Bio-RFX: Refining Biomedical Extraction via Advanced Relation Classification and Structural Constraints

Minjia Wang[1,2], Fangzhou Liu[1], Xiuxing Li[3,4*],
Bowen Dong[1], Zhenyu Li[1], Tengyu Pan[1], Jianyong Wang[1*]
[1]Tsinghua University [2]Harvard University
[3]School of Computer Science & Technology, Beijing Institute of Technology
[4]Key Laboratory of Brain-Machine Intelligence Technology,
Ministry of Education, Nanjing University of Aeronautics and Astronautics
minjiawang@g.harvard.edu
{lfz24,dbw22,zy-li21,pty23}@mails.tsinghua.edu.cn
xxl@bit.edu.cn jianyong@tsinghua.edu.cn

Abstract

The ever-growing biomedical publications magnify the challenge of extracting structured data from unstructured texts. This task involves two components: biomedical entity identification (Named Entity Recognition, NER) and their interrelation determination (Relation Extraction, RE). However, existing methods often neglect unique features of the biomedical literature, such as ambiguous entities, nested proper nouns, and overlapping relation triplets, and underutilize prior knowledge, leading to an intolerable performance decline in the biomedical domain, especially with limited annotated training data. In this paper, we propose the Biomedical Relation-First eXtraction (Bio-RFX) model by leveraging sentence-level relation classification before entity extraction to tackle entity ambiguity. Moreover, we exploit structural constraints between entities and relations to guide the model's hypothesis space, enhancing extraction performance across different training scenarios. Comprehensive experimental results on biomedical datasets show that Bio-RFX achieves significant improvements on both NER and RE tasks. Even under the low-resource training scenarios, it outperforms all baselines in NER and has highly competitive performance compared to the state-of-the-art fine-tuned baselines in RE [1].

1 Introduction

Biomedical literature is a vital resource for research, but the surge in publications makes manual tracking of advances difficult. Consequently, there's growing interest in methods for automatic extraction of structured information from these texts. This involves identifying biomedical entities and their relations from plain texts, namely Named Entity Recognition (NER) and Relation Extraction (RE), as illustrated in Figure 1. These structured data can be applied to several downstream tasks and real-world circumstances in academia and industry.

The keystone of entity and relation extraction hinges on proficiently modeling textual data, which includes deriving meaningful biomedical text representations and developing methods to utilize them. The adaptation of BERT (Devlin et al., 2019) architectures to the biomedical field, including pretraining and additional training, has seen significant success in recent years. However, two substantial challenges remain in this domain.

Firstly, learning effective representations is challenging in low-resource scenarios. Neural network-based strategies depend on substantial quantities of labeled training data, a prerequisite often elusive in the biomedical domain. This is mainly due to the labor-intensive, time-consuming, and error-prone nature of manually annotating biomedical text data. Detailed reading and interpretation are required for annotation, and reliable annotations often necessitate domain experts or multiple annotation rounds.

Some studies focus on incorporating biomedical knowledge graphs (KGs) like UMLS (Bodenreider, 2004) into training data to improve cross-domain adaptability (Zhang et al., 2021). Nonetheless, this approach is subject to several limitations. Biomedical KGs like UMLS, can be sizeable (27.1 GB), leading to large storage space and heavy computational costs. Besides, while most biomedical information extraction tasks focus on extracting fine-grained relations between coarse-grained entities, KGs often struggle to differentiate between relation types. For instance, all relation types in DrugProt (Miranda et al., 2021) and DrugVar (Peng et al., 2017) datasets are classified as the same type (*interact-with*) in UMLS. This classification significantly diminishes the instructive value of prior

[*] Corresponding authors.

[1] The source code of this paper can be obtained from https://github.com/Tschal-rsa/bio-rfx.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 10524–10539
November 12-16, 2024 ©2024 Association for Computational Linguistics

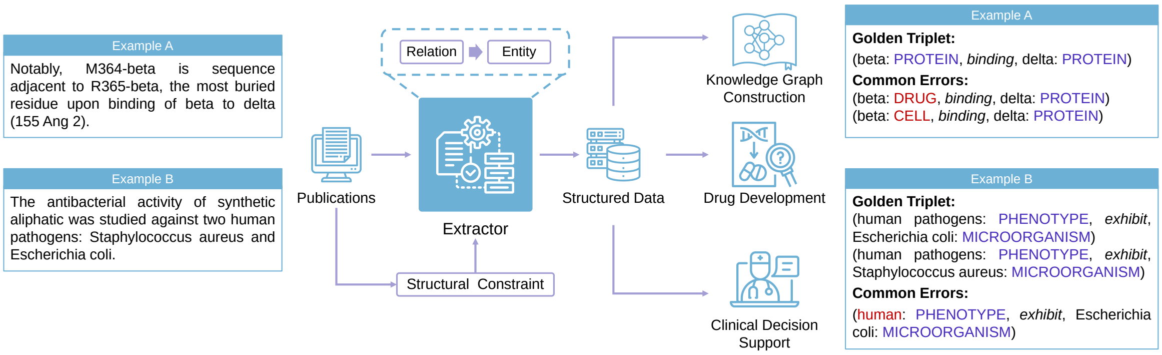

Example A
Notably, M364-beta is sequence adjacent to R365-beta, the most buried residue upon binding of beta to delta (155 Ang 2).
Example B
The antibacterial activity of synthetic aliphatic was studied against two human pathogens: Staphylococcus aureus and Escherichia coli.
Relation
Entity
Publications
Extractor
Structural Constraint
Structured Data
Knowledge Graph Construction
Drug Development
Clinical Decision Support
Example A
Golden Triplet:
(beta: PROTEIN, binding, delta: PROTEIN)
Common Errors:
(beta: DRUG, binding, delta: PROTEIN)
(beta: CELL, binding, delta: PROTEIN)
Example B
Golden Triplet:
(human pathogens: PHENOTYPE, exhibit, Escherichia coli: MICROORGANISM)
(human pathogens: PHENOTYPE, exhibit, Staphylococcus aureus: MICROORGANISM)
Common Errors:
(human: PHENOTYPE, exhibit, Escherichia coli: MICROORGANISM)

Figure 1: Automatic entity and relation extraction from biomedical publications. Example A illustrates ambiguous entities and Example B shows perplexing nested biomedical proper nouns.

knowledge in KGs, especially when using triplets from KGs in extraction tasks (Han et al., 2016; Bastos et al., 2021).

Secondly, biomedical literature's unique features necessitate domain-specific model design, an area less explored than text representations. The performance of general-domain models drops dramatically when adapting to biomedical contexts due to the stylized writing and domain-specific terminology. Moreover, biomedical entities can be ambiguous, with the same phrases recognized as different entities depending on context and relationships with other entities. For instance, in Figure 1 Example A, *beta* and *delta* could refer to various entities, but their *binding* relation suggests they're proteins. Furthermore, overlapping proper nouns can perplex models, making entity detection challenging. In Figure 1 Example B, both *human* and *human pathogens* are valid entities, but only the latter should be extracted under the *exhibit* relation type. These factors make it hard for general-domain models to effectively handle biomedical literature's distinctive features.

To address these issues, we proposed **<u>Bio</u>**medical **<u>R</u>**elation-**<u>F</u>**irst e**<u>X</u>**traction (**Bio-RFX**) model, wherein hypothesis space is constrained by prior knowledge. This architecture, inspired by the strong structural knowledge implications among relational triplets, first predicts the relation types that appeared in the sentence. It then extracts relevant entities satisfying such structure through a question-answering approach. A question is generated based on the relation type, with the original sentence as context, and related entities form a multi-span answer. We then predict the sentence's valid entity count and remove false entities using the text-NMS algorithm (Hu et al., 2019). Finally, relations between entities are generated according to structural constraints.

This approach is capable of tackling specific issues in biomedical texts. For ambiguous entities, the predicted relation information guides entity type identification. For perplexing entities, overlapping terms are eliminated by the text-NMS algorithm, enhancing specificity.

We evaluate our method on four biomedical datasets: DrugProt (Miranda et al., 2021), DrugVar (Peng et al., 2017), BC5CDR (Li et al., 2016) and CRAFT (Cohen et al., 2017). Experimental results show that our model achieves the best average rank among all the models. Our model also surpasses the previous state-of-the-art, improving NER and RE F1 scores by up to 2.91% and 1.86% respectively.

The main contributions of this paper include:

- We unveil an efficient biomedical relation-first extraction framework, meticulously crafted for extracting entities and relations from biomedical literature in low-resource settings.
- We construct a relation-first model to adapt to the features of biomedical texts and innovatively utilize prior knowledge to constrain the hypothesis space of the model.
- Comprehensive experimental results show that our model significantly outperforms baseline models on biomedical datasets under different settings.
- To the best of our knowledge, our work marks the inaugural endeavor in extracting both entities and relations from biomedical literature

under the scenarios characterized by limited training data.

2 Related Work

Researchers have proposed a multitude of methods for extracting entities and relations, the majority of which fall under either pipeline or joint methods.

2.1 Pipeline Method

The pipeline method is categorized into three approaches based on the order of data extraction.

The first approach starts with NER to identify entities in a sentence, and then classifies each extracted entity pair into different types of relations. To attain representations for entities and relations at various levels, FCM (Gormley et al., 2015) uses a compositional embedding with hand-crafted and learned features. PURE (Zhong and Chen, 2021) inserts predicted entity label marks into the input sentence before RE to integrate semantic information provided by entity types. PL-Marker (Ye et al., 2022) uses a neighborhood-oriented packing strategy and a subject-oriented packing strategy, and Fabregat et al. (2023) first trains a NER model and then transfers the model's weights to the RE model. These methods, while easy to implement, often ignore either the overlapping relation triplets or the important inner structure behind the text.

To tackle these challenges, the second approach is proposed. The model first detects all potential subject entities in a sentence and then recognizes object entities in relation to each relation. CasRel (Wei et al., 2020) regards relations as functions that map subjects to objects and identifies subjects and objects in a sequence-tagging manner. Multi-turn QA (Li et al., 2019) formulates entity and relation extraction as a question-answering task, sequentially generating questions on subject entities, relations, and object entities. ETL-Span (Yu et al., 2020) designs a subject extractor and an object-relation extractor and decodes the entity spans by token classification and heuristic matching algorithm. The sparsity of relations in real-life sentences can lead to redundancy when using the above methods. This is because these methods predict relations for entity pairs that don't actually have any, or they enumerate all relation types, even when many of them are not present.

The third approach addresses this problem by running relation detection at a sentence level before entity extraction. RERE (Xie et al., 2021) predicts potential relations and performs a relation-specific sequence-tagging task to extract entities. PRGC (Zheng et al., 2021) adds a global correspondence for triplet decoding. Our method, Bio-RFX, differs in the following aspects. We use independent encoders for entity and relation extraction, aiding in learning task-specific contextual representations. Besides, instead of directly applying relation representations, we generate a query related to the relation type and targeted entity types. This approach naturally models the connection between entity and relation, allowing us to leverage fully-fledged machine reading comprehension models. Furthermore, focusing on domain-specific issues, such as nested or overlapping proper nouns and biomedical terms, we implemented a text-NMS algorithm to improve extraction specificity.

2.2 Joint Method

Another task formulation is building joint models that simultaneously extract entities and relations. Recent research has focused on neural network-based models and has yielded promising results. For instance, a joint extraction task can be converted to a sequence tagging problem by designing token labels that encapsulate information on entities and the relation they hold (Zheng et al., 2017). However, these methods failed to extract overlapping entities and relation triplets, which are ubiquitous in the biomedical domain.

To tackle the aforementioned challenge, subsequent works introduced various enhancement mechanisms via modeling input texts in a spatial rather than traditional sequential manner. TPLinker (Wang et al., 2020) regards extraction as matrix tagging instead of sequence tagging, and links token pairs with a handshake tagging scheme. OneRel (Shang et al., 2022) enumerates all the token pairs and relations and predicts whether they belong to any factual triplets. SPN (Sui et al., 2023) formulates joint extraction as a direct set prediction problem. REBEL (Huguet Cabot and Navigli, 2021) takes a seq2seq approach, translating the triplets as a sequence of tokens to be decoded by the model. DeepStruct (Wang et al., 2022) pretrains language models to generate triplets from texts and performs joint extraction in a zero-shot manner. Graph structures are also widely applied. KECI (Lai et al., 2021) first constructs an initial span graph from the text, then uses an entity linker to form a biomedical knowledge graph. It uses an attention mechanism to refine the initial span graph and the knowledge graph into a refined graph for final predictions. SpanBioER (Fei et al., 2020) is also a span-graph neural model that formulates the task as relation triplets prediction and builds the entity graph by enumerating candidate entity spans.

However, joint models have several drawbacks. These spatial approaches suffer from a high computational complexity. Besides, NER and RE are distinct tasks, thus sharing representations between entities and relations undermines performance (Zhong and Chen, 2021). In comparison, it is much easier to divide joint extraction into several submodules and conquer each of them separately.

3 Method

In this section, we detail the proposed Bio-RFX, as illustrated in Figure 2. The framework contains four key components: (1) **Relation Classifier** predicts all the relation types that the input sentence expresses by performing a multi-label classification task. (2) **Entity Span Detector** extracts subject and object entities for each relation in a sentence using a relation-specific question. (3) **Entity Number Predictor** predicts the number of entities with a regression task in a question-answering manner. (4) **Pruning Algorithm** filters the candidate entities by the predicted entity number.

3.1 Relation Classification

For relation extraction, we detect relations at the sentence level. This helps to avoid the prediction of relations that are not actually present. As shown in Figure 2, for each relation type in the dataset, we will detect if the relation is expressed in the sentence respectively, which is a multi-label classification task. Our model first constructs a contextualized representation for each input token $x_i \in x = \{x_1, x_2, ..., x_n\}$ using SciBERT (Beltagy et al., 2019). To be more specific, we construct an input sequence $[[\text{CLS}], x, [\text{SEP}]]$, feed it into the encoder and obtain the output token representation matrix $\boldsymbol{H} = [\boldsymbol{h}_0, \boldsymbol{h}_1, \dots, \boldsymbol{h}_n, \boldsymbol{h}_{n+1}] \in \mathbb{R}^{d \times (n+2)}$, where d indicates the hidden dimension. We then use $\boldsymbol{h}_0 \in \mathbb{R}^d$ to represent the semantic information of the sentence. Next, the sentence representation is fed into $|T_r|$ classifiers independently to determine whether the sentence expresses relation τ_r, where $\tau_r \in T_r$ and T_r is the set of relation types in the dataset D. For relation τ_r, the output of the classifier $\hat{p}_r$ can be defined by $\hat{p}_r = \sigma(\boldsymbol{W}_r \boldsymbol{h}_0 + \boldsymbol{b}_r)$, where $\boldsymbol{W}_r, \boldsymbol{b}_r$ are trainable model parameters. σ is the sigmoid activation function. For each relation τ_r, we employ the cross-entropy loss to optimize the training process. Let p_r denote the ground truth from annotation; $p_r = 1$ is used to represent that relation τ_r has appeared in the sentence and vice versa. Therefore, the loss function for the relation classifier can be defined as:

$$\mathcal{L}_{\text{rel}} = -\sum_{x \in D} \sum_{r=1}^{|T_r|} p_r \log \hat{p}_r. \tag{1}$$

3.2 Entity Extraction

3.2.1 Entity Detection

We formulate entity detection as span extraction from the sentence. This approach is inspired by machine reading comprehension models that extract answer spans from the context. For the first step, we design a question for entity detection. For NER, we generate a question q using predefined templates with all the entity types in T_e. For example, if $T_e = \{$*null*, *chemical*, *gene*, *variant*$\}$, then $q =$ *What are the chemicals, genes, and variants in the sentence?* RE is more complicated since the strong structural constraints between entity types and relation types should not be ignored. For RE, the question is specific to each relation type τ_r that appeared in the sentence. Given a relation type τ_r, let $T_{re} \subseteq T_e \times T_e$ denote the set of allowed subject and object entity type pairs. We obtain T_{re} by enumerating all the possible triplets in the dataset as prior knowledge, which is undemanding since the relation types are fine-grained while the entity types are coarse-grained, resulting in a limited size of T_{re}. Suppose $\tau_r =$ *activator*, then $T_{re} = \{\langle$*chemical, gene*$\rangle\}$. The question is generated with T_{re}, i.e. $q_r =$ *What gene does the chemical activate?* We also explored other prompting techniques in Appendix A. Based on the question, we regard the sentence x as context and build the input sequence $[[\text{CLS}], q_r, [\text{SEP}], x, [\text{SEP}]]$. Next, we compute the representation of each span $s \in S$ in sentence x. Let FFNN be a feed-forward neural network, and $\boldsymbol{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \dots, \boldsymbol{h}_N]$ be the token representation matrix for the input sequence, where N denotes the number of tokens in the sequence. We obtain the representation $\boldsymbol{s}$ for s using an attention mechanism over tokens (Lee et al., 2017):

$$a_t = \frac{\exp\left(\text{FFNN}_\alpha(s_t^*)\right)}{\sum\limits_{k=l_S}^{l_E} \exp\left(\text{FFNN}_\alpha(s_k^*)\right)}, \tag{2}$$

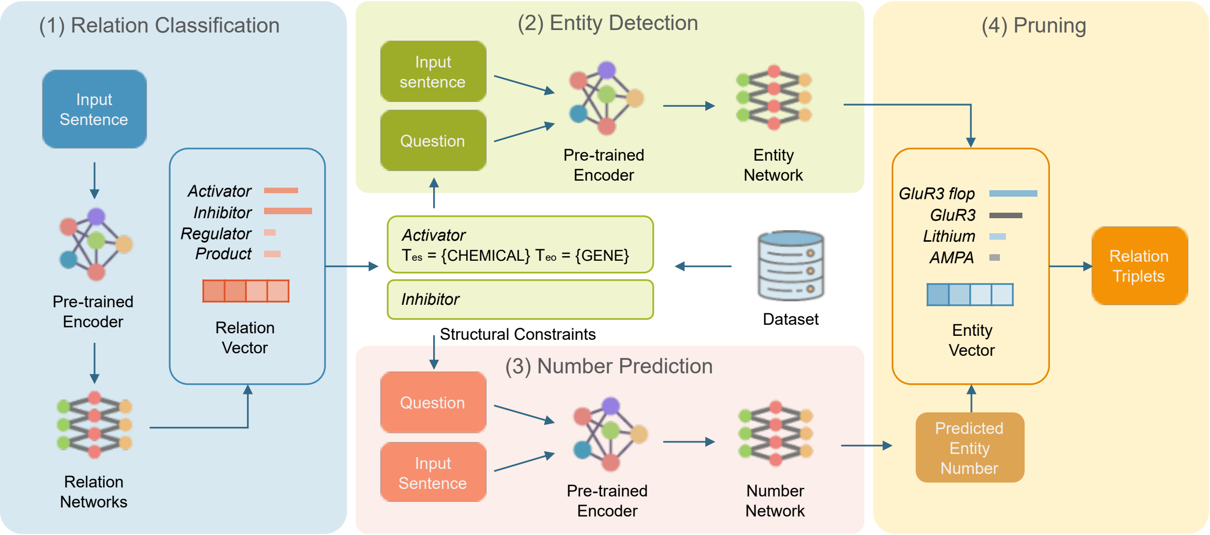


Figure 2: The overall framework of Bio-RFX. (1) The relation classifier predicts that there are two relations in the sentence, *Activator* and *Inhibitor*. (2–4) Relation-specific entity extraction is performed for each of the predicted relation types. To be more specific, (2) the entity detector extracts all the entities that satisfy the structural constraints via a question-answering manner, and (3) the number predictor outputs the number of spans similarly. (4) The relation triplets are generated by excluding the overlapping perplexing entities.

$$\boldsymbol{s} = [\boldsymbol{h}_{l_S}, \sum_{t=l_S}^{l_E} a_t \boldsymbol{h}_t, \boldsymbol{h}_{l_E}, \Phi(w)], \tag{3}$$

where s^* denotes the concatenation of all the tokens in the span s; weight a_t denotes the normalized attention score; l_S, l_E denote the start and end position for span s respectively; and $\Phi(w)$ is a learnable width embedding for the span width $w = l_E - l_S$. Then, for NER, we compute the probability $\hat{p}_e$ that span s is an entity of type τ_e using a FFNN with GELU activation function, namely $\hat{p}_e = \mathrm{FFNN}_e(\boldsymbol{s})$. The loss function is defined in the following equation:

$$\mathcal{L}_{\text{ent}} = -\sum_{x \in D} \sum_{s \in S} \sum_{e=0}^{|T_e|} w_e p_e \log \hat{p}_e. \tag{4}$$

For RE, the input sequence is relation-specific. We compute the probability $\hat{p}_{re}$ that span s is a subject or object entity of type τ_e allowed by the relation type τ_r, thus the loss function is:

$$\mathcal{L}_{\text{ent}} = -\sum_{x \in D} \sum_{s \in S} \sum_{e=0}^{|T_e|} \sum_{\substack{r=1 \\ \tau_r \in R_x}}^{|T_r|} w_e p_{re} \log \hat{p}_{re}. \tag{5}$$

In both cases, w_e is a hyperparameter used to handle the overwhelming negative entity labels, i.e. for *null* entity, we set $w_e = 0.1$, and $w_e = 1$ for all other entities.

3.2.2 Number Prediction

To exclude perplexing entities from the output, we implement textual Non-Maximum Suppression (text-NMS) algorithm (Hu et al., 2019), which requires us to predict the number of potential entities in a sentence x. We formulate the regression task in a question-answering manner. In the above example, for NER, we have q = *How many chemicals, genes, and variants are there in the sentence?* For RE, for each subject-object pair in T_{re}, a unique question is generated. For instance, τ_r = *activator*, $T_{re} = \{\langle$*chemical, gene*$\rangle\}$, then q_r = *How many chemicals and genes are there in the sentence with relation activation?* The question and the sentence are concatenated together using [CLS] and [SEP] to form the input sequence. Similar to Section 3.1, we obtain the representation vector $\boldsymbol{h}_0$ for the input sequence and then acquire the predicted number $\hat{k}$ of potential entities with $\hat{k} = \mathrm{FFNN}_n(\boldsymbol{h}_0)$.

We use k to denote the number of ground truth entities in a sentence. The loss function for number prediction in NER is the mean squared loss, which can be defined as:

$$\mathcal{L}_{\text{num}} = \sum_{x \in D} (k - \hat{k})^2. \tag{6}$$

For RE, it is slightly different concerning relations. We define k_r as the number of subjects and objects with relation τ_r, and entities presented in multiple triplets are only counted once. The loss is defined as:

$$\mathcal{L}_{\text{num}} = \sum_{x\in D} \sum_{\substack{r=1 \\ \tau_r \in R_x}}^{|T_r|} (k_r - \hat{k}_r)^2. \tag{7}$$

3.2.3 Pruning Algorithm

After extracting spans, we adopt the text-NMS algorithm to heuristically prune redundant and perplexing entities. First, for each span s, we obtain the confidence score $\lambda(s) = 1 - \hat{p}_{e=0}$, namely the probability of not being a *null* entity. Next, spans in S are sorted by descending confidence scores. A new set $\hat{S}$ is initialized as the final span prediction. We select the span s_i with the highest confidence score, add s_i to $\hat{S}$, remove any remaining span $s_j \in S$ that overlaps with s_i from S, and remove s_i from S as well. The text-level F1 score indicates the degree of overlapping. This process repeats until either $|\hat{S}|$ reaches k, i.e. the number of entities, or S is empty. The algorithm is detailed in Algorithm 1 in Appendix B.

We then generate relation triplets with the spans in $\hat{S}$. Instead of adopting a nearest-matching method (Xie et al., 2021), we match all the possible subjects and objects to address the overlapping triplets in biomedical texts. To be more specific, for relation τ_r, each $\langle \tau_{es}, \tau_{eo} \rangle \in T_{re}$ is converted to a relation triplet $\langle \tau_{es}, \tau_r, \tau_{eo} \rangle$ as the final result.

4 Experiments and Analysis

In this section, we validate our model's effectiveness through extensive sentence-level NER and RE experiments. We begin with the experimental setup, followed by performance evaluation and analysis. We then explore our method's efficacy in a low-resource setting and conclude with an ablation study to highlight the impact of each submodule in our framework.

4.1 Experimental Settings

4.1.1 Datasets

We empirically evaluate related methods on four datasets: DrugProt (Miranda et al., 2021), DrugVar (Peng et al., 2017), BC5CDR (Li et al., 2016) and CRAFT (Cohen et al., 2017). More details and preprocessing methods are presented in Appendix C.

4.1.2 Baselines

We evaluate our model by comparing with several models that are capable of both entity and relation extraction on the same datasets, which are strong models designed for general domain (PURE (Zhong and Chen, 2021), TPLinker-plus (Wang et al., 2020) and PL-Marker (Ye et al., 2022)) and biomedical domain (KECI (Lai et al., 2021) and SpanBioER (Fei et al., 2020)). Some of the competitive relation-first approaches, such as PRGC (Zheng et al., 2021), use ground truth entities as input, while the other methods use the raw text as input, therefore making them unsuitable for baseline models.

Recent studies demonstrate generative methods' effectiveness in extractive tasks. Thus, we include REBEL (Huguet Cabot and Navigli, 2021) and GPT-4 (OpenAI, 2023) in our set of baselines. REBEL is unable to identify entities that are not part of any relation triplets, so we only report the metrics for RE. Please refer to Appendix D for implementation details. We also detail the experimental settings of GPT-4 in Appendix E.

4.1.3 Evaluation Metrics

We use micro F1 score and average rank for both NER and RE evaluation. When computing the micro F1 score, an entity is considered matched if the whole span and entity type match the ground truth, and a relation triplet is regarded as correct if the relation type, subject entity, and object entity are all correct. Following Demšar (2006) and Wang et al. (2024), we also obtain the average rank of each model for comparison across all datasets.

4.2 Main Results

Table 1 shows the micro F1 scores of all models on the four datasets. The results demonstrate that our model achieves the best result in NER and RE in average rank. Our model obtains an absolute F1 gain of up to 1.34% compared with previous state-of-the-art in NER, and 1.86% in RE. It significantly outperforms most of the other baselines in both tasks (see Appendix G for significance analysis). On DrugProt, KECI achieves competitive performance in RE but performs poorly in NER. KECI's graphical structure enables it to generate more accurate relation triplets compared to our simple generating method. However, its training process depends heavily on a large amount of annotated relations, leading to unsatisfactory results on other datasets. Conversely, on a more practical biomedical dataset with insufficient annotated training data, Bio-RFX performs better.

We can draw several conclusions from the observations. Firstly, Bio-RFX achieves superior per-

Table 1: The average micro F1 scores (%) and ranks of models calculated over 5 runs on biomedical datasets. The best results are in bold, and the second-best results are in italic with an underline.

Model	DrugProt		DrugVar		BC5CDR		CRAFT	**Avg. Rank**	
	NER	RE	NER	RE	NER	RE	NER	NER	RE
TPLinker-Plus	*90.96*	70.03	79.87	62.97	89.47	72.23	93.53	4.50	5.00
KECI	87.73	**80.39**	74.55	62.96	85.76	68.44	93.62	5.75	4.67
PURE	90.63	70.00	80.59	65.26	*91.78*	**75.78**	94.17	*3.25*	3.33
SpanBioER	88.56	65.38	*81.82*	*68.21*	91.39	73.76	94.30	3.50	4.00
REBEL	-	45.71	-	59.70	-	72.62	-	-	6.33
PL-Marker	90.62	70.05	80.77	65.63	91.56	*74.68*	*94.44*	3.00	*2.67*
GPT-4	66.62	27.73	66.05	14.87	77.67	59.74	54.97	7.00	8.00
Bio-RFX	**91.75**	*70.16*	**83.16**	**70.07**	**91.96**	74.49	**94.90**	**1.00**	**2.00**

formance compared to baselines for biomedical datasets, indicating that individual encoders can effectively learn precise representations for biomedical texts. Besides, in datasets that have annotation discrepancies with knowledge bases, strong structural constraints in the biomedical domain can indeed help outperform traditional methods that fuse KGs into the model. Moreover, despite the numerous emergent abilities of large language models, designing task-specific architectures and fine-tuning remain essential for biomedical RE.

4.3 Low-Resource Setting

We conducted experiments to explore our method's effectiveness in a low-resource scenario. We randomly selected 10% and 4% samples from DrugProt, and 50% and 20% samples from DrugVar to construct new datasets. The results are shown in Table 2. Compared to previous methods, Bio-RFX improves the NER and RE F1 by up to 2.91% and 1.75% absolute across all datasets. RE in the biomedical domain under low-resource settings is challenging, and performance varies with the datasets. Bio-RFX secures an average rank of 1.00 in NER and 2.00 in RE, outperforming all models.

Compared with pipeline and joint methods, our model excels in the following aspects: (1) Dividing complicated tasks into several submodules significantly decreases the difficulty and improves the stability of training. Joint methods with intricate tagging schemes struggle with scarce training data. For instance, TPLinker-plus combines information from the whole triplet and the whole span to construct labels for span pair, resulting in 4 variants per relation type. Hence, the $4|T_r|$-class classification task contributes to great learning difficulty and significant performance drop in low-resource settings. Moreover, methods that utilize span extraction and special tokens (such as PURE and PL-Marker) exhibit poor training stability. As the size of the training set decreases from 500 to 200, the standard deviation of the RE score for PL-Marker increases to 184%, while that of Bio-RFX rises to an average of 99%. On the contrary, our divide-and-conquer philosophy is more effective because task-specific representation helps to achieve better performance and stabilize the training process. (2) KG-enhanced joint methods are affected by noisy prior knowledge from KGs when training data is limited. In biomedical datasets, the definition for a *null* entity varies greatly, as specific entities (e.g., qualitative concepts such as *revealed* or *active*) are likely to be considered as a *null* entity if they are not the primary focus of the dataset. Comprehensive KGs incorrectly recognize these entities when training samples are small. To support this argument, we find that KECI has lower precision and higher recall across the experiments, while our model shows the opposite. Using an extensive knowledge base as prior knowledge in low-resource scenarios leads to overfitting to KGs, and constraining the hypothesis space of the model is a much more preferable alternative. (3) Generative models linearize triplets into a sequential order, posing challenges for overlapping triplets in biomedical literature. Although in NER, GPT-4 can achieve comparable performance with models fine-tuned on specific datasets, the performance gap in RE is intolerable. Relation extraction, which aims to identify interactions between entities, might not be suitable to be directly formulated as a sequence generation task. A classification approach like Bio-RFX is more effective.

Table 2: The average micro F1 scores (%) and ranks of models calculated over 5 runs on biomedical datasets under a low-resource setting. The best results are in bold, and the second-best results are in italic with an underline. The number in the bracket indicates the approximate size of the training set.

Model	DrugVar (500)		DrugVar (200)		DrugProt (500)		DrugProt (200)		**Avg. Rank**	
	NER	RE	NER	RE	NER	RE	NER	RE	NER	RE
TPLinker-Plus	76.99	59.38	69.35	13.42	83.88	48.39	81.64	28.17	4.50	6.00
KECI	73.12	59.23	65.37	50.88	75.06	41.87	71.62	39.07	6.00	5.00
PURE	76.69	58.34	72.63	48.77	*89.86*	**59.60**	83.96	54.58	3.50	3.25
SpanBioER	*78.16*	*60.42*	73.15	48.49	87.43	51.02	82.14	41.59	3.25	4.25
REBEL	-	55.78	-	47.11	-	53.30	-	51.91	-	5.25
PL-Marker	76.79	56.66	*73.58*	**51.44**	89.46	*58.41*	*86.10*	**56.67**	*2.75*	*2.50*
GPT-4	61.86	12.62	61.97	6.94	67.29	26.25	69.80	32.26	7.00	7.75
Bio-RFX	**80.64**	**62.17**	**73.80**	*51.23*	**89.90**	54.37	**89.01**	*56.20*	**1.00**	**2.00**

We observe that Bio-RFX performs better on DrugProt (200) than DrugProt (500), likely due to their statistical differences. The average relation triplets per sentence for DrugProt, DrugProt (500), and DrugProt (200) are 2.7, 1.2, and 2.3, respectively. The sparsity of relation triplets hampers the relation classifier's performance, creating a bottleneck in overall extraction.

4.4 Ablation Study

This subsection examines the impact of structural constraints and the number predictor in our framework. Table 3 presents the micro F1 scores of the ablated and full models.

Bio-RFX (- SC / Structural Constraints) removes the structural constraints for relation triplet generation. Instead of enumerating each $\langle \tau_{es}, \tau_{eo} \rangle \in T_{re}$ for relation τ_r to produce relation triplets, we regard each entity pair in $T_{ev} \times T_{ev}$ as a subject-object pair for relation τ_r, where T_{ev} is the set of valid and not-*null* entities. Structural constraints only affect relation triplet generation, leaving NER results unchanged.

Bio-RFX (- NP / Number Predictor) removes the number predictor and uses the average number of entities in a sentence as the threshold for the text-NMS algorithm during inference.

Bio-RFX (- PA, NP / Pruning Algorithm, Number Predictor) removes the text-NMS pruning algorithm, and uses the output of the entity detector as the final results. In this case, the prediction of the number predictor is not utilized.

The results suggest that structural constraints, number prediction, and the pruning algorithm are crucial for enhancing the model's performance. Of these, the strong structural constraints between entity types and relation types are the most beneficial. This highlights the model's ability to handle complex entities while leveraging the structural constraints of relation triplets in biomedical literature.

To assess the model's comprehension of ambiguous biomedical entities, we study several typical cases. The results are presented in Appendix H.

5 Conclusion

This paper introduces Bio-RFX, a novel biomedical entity and relation extraction method, using structural constraints for relation triplets to constrain the hypothesis space. The model tackles ambiguous entities and redundant relation prediction using a relation-first extraction approach, and uses a heuristic pruning algorithm for precise recognition of complex overlapping entity spans. Experimental results on real-world biomedical datasets with abundant and limited training data show that Bio-RFX outperforms the state-of-the-art methods in NER, and has highly competitive performance in RE.

6 Limitations

Despite the significant advancements in biomedical entity and relation extraction, several challenges persist. Our work has certain limitations that provide avenues for future exploration:

1. The current capability of Bio-RFX is limited to using structural constraints obtained by statistical features. Future work could expand this by incorporating other knowledge representation methods.

Table 3: Ablation study on biomedical datasets. Results with blue backgrounds indicate inferior performance.

Dataset		Bio-RFX	Bio-RFX (- SC)	Bio-RFX (- NP)	Bio-RFX (- PA, NP)
DrugVar	NER	83.16	83.16	83.72	81.94
	RE	70.07	37.10	71.61	71.12
DrugVar (500)	NER	80.64	80.64	79.81	79.79
	RE	62.17	29.25	60.13	58.48
DrugVar (200)	NER	73.80	73.80	73.03	73.56
	RE	51.23	25.56	49.07	52.57
DrugProt	NER	91.75	91.75	90.62	82.88
	RE	70.16	26.91	64.45	53.71
DrugProt (500)	NER	89.90	89.90	89.66	75.58
	RE	54.37	19.62	53.73	51.58
DrugProt (200)	NER	89.01	89.01	90.93	80.41
	RE	56.20	26.36	47.41	54.44

2. The method's effectiveness in generating questions or hints for relation-specific tasks could be improved. This would allow for better utilization of the rich semantic information provided by pre-trained encoders.

3. The pipeline training approach used by Bio-RFX may lead to error propagation, causing a discrepancy between training and testing. This issue will be addressed in future work.

Acknowledgements

This work was supported in part by *National Key Research and Development Program of China under Grant No. 2020YFA0804503*, *National Natural Science Foundation of China under Grant No. 62272264*, *National Natural Science Foundation of China, NO. 62306300*, *Fundamental Research Funds for the Central Universities, NO. NJ2024029*, and Beijing Academy of Artificial Intelligence (BAAI).

References

Anson Bastos, Abhishek Nadgeri, Kuldeep Singh, Isaiah Onando Mulang, Saeedeh Shekarpour, Johannes Hoffart, and Manohar Kaul. 2021. Recon: Relation extraction using knowledge graph context in a graph neural network. In *Proceedings of the Web Conference 2021*, WWW '21, page 1673–1685, New York, NY, USA. Association for Computing Machinery.

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

Olivier Bodenreider. 2004. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32(suppl_1):D267–D270.

K. Bretonnel Cohen, Karin Verspoor, Karën Fort, Christopher Funk, Michael Bada, Martha Palmer, and Lawrence E. Hunter. 2017. *The Colorado Richly Annotated Full Text (CRAFT) Corpus: Multi-Model Annotation in the Biomedical Domain*, pages 1379–1394. Springer Netherlands, Dordrecht.

Janez Demšar. 2006. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7(1):1–30.

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.

Rodrigo Dienstmann, In Sock Jang, Brian Bot, Stephen Friend, and Justin Guinney. 2015. Database of Genomic Biomarkers for Cancer Drugs and Clinical Targetability in Solid Tumors. *Cancer Discovery*, 5(2):118–123.

Hermenegildo Fabregat, Andres Duque, Juan Martinez-Romo, and Lourdes Araujo. 2023. Negation-based transfer learning for improving biomedical named entity recognition and relation extraction. *Journal of Biomedical Informatics*, 138:104279.

Hao Fei, Yue Zhang, Yafeng Ren, and Donghong Ji. 2020. A span-graph neural model for overlapping entity relation extraction in biomedical texts. *Bioinformatics*, 37(11):1581–1589.

Matthew R. Gormley, Mo Yu, and Mark Dredze. 2015. Improved relation extraction with feature-rich compositional embedding models. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1774–1784, Lisbon, Portugal. Association for Computational Linguistics.

Xu Han, Zhiyuan Liu, and Maosong Sun. 2016. Joint representation learning of text and knowledge for knowledge graph completion.

Minghao Hu, Yuxing Peng, Zhen Huang, and Dongsheng Li. 2019. A multi-type multi-span network for reading comprehension that requires discrete reasoning. 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 1596–1606, Hong Kong, China. Association for Computational Linguistics.

Pere-Lluís Huguet Cabot and Roberto Navigli. 2021. REBEL: Relation extraction by end-to-end language generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2370–2381, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization.

Tibor Kiss and Jan Strunk. 2006. Unsupervised multilingual sentence boundary detection. *Comput. Linguist.*, 32(4):485–525.

Tuan Lai, Heng Ji, ChengXiang Zhai, and Quan Hung Tran. 2021. Joint biomedical entity and relation extraction with knowledge-enhanced collective inference. 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 6248–6260, Online. Association for Computational Linguistics.

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

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. 2016. BioCreative V CDR task corpus: a resource for chemical disease relation extraction. *Database*, 2016:baw068.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019. Entity-relation extraction as multi-turn question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1340–1350, Florence, Italy. 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.

OpenAI. 2023. Gpt-4 technical report.

Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen-tau Yih. 2017. Cross-sentence n-ary relation extraction with graph LSTMs. *Transactions of the Association for Computational Linguistics*, 5:101–115.

Yu-Ming Shang, Heyan Huang, and Xianling Mao. 2022. Onerel: Joint entity and relation extraction with one module in one step. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):11285–11293.

Dianbo Sui, Xiangrong Zeng, Yubo Chen, Kang Liu, and Jun Zhao. 2023. Joint entity and relation extraction with set prediction networks. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–12.

Chenguang Wang, Xiao Liu, Zui Chen, Haoyun Hong, Jie Tang, and Dawn Song. 2022. DeepStruct: Pretraining of language models for structure prediction. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 803–823, Dublin, Ireland. Association for Computational Linguistics.

Yucheng Wang, Bowen Yu, Yueyang Zhang, Tingwen Liu, Hongsong Zhu, and Limin Sun. 2020. TPLinker: Single-stage joint extraction of entities and relations through token pair linking. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1572–1582, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Zhuo Wang, Wei Zhang, Ning Liu, and Jianyong Wang. 2024. Learning interpretable rules for scalable data representation and classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(2):1121–1133.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Zhepei Wei, Jianlin Su, Yue Wang, Yuan Tian, and Yi Chang. 2020. A novel cascade binary tagging framework for relational triple extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1476–1488, Online. Association for Computational Linguistics.

Chenhao Xie, Jiaqing Liang, Jingping Liu, Chengsong Huang, Wenhao Huang, and Yanghua Xiao. 2021. Revisiting the negative data of distantly supervised relation extraction. 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 3572–3581, Online. Association for Computational Linguistics.

Deming Ye, Yankai Lin, Peng Li, and Maosong Sun. 2022. Packed levitated marker for entity and relation extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4904–4917, Dublin, Ireland. Association for Computational Linguistics.

Bowen Yu, Zhenyu Zhang, Xiaobo Shu, Yubin Wang, Tingwen Liu, Bin Wang, and Sujian Li. 2020. Joint extraction of entities and relations based on a novel decomposition strategy. In *Proc. of ECAI*.

Hongyi Yuan, Zheng Yuan, Ruyi Gan, Jiaxing Zhang, Yutao Xie, and Sheng Yu. 2022. BioBART: Pretraining and evaluation of a biomedical generative language model. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 97–109, Dublin, Ireland. Association for Computational Linguistics.

Jiawen Zhang, Jiaqi Zhu, Yi Yang, Wandong Shi, Congcong Zhang, and Hongan Wang. 2021. Knowledge-enhanced domain adaptation in few-shot relation classification. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21, page 2183–2191, New York, NY, USA. Association for Computing Machinery.

Hengyi Zheng, Rui Wen, Xi Chen, Yifan Yang, Yunyan Zhang, Ziheng Zhang, Ningyu Zhang, Bin Qin, Xu Ming, and Yefeng Zheng. 2021. PRGC: Potential relation and global correspondence based joint relational triple extraction. 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 6225–6235, Online. Association for Computational Linguistics.

Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. 2017. Joint extraction of entities and relations based on a novel tagging scheme. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1227–1236, Vancouver, Canada. 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.

A Prompt Techniques

We have explored the following prompt techniques. However, incorporating these prompt modules has negatively impacted the model's performance. In contrast, our designed question template turned out to be more effective.

A.1 Term Definitions

We enrich the question with definitions of types of entities and relations to provide the model with semantic information in the biomedical domain. For instance, the relation-specific question *What gene does the chemical activate?* is followed by the definition of activator obtained from the Free Medical Dictionary[2], i.e., *An activator is a substance that makes another substance active or reactive, induces a chemical reaction, or combines with an enzyme to increase its catalytic activity.* The results are shown in Table 4, i.e. Bio-RFX (+Definition). It can be observed that the micro F1 scores for NER and RE decreased. We believe the contextualized knowledge representation during the pre-training process is sufficient, and the rigid definitions merely introduce noise to data distribution.

A.2 UMLS Markers

External biomedical knowledge is also considered when designing prompts. We use UMLS Metamap, a handy toolkit based on a biomedical knowledge graph, to match the biomedical terms in the text and insert unique markers both before and after the terms. Take the following sentence as an example.

> Some clinical evidences suggested that pindolol can be effective at producing a shortened time to onset of antidepressant activity.

In this sentence, *pindolol* is recognized by Metamap as a pharmacologic substance. When type-specific markers are used, the result is:

[2] https://medical-dictionary.thefreedictionary.com/

Table 4: The absolute differences in micro F1 (%) after adding term definitions in prompts.

Dataset		Bio-RFX (+ Definition)
DrugVar	NER	-0.26
	RE	0.42
DrugVar(500)	NER	-1.33
	RE	-0.68
DrugVar(200)	NER	-3.71
	RE	-5.33
DrugProt	NER	-1.08
	RE	-14.43
DrugProt(500)	NER	-1.00
	RE	-5.08
DrugProt(200)	NER	1.92
	RE	4.70

> Some clinical evidences suggested that <DRUG> pindolol </DRUG> can be effective at producing a shortened time to onset of antidepressant activity.

On the DrugProt dataset, we observed a 3.02% and 6.45% decrease in micro F1 scores for NER and RE, respectively. Several reasons may contribute to this experience results. To begin with, the entity types in Metamap and the entity types in the datasets are quite different, posing a challenge for entity linking. Another reason is that the matching method is mainly based on the syntax tree and searching, thus the matching accuracy is not satisfactory. In the following example, the term *of* is erroneously identified as a gene (OF (TAF1 wt Allele)) due to its ambiguous nature, which subsequently hampers the overall performance. Moreover, Metamap extracts all the entities without being conscious of the relation type expressed in the sentence, misleading our entity detector.

> ... <CHEMICAL> isoprenaline </CHEMICAL> - induced maximal relaxation (E (max)) <GENE> **of** </GENE> <CHEMICAL> methacholine </CHEMICAL> - contracted preparations in a concentration dependent fashion ...

B Textual NMS Algorithm

A detailed description of the algorithm is presented in Algorithm 1.

Algorithm 1 Textual Non-Maximum Suppression

```
Require: spans $S$, span number threshold $k$;
Ensure: pruned spans $\hat{S}$;
  Sort $S$ in descending order of span scores;
  $\hat{S} = \{\}$;
  while $S \neq \{\}$ and $|\hat{S}| < k$ do
      for $s_i$ in $S$ do
          $\hat{S} = \hat{S} \cup \{s_i\}$;
          $S = S - \{s_i\}$;
          for $s_j$ in $S$ do
              if F1$(s_i,\ s_j) > 0$ then
                  $S = S - \{s_j\}$;
              end if
          end for
      end for
  end while
```

C Datasets and Preprocessing

We will briefly review all the datasets below and state the preprocessing methods we have applied. All the datasets we use are publicly available and designed to advance research in information extraction. The statistics of the datasets are listed in Table 5.

1. **DrugVar** is a subset of N-ARY datasets proposed in Peng et al. (2017) and mainly focuses on extracting fine-grained interactions between drugs and variants. The dataset was constructed by first obtaining biomedical literature from PubMed Central[3] and then identifying entities and relations with distant supervision from Gene Drug Knowledge Database (Dienstmann et al., 2015) and Clinical Interpretations of Variants In Cancer[4] knowledge bases. The original dataset is designed for document-level information extraction, which is beyond the scope of this paper. Thus we split the texts into sentences with Punkt (Kiss and Strunk, 2006) sentence tokenizer and ignore all the cross-sentence relation triplets.

2. **DrugProt** is a track in BioCreative VII and focuses on extracting a variety of important associations between drugs and genes/proteins to understand gene regulatory and pharmacological mechanisms. The data is collected

[3] http://www.ncbi.nlm.nih.gov/pmc/

[4] http://civic.genome.wustl.edu/

Table 5: Statistics of datasets.

Dataset	#Ent Type	#Rel Type	#Ent	#Rel	#Train	#Valid
DrugProt	3	6	40,185	20,800	6,273	1,377
DrugVar	3	4	2,760	1,583	929	267
BC5CDR	2	1	11,241	5,520	2,184	1,131
CRAFT (SO)	198	N/A	11,932	N/A	5,062	1,890

from PubMed abstracts and then manually labeled by domain experts. We also perform sentence segmentation during preprocessing. We merge some of the relation types so that all the refined relation labels are at the same level in the relation concept hierarchy.

3. **BC5CDR** is the supporting corpus of the BioCreative V Chemical Disease Relation (CDR) task. It aims to extract chemical-disease relations from PubMed articles. In a similar manner, the articles are split into sentences for sentence-level extraction tasks.

4. **CRAFT** is a manually annotated corpus consisting of 97 full-text biomedical journal articles. Each article is a member of the PubMed Central Open Access Subset[5]. The subtask for Sequence Ontology (SO) is utilized in this work. It's important to note that this dataset lacks relation annotations, hence, only Named Entity Recognition (NER) is performed.

D Implementation Details

For a fair comparison, all the BERT-based models use *scibert-scivocab-cased* (Beltagy et al., 2019) as the pre-trained Transformer encoder. REBEL(Huguet Cabot and Navigli, 2021) uses *BioBART-base* (Yuan et al., 2022) as the pre-trained encoder.

We consider spans with up to L = 8 words, which covers 97.89% of the entities on average in the datasets. We train our models with Adam (Kingma and Ba, 2017) optimizer of a linear scheduler with a warmup ratio of 0.1. We train the relation classifier, entity detector, and number predictor for 100 epochs, and a learning rate of 1e-5 and a batch size of 8. We use gold relations and entity numbers to train the entity detector and the predicted relations and numbers during inference. To be more specific, for each relation, if the probability obtained by the relation classifier is above the relation-specific threshold, then the sentence will be classified as positive, which means the sentence is expressing this relation. Otherwise, it will be classified as negative. The relation-specific threshold can be optimized by maximizing the classification F1 score on the validation set.

The training process of each component takes 12 hours at most on one NVIDIA GeForce RTX 3090. The model sizes of the relation classifier, entity detector, and number predictor are 420MB, 423MB, and 434MB respectively.

E Experimental Settings of GPT-4

With the rapid development of Large Language Models (LLMs), it is necessary to discuss the potential of LLMs for our task. We choose GPT-4 (OpenAI, 2023) to jointly conduct few-shot NER and RE on biomedical texts.

To inform GPT-4 about its role and our task, we first send a system message, i.e. *You are stepping into the role of an expert assistant specialized in biomedicine. Your primary task is to accurately extract entities and relations from biomedical texts and respond to users' queries with clear, concise, and precise answers.*

After the system message, we provide GPT-4 with 5 examples. Each example contains a question section and an answer section. A question section consists of four parts:

1. The biomedical text where we extract entities and relations.

2. The entity and relation types specified by the dataset.

3. The structural constraints between the entity and relation types.

4. A question guiding GPT-4 to provide the answer.

An answer section consists of two parts:

[5] http://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/

1. The entities detected from the text. To facilitate entity extraction, we inform GPT-4 to generate highly structured answers, e.g. `<BCRP|GENE>` represents an entity *BCRP* of type GENE. In practice, we perform Chain of Thought (Wei et al., 2022) prompting to enhance accuracy.

2. The relation triplets extracted from the text. Similar to entity detection, GPT-4 intends to generate structured answers, e.g. `<Menthol|CHEMICAL|TRPM8|GENE|activator>` represents an *activator* relation, whose subject and object are *Menthol* and *TRPM8*.

Finally, we form a question section based on the biomedical text and send it to GPT-4. We perform regular expression matching on the response message to retrieve the answers. The evaluation metrics are consistent with the previous sections, i.e. an entity is considered matched if the whole span and entity type match the ground truth, and a relation triplet is regarded correct if the relation type and both subject entity and object entity are all correct. The source code is publicly available at `https://github.com/Tschal-rsa/bio-re-gpt`.

F Error Analysis of GPT-4

Upon examining the predictions from GPT-4, we identified two primary types of errors:

1. Misinterpretation of labels based on their literal meaning. For instance, in the field of biology, an 'activator' is a protein that enhances the transcription of a gene or a set of genes. However, GPT-4 tends to interpret it as a substance that activates or makes other substances operative. If a text indicates that gene A facilitates gene B, the model incorrectly classifies the triplet as (gene A, activator, gene B). Our approach mitigates this issue by fine-tuning the model to provide context for the label, and by applying structural constraints, such as recognizing that activation only occurs between a chemical and a gene.

2. Coreference issues leading to a low recall rate. GPT-4 may merge triplets if the subjects (or objects) in the triplets are actually different names of the same biomedical entity. While this approach is reasonable to some degree, it introduces an additional challenge for recognizing coreference. Fine-tuning GPT-4 to explicitly generate all the names in the prediction might address this problem.

G Significance Tests

In this section, we detail the significance test between Bio-RFX and baselines. Note that we exclude GPT-4 from our baselines here since it is not feasible to fine-tune it on our datasets.

The details of the experiments are addressed as follows. First, we choose 5 seeds randomly, train Bio-RFX and all the baseline models with each seed, and record the corresponding performances. Then, we perform one-tailed paired t-tests between Bio-RFX and each baseline model. Considering that the sample size is small, we also perform one-tailed Wilcoxon signed rank test, which is a non-parametric counterpart of the paired samples t-test. For each baseline model:

1. We compute the difference in performance between Bio-RFX and the baseline model so that we obtain 5 difference measures d_i $(i = 1, 2, \ldots, 5)$.

2. We compute the t statistic under the null hypothesis that Bio-RFX and the compared baseline have equal performance:

$$t = \frac{\bar{d} - 0}{s/\sqrt{5}} = \frac{\sqrt{5}\bar{d}}{\sqrt{\frac{1}{4}\sum_{i=1}^{5}(d_i - \bar{d})^2}},$$

where $\bar{d}$ and s are the sample mean and standard deviation of the difference measures, respectively.

3. We compute the Wilcoxon signed-rank test statistic W under the same null hypothesis:

$$R^+ = \sum_{i=1}^{n} rank(d_i) \quad \text{for} \quad d_i > 0$$

$$R^- = \sum_{i=1}^{n} rank(d_i) \quad \text{for} \quad d_i < 0$$

$$W = \min\left(\sum_{i=1}^{n} R_i^+, \sum_{i=1}^{n} R_i^-\right)$$

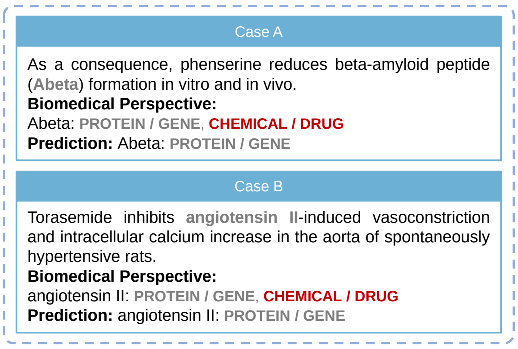
4. For t-test, we compute the p-value and compare it to the significance level $\alpha = 0.05$. If the $p_t < 0.05$ or $t > 2.132$, we reject the null hypothesis.

5. For Wilcoxon signed rank test, we compute the p-value and compare it to the significance level $\alpha = 0.1$, because 0.1 is the minimum p-value for 5 samples. If the $p_W < 0.1$ or $W = 0$, we reject the null hypothesis.

The statistics and p-values between Bio-RFX and the baseline models are shown in Table 6 and 7. We can observe that for t-test, most of the p-values are below $\alpha = 0.05$ (and the corresponding t statistics are above 2.132), rejecting the null hypothesis under both general and low-resource settings.

H Case Study

Here we present several cases to gain deeper insights into the model's ability to handle ambiguous entities.



Case A

As a consequence, phenserine reduces beta-amyloid peptide (**Abeta**) formation in vitro and in vivo.
Biomedical Perspective:
Abeta: **PROTEIN / GENE**, **CHEMICAL / DRUG**
Prediction: Abeta: **PROTEIN / GENE**

Case B

Torasemide inhibits **angiotensin II**-induced vasoconstriction and intracellular calcium increase in the aorta of spontaneously hypertensive rats.
Biomedical Perspective:
angiotensin II: **PROTEIN / GENE**, **CHEMICAL / DRUG**
Prediction: angiotensin II: **PROTEIN / GENE**

Figure 3: Case study for ambiguous biomedical entities.

Figure 3 illustrates cases of ambiguous entities in the DrugProt dataset. In case A, *Abeta* is a chemical in the form of a peptide, as well as processed from the Amyloid precursor protein. In case B, *angiotensin II* is both a medication used to increase blood pressure and a type of protein. Since DrugProt focuses on extracting drug-gene/protein interactions, both of them are considered to be proteins in the context. With the structural constraints, our model can correctly predict the ground truth labels.

Table 6: Significance tests on biomedical datasets. Results with blue backgrounds indicate that Bio-RFX significantly outperforms the baseline model.

Model		DrugProt		DrugVar		BC5CDR		CRAFT
		NER	RE	NER	RE	NER	RE	NER
TPLinker-Plus	t	9.31	0.64	5.40	5.52	16.58	10.74	9.00
	p_t	0.0004	0.2789	0.0028	0.0026	0.0000	0.0002	0.0004
	W	0	5	0	0	0	0	0
	p_W	0.0625	0.6250	0.0625	0.0625	0.0625	0.0625	0.0625
KECI	t	5.32	-5.14	33.76	5.99	46.91	40.75	15.29
	p_t	0.0030	0.0034	0.0000	0.0020	0.0000	0.0000	0.0001
	W	0	0	0	0	0	0	0
	p_W	0.0625	0.0625	0.0625	0.0625	0.0625	0.0625	0.0625
PURE	t	17.54	0.51	8.13	8.78	4.03	-8.82	24.35
	p_t	0.0000	0.3177	0.0006	0.0005	0.0079	0.0005	0.000
	W	0	7	0	0	0	0	0
	p_W	0.0625	0.8125	0.0625	0.0625	0.0625	0.0625	0.0625
SpanBioER	t	41.94	17.98	3.76	2.39	4.43	2.17	8.14
	p_t	0.0000	0.0000	0.0099	0.0375	0.0057	0.0481	0.0006
	W	0	1	0	0	0	3	0
	p_W	0.0625	0.1250	0.0625	0.0625	0.0625	0.3125	0.0625
REBEL	t	-	65.89	-	13.21	-	7.24	-
	p_t	-	0.0000	-	0.0001	-	0.0010	-
	W	-	0	-	0	-	0	-
	p_W	-	0.0625	-	0.0625	-	0.0625	-
PL-Marker	t	10.34	0.28	6.43	2.38	2.72	-0.61	6.71
	p_t	0.0002	0.3981	0.0015	0.0381	0.0264	0.2872	0.0013
	W	0	6	0	0	0	6	0
	p_W	0.0625	0.8125	0.0625	0.0625	0.0625	0.8125	0.0625

Table 7: Significance tests on biomedical datasets under low-resource setting. Results with blue backgrounds indicate that Bio-RFX significantly outperforms the baseline model.

Model		DrugVar(500)		DrugVar(200)		DrugProt(500)		DrugProt(200)	
		NER	RE	NER	RE	NER	RE	NER	RE
TPLinker-Plus	t	8.34	3.92	3.26	9.97	3.67	4.42	7.37	18.62
	p_t	0.0006	0.0086	0.0155	0.0003	0.0106	0.0057	0.0009	0.0000
	W	0	0	0	0	0	0	0	0
	p_W	0.0625	0.0625	0.0625	0.0625	0.0625	0.0625	0.0625	0.0625
KECI	t	10.57	2.85	7.10	0.26	16.16	4.54	16.61	17.61
	p	0.0002	0.0232	0.0010	0.4035	0.0000	0.0052	0.0000	0.0000
	W	0	0	0	7	0	0	0	0
	p_W	0.0625	0.0625	0.0625	1.0000	0.0625	0.0625	0.0625	0.0625
PURE	t	23.44	2.95	1.96	1.72	0.20	-3.37	13.69	2.11
	p_t	0.0000	0.0210	0.0605	0.0804	0.4243	0.0140	0.0001	0.0512
	W	0	0	1	1	5	1	0	1
	p_W	0.0625	0.0625	0.1250	0.1250	0.6250	0.1250	0.0625	0.1250
SpanBioER	t	12.25	3.30	1.37	5.03	18.10	3.71	19.22	26.16
	p_t	0.0001	0.0149	0.1209	0.0037	0.0000	0.0103	0.0000	0.0000
	W	0	0	4	0	0	0	0	0
	p_W	0.0625	0.0625	0.4375	0.0625	0.0625	0.0625	0.0625	0.0625
REBEL	t	-	5.23	-	2.42	-	0.87	-	3.13
	p_t	-	0.0032	-	0.0364	-	0.2155	-	0.0176
	W	-	0	-	1	-	4	-	0
	p_W	-	0.0625	-	0.1250	-	0.4375	-	0.0625
PL-Marker	t	10.41	9.88	0.39	-0.15	1.60	-6.75	6.48	-0.54
	p_t	0.0002	0.0003	0.3599	0.4439	0.0921	0.0013	0.0015	0.3099
	W	0	0	6	7	2	0	0	6
	p_W	0.0625	0.0625	0.8125	1.000	0.1875	0.0625	0.0625	0.8125