

# PETAILOR: Improving Large Language Model by Tailored Chunk Scorer in Biomedical Triple Extraction

Mingchen Li, M.Chen, Huixue Zhou, Rui Zhang

University of Minnesota Twin Cities  
{li003378, zhou1742, zhan1386}@umn.edu

## Abstract

The automatic extraction of biomedical entities and their interaction from unstructured data remains a challenging task due to the limited availability of expert-labeled standard datasets. In this paper, we introduce PETAILOR, a retrieval-based language framework that is augmented by tailored chunk scorer. Unlike previous retrieval-augmented language models (LM) that retrieve relevant documents by calculating the similarity between the input sentence and the candidate document set, PETAILOR segments the sentence into chunks and retrieves the relevant chunk from our pre-computed chunk-based relational key-value memory. Moreover, in order to comprehend the specific requirements of the LM, PETAILOR adapts the tailored chunk scorer to the LM. We also introduce GM-CIHT, an expert annotated biomedical triple extraction dataset with more relation types. This dataset is centered on the non-drug treatment and general biomedical domain. Additionally, we investigate the efficacy of triple extraction models trained on general domains when applied to the biomedical domain. Our experiments reveal that PETAILOR achieves state-of-the-art performance on GM-CIHT.<sup>1</sup>

## 1 Introduction

Biomedical triple extraction aims to predict triples that are related to a given sentence, where each triple can be described with a (*head entity, relation, tail entity*). It's an important task for knowledge graph construction (Fellbaum, 2010; Li et al., 2020) and its downstream applications, such as link prediction (Li et al., 2022), drug repurposing (Zhang et al., 2021), question answering (Li and Ji, 2022), etc.

Despite significant progress in recent years, the challenge of obtaining high-quality biomedical

triple extraction datasets with more valuable relation types by medical experts places constraints on model development. For instance, the BioRelEx dataset (Khachatryan et al., 2019) contains only three relation types, while the ADE dataset (Gurulingappa et al., 2012) includes just one relation type.

Regarding another challenge, the majority of efforts about triple extraction are directed towards employing either the fill-formed method (Tang et al., 2022; Shang et al., 2022) or generation methods (Lu et al., 2022; Gao et al., 2023a). Particularly, UniRel (Tang et al., 2022) employs the output probability matrix of an Auto-Encoding language model to assess both entity-entity interactions and entity-relation interactions, while UIE (Lu et al., 2022) transfer the different information extraction tasks such as named entity recognition (Li and Zhang, 2023; Li et al., 2023a), relation extraction (Sui et al., 2023; Chen et al., 2023), triple extraction (Shang et al., 2022; Lu et al., 2022) to a unified generation framework. However, the complexity of medical sentence descriptions and the reliance on domain-specific knowledge for relation definitions make it challenging for the model to accurately predict the correct relation type. This observation is further substantiated by our experiments. (refer to Section 6.2).

Alternatively, as the development of the language model, the retrieval-augmented language model (Lewis et al., 2020; Li and Huang, 2023) can retrieve knowledge from an external datastore when needed, potentially reducing the difficulty of relation identification. This observation is further substantiated by our experiments. (refer to Section 6.2). Previous methods involving retrieval-augmented language models typically require a fixed retriever, such as K-nearest neighbors (KNN), to retrieve the most relevant document for the input sentence. However, this approach lacks tunability to adapt LM. Instead, some approaches treat

<sup>1</sup>The data, code, and API will be available here: <https://github.com/ToneLi/PETAILOR-for-bio-triple-extraction>.

the LM as a black box and allow for tunability in the retriever itself, as demonstrated by (Shi et al., 2023). However, these methods mainly perform knowledge retrieval from unlabelled sentence level, which means that potentially retrievable information is not effectively utilized for triple extraction.

In this study, we initially introduce a novel biomedical triple extraction dataset GM-CIHT (General BioMedical and Complementary and Integrative Health<sup>2</sup> Triple) characterized by its high-quality annotations and comprehensive coverage of relation types, this dataset is created by repurposing an established expert-annotated dataset and annotating 2,450 sentences of complementary and integrative health from two medical experts. Additionally, we introduce PETAILOR (Personal Tailor), a new retrieval-argument LM framework that adapts the tailored chunk scorer to LM. Taking inspiration from the fact that only consecutive words (chunks) in the sentence are relevant to the relation type. The knowledge from pre-constructed relational Key-Value Memory is in the format of (relation description chunk, relation type). Given an input context, PETAILOR first constructs the relational Key-Value Memory about the relation description chunk and its relevant relation type. Then, PETAILOR retrieves the most relevant key-value pair from the constructed Key-Value Memory for the input context by using our proposed tailored chunk scorer. After that, The retrieved pair are added to the input context and provided as input to the tunable LM. In our work, the chunk scorer is trained by leveraging the supervision signals from the LM. The core idea is to customize the chunk scorer to align with the LM, creating a personalized and tailored chunk retrieval model for the LM. The experimental results demonstrate the effectiveness of our proposed PETAILOR framework with a significant improvement over the strong baselines. The contributions of this work can be summarized as:

- We introduce a biomedical triple extraction dataset GM-CIHT with high-quality annotations and comprehensive coverage of relation types.
- To the best of our knowledge, we are the first to incorporate the external (chunked relation description, relation type) knowledge into LM.

<sup>2</sup>In our paper, it is same as the concept: non-drug therapy.

- We introduce a tailored chunk scorer to adapt the needs of LM by leveraging the LM output as a signal.
- We conduct a thorough analysis of our method, including an ablation study, demonstrating the robustness of our framework.

## 2 Related Work

### 2.1 Biomedical Triple Extraction Datasets

Biomedical triple extraction plays a crucial role in the construction of the medical knowledge graph and its downstream applications. However, there is a notable scarcity of prior research (Khachatrian et al., 2019; Gurulingappa et al., 2012; Taboureau et al., 2010; Segura-Bedmar et al., 2013; Gao et al., 2019; Lee et al., 2013; Van Mulligen et al., 2012; Cheng et al., 2008) focused on biomedical dataset building through expert annotation. For example, the BioRelEx (Khachatrian et al., 2019) annotates 2,010 sentences extracted from biomedical literature, specifically addressing binding interactions involving proteins and/or biomolecules. This dataset is annotated for three distinct types of binding interaction relations. ADE (Gurulingappa et al., 2012) involves the manual annotation of 4,272 sentences from medical reports, specifically focusing on descriptions of drug-related adverse effects. In this dataset, two entity types (Adverse-Effect and Drug) are predefined, with a single relation type (Adverse-Effect) being identified. Despite the success, the relation type coverage and data quantity is still a big challenge. Our work addresses these concerns by introducing GM-CIHT dataset as well as proving the high relation type coverage with expert annotation.

### 2.2 Triple Extraction Methods

Most previous studies about triple extraction are directed towards employing either the fill-formed method (Tang et al., 2022; Shang et al., 2022; Liu et al., 2023) or generation methods (Lu et al., 2022; Gao et al., 2023a; Fei et al., 2022; Lou et al., 2023; Tan et al., 2022). For example, UniRel (Tang et al., 2022) models entity-entity interactions and entity-relation interactions in one single Interaction Map, which is predicted by the output probability matrix of auto-encoding language models, such as BERT. OneRel (Shang et al., 2022) introduces a Rel-Spec Horns Tagging strategy to predict the entities and their interaction. (Lu et al., 2022) propose the Unified Information Extraction Framework (UIE),

which possesses the ability to universally model various IE tasks, adaptively generate specific structures, and uniformly acquire general IE capabilities from a range of knowledge sources. To address the challenges of UIE neglecting syntax structure information, such as dependency trees, Fei et al. (Fei et al., 2022) introduce LasUIE, a unified Latent Adaptive Structure-aware Information Extraction framework. E2H (Gao et al., 2023a) employs a three-stage approach to enhance entity recognition, relation recognition, and triple extraction capabilities. Compared with all these studies, our work focuses on using the retrieval-augmented language model to encourage the model to generate the triple with high accuracy.

### 2.3 Retrieval-augmented Models

Many studies (Lewis et al., 2020; Guu et al., 2020; Gao et al., 2023b; Li et al., 2023b; Ram et al., 2023; Shi et al., 2023; Li and Huang, 2023), have been proposed to using retrieved relevant information from various knowledge stores to better understand the text or generate the expected output. For example, KIRST (Li and Huang, 2023) dynamic injects retrieved entity and attribute knowledge from the knowledge graph when generating the entity or attribute in the task of entity stage changes. RAG (Lewis et al., 2020) uses the Maximum Inner Product Search (MIPS) to find the top-K documents which are combined with a query to predict the output answers. To enhance retrieval capability, REALM (Guu et al., 2020) employs a gradient-based method to reward the retriever, leading to improved prediction accuracy, while RePLUG (Shi et al., 2023) treats the language model as a black box and enhances retrieval performance by leveraging the language model’s output. COK (Li et al., 2023b) aims to create a chain of sub-questions and their corresponding queries using a large language model (LLM). Subsequently, they retrieve knowledge relevant to these sub-questions. Inspired by these studies, we retrieve the knowledge from pre-constructed (relation chunk description, relation type) memory pairs and further optimize the retriever by the language model to obtain the pairs that adapt to LM’s needs.

## 3 Problem Statement

In the context of triplet extraction, our objective is to extract the triple (*head entity, relation, tail entity*) from the input context  $x$ . To thoroughly assess

the model’s extraction capabilities, we introduce a supplementary task, relation extraction, with the objective of predicting the relation type between the provided head entity and tail within the input context. It’s crucial to emphasize that both the training and testing datasets for these tasks consist of same predefined sets of relations.

## 4 GM-CIHT

Due to the annotation costs, dataset availability is a primary bottleneck for biomedical information extraction. Annotating over a thousand pieces of data is a demanding task for experts. In order to reduce the burden of labeling, we propose repurposing an established expert-annotated dataset Medcial triple Classification Dataset (MTCD) (Zhang et al., 2021) from the broader medical domain. Simultaneously, we have also undertaken the annotation of 2,450 datasets on non-drug therapy, a task completed by two medical experts. For the required domain expertise, we enlisted the services of two medical experts who worked for 12 weeks. Each expert worked 25 hours per week at a rate of \$26 per hour.

### 4.1 Repurposing MTCD

MTCD is a dataset for the task of medical triple classification, it comprises 4,352 positives, and 2,140 negative training pairs<sup>3</sup> from SemmedDB (Kilicoglu et al., 2012) which annotated by medical expert with the prior experience in medical annotations and 20 relation types. As the nature of the task in triple classification differs from triple extraction, in our study, we initially selected 4,352 positive data instances from MTCD as the source dataset. Furthermore, since a single sentence may contain multiple triples, we cross-reference the source sentences in SemmedDB to determine how many triples are associated with each sentence and accordingly relabel the sentences. We also designed the following steps to correct the annotation errors of MTCD:

- We first remove all useless characters, such as 7 or 8 spaces between two words, which will influence the generation of a head or tail entity.
- We then make sure all the surface names of the head and tail entities are in the sentence.

<sup>3</sup>When presented with a sentence and a triple, the model’s task is to determine whether this triple is a positive or negative match for the sentence.

Relation Type	Definition
ASSOCIATED WITH	CIH therapies that have a correlation or connection with specific chemicals or genes, either directly or indirectly, without necessarily altering their function.
DISRUPTS	CIH therapies that interfere with or disturb the normal function or balance of particular chemicals or genes, either intentionally or unintentionally.
INHIBITS	CIH therapies that suppress or reduce the production, release, or activity of certain chemicals or genes.
STIMULATES	CIH therapies that promote or enhance the production, release, or activity of specific chemicals or genes.
TREATS	CIH therapies applies a remedy with the diseases or symptoms of effecting a cure or managing a condition.
DOES NOT TREAT	CIH therapies do not applies a remedy with the diseases or symptoms of effecting a cure or managing a condition.
AFFECTS	Refers to the direct or indirect influence or impact (positive or negative) that CIH therapies have on the disease or syndrome, its symptoms, or the overall well-being of the individual.

Table 1: A part of Annotation Guideline, and it follows the relation type definition.

## 4.2 Non-Drug Therapy Annotation

The CIHT (Complementary and Integrative Health Triple) dataset focuses on the relationship between complementary and integrative health (CIH) therapies and their impact on diseases, genes, gene products, and chemicals. Our definition and reference for CIH therapies are drawn from CIHLex (Zhou et al., 2023).

To gather data for our dataset, we extracted information from abstracts in the PubMed bibliographic database. We employed all the CIHLex terms as search criteria to retrieve articles related to CIH. For instance, we used phrases like *Ear acupuncture OR Laser Acupuncture* to find articles related to *Ear acupuncture*. Out of the initial pool of articles retrieved using CIHLex terms, we further refined our selection using PubTator (Wei et al., 2013) to identify abstracts containing terms related to diseases, genes, and chemicals. Subsequently, from this refined subset, we randomly selected 400 abstracts to be included in our dataset.

The process of creating and refining annotation guidelines (as shown in Table 1) involved the participation of three individuals with medical backgrounds. Notably, one of them holds a Doctor of Chiropractic (DC) degree.

To assess the agreement between annotators, two annotators independently reviewed a common set of 10% of the notes. We employed Cohen’s Kappa for token-based annotation to evaluate the inter-annotator agreement for the CIHT dataset annotations. The remaining notes were then individually annotated by each annotator separately.

## 4.3 Data Statistics and Comparison with other datasets

In Table 2, we show the statistics for GM-CIHT as well as the comparable value for seven widely-

used medical triple extraction datasets. GM-CIHT mainly differs from other datasets from three aspects: 1) It encompasses a wider range of relation types. 2) The GM-CIHT dataset comprises 4,912 labeled sentences, surpassing the size of all other datasets; 3) The relation types is characterized by a higher degree of universality and granularity. In contrast, other datasets tend to feature coarser-grained relations with relatively lower medical significance. For instance, in BioRelEx (Khachatryan et al., 2019), relation types are defined as simple protein-protein interactions, and the relation type chemical-protein interaction is defined in CHEMPROT (Taboureau et al., 2010).

Dataset	# Entities	#Relation Types	# sentences
BioRelEx (Khachatryan et al., 2019)	9,871	3	2,010
ADE (Gurulingappa et al., 2012)	11,070	2	4,272
CHEMPROT (Taboureau et al., 2010)	–	14	3,895
DDI (Segura-Bedmar et al., 2013)	13,107	5	–
COMAGC (Lee et al., 2013)	541	15	821
EUADR (Van Mulligen et al., 2012)	339	4	355
PolySearch (Cheng et al., 2008)	255	2	522
GM-CIHT	5,644	22	4,912

Table 2: Comparing GM-CIHT to the seven commonly used medical triple extraction datasets.

## 5 PETAILOR

We introduce PETAILOR, a new retrieval-augmented language model, and adapt the tailored chunk scorer to the language model. In addition to triple extraction, we also evaluate the model’s performance in the relation extraction.

As shown in Figure 1, PETAILOR first constructs the **Relational Key-Value Memory (RKVM)** from the valid/dev dataset, the *Key* corresponds to the relation description chunk, and the *Value* represents its associated relation type. Then we split each input context into several input chunks, and the top-1 relevant key-value pair for each input chunk is retrieved from the Relational Key-Value Memory by **chunk Retriever**, the relevant key-value pairs are used to construct the retrieved database for each input context and enhance diversity. Next, we employ the **Tailored Chunk Scorer** to calculate the weight value associated with each key-value pair in retrieved database and the input context. Simultaneously, both the input context and the retrieved key-value pairs are fed into the LM in parallel. The training of the Tailored Chunk Scorer is guided by **Language Model**. Finally, the key-value pair from the retrieved database with the



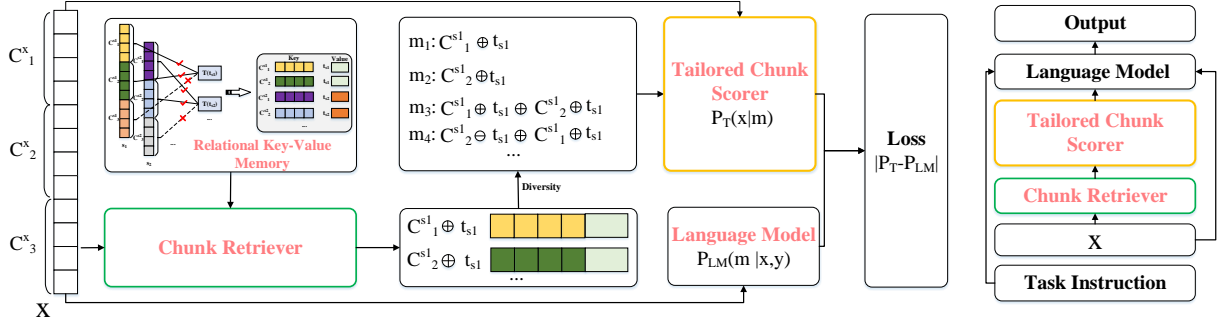


Figure 1: Overview of the PETAILOR.

highest weight score, instruction, and input context is fed into LM to predict the triple or relation.

### 5.1 Relational Key-Value Memory Construction

In contrast to previous retrieved language models that retrieve relevant information at the document or sentence level without labels, our approach focuses on retrieving information at the chunk level with labels. As comprehending the text description of relation types in chunk format, which consists of consecutive words in a sentence, can potentially improve the performance of relation identification. In our work, we build a chunk-based relational key-value memory to improve the accuracy of retrieved knowledge. More specially, we choose the valid/dev dataset as the source dataset  $D$ , which includes  $\{s, t\}$  pairs,  $s$  represents the sentence from the source dataset,  $t$  corresponds to the relation types associated with  $s$ . Then the  $s$  is split into several chunks in the length of  $m = l/v$ ,  $l$  is the length of sentence  $s$ . Next, we calculate the similarity of each relation type  $t$  and chunks in sentence  $s$ , by using the cosine similarity function  $S(\cdot)$ ,

$$S(\mathbf{E}(T(t_s)), \mathbf{E}(C_v^s))$$

where  $T(t_s)$  is the text description of relation type  $t$  which relevant to sentence  $s$ ,  $C_v^s$  is the  $v$ -th chunk in sentence  $s$ .  $E$  is the encoder function, we used MedLLaMA 13B as  $E$  in our work. Subsequently, for each *Value* corresponding to a relation type, we determine its associated *Key* by selecting the top two chunks from the sentence related to that particular *Value*. Relational Key-Value Memory  $M$  can be defined as:

$$M_s = \left\{ \underbrace{C_{top1}^s}_{key} \oplus \underbrace{t_s}_{value}, \underbrace{C_{top2}^s}_{key} \oplus \underbrace{t_s}_{value} \right\}$$

$$M = \{M_s\}$$

Such as, in Figure 1, for relation type  $t_{s1}$ , the key-value pairs are  $(C_1^{s1}, t_{s1})$  and  $(C_2^{s1}, t_{s1})$ .

### 5.2 Chunk Retriever

Given the split chunks  $\{C_i^x\}$  of the input context  $x$ ,  $C_i^x$  has the same length  $m$  with the chunk  $C_v^s$  in the Relational Key-Value Memory. the chunk Retriever aims to retrieve a small set of *key-value* (k-v) pairs from the Relational Key-Value Memory  $M$  by calculating the similarity between each *key*  $k$  in  $M$  and input chunk  $\{C_v^x\}$ . Following prior work (Li and Ji, 2022), we use a dense retriever based on the dual encoder architecture, where an encoder is used to encode each *key*  $k$  in  $M$  and each chunk  $C_i^x$  in input context  $x$ . In our work, we use MedLLaMA 13b as our encoder. Specifically, the encoder maps each *key*  $k$  and chunk  $C_i^x$  to the embeddings  $\mathbf{E}(k)$  and  $\mathbf{E}(C_i^x)$ . The similarity between the input chunk embedding and *key* embedding is computed by the cosine similarity:

$$s(k, C_i^x) = S(\mathbf{E}(k), \mathbf{E}(C_i^x))$$

Subsequently, the retrieved relational *key-value* pairs  $M_s^r$  for input context  $x$  are formed by including each *key-value* pair with the highest similarity score for each input chunk in  $s$ . More importantly, to improve the diversity of the retrieved *key-value* pair in  $M_s^r$ , we used all combinations of the *key-value* pairs in  $M_s^r$  as the diverse retrieved *key-value* pairs  $\hat{M}_s^r$ , for example, as shown in Figure 1,  $M_s^r : (C_1^{s1} \oplus t_{s1}, C_2^{s1} \oplus t_{s1}) \rightarrow \hat{M}_s^r : (C_1^{s1} \oplus t_{s1}, C_2^{s1} \oplus t_{s1}, C_1^{s1} \oplus t_{s1} \oplus C_2^{s1} \oplus t_{s1}, C_2^{s1} \oplus t_{s1} \oplus C_1^{s1} \oplus t_{s1})$ , otherwise, the sentence

from source dataset  $D$  with the highest cosine similarity with  $x$  also is regarded as the *key* in  $\hat{M}_s^r$ . To adapt various tasks, such as triple extraction, we augment each *value* in  $\hat{M}_s^r$  by adding the head entity and tail entity. This is done by randomly selecting the head and tail entity associated with the *value* from the source dataset  $D$ . The chosen head or tail entity is then appended to both the beginning and end positions of the *key* (relational chunk). after that the diverse retrieved relational *key-value* pairs of context input  $x$  are represented by  $\hat{M}_s^r = \{m_1, m_2, \dots, m_j\}$ ,  $j$  is the size of  $\hat{M}_s^r$ ,  $m_j$  is the combined *key-value*, such as in Figure 1,  $m_3 = C_1^{s1} \oplus t_{s1} \oplus C_2^{s1} \oplus t_{s1}$ .

### 5.3 Tailored chunk Scorer

There exists a different degree of correlation between input context  $x$  and  $m_j$  in  $\hat{M}_s^r$ . The aim of Tailored chunk Scorer is to learn the weight value between input context  $x$  and each  $m_j$ . The context input  $x$  and its relevant memory  $m_j$  are individually initialized by their embedding which is calculated by using the MedLLaMA 13b. Specifically, The context input  $x$  and each of its relevant memory  $m_j$  are encoded into the context input embedding  $\mathbf{E}(x)$  and memory embedding  $\mathbf{E}(m_j)$ . Let  $X^0 = (x_1, \dots, x_k)$  denote the input embedding matrix of Tailored chunk Scorer, where  $x_1$  refers to  $\mathbf{E}(x)$ ,  $x_{2:k}$  refers to memory embedding  $\mathbf{E}(m_{1:j})$ . An  $L$ -layer Transformer takes  $X^0$  as input and produces the semantic latent representation  $X^l = (h_1^l, \dots, h_k^l)$  of context input and memory at the  $l$ -th layer. Specifically, the Transformer employs a multi-head self-attention to project the output of the  $(l-1)$ -th layer to a query  $Q_l$  and a set of key-value  $(K_l - V_l)$  pairs,

$$Q_l = X^{l-1}W_l^q, K_l = X^{l-1}W_l^k, V_l = X^{l-1}W_l^v$$

where  $(W_l^q, W_l^k, W_l^v) \in \mathbb{R}^{d_{model} \times d_k}$  are learnable weight matrices. The output of a weight matrix for the input context  $x$  and its relevant memory  $m_j$  is calculated as:

$$\tilde{W}_x = softmax \left( \frac{Q_l K_l^T}{\sqrt{d_k}} \right)$$

$\tilde{W}_x \in \mathbb{R}^{k \times k}$  can be interpreted as a weight matrix that captures the correlations among the context input and its relevant memory. The memory retrieval likelihood  $P_T(m|x)$  is calculated by calculating the highest weight value between memory  $m_j$  and input context  $x$ .

### 5.4 Training the Tailored chunk Scorer

We use the LM as a scoring function to help train the Tailored chunk Scorer and measure how much each memory could improve the LM perplexity, in the training process, the memory that makes the LM's output as close as possible to the ground truth is considered to be providing the memory that the LM needs. Specifically, we first compute  $P_{LM}(y|m_j, x)$ , the LM probability of the ground truth output  $y$  given the input context  $x$  and a memory  $m_j$ . The higher the probability, the better the memory  $m_j$  is at improving the LM's perplexity. So we compute the LM likelihood of each memory  $m$  as follows:

$$P_{LM}(m|x, y) = \max(P_{LM}(y|m_1, x), \dots, P_{LM}(y|m_j, x))$$

The Tailored chunk Scorer is trained by minimizing the loss function between the memory retrieval likelihood and LM likelihood:

$$L = \frac{1}{|D|} \sum_{x \in D} |P_T(m|x) - P_{LM}(m|x, y)|$$

where  $D$  is a set of input contexts.

### 5.5 Training the Information Extractor

First, we construct the two different instruction-based datasets for triple extraction and relation extraction. Specifically, as the example of triple extraction, the dataset contains four components: 1) Instruction ( $I$ ), which guides LM to generate the triples for each sentence, the instruction is manually defined. 2) Example, we employ the trained Tailored chunk Scorer to assign weights to each memory in  $\hat{M}_s^r$  for each input context  $x$ , the memory  $\bar{m}$  with the highest weight score is regarded as the example. 3) Input context ( $x$ ). 4) gold triples  $y$ . LM will be trained by,

$$Q(y|I \oplus \bar{m} \oplus x)$$

In the generation progress, instruction  $I$ , example  $m$ , and input context  $x$  are fed into the LM to generate the triple  $y$  of the  $x$ . It has the same progress when training the model of the relation extraction task.

## 6 Experiments

We conducted performance evaluations of our model in two distinct information extraction tasks,

triple extraction, and relation extraction. In all settings, PETAILOR significantly outperforms the strong baselines on the benchmark dataset GM-CIHT, showing the effectiveness and generality of our approach.

## 6.1 Evaluation Metrics

Same as (Tang et al., 2022; Zeng et al., 2019), triple is regarded as correct when its relation, the head entity and the tail entity are all correct. For example, in the sentence: *Infusion of prostacyclin (PGI2) reportedly attenuates renal ischemic injury in the dog and the rat.*, triple  $\langle \text{Infusion}, \text{treats}, \text{rat} \rangle$  is regarded as correct while  $\langle \text{injury}, \text{treats}, \text{rat} \rangle$  is not. we report the standard Micro Precision, Recall, and F1-score on the test set. In the task of relation extraction, we have the same evaluation metric as the task of triple extraction.

## 6.2 Triple Extraction

### 6.2.1 Datasets

The constructed biomedical triple extraction GM-CIHT is used as the benchmark dataset in our work.

### 6.2.2 Baselines

We selected six models as the baselines for our triple extraction task, which are the state-of-the-art models for the eight commonly used triple extraction datasets, as detailed in Table 3. They are UniRel (Tang et al., 2022), OneRel (Shang et al., 2022), UIE (base) (Lu et al., 2022), UIE (large) (Lu et al., 2022), E2H (base) (Gao et al., 2023a), E2H (large) (Gao et al., 2023a). These models are categorized as small language models (SLM) in our work.

We also compare the performance of PETAILOR with several strong baselines based on the state-of-the-art pre-trained large language models (LLM), including **1) GPT-3.5/4**: GPT-3.5-turbo (P1), GPT-3.5-turbo (P2), GPT-4 (P1), GPT-4 (P2). In these models, we first design prompts to instruct the GPT models to predict the triples for each input sentence, we also provide the relation definition for each relation in the instruction. The difference between prompt (P1) and prompt (P2) lies in the output format. For detailed information, please refer to Appendix C. **2) RT-n**: We also use RT (Li and Zhang, 2023), a retrieval and chain of thoughts-based method to extract the relation. Since this model focuses on relation extraction, we present the performance results for head and tail entities based on GPT-4 (P1), which is the top-performing

model across GPT-3.5/4 (P1/P2). **3) LLAMA-based model**: Llama 13B (Touvron et al., 2023), PMC-LLaMA 7B (Wu et al., 2023), MedLLaMA 13B (Wu et al., 2023). When designing the instruction in these models, the examples are randomly selected. **4) KNN-LLM-n**: We have also incorporated a KNN-retrieval-based method as the baseline, built upon the MedLLaMA 13B, which is the top-performing model across **3) LLAMA-based model**,  $n$  represents the top- $n$  sentences obtained through KNN. We introduce **5) KNN-LLM-RT**, where examples are constructed using a combination of the methods employed in RT and KNN. We also design a variant of PETAILOR where we remove the tailored chunk scorer and instead employ cosine similarity to select the most relevant chunks from constructed relational key-value memory for the input sentence (denoted as PETAILOR w/o TCS).

### 6.2.3 Results and Discussion

Table 4 presents the experiment results of various approaches based on the metrics, including Precision, Recall, and F1 on the task of triple extraction. We also provide individual results for the head entity, tail entity, and relation, based on the output of different triple extraction models. We have the following observations: (1) our PETAILOR significantly outperforms all the strong baselines and its variant across all evaluation metrics; (2) The non-finetuned LLMs are incapable of extracting the precise triple from biomedical sentences, as exemplified by GPT-4 (P1); (3) Using SLM models in the biomedical domain reveals their reduced effectiveness, even though they have demonstrated strong performance in the general domain. This finding also supports the conclusion presented in the Introduction section

(4) By comparing the performance of KNN-LLM- $n$ , we discovered that the value of  $n$  in top- $n$  and the performance of LLM are not directly proportional. (5) Our findings indicate that the KNN-based language model is highly effective in the biomedical triple extraction task, significantly enhancing the overall performance of triple extraction. (6) by adding the tailored chunk scorer, our PETAILOR approach significantly improves over the baseline PETAILOR w/o TCS, especially on F1 value of all evaluation metrics, demonstrating the effectiveness of the learned chunk scorer in adapting the LM to select more useful chunks.

Dataset	Domain	Metric	Comparable SOTA	Method	Micro F1	GM-CIHT
NYT	News	Triple Micro F1	UniRel (Tang et al., 2022)	Table-Filling	93.7	18.60
NYT*	News	Triple Micro F1	QIDN (Tan et al., 2022)	Query-Based	92.9	–
WebNLG	Encyclopedia	Triple Micro F1	UniRel (Tang et al., 2022)	Table-Filling	94.7	18.60
WebNLG*	Encyclopedia	Triple Micro F1	OneRel (Shang et al., 2022)	Table-Filling	94.3	44.52
14-res	Reviews	Triple Micro F1	LasUIE/E2H/UIE/USM (Lu et al., 2022; Lou et al., 2023)	Generation-Based	75.2/75.92/74.52/77.26	–
14-lap	Reviews	Triple Micro F1	E2H/UIE/USM (Lu et al., 2022; Lou et al., 2023)	Generation-Based	65.98/63.88/65.51	29.00
15-res	Reviews	Triple Micro F1	E2H/UIE/USM (Lu et al., 2022; Lou et al., 2023)	Generation-Based	68.80/67.15/69.86	–
16-res	Reviews	Triple Micro F1	E2H/UIE/USM (Lu et al., 2022; Lou et al., 2023)	Generation-Based	75.46/75.07/78.25	–

Table 3: The performance of the state-of-the-art model on eight commonly used public benchmark datasets for triple extraction. We reproduce these models by using their provided source code. QIDN and USM does not provide the source code, so we ignore them as our baselines.

	Approach	Triple			Head Entity			Tail Entity			Relation		
		Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
SLM	UniRel (Tang et al., 2022)	36.36	12.50	18.60	65.18	24.83	35.96	56.14	21.12	30.69	86.60	34.15	48.98
	OneRel (Shang et al., 2022)	47.97	52.01	44.52	65.26	71.54	60.00	68.16	75.46	62.15	74.70	82.94	67.96
	UIE (base) (Lu et al., 2022)	34.81	30.59	32.56	76.04	66.81	71.13	79.26	69.63	74.13	42.96	37.74	40.18
	UIE (large) (Lu et al., 2022)	33.99	29.93	31.83	77.83	68.55	72.90	79.56	70.07	74.51	43.10	37.96	40.37
	E2H (base) (Gao et al., 2023a)	31.29	26.68	28.80	72.52	61.82	66.74	80.15	68.32	73.77	40.96	34.92	37.70
	E2H (large) (Gao et al., 2023a)	31.47	26.89	29.00	74.87	63.99	69.00	78.17	66.81	72.04	41.62	35.57	38.36
LLM	GPT-3.5-turbo (P1)	7.27	4.08	5.23	38.13	21.08	27.15	25.77	14.41	18.48	35.08	18.71	24.40
	GPT-3.5-turbo (P2)	6.46	6.02	6.23	36.04	32.47	34.16	25.58	23.66	24.58	30.51	27.10	28.70
	GPT-4 (P1)	9.51	9.25	9.38	36.94	35.91	36.42	26.33	25.59	25.95	41.37	40.22	40.79
	GPT-4 (P2)	9.48	9.46	9.47	35.56	35.48	35.52	25.00	24.95	24.97	42.45	42.37	42.41
	RT-1 (Li and Zhang, 2023)	9.73	9.46	9.60	36.94	35.91	36.42	26.33	25.59	25.95	47.31	47.31	47.31
	RT-5 (Li and Zhang, 2023)	9.51	9.25	9.38	36.94	35.91	36.42	26.33	25.59	25.95	52.04	52.04	52.04
	RT-15 (Li and Zhang, 2023)	9.07	8.81	8.94	36.94	35.91	36.42	26.33	25.59	25.95	48.17	48.17	48.17
	RT-20 (Li and Zhang, 2023)	9.29	9.03	9.16	36.94	35.91	36.42	26.33	25.59	25.95	50.75	50.75	50.75
	PMC-LLaMA 7B (Wu et al., 2023)	57.14	25.81	35.56	89.52	40.43	55.70	68.09	30.75	42.37	82.38	37.20	51.26
	Llama 13B (Touvron et al., 2023)	11.70	9.46	10.46	57.71	46.66	51.61	18.88	15.26	16.88	71.27	57.63	63.73
	MedLLaMA 13B (Wu et al., 2023)	58.70	36.98	45.38	90.44	56.99	69.92	69.28	43.65	53.56	83.27	52.47	64.37
	KNN-LLM-1	70.96	70.96	70.96	88.38	88.38	88.38	87.09	87.09	87.09	78.92	78.92	78.92
	KNN-LLM-2	69.04	62.37	65.54	89.52	80.86	84.97	80.24	72.47	76.16	82.85	74.83	78.64
	KNN-LLM-3	69.54	69.24	69.39	90.06	89.67	89.87	81.85	81.50	81.68	82.51	82.15	82.32
	KNN-LLM-4	48.81	48.81	48.81	83.44	83.44	83.44	62.36	62.36	62.36	72.90	72.90	72.90
	KNN-LLM-RT	62.02	59.35	60.66	82.47	78.92	80.65	77.75	74.41	76.04	79.77	76.34	78.02
	PETAILOR (Our Approach)	<b>76.88</b>	<b>76.56</b>	<b>76.72</b>	<b>92.22</b>	<b>91.83</b>	<b>92.03</b>	<b>90.06</b>	<b>89.68</b>	<b>89.87</b>	83.37	<b>83.01</b>	<b>83.19</b>
	PETAILOR w/o TCS	75.21	75.05	75.13	89.67	76.55	82.59	84.88	72.47	78.19	<b>83.87</b>	71.61	77.26

Table 4: Results of various approaches for triple extraction on GM-CIHT. In RT- $n$ ,  $n$  refers to the quantity of retrieved relevant sentences. **PETAILOR** improve **ONEREL** and **KNN-LLM** by **32.20%** and **5.76%** respectively.

### 6.3 Relation Extraction

#### 6.3.1 Dataset

The biomedical triple extraction GM-CIHT serves as the source data for the relation extraction task. In this task, we utilize the input context, head entity, and tail entity as model inputs, with the model’s output indicating the relationship between the input head entity and tail entity.

#### 6.3.2 Baselines

We compare the performance of PETAILOR with several strong baselines based on the state-of-the-art pre-trained large language models (LLM), including 1) **GPT-4**: In GPT-4, we also design prompts to instruct the GPT models to predict the relations for each input sentence. For detailed information, please refer to Appendix C. 2) **RT-n**: we use the same relation method RT as we used in

triple extraction. 3) **LLAMA-based model**: Llama 13B (Touvron et al., 2023), PMC-LLaMA 7B (Wu et al., 2023), MedLLaMA 13B (Wu et al., 2023). When designing the instruction in these models, the examples are randomly selected. 4) **KNN-LLM-n**: We have also incorporated a KNN-retrieval-based method as the baseline, built upon the MedLLaMA 13B. We also design one variant of PETAILOR where we remove the tailored chunk scorer and instead employ cosine similarity to select the most relevant chunks from constructed relational key-value memory for the input sentence (denoted as PETAILOR w/o TCS).

#### 6.3.3 Results and Discussion

Table 5 presents the experiment results of various approaches on the relation extraction task. We have the following observations: (1) our PETAI-



Approach	Relation		
	Precision	Recall	F1
GPT-4	48.17	48.17	48.17
PMC-LLaMA 7B (Wu et al., 2023)	84.51	84.51	84.51
Llama 13B (Touvron et al., 2023)	86.63	86.45	86.54
MedLLaMA 13B (Wu et al., 2023)	86.24	86.24	86.24
RT-1 (Li and Zhang, 2023)	44.86	46.02	45.44
RT-5 (Li and Zhang, 2023)	44.91	46.45	45.67
RT-10 (Li and Zhang, 2023)	42.80	43.44	43.12
RT-20 (Li and Zhang, 2023)	44.94	45.81	45.37
KNN-LLM-1	85.16	85.16	85.16
KNN-LLM-2	86.02	86.02	86.02
KNN-LLM-3	84.73	82.37	83.53
KNN-LLM-4	84.51	84.51	84.51
PETAILOR	<b>86.66</b>	<b>86.66</b>	<b>86.66</b>
PETAILOR w/o TCS	85.59	85.59	85.59

Table 5: Results of various approaches for relation extraction on GM-CIHT. **PETAILOR** improve **KNN-LLM** by **1.5%**.

LOR significantly outperforms all the strong baselines and its variant across all evaluation metrics; (2) Although MedLLaMA 13B is trained on a biomedical dataset, its performance is inferior to that of LLaMA 13B. This underscores the notion that models trained on medical data are not inherently more effective than models trained without medical data in certain tasks, such as relation extraction. (3) Despite we built the PETAILOR by the MedLLaMA 13B, it still can get a better performance, it shows the effectiveness of our model. (4) When assessing the performance of KNN-LLM- $n$  and considering the results from triple extraction, we have a different discovery: when the value of  $n$  is between 1 and 2, the value of  $n$  in top- $n$  and the LLM’s performance are directly proportional. However, when  $n$  exceeds 2, the results are opposite. (5) by adding the tailored chunk scorer, our PETAILOR approach also significantly improves over the baseline PETAILOR w/o TCS in the relation extraction task, demonstrating the effectiveness of the learned chunk scorer could adapt the LLM to select more useful chunks in the task of relation extraction.

## 7 Analysis

### 7.1 The Impact of chunk Numbers

In our proposed model, there is a parameter that controls the number of chunks:  $m \in \{3, 4, 5\}$  when constructing the relational key-value memory, which represents the length of each chunk. This parameter aids in retrieving relational information, considering that the input sentence often contains noise that can impact relation identification. We

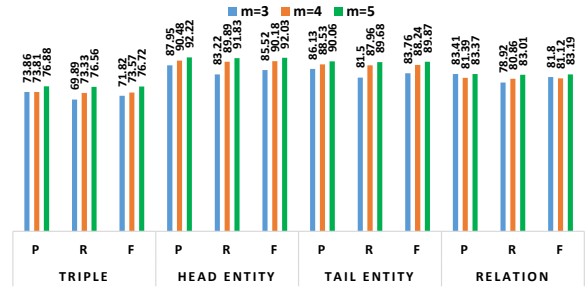


Figure 2: Precision (P), Recall (R), F1 results with different chunk length  $m$  settings in the task of triple extraction

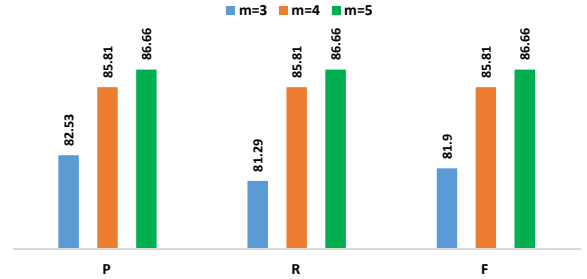


Figure 3: Precision (P), Recall (R), F1 results with different chunk length  $m$  settings in the task of relation extraction

analyze the impact of the different  $m$  in the task of triple extraction and relation extraction, as shown in Figure 2 and Figure 3. As we can see, in these two tasks, when  $m = 5$ , PETAILOR can construct the more rich relation-based information, and retrieve the more relevant relation chunk to the input sentence. In addition, the performance of  $m = 5$  is better than  $m = 3$  or  $m = 4$ , also indicating that if the granularity of relation chunk is too coarse, it will impact relation recognition and consequently affect model performance.

### 7.2 The Impact of chunk Choose Method

In our work, we use the trained chunk scorer to get the most relevant chunk of the input sentence from selected candidate relation chunks, there are two strategies to construct the candidate relation chunks, 1) by calculating the similarity between the sentence chunk and the relation chunk in relation chunk key-value memory, the top-1 relation chunk is chosen as the candidate relation chunk. We denote this method as cosine selection (CS). 2) Randomly select several relation chunks from the relational key-value memory. We denote this method as random selection (RS). As shown in Figure 4 and Figure 5. One interesting observation

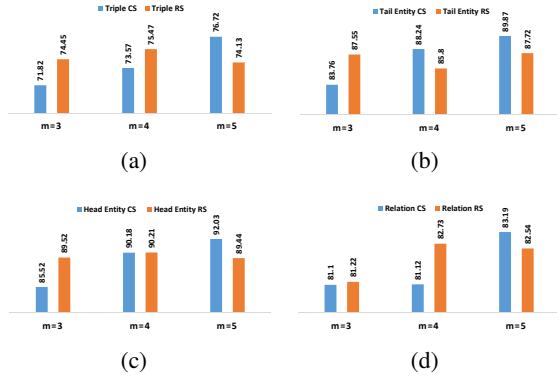


Figure 4: F1 results with different retrieved chunk method in the task of triple extraction

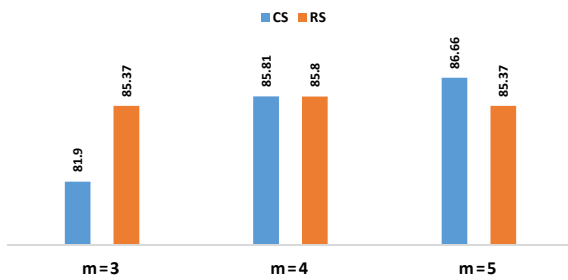


Figure 5: F1 results with different retrieved chunk methods in the task of relation extraction.

is that when chunk size  $m$  is less than 5, the CS strategy is better than the RS strategy in the task of triple extraction and relation extraction. This suggests that CS strategy may improve a language model’s generation ability when setting a larger chunk size.

## 8 Conclusion

In this paper, we introduce a novel biomedical triple extraction framework called PETAILOR. Unlike the traditional retrieval-argument language model, our framework retrieval the knowledge from the pre-computed relational Key-Value Memory and adapts the Tailored Chunk Scorer to the language model (LM). Additionally, we create a biomedical triple extraction dataset with extensive relation type coverage and expert annotations. Experimental results show that our framework achieves consistent improvements over various baselines. Our method is also flexible, we intend to explore its effectiveness on other NLP tasks including text classification.

## 9 Authorship contribution statement

**First author:** Conducting related research, identifying current work challenges, implementing

baseline models, assisting in dataset construction, proposing ideas, developing frameworks, coding, writing papers, and revising manuscripts. **Third author:** Assisting in dataset construction, providing medical expertise, writing papers, and revising manuscripts. **Corresponding author:** Supervision, Writing – review & editing.

## Acknowledgements

This work was supported by the National Institutes of Health’s National Center for Complementary and Integrative Health grant number R01AT009457 and National Institute on Aging grant number R01AG078154. The content is solely the responsibility of the authors and does not represent the official views of the National Institutes of Health. We would also like to express our gratitude to all the medical annotators who contributed to the creation of the dataset.

## A Data statistics of each relation type

In Table 6, we present the sentence statistics for our 22 defined relation types within GM-CIHT.

relation type	Sentences	relation type	Sentences
INTERACTS WITH	1019	PREVENTS	107
TREATS	726	PRECEDES	103
PROCESS OF	686	COMPLICATES	101
INHIBITS	345	ASSOCIATED WITH	89
STIMULATES	298	CAUSES	76
USES	293	PREDISPOSES	61
COEXISTS WITH	256	MANIFESTATION OF	54
ADMINISTERED TO	175	AUGMENTS	53
DIAGNOSES	152	DISRUPTS	51
AFFECTS	117	DOES NOT TREAT	24
PRODUCES	116	SYMPTOM OF	10

Table 6: Statistics of GM-CIHT.

## B Relation Type Definition

We list the definitions of 22 relation types by referring (Kilicoglu et al., 2011).

- 1. CAUSES: Brings about a condition or an effect. Implied here is that an agent, such as for example, a pharmacologic substance or an organism, has brought about the effect. This includes induces, effects, evokes, and etiology. Neurocysticercosis (NCC) is one of the major causes of neurological disease
- 2.COMPLICATES: Causes to become more severe or complex, or results in adverse effects.

- 3. USES: Employs in the carrying out of some activity. This includes applies, utilizes, employs, and avails.
- 4. STIMULATES: Increases or facilitates the action or function of (substance interaction).
- 5. DISRUPTS: Alters or influences an already existing condition, state, or situation. Produces a negative effect on.
- 6. TREATS: Applies a remedy with the object of effecting a cure or managing a condition.
- 7. COEXISTS\_WITH: Occurs together with, or jointly. Food intolerance-related constipation is characterized by proctitis.
- 8. MANIFESTATION\_OF: That part of a phenomenon which is directly observable or concretely or visibly expressed, or which gives evidence to the underlying process. This includes expression of, display of, and exhibition of.
- 9. INTERACTS\_WITH: Substance interaction.
- 10. ADMINISTERED\_TO: Given to an entity, when no assertion is made that the substance or procedure is being given as treatment.
- 11. PREVENTS: Stops, hinders or eliminates an action or condition.
- 12. PREDISPOSES: To be a risk to a disorder, pathology, or condition.
- 13. INHIBITS: Decreases, limits, or blocks the action or function of (substance interaction).
- 14. AUGMENTS: Expands or stimulates a process.
- 15. PRODUCES: Brings forth, generates or creates. This includes yields, secretes, emits, biosynthesizes, generates, releases, discharges, and creates.
- 16. PROCESS\_OF: Disorder occurs in (higher) organism.
- 17. PRECEDES: Occurs earlier in time. This includes antedates, comes before, is in advance of, predates, and is prior to.

**System Prompt:**  
please extract the triple from a sentence, the triplet is [head entity, relation, tail entity], the element relation denotes the relationship between head entity and tail entity. I will provide you the definition of the triple you need to extract, the sentence from where you extract the triples (head entity, relation, tail entity) and the output format with examples. The extracted relation must align with my established relation set.

**User Prompt:** Are you clear about your role?

**Assistant Prompt:** Sure, I'm ready to help you with your triple extraction task. Please provide me with the necessary information to get started.

**Guidelines Prompt:**  
Relation Definition (please check the Appendix.B)

**P1:**  
**Output Format:**  
[head entity, relation, tail entity]  
the relation must be the relation in our relation definition  
If no triple are presented in any categories keep it None  
**Examples:**  
**Input:** The effect of bicarbonate on liver alcohol dehydrogenase.  
**Output:** [bicarbonate, INTERACTS\_WITH, Alcohol]

**P2:**  
**Output Format:**  
head entity(relation)tail entity  
the relation must be the relation in our relation definition  
If no triple are presented in any categories keep it None  
**Examples:**  
**Input:** The effect of bicarbonate on liver alcohol dehydrogenase.  
**Output:** bicarbonate(INTERACTS\_WITH)alcohol

Figure 6: Example of Prompt 1 and Prompt 2 defined for GPT-based models on the task of triple extraction

**System Prompt:**  
You are an excellent linguist. The task is to predict relationship between the given head entity and tail entity within a given sentence; this relation which must be in ('PRECEDES', 'PREDISPOSES', 'TREATS', 'AFFECTS', 'INTERACTS\_WITH', 'INHIBITS', 'PRODUCES', 'ADMINISTERED\_TO', 'PROCESS\_OF', 'AUGMENTS', 'PREVENTS', 'DIAGNOSES', 'COEXISTS\_WITH', 'ASSOCIATED\_WITH', 'DISRUPTS', 'CAUSES', 'COMPLICATES', 'SYMPTOM\_OF', 'DOES\_NOT\_TREAT', 'STIMULATES', 'MANIFESTATION\_OF', 'USES')

**User Prompt:** Are you clear about your role?

**Assistant Prompt:** Sure, I'm ready to help you with your relation extraction task. Please provide me with the necessary information to get started.

**Guidelines Prompt:**  
Relation Definition (please check the Appendix.B)

**Output Format:**  
relation  
the relation must be the relation in our relation definition  
If no relation are presented in any categories keep it None  
**Examples:**  
**Input:** In the sentence The effect of bicarbonate on liver alcohol dehydrogenase, the relationship between bicarbonate and alcohol dehydrogenase is?  
**Output:** INTERACTS\_WITH

Figure 7: Instruction defined for GPT-based models on the task of relation extraction

- 18. AFFECTS: Produces a direct effect on. Implied here is the altering or influencing of an existing condition, state, situation, or entity. This includes has a role in, alters, influences, predisposes, catalyzes, stimulates, regulates, depresses, impedes, enhances, contributes to, leads to, and modifies.
- 19. DIAGNOSES: Distinguishes or identifies the nature or characteristics of.
- 20. ASSOCIATED\_WITH: Has a relationship to (genedisease relation).
- 21. DOES\_NOT\_TREAT: antonyms of TREATS
- 22. SYMPTOM\_OF: departure from normal function or feeling which is noticed by a patient, reflecting the presence of an unusual state, or of a disease; subjective, observed by the patient, cannot be measured directly

## C Instruction of triple extraction and relation extraction by GPT API

Instructions for triple extraction and relation extraction using the GPT API can be found in Figure 6 and Figure 7.

## References

- Hui Chen, Pengfei Hong, Wei Han, Navonil Majumder, and Soujanya Poria. 2023. Dialogue relation extraction with document-level heterogeneous graph attention networks. *Cognitive Computation*, pages 1–10.
- Dean Cheng, Craig Knox, Nelson Young, Paul Stothard, Sambasivarao Damaraju, and David S Wishart. 2008. Polysearch: a web-based text mining system for extracting relationships between human diseases, genes, mutations, drugs and metabolites. *Nucleic acids research*, 36(suppl\_2):W399–W405.
- Hao Fei, Shengqiong Wu, Jingye Li, Bobo Li, Fei Li, Libo Qin, Meishan Zhang, Min Zhang, and Tat-Seng Chua. 2022. Lasuie: Unifying information extraction with latent adaptive structure-aware generative language model. *Advances in Neural Information Processing Systems*, 35:15460–15475.
- Christiane Fellbaum. 2010. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer.
- Chang Gao, Wenxuan Zhang, Wai Lam, and Lidong Bing. 2023a. Easy-to-hard learning for information extraction. *arXiv preprint arXiv:2305.09193*.
- Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, et al. 2023b. Rarr: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508.
- Tianyu Gao, Xu Han, Hao Zhu, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2019. Fewrel 2.0: Towards more challenging few-shot relation classification. *arXiv preprint arXiv:1910.07124*.
- Harsha Gurulingappa, Abdul Mateen Rajput, Angus Roberts, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. 2012. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *Journal of biomedical informatics*, 45(5):885–892.
- Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.
- Hrant Khachatrian, Lilit Nersisyan, Karen Hambardzumyan, Tigran Galstyan, Anna Hakobyan, Arsen Arakelyan, Andrey Rzhetsky, and Aram Galstyan. 2019. Biorelex 1.0: Biological relation extraction benchmark. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 176–190.
- Halil Kilicoglu, Graciela Rosemblat, Marcelo Fiszman, and Thomas C Rindflesch. 2011. Constructing a semantic predication gold standard from the biomedical literature. *BMC bioinformatics*, 12(1):1–17.
- Halil Kilicoglu, Dongwook Shin, Marcelo Fiszman, Graciela Rosemblat, and Thomas C Rindflesch. 2012. Semmeddb: a pubmed-scale repository of biomedical semantic predications. *Bioinformatics*, 28(23):3158–3160.
- Hee-Jin Lee, Sang-Hyung Shim, Mi-Ryoung Song, Hyunju Lee, and Jong C Park. 2013. Comagc: a corpus with multi-faceted annotations of gene-cancer relations. *BMC bioinformatics*, 14:1–17.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.
- Mingchen Li, Junfan Chen, Samuel Mensah, Nikolaos Aletras, Xiulong Yang, and Yang Ye. 2022. A hierarchical n-gram framework for zero-shot link prediction. *arXiv preprint arXiv:2204.10293*.
- Mingchen Li and Lifu Huang. 2023. Understand the dynamic world: An end-to-end knowledge informed framework for open domain entity state tracking. *arXiv preprint arXiv:2304.13854*.
- Mingchen Li and Shihao Ji. 2022. Semantic structure based query graph prediction for question answering over knowledge graph. *arXiv preprint arXiv:2204.10194*.
- Mingchen Li, Yang Ye, Jeremy Yeung, Huixue Zhou, Huaiyuan Chu, and Rui Zhang. 2023a. W-procer: Weighted prototypical contrastive learning for medical few-shot named entity recognition. *arXiv preprint arXiv:2305.18624*.
- Mingchen Li and Rui Zhang. 2023. How far is language model from 100% few-shot named entity recognition in medical domain. *arXiv preprint arXiv:2307.00186*.
- Mingchen Li, Zili Zhou, and Yanna Wang. 2020. Multi-fusion chinese wordnet (mcw): Compound of machine learning and manual correction. *arXiv preprint arXiv:2002.01761*.
- Xingxuan Li, Ruochen Zhao, Yew Ken Chia, Bosheng Ding, Lidong Bing, Shafiq Joty, and Soujanya Poria. 2023b. Chain of knowledge: A framework for grounding large language models with structured knowledge bases. *arXiv preprint arXiv:2305.13269*.
- Chengyuan Liu, Fubang Zhao, Yangyang Kang, Jingyuan Zhang, Xiang Zhou, Changlong Sun, Fei Wu, and Kun Kuang. 2023. Rexuie: A recursive method with explicit schema instructor for universal information extraction. *arXiv preprint arXiv:2304.14770*.
- Jie Lou, Yaojie Lu, Dai Dai, Wei Jia, Hongyu Lin, Xi-anpei Han, Le Sun, and Hua Wu. 2023. Universal information extraction as unified semantic matching. *arXiv preprint arXiv:2301.03282*.



- Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. *arXiv preprint arXiv:2203.12277*.
- Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *arXiv preprint arXiv:2302.00083*.
- Isabel Segura-Bedmar, Paloma Martínez Fernández, and María Herrero Zazo. 2013. Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013). Association for Computational Linguistics.
- Yu-Ming Shang, Heyan Huang, and Xianling Mao. 2022. Onerel: Joint entity and relation extraction with one module in one step. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11285–11293.
- Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2023. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*.
- 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*.
- Olivier Taboureau, Sonny Kim Nielsen, Karine Audouze, Nils Weinhold, Daniel Edsgård, Francisco S Roque, Irene Kouskoumvekaki, Alina Bora, Ramona Curpan, Thomas Skøt Jensen, et al. 2010. Chemprot: a disease chemical biology database. *Nucleic acids research*, 39(suppl\_1):D367–D372.
- Zeqi Tan, Yongliang Shen, Xuming Hu, Wenqi Zhang, Xiaoxia Cheng, Weiming Lu, and Yueting Zhuang. 2022. Query-based instance discrimination network for relational triple extraction. *arXiv preprint arXiv:2211.01797*.
- Wei Tang, Benfeng Xu, Yuyue Zhao, Zhendong Mao, Yifeng Liu, Yong Liao, and Haiyong Xie. 2022. Unirel: Unified representation and interaction for joint relational triple extraction. *arXiv preprint arXiv:2211.09039*.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
- Erik M Van Mulligen, Annie Fourier-Reglat, David Gurwitz, Mariam Molokhia, Ainhua Nieto, Gianluca Trifiro, Jan A Kors, and Laura I Furlong. 2012. The eu-adr corpus: annotated drugs, diseases, targets, and their relationships. *Journal of biomedical informatics*, 45(5):879–884.
- Chih-Hsuan Wei, Hung-Yu Kao, and Zhiyong Lu. 2013. Pubtator: a web-based text mining tool for assisting biocuration. *Nucleic acids research*, 41(W1):W518–W522.
- Chaoyi Wu, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2023. Pmc-llama: Further fine-tuning llama on medical papers. *arXiv preprint arXiv:2304.14454*.
- Xiangrong Zeng, Shizhu He, Daojian Zeng, Kang Liu, Shengping Liu, and Jun Zhao. 2019. Learning the extraction order of multiple relational facts in a sentence with reinforcement learning. 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 367–377.
- Rui Zhang, Dimitar Hristovski, Dalton Schutte, Andrej Kastrin, Marcelo Fiszman, and Halil Kilicoglu. 2021. Drug repurposing for covid-19 via knowledge graph completion. *Journal of biomedical informatics*, 115:103696.
- Huixue Zhou, Robin Austin, Sheng-Chieh Lu, Greg Silverman, Yuqi Zhou, Halil Kilicoglu, Hua Xu, and Rui Zhang. 2023. Complementary and integrative health lexicon (cihlex) and entity recognition in the literature. *arXiv preprint arXiv:2305.17353*.