho_T, \rho_D^r).$$

The full architecture is shown in Figure 2 with the two encoders handling the input and instance-specific label descriptions separately.

## 3 Experimental Setup

### 3.1 Datasets

We looked for public RE datasets that encompass a variety of relation types with apt predicate definitions and landed on three: SciERC, ChemProt, and BioRED with stats as shown in Table 1.

- **SciERC** (Luan et al., 2018): This dataset is created from AI conference or workshop paper abstracts and includes annotations for both entities and relationships offering predicates common in scientific discourse.

| Dataset | # Predicates | # Train | # Dev | # Test |
|---|---|---|---|---|
| SciERC | 7 | 350 | 50 | 100 |
| ChemProt | 5/13 | 1020 | 612 | 800 |
| BioRED | 8 | 400 | 100 | 100 |

Table 1: Statistics of datasets used (columns 3–5 are numbers of abstracts, not relations).

- **ChemProt** (Krallinger et al., 2017): This dataset is designed for chemical-protein interaction detection in biomedical literature and was created as part of the BioCreative shared task series. Although entities were annotated with the potential to be connected by one of ten predicates, only five are consistently used for evaluation following the shared task conventions, due to their relative importance and relevance in the context of chemical-protein interactions. These five predicates were further subdivided into a total of 13 fine-grained predicates, which characterize further nuances in interaction types (such as distinguishing between different kinds of upregulators or activators.) We test our methods with both schemes

(the five and 13 predicate variations).

- **BioRED** (Luo et al., 2022): This is a more recent and broad scoped biomedical RE dataset that includes four distinct entity types and eight different predicates. In its original form, entity normalization to standardized vocabularies is also expected. We adapt the dataset to fit our needs by treating it as a conventional RE task. In this adaptation, we address the issue of multiple entity mentions associated with a single entity ID by splitting these mentions into separate relations. This modification ensures that each entity mention is treated independently, simplifying the RE process under our current system capabilities. However, it does not reduce task difficulty because it is evaluated based on obtaining relations between all spans corresponding to both subject and object entities.

SciERC and ChemProt deal with sentence-level relations (the participating entities are within the same sentence) but the surrounding context of the full abstract maybe needed to extract the relations. BioRED is a more general document level dataset and includes cross-sentence relations.

### 3.2 Baseline methods and prior efforts

We used the basic Google BERT model (Devlin et al., 2019) and its biomedical variants BioBERT (Lee et al., 2019) and PubMedBERT (Gu et al., 2021) (trained on PubMed corpora) as our baselines. While BioBERT uses the original BERT tokenizer, PubMedBERT's vocabulary is built from scratch and has shown improvements in the past over BERT and BioBERT. Another recent popular method that revived pipelines by using entity role and type specific markers is the Princeton University Relation Extraction (PURE) framework (Zhong and Chen, 2021). PURE uses a BERT model as its base and builds on it with special tokens for entity boundaries. We also compare with two other prior efforts. The first by Su et al. (2021) introduced a novel method that enhances RE capabilities with contrastive learning for data augmentation. This approach refines text representations derived from the BERT model specifically for RE tasks. Wan et al. (2023) developed GPT-RE, a new RE system that integrates GPT-3, using an LLM as an instance-aware retrieval mechanism to obtain relevant demonstrations from the training dataset. The demonstrations are then used for in-context learning to predict outputs with GPT-3.

### 3.3 Base encoder choices and settings

Our method relies on two encoder models working together. Following prior studies (Zhong and Chen, 2021; Wan et al., 2023), we use SciBERT (Beltagy et al., 2019) as the encoder for the AI abstract RE dataset SciERC. SciBERT has been pre-trained on a corpus of computer science and biomedical full text articles, which makes SciBERT well-suited for the SciERC dataset. For BioRED and ChemProt, we use PubMedBERT as the encoder. In all our experiments, the encoders are all the same size as BERT-base, which contains approximately 110 million parameters.

All of our experiments were conducted using a consistent training regimen across different models. We arrived at $\alpha = 0.5$ with experiments on the development datasets for weighting the two losses in Equation (3) (more in Section 4.3). Each model was trained for 10 epochs, a batch size of 4 with a single run. We experimented with learning rates 1e-5 and 2e-5, to optimize performance and adaptability across various tasks and datasets. We used one NVIDIA V100 GPU trained for roughly 10 hours per experiment.

## 4 Results and Discussion

### 4.1 Main results

The main results in Table 2 show our method (last row) leads to performance enhancements ranging from 1% to 2.1% in micro F-score over prior methods. Since BioRED was introduced after papers from the first three rows were published, we trained and ran their code on it; this was also done for the 13-class version of ChemProt. Since the paradigm used by Su et al. (2021) (augmentation) and Wan et al. (2023) (GPT-3 calls) were quite different from ours, we did not run new experiments with them and simply reported the results from their papers, when available. In the penultimate row, we show the scores from a standard dual-encoder model without instance-adaptation. That is, we simply used the static predicate definitions without instantiating them with the entities from the input. The instance specific version shows nontrivial gains of 2.5% on the SciERC dataset and 3.6% on the BioRED dataset. This shows that instance-adaptation features are crucial for this method.

Additionally, our experiments with the BioRED

| Methods | Encoder | SciERC | ChemProt[13] | ChemProt[5] | BioRED |
|---|---|---|---|---|---|
| Devlin et al. (Devlin et al., 2019) | BERTBASE | 65.2 | 68.2 | 73.7 | 34.7 |
| Lee et al. (Lee et al., 2019) | BioBERT | - | 71.8 | 76.5 | 38.7 |
| Gu et al. (Gu et al., 2021) | PubMedBERT | - | 72.3 | 77.2 | 48.3 |
| Su et al. (Su et al., 2021) | PubMedBERT | - | - | 78.7 | - |
| GPT-RE (Wan et al., 2023) | SciBERT | 69.0 | - | - | - |
| PURE (Zhong and Chen, 2021) | SciBERT/ PubMedBERT | 68.5 | 72.5 | 78.7 | 51.4 |
| **Dual-Encoder (*Ours*)** | SciBERT/ | 68.6 | 72.6 | 79.5 | 48.9 |
| **Dual-Encoder+Adapt. (*Ours*)** | PubMedBERT | **71.1** | **73.5** | **79.8** | **52.5** |

Table 2: We compare the micro-F1 score, a common metric for evaluating the accuracy of classification models. Dual-Encoder+Adapt refers to our full model with instance-adaptation. The ChemProt[13] and ChemProt[5] columns refer to the 13-class and 5-class variants of the dataset, respectively. Pretrained biomedical encoders (BioBERT and PubMedBERT) are not used for the AI dataset (SciERC).

dataset were conducted at the mention-level, rather than at the entity identifier level. This evaluation choice was driven by our focus on the granularity of mention-specific data rather than on broader entity identifiers, dealing with which is an orthogonal issue of entity linking.

### 4.2 Ablation of the [CLS] component

Recall that the first input encoder's [CLS] token output was included as part of the predicate description representation (Section 2.2) to enhance its instance specific aspects. In the 2nd row of Table 3, we see that removing this component dips the performance by 0.7 points in F-score for the ChemProt dataset indicating a modest influence on eventual performance.

| Model | ChemProt[5] $F_1$ |
|---|---|
| **Full dual-encoder model** | **79.8** |
| w/o. [CLS] concatenation | 79.1 |
| w/o. cross-entropy loss | 78.7 |
| w/o. dual-encoder | 78.5 |

Table 3: ChemProt (5-class) ablated $F_1$ scores on the test set.

### 4.3 Ablation of cross-entropy loss

Integrating cross-entropy loss alongside our contrastive loss proved helpful for enhancing the model's ability to establish more effective relation representations. Traditional approaches in RE have relied on cross-entropy loss due to its effectiveness in clustering input embeddings of the same class closely together, thereby improving overall model performance. From row 3 of Table 3, we see that dropping it leads to a 1.1% dip in performance.

| $L_{ce}$ weight | $P$ | $R$ | $F_1$ |
|---|---|---|---|
| $\alpha = 0.1$ | 76.1 | 87.1 | 81.3 |
| $\alpha = 0.3$ | 81.2 | 82.8 | 81.9 |
| $\alpha = 0.5$ | 82.9 | 81.8 | 82.3 |
| $\alpha = 0.7$ | 80.9 | 83.2 | 82.1 |

Table 4: Different proportions of cross-entropy loss on ChemProt (5-class) development set.

We examined the impact of cross-entropy loss on overall model performance by varying its weight in the combined loss function (Equation (3)) to determine and fix its value for final training as part of hyperparameter optimization. The results are presented in Table 4 for the ChemProt development dataset. Assigning a small weight to this loss ($\alpha = 0.1$) and hence using more of the contrastive component leads to the highest recall; but the highest F-score is reached when $\alpha = 0.5$ (equal weighting), with some compromise in recall but with a better jump in precision sufficient enough to lead to an overall F1 gain of 1%.

### 4.4 Ablation of the dual encoder

Instead of the two separate encoders for input text and predicate description representation, we used the same encoder (hence shared parameters). Row 4 of Table 3 shows that this results in a 1.3% dip in performance. This maybe due to potential trade-off between system efficiency and performance. The dual-encoder configuration seems to provide superior performance by leveraging specialized processing streams for input and description texts.

### 4.5 Assessing different description texts

Although we used predicate definitions created by RE dataset creators, they are the nevertheless a single symbolic and discrete form. We wondered about how the scores would change if we rephrased them by prompting GPT-4 to "rewrite" them without any special instructions. We also tested a simple baseline that literally has the predicate name and entity spans fill the slots in this template: `@predicate@: @subject@, @object@`.

| **Versions** | `ChemProt`$^{13}$ $F_1$ |
|---|---|
| ORIGINAL | 73.5 |
| REWRITE | 73.4 |
| SIMPLE | 72.9 |

Table 5: `ChemProt (13-class)` scores with different description texts.

As the associated scores in Table 5 show, there is not much difference in performance with rewrites. However, surprisingly, the simple baseline is worse than the original definition by only $0.6\%$. This small dip highlights the effectiveness of even using templated forms with the tokens indicating the predicate name. But it is important to note that predicate names in ChemProt dataset are highly specific with unambiguous meanings such as `UPREGULATOR` and `ANTAGONIST`. Pretrained encoders such as PubMedBERT might already have decent representations for them that carry substantial semantic signal. However, it is not clear if the simple baseline holds as well with predicate names that have a broader meaning, where explicit detailed definitions might be needed for more gains.

## 5 Error Analysis

Although our approach does not incorporate the NER step, there are some notable errors purely for the RE component. One issue is the model's failure to accurately infer the meaning from complex or ambiguous contexts. Consider this example from the SciERC dataset: *"The proposed detectors are able to capture large-scale structures and distinctive textured patterns, and exhibit strong invariance to rotation, illumination variation, and blur."* While our model successfully identified the `USED-FOR` relation between *detectors* (subject) and entities *large-scale structures* and *distinctive textured patterns* (object), it was not able to predict the same gold `USED-FOR` relation with object entities *rotation*, *illumination variation* and *blur*. Since the sentence does not explicitly state that the *detectors* are used for *rotation*, *illumination variation* and *blur*, it might have missed potential implied links. On the other hand, one could argue from the original input that maybe the ground truth *detectors* `USED-FOR` relations with these entities may not be entirely accurate — *detectors* are overcoming the barriers of *blur* and *rotation* to excel at capturing *large-scale structures* and *distinctive textured patterns* and not necessarily being used to detect or capture blur/rotation.

Next, consider this input from the ChemProt dataset: *"Down-regulation of prostate-specific antigen (PSA) expression, an AR-target gene, by estramustine and bicalutamide was accompanied by the blockade of the mutated androgen receptor."* The model was able to identify the `INDIRECT-DOWNREGULATOR` relation between drugs *estramustine* and *bicalutamide* (subjects) and the protein *PSA* (object). But it failed to spot the same relation of those drugs with the object protein *AR*, which appears to be implicitly stated. Considering ChemProt extraction sometimes invovles the full abstract, other sentences surrounding this may offer indirect clues about the relation. However, upon examining the full abstract, we are not able to see stronger evidence than what is already present in the sentence shown here.

Another common error pattern is due to the model's lack of grasp of deep domain knowledge, particularly in biomedical datasets when information is densely packed. For example, consider this pithy ChemProt input: *"In vivo, agonist actions of yohimbine at 5-HT(1A) sites are revealed by WAY 100,635-reversible induction of hypothermia in the rat."* Here the gold relation is the `ANTAGONIST` link between subject *WAY 100,635* and object *5-HT(1A)*. The agonist actions of the chemical yohimbine on 5-HT(1A) result in hypothermia and the fact that

this hypothermia can be reversed by *WAY 100,635* indicates that it is playing an antagonist role for *5-HT(1A)*. This a complex expression involving the intermediate entity yohimbine and an unusual looking phrase *WAY 100,635-reversible* that densely packs meaning that typically needs a new sentence to convey explicitly. This may simply be a case of a highly complex example needing multi-hop reasoning that needed to be carried out with a compact input text.

A significant challenge in the BioRED dataset arises from the hierarchical nature of predicates where a more general `ASSOCIATION` relation is confused with specific `POSITIVE/NEGATIVE CORRELATION` relations. If the model fails to latch on to specific relations, it may default to the general predicate. Consider the input: *"The 2:1 atrioventricular block improved to 1:1 conduction only after intravenous lidocaine infusion or a high dose of mexiletine, which also controlled the ventricular tachycardia. A novel, spontaneous LQTS-3 mutation was identified in the transmembrane segment 6 of domain IV of the Na(v)1.5 cardiac sodium channel, with a G–>A substitution at codon 1763, which changed a valine (GTG) to a methionine (ATG). ...."* Here the text mentions that the administration of the drug *lidocaine* improved the patient's condition, and separately, a novel mutation (*LQTS-3*) was identified in patients suffering from arrhythmias. The gold relation is `NEGATIVE CORRELATION` of *lidocaine* with *LQTS-3 mutation* but the model predicted `ASSOCIATION`, the generic predicate, which is only supposed to be used when the correlation type cannot be discerned from the input. However, although *lidocaine* and *LQTS-3* are never mentioned in the same sentence, through the full abstract that has over 250 words, one can see the *lidocaine* is blocking the function of this mutation in causing arrhythmias and as such has a negative correlation with it.

## 6 Conclusion

In this paper, we introduce a new approach to RE that uses a dual-encoder architecture that compares input text to instance-adapted canonical descriptions of the predicates. Experiments with a equally weighted joint contrastive and cross entropy loss show that this approach improves over prior methods for three scientific RE datasets including AI and biomedical abstract inputs. Ablation experiments also reveal that each component of the model plays a nontrivial role in the overall performance. We conclude with a few future research directions.

- As discussed in Section 2.3, we only use the cosine scores of the two encoder representations to pick the right predicate at test time. Since the cross entropy loss helped during training, a better way to integrate output softmax probability estimates with normalized cosine scores could lead to more performance improvements.

- Many RE use-cases have hierarchical predicate structures which were also observed in ChemProt and BioRED datasets in our paper. More involved learning strategies that leverage label hierarchies, potentially with graph convolutional nets, may be needed. Another training loss term that imposes penalties for violating hierarchical constraints could lead to better regularization and fewer errors arising from distant predicates from the hierarchy.

## Limitations

Despite the overall positive results, our model has some limitations. Although the description texts are derived from the annotation guidelines, they require an additional step to make them compatible with our model, such as manually inserting placeholders for the subject and object entities. The instance adaptation provided by our current approach is limited to canonical definitions. This issue could be mitigated by using LLMs to produce more refined descriptions.

## References

Xuguang Ai and Ramakanth Kavuluru. 2023. End-to-end models for chemical–protein interaction extraction: Better tokenization and span-based pipeline strategies. In *2023 IEEE 11th International Conference on Healthcare Informatics (ICHI)*, pages 610–618. IEEE.

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

Razvan Bunescu and Raymond Mooney. 2005. A shortest path dependency kernel for relation extraction. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in*

*Natural Language Processing*, pages 724–731, Vancouver, British Columbia, Canada. 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.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23.

Zhijiang Guo, Yan Zhang, and Wei Lu. 2019. Attention guided graph convolutional networks for relation extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 241–251.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the association for computational linguistics*, 8:64–77.

Nanda Kambhatla. 2004. Combining lexical, syntactic, and semantic features with maximum entropy models for information extraction. In *Proceedings of the ACL interactive poster and demonstration sessions*, pages 178–181.

Martin Krallinger, Obdulia Rabal, Saber A Akhondi, Martın Pérez Pérez, Jesús Santamaría, Gael Pérez Rodríguez, Georgios Tsatsaronis, Ander Intxaurrondo, José Antonio López, Umesh Nandal, et al. 2017. Overview of the biocreative vi chemical-protein interaction track. In *Proceedings of the sixth BioCreative challenge evaluation workshop*, volume 1, pages 141–146.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Guozheng Li, Peng Wang, and Wenjun Ke. 2023. Revisiting large language models as zero-shot relation extractors. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6877–6892.

Chen Lin, Timothy Miller, Dmitriy Dligach, Steven Bethard, and Guergana Savova. 2019. A bert-based universal model for both within-and cross-sentence clinical temporal relation extraction. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 65–71.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreferencefor scientific knowledge graph construction. In *Proc. Conf. Empirical Methods Natural Language Process. (EMNLP)*.

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.

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

Thien Huu Nguyen and Ralph Grishman. 2015. Relation extraction: Perspective from convolutional neural networks. In *Proceedings of the 1st workshop on vector space modeling for natural language processing*, pages 39–48.

Ellen Riloff et al. 1993. Automatically constructing a dictionary for information extraction tasks. In *AAAI*, volume 1, pages 2–1. Citeseer.

Peng Su, Yifan Peng, and K Vijay-Shanker. 2021. Improving bert model using contrastive learning for biomedical relation extraction. In *Proceedings of the 20th Workshop on Biomedical Language Processing*, pages 1–10.

Ngoc Thang Vu, Heike Adel, Pankaj Gupta, and Hinrich Schütze. 2016. Combining recurrent and convolutional neural networks for relation classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 534–539.

Somin Wadhwa, Silvio Amir, and Byron C Wallace. 2023. Revisiting relation extraction in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15566–15589.

Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi. 2023. Gpt-re: In-context learning for relation extraction using large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3534–3547.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. 2002. Kernel methods for relation extraction. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 71–78. Association for Computational Linguistics.

Zexuan Zhong and Danqi Chen. 2021. A frustratingly easy approach for entity and relation extraction. In *Proceedings of the 2021 Conference of the North*

*American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 50–61, Online. Association for Computational Linguistics.