Open Relation Extraction with Non-Existent and Multi-Span Relationships

Huifan Yang, Da-Wei Li, Zekun Li, Donglin Yang and Bin Wu

April 24, 2022

Open Relation Extraction With Non-existent and Multi-span Relationships

Huifan Yang[1], **Da-Wei Li**[2], **Zekun Li**[1], **Donglin Yang**[1], **Bin Wu**[1]

[1]Beijing Key Laboratory of Intelligence Telecommunication Software and Multimedia,
Beijing University of Posts and Telecommunications

[2]Microsoft Software Technology Center Asia

{huifunny, lizekun, iceberg, wubin}@bupt.edu.cn, daweilee@microsoft.com

Abstract

Open relation extraction (ORE) aims to assign semantic relationships among arguments, essential to the automatic construction of knowledge graphs (KG). The previous ORE methods and some benchmark datasets consider a relation between two arguments as definitely existing and in a simple single-span form, neglecting possible non-existent relationships and flexible, expressive multi-span relations. However, detecting non-existent relations is necessary for a pipelined information extraction system (first performing named entity recognition then relation extraction), and multi-span relationships contribute to the diversity of connections in KGs. To fulfill the practical demands of ORE, we design a novel **Q**uery-based M**u**lti-head **O**pen **R**elation **E**xtractor (**QuORE**) to extract single/multi-span relations and detect non-existent relationships effectively. Moreover, we re-construct some public datasets covering English and Chinese to derive augmented and multi-span relation tuples. Extensive experiment results show that our method outperforms the state-of-the-art ORE model LOREM in the extraction of existing single/multi-span relations and the overall performances on four datasets with non-existent relationships. Our code and data are publicly available[1].

1 Introduction

Relation extraction (RE) from unstructured text is fundamental to a variety of downstream tasks, such as constructing knowledge graphs and computing sentence similarity. Conventional closed relation extraction considers only a predefined set of relation types on small and homogeneous corpora, which is far less effective when shifting to general-domain text mining that has no limits in relation types or languages. To alleviate the constraints of closed RE, Banko et al. introduce a new paradigm: open relation extraction (ORE), predicting a text span as the semantic connection between arguments from within a context, where a span is a contiguous subsequence. However, the previous works and some widely-used datasets regard an arbitrary argument pair as having a definitely existing relation by default and constrain their task to extract a simple single-span relation between two arguments, ignoring possible non-existent relationships and flexible multi-span relations. To extend ORE for practical demands of general-domain text mining, this

[1]https://github.com/farahhuifanyang/QuORE

Context (*English*): *Donna Karan* was born in *Long Island*, *New York*. *She has a special comprehension of New York, known by the world as a cosmopolitan city.*

Given Argument Tuple	Open Relation
(*Donna Karan, Long Island*)	(*born in*) Single-span
(*a special comprehension, the world*)	Non-existent
(*Donna Karan, a special comprehension, New York*)	(*has, of*) Multi-span
(*New York, the world, a cosmopolitan city*)	(*known by, as*) Multi-span

Context (*Chinese*): *唐娜·凯伦出生于纽约长岛，对纽约这个闻名世界的大都会有着一份特殊的感悟* 。

Given Argument Tuple	Open Relation
(*唐娜·凯伦, 长岛*)	（*出生于*） Single-span
(*世界, 一份特殊的感悟*)	Non-existent
(*唐娜·凯伦, 纽约, 一份特殊的感悟*)	（*对, 有着*） Multi-span

Figure 1: An illustration of our task: open relation extraction with single-span, multi-span, and non-existent relationships. (We present cases in English and Chinese due to the datasets of the two languages used in this paper.)

paper describes the further tasks of ORE with non-existent and multi-span relationships, and proposes a novel query-based multi-head open relation extractor (QuORE) developed from our re-constructed datasets.

We illustrate the ORE task of this paper in Figure 1. Provided a context and an argument tuple, open relation extraction identifies a single span or multiple spans to specify a semantic connection among the arguments, or detects a non-existent relationship if the arguments do not entail an open relation. (An argument is a text span representing an adverbial, adjectival, nominal phrase, and so on, which is not limited to an entity.)

Conventional ORE systems are largely based on syntactic patterns and heuristic rules that depend on external tools of natural language processing (e.g., PoS-taggers) and language-specific relation formations. For example, ReVerb (Fader, Soderland, and Etzioni 2011), ClausIE (Corro and Gemulla 2013), OpenIE4 (Mausam 2016) for English and

CORE (Tseng et al. 2014), ZORE (Qiu and Zhang 2014) for Chinese, leverage external tools to obtain part-of-speech tags or dependency features and generate syntactic patterns to extract relational facts. These pattern-based approaches cannot handle the complexity and diversity of languages well, and the extraction is usually far from satisfactory. To alleviate the burden of designing manual features, some neural ORE models have been proposed, typically adopting the methods of either sequence labeling or span selection. But their tasks still do not consider non-existent or multi-span relations. MGD-GNN (Lyu et al. 2021) for Chinese ORE constructs a multi-grained dependency graph and utilizes a span selection model to predict based on character features. Compared with our method, MGD-GNN heavily relies on dependency information and cannot deal with multiple languages. Jia, Xiang, and Chen transform English ORE into a sequence labeling process and present a hybrid neural network NST, whereas a dependency on PoS-taggers may introduce error propagation to NST. Improving NST, the current state-of-the-art ORE model LOREM (Harting, Mesbah, and Lofi 2020) works as a multilingual-embedded sequence-labeling method based on CNN/BiLSTM. Identical to our model, LOREM does not rely on language-specific knowledge or external tools. However, LOREM is restricted to binary-argument relation extraction and tends to output single spans rather than multi-span relations. Furthermore, based on our comparison of architectures in Section 4.1, LOREM suffers from inherent problems in learning long-range sequence dependencies (Vaswani et al. 2017), basic to computing token relevances to gold relations.

The benchmark ORE datasets in English (*En*) and Chinese (*Zh*) include OpenIE4$^{\text{En}}$ (Mausam 2016), LSOIE-wiki$^{\text{En}}$, LSOIE-sci$^{\text{En}}$ (Solawetz and Larson 2021), COER$^{\text{Zh}}$ (Tseng et al. 2014), and SAOKE$^{\text{Zh}}$ (Sun et al. 2018), whose contexts are complex or multiple sentences. Nevertheless, the five datasets do not contain tuples with non-existent relations. Moreover, tuples in the datasets except SAOKE$^{\text{Zh}}$ are in a fixed triple form of (*Arg_1, Single-span Rel., Arg_2*) without tuples of multi-span relations and possible polyadic arguments. To our best knowledge, SAOKE$^{\text{Zh}}$ is the only multi-span Chinese ORE dataset, and there are no English datasets with multi-span relations so far. We need to clarify that the triple-form data take the majority of tuples in common KGs and their extraction accuracy is the most influential part of the overall performance. But it is also significant to measure the multi-span extraction ability of an extractor since the multi-span form can express rich semantics and constitute diverse relationships in existing languages.

The motivations of our work are as follows. **(1) Non-existent relations**: Most knowledge graphs are constructed using a pipeline of named entity recognition (NER), relation extraction (with open relation normalization), and entity linking (Martínez-Rodríguez, López-Arévalo, and Ríos-Alvarado 2018). However, an arbitrary argument tuple identified from NER does not necessarily entail a semantic relationship, especially when the arguments are separated distantly from each other in a long context (typically in complex-sentence-level or document-level relation extraction). Thus, it is significant to facilitate ORE methods with the twofold ability to extract existing relation spans and detect non-existent relationships. **(2) Multi-span relations**: Some languages, such as Chinese, express the connections among the arguments in a flexible way of single and multiple spans, as illustrated in Figure 1. Multi-span relations are often linked to polyadic arguments from our observation of SAOKE$^{\text{Zh}}$. The multi-span form of relations generally represents richer and more precise meanings, thus playing an essential role in the diverse relationships in KGs. The above two further tasks of ORE are practical and vital demands of KG construction and general-domain information extraction, but neglected by the previous works. **(3) Data augmentation**: Since there are no datasets with non-existent relationships, we re-construct four public ORE datasets covering English and Chinese to derive the corresponding augmented datasets with non-existent relations for training models. **(4) Proposed QuORE framework**: We design a query-based multi-head framework QuORE to extract single/multi-span relations and detect non-existent relationships effectively. Given an argument tuple and its context, we first create a query template containing the argument information and derive a contextual representation of query and context via a pre-trained language model BERT (Devlin et al. 2019), which provides a deep understanding of query and context, and models the information interaction between them. The two sub-modules of multi-head framework are SSE (Single-span Extraction) and QASL (Query-based Sequence Labeling), which have different specialties in extraction. The sub-module SSE is used for effective single-span relation extraction, while QASL is designed for labeling non-existent relation sequences and multi-span relationships. Finally, our multi-head framework dynamically decides which sub-module (i.e., head) to predict a relation depending on the input.

To summarize, the main contributions of this work are:

- We define two further tasks of open relation extraction with non-existent and multi-span relationships considering the practical demands of ORE.
- By re-constructing some existing ORE datasets, we derive and publicize four augmented datasets with non-existent relationships and a multi-span relation dataset.
- We propose a query-based multi-head framework QuORE to extract single/multi-span relations and detect non-existent relationships effectively. We conduct extensive experiments, showing that our models outperform the state-of-the-art ORE method LOREM in the extraction of existing single/multi-span relations and the overall performances of non-existent relationships. We also give an in-depth analysis of the functions of QuORE sub-modules.

2 QuORE Framework

An overview of our QuORE framework is visualized in Figure 2. Given a context and an argument tuple, we first create a query from the arguments based on a template and encode the combination of query and context using the pre-trained BERT. The multi-head selection process dynamically determines a sub-module (SSE or QASL) to output a predicted open relation depending on the input.

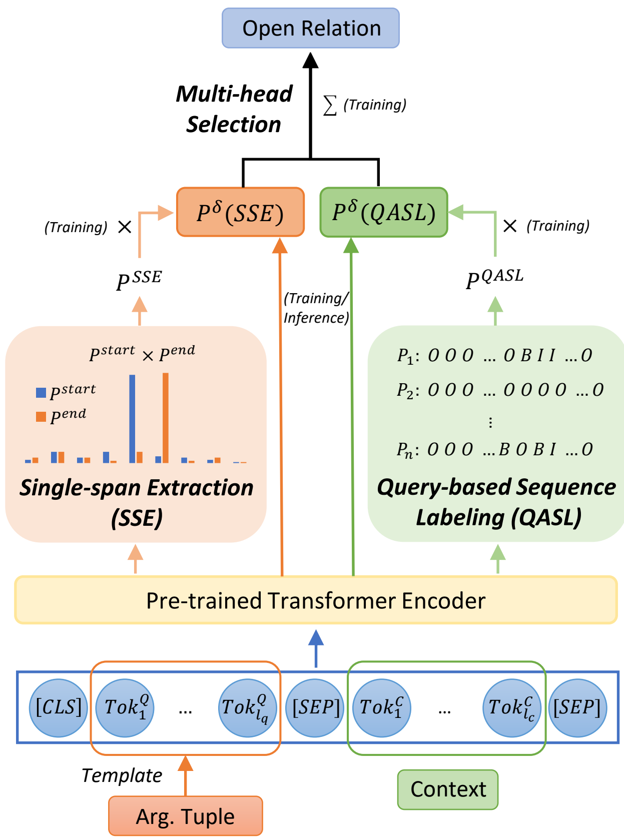


Figure 2: An overview of QuORE framework

2.1 Task Description

Given a context $\boldsymbol{C}$ and an argument tuple $\boldsymbol{A} = (A_1, A_2, ..., A_n)$ in $\boldsymbol{C}$, an ORE model needs to find the semantic relationship within the tuple $\boldsymbol{A}$ and detect non-existent relation cases, where n is the number of arguments. We denote the context as a word token sequence $\boldsymbol{C} = \{x_i^c\}_{i=1}^{l_c}$ and an argument as a text span $A_k = \{x_i^{a_k}\}_{i=1}^{l_{a_k}}$, where l_c is the context length and l_{a_k} is the k-th argument length. Our goal is to predict a relation $\boldsymbol{R} = (R_1, R_2, ..., R_u)$, a tuple of spans in the context, or return an empty string if there is no open relation, where u is the span number of $\boldsymbol{R}$, a relation span $R_j = \{x_i^{r_j}\}_{i=1}^{l_{r_j}}$ and l_{r_j} is the j-th relation span length.

2.2 Query Template Creation

Provided an argument tuple $(A_1, A_2, ..., A_n)$, we adopt a rule-based method to create the query template

$$\boldsymbol{T} = \langle s_1 \rangle\, A_1 \,\langle s_2 \rangle\, A_2 ... \langle s_n \rangle\, A_n \,\langle s_{n+1} \rangle \tag{1}$$

having $n+1$ slots for a possible relation $(R_1, R_2, ..., R_{n+1})$ with the span number up to $n+1$, where $\langle s_i \rangle$ indicates the i-th slot. The tokens filling a slot are separators of the adjacent arguments (e.g., double-quotes, a comma, or words of natural languages) or a placeholder for a relation span (e.g., a question mark or words of natural languages). In this paper, we adopt the question-mark (QM) style template $\boldsymbol{T}_{QM}$, taking the form of a structured argument-relationship tuple.

$$\boldsymbol{T}_{QM} = \text{“}A_1\text{”?“}A_2\text{”?...?“}A_n\text{”} \tag{2}$$

We also construct other templates, such as the w/o QM style, comma style, and language-specific natural-language (NL) style, but obtain marginal performance differences, where the experiments and analysis can be found in Appendix A.

2.3 Encoder

Given a context $\boldsymbol{C} = \{x_i^c\}_{i=1}^{l_c}$ with l_c tokens and a query $\boldsymbol{Q} = \{x_j^q\}_{j=1}^{l_q}$ with l_q tokens, we employ a pre-trained language model BERT (Devlin et al. 2019) as the encoder to learn the contextual representation for each token. First, we concatenate the query $\boldsymbol{Q}$ and the context $\boldsymbol{C}$ to derive the input $\boldsymbol{I}$ of encoder:

$$\boldsymbol{I} = \{[CLS], x_1^q, ..., x_{l_q}^q, [SEP], x_1^c, ..., x_{l_c}^c, [SEP]\} \tag{3}$$

where $[CLS]$ and $[SEP]$ denote the beginning token and the segment token, respectively.

Next, we generate the initial embedding $\boldsymbol{e}_i$ for each token by summing its word embedding $\boldsymbol{e}_i^w$, position embedding $\boldsymbol{e}_i^p$, and segment embedding $\boldsymbol{e}_i^s$. The sequence embedding $\boldsymbol{E} = \{\boldsymbol{e}_1, \boldsymbol{e}_2, ..., \boldsymbol{e}_m\}$ is then fed into the deep Transformer layers to learn contextual representations with **long-range sequence dependencies via the self-attention mechanism** (Vaswani et al. 2017). Finally, we obtain the last-layer hidden states $\boldsymbol{H} = \{\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_m\}$ as the contextual representation for the input sequence $\boldsymbol{I}$, where $\boldsymbol{h}_i \in \mathbb{R}^{d_h}$ and d_h indicates the dimension of the last hidden layer of BERT. The length of the sequences $\boldsymbol{I}$, $\boldsymbol{E}$, $\boldsymbol{H}$ is denoted as m where $m = l_q + l_c + 3$.

2.4 Single-span Extraction (SSE) Head

Span Prediction The SSE head aims to find a single span in a context as open relation. We utilize two learnable parameter matrices (feed-forward networks) $f_{start} \in \mathbb{R}^{d_h}$ and $f_{end} \in \mathbb{R}^{d_h}$ followed by the softmax normalization, then take each contextual token representation $\boldsymbol{h}_i$ in $\boldsymbol{H}$ as the input to produce the probability of each token i being selected as the start/end of relation span:

$$p_i^{start} = softmax(f_{start}(\boldsymbol{h}_1), ..., f_{start}(\boldsymbol{h}_m))_i \tag{4}$$

$$p_i^{end} = softmax(f_{end}(\boldsymbol{h}_1), ..., f_{end}(\boldsymbol{h}_m))_i \tag{5}$$

We denote $\boldsymbol{p}^{start} = \{p_i^{start}\}_{i=1}^m$ and $\boldsymbol{p}^{end} = \{p_i^{end}\}_{i=1}^m$.

Training The training objective of span prediction is defined as minimizing the cross entropy loss for the start and end selections,

$$p_k^{SSE} = p_{y_k^s}^{start} \times p_{y_k^e}^{end} \tag{6}$$

$$\mathbb{L}^{SSE} = -\frac{1}{N}\sum_k^N \log p_k^{SSE} \tag{7}$$

where y_k^s and y_k^e are respectively ground-truth start and end positions of example k. N is the number of examples.

Inference In the inference process, the relation span is extracted by finding the indices (s, e):

$$(s, e) = \underset{s \leq e}{\arg\max}(p_s^{start} \times p_e^{end}) \tag{8}$$

2.5 Query-based Sequence Labeling (QASL) Head

We tackle multi-span and non-existent ORE with query-based sequence labeling. The QASL head utilizes the same contextual representation $\boldsymbol{H}$ from the encoder as the SSE head, but instead of predicting the start and end probabilities, it outputs a probability distribution over a set of labels for each token. We use $\mathbb{V}$ to denote a label set. Formally, given $\boldsymbol{H}$, for each of the m tokens, the QASL head computes the probability of a label assigned to the i-th token as

$$\boldsymbol{p}_i^{label} = softmax(\boldsymbol{h}_i \boldsymbol{W}_{\mathbb{V}} + \boldsymbol{b}_{\mathbb{V}}) \tag{9}$$

where d_h is the hidden dimension of the encoder, $\boldsymbol{W}_{\mathbb{V}} \in \mathbb{R}^{d_h \times |\mathbb{V}|}$ and $\boldsymbol{b}_{\mathbb{V}} \in \mathbb{R}^{|\mathbb{V}|}$ are trainable parameters.

In this paper, we experiment with the `BIO` labeling scheme (Sang 2000; Huang, Xu, and Yu 2015), including three label types: `B` (beginning of a relation span), `I` (internal word in a relation span), and `O` (not part of a relation span). The labeling for a sequence of a non-existent relationship is all `O`. We illustrate the labeling for a sequence consisting of a multi-span relation with an instance from Figure 1: "*Donna*$_{\texttt{O}}$ *Karan*$_{\texttt{O}}$ *was*$_{\texttt{O}}$ *born*$_{\texttt{O}}$ *in*$_{\texttt{O}}$ *Long*$_{\texttt{O}}$ *Island*$_{\texttt{O}}$ *,*$_{\texttt{O}}$ *New*$_{\texttt{O}}$ *York*$_{\texttt{O}}$ *.*$_{\texttt{O}}$ *She*$_{\texttt{O}}$ *has*$_{\texttt{O}}$ *a*$_{\texttt{O}}$ *special*$_{\texttt{O}}$ *comprehension*$_{\texttt{O}}$ *of*$_{\texttt{O}}$ *New*$_{\texttt{O}}$ *York*$_{\texttt{O}}$ *,*$_{\texttt{O}}$ *known*$_{\texttt{B}}$ *by*$_{\texttt{I}}$ *the*$_{\texttt{O}}$ *world*$_{\texttt{O}}$ *as*$_{\texttt{B}}$ *a*$_{\texttt{O}}$ *cosmopolitan*$_{\texttt{O}}$ *city*$_{\texttt{O}}$ *.*$_{\texttt{O}}$"

Training We train the QASL head for predicting the correct sequence labeling $\boldsymbol{L} = (y_1, y_2, ..., y_m)$ corresponding to a ground-truth relation by minimizing the loss $\mathbb{L}^{QASL}$:

$$p_k^{QASL} = p(\boldsymbol{L}|\boldsymbol{H}_k) = \prod_{i=1}^{m} p(y_i|y_1, ..., y_{i-1}, \boldsymbol{H}_k) \tag{10}$$

$$\mathbb{L}^{QASL} = -\frac{1}{N}\sum_{k}^{N} \log p_k^{QASL} \tag{11}$$

where m denotes the length of $\boldsymbol{L}$, and $\boldsymbol{H}_k$ is the encoding output of example k. N is the number of examples.

Inference At test time, we would like to find and decode the most likely labeling $\hat{\boldsymbol{L}}$ from all the valid labelings. For the `BIO` scheme, the set of all valid labelings $\mathbb{S}$ includes all labelings that do not have an `I` after an `O`. Given $\boldsymbol{H}$, we hope to find:

$$\hat{\boldsymbol{L}} = \underset{\boldsymbol{L}_s \in \mathbb{S}}{\arg\max}\, p(\boldsymbol{L}_s|\boldsymbol{H}) \tag{12}$$

where the used maximization method is the Viterbi algorithm (Viterbi 1967) that performs in linear time.

2.6 Multi-head Framework

We employ a multi-head framework to handle diverse open relations in single/multi-span or non-existent forms. A head $\boldsymbol{\theta}$ is a module that inputs the contextual representation $\boldsymbol{H}$ and calculates a probability distribution over predictions. To derive a relation $\boldsymbol{R}_k$ for the example k having a context $\boldsymbol{C}_k$ and an argument tuple $\boldsymbol{A}_k$, a head $\boldsymbol{\theta}$ computes

$$p_k^{\boldsymbol{\theta}} = p^{\boldsymbol{\theta}}(\boldsymbol{R}_k|\boldsymbol{A}_k, \boldsymbol{C}_k) = p^{\boldsymbol{\theta}}(\boldsymbol{R}_k|\boldsymbol{H}_k) \tag{13}$$

In addition, we train an extra module $\boldsymbol{\delta}$ to decide which head to use for each example:

$$p_k^{\boldsymbol{\delta}} = p^{\boldsymbol{\delta}}(\boldsymbol{\theta}|\boldsymbol{A}_k, \boldsymbol{C}_k) = p^{\boldsymbol{\delta}}(\boldsymbol{\theta}|\boldsymbol{H}_k) \tag{14}$$

Training The training object of multi-head framework is to minimize the loss $\mathbb{L}^{MH}$ corresponding to the cumulative probability of each head's prediction:

$$p(\boldsymbol{R}_k|\boldsymbol{A}_k, \boldsymbol{C}_k) = \sum_{\boldsymbol{\theta}} p^{\boldsymbol{\delta}}(\boldsymbol{\theta}|\boldsymbol{A}_k, \boldsymbol{C}_k) \times p^{\boldsymbol{\theta}}(\boldsymbol{R}_k|\boldsymbol{A}_k, \boldsymbol{C}_k) \tag{15}$$

$$\mathbb{L}^{MH} = -\frac{1}{N}\sum_{k}^{N} \log p(\boldsymbol{R}_k|\boldsymbol{A}_k, \boldsymbol{C}_k) \tag{16}$$

Inference In the inference process, we first predict which head to function using Equation 14, then output the prediction from the chosen head.

With this framework, we integrate both SSE and QASL heads, dynamically deciding which head to utilize based on the input and achieving different forms of open relations.

3 Experimental Setup

We propose the following hypotheses and design a set of experiments to examine the performances of QuORE multi-head framework and its sub-modules. We choose LOREM as the comparison baseline because it is the state-of-the-art ORE method and the only multi-lingual neural model capable of processing English and Chinese texts. We arrange the hypotheses based on the following considerations: $\mathbf{H_1}$: As stated in Introduction, the extraction of (*Arg*$_1$*, Single-span Rel., Arg*$_2$) is the most influential part of the overall performance since the triple form takes the majority of tuples in KGs. Thus, we first evaluate on the original single-span OpenIE4$^{\text{En}}$, LSOIE-wiki$^{\text{En}}$, LSOIE-sci$^{\text{En}}$, COER$^{\text{Zh}}$ datasets. $\mathbf{H_2}$: As a follow-up to $\mathbf{H_1}$, we augment the above four datasets and investigate the twofold abilities of existing and non-existent relation extraction. $\mathbf{H_3}$ explores the multi-span relation extraction capabilities on the existing multi-span-style SAOKE$^{\text{Zh}}$.

- $\mathbf{H_1}$: For extracting existent single-span open relations, QuORE can outperform the SOTA model LOREM with notable performances of the multi-head framework and the SSE model.
- $\mathbf{H_2}$: For open relation extraction with non-existent relationships, our multi-head framework can process an input using a suitable sub-module, flexibly determining whether to extract or not and exceeding LOREM in overall performances.
- $\mathbf{H_3}$: For a challenging Chinese ORE dataset SAOKE$^{\text{Zh}}$ with multi-span relations and polyadic arguments, QuORE shows effectiveness in extracting relations with a diverse number of spans.

	OpenIE4$^{\text{En}}$			LSOIE-wiki$^{\text{En}}$			LSOIE-sci$^{\text{En}}$			COER$^{\text{Zh}}$			SAOKE$^{\text{Zh}}$	
	#Con	#Tup$_{\text{ori}}$	#Tup$_{\text{aug}}$	#Con	#Tup$_{\text{ori}}$	#Tup$_{\text{aug}}$	#Con	#Tup$_{\text{ori}}$	#Tup$_{\text{aug}}$	#Con	#Tup$_{\text{ori}}$	#Tup$_{\text{aug}}$	#Con	#Tup$_{\text{ori}}$
Train	11998	21295	30143	11872	11872	19145	11983	11983	18931	12000	13914	20463	11680	24451
Dev	1500	2660	3749	3142	3142	5045	1498	1498	2376	1500	1746	2572	1480	3059
Test	1499	2567	3647	3202	3202	5065	1499	1499	2380	1500	1771	2633	1461	3126

Table 1: Statistics of five datasets and corresponding augmentations. *(Con: Contexts; Tup: Tuples; ori: original; aug: augmented.)*

3.1 Datasets and Augmentation

We evaluate the performances of our proposed QuORE framework on five public datasets covering English (*En*) and Chinese (*Zh*). Table 1 lists the statistics of the used training, development and test sets.

- **OpenIE4$^{\text{En}}$**[2] was bootstrapped from extractions of OpenIE4 (Mausam 2016) from Wikipedia and annotated with POS and dependency information by Zhan and Zhao.
- **LSOIE-wiki$^{\text{En}}$** and **LSOIE-sci$^{\text{En}}$**[3] (Solawetz and Larson 2021) were algorithmically re-purposed from the QA-SRL BANK 2.0 dataset (FitzGerald et al. 2018), covering the domains of Wikipedia and science, respectively.
- **COER$^{\text{Zh}}$**[4] is a high-quality Chinese knowledge base, created by an unsupervised open extractor (Tseng et al. 2014) from heterogeneous web text.
- **SAOKE$^{\text{Zh}}$**[5] (Sun et al. 2018) is a human-annotated large-scale dataset for Chinese open information extraction, containing a single/multi-span relation and binary/polyadic arguments in a tuple.

In the data cleaning process, we only retain argument-relation tuples whose components are spans of a context. From our statistics, a tuple from OpenIE4$^{\text{En}}$, LSOIE-wiki$^{\text{En}}$, LSOIE-sci$^{\text{En}}$, and COER$^{\text{Zh}}$ is made up of a single-span relation and binary arguments. After preprocessing, we reconstruct SAOKE$^{\text{Zh}}$ into a dataset with single/multi-span relations and binary/polyadic arguments.

To adapt the data to our setting of ORE with non-existent relations, we augment the datasets that only contain binary-argument tuples. We implement a rule-based augmentation method, as shown in Algorithm 1. In this way, we add 0 or 1 non-existent relation to each context and consider the arguments having an order to specify the direction of a relation. The number of tuples before and after augmentation are recorded in Table 1 as #Tup$_{\text{ori}}$ and #Tup$_{\text{aug}}$, respectively. By randomly dividing a dataset, we obtain close ratios $R_{o/a} = \frac{\#Tup_{\text{ori}}}{\#Tup_{\text{aug}}}$ among the training, development, and test sets. The percents $R_{o/a}$ (%) of test sets in OpenIE4$^{\text{En}}$, LSOIE-wiki$^{\text{En}}$, LSOIE-sci$^{\text{En}}$, and COER$^{\text{Zh}}$ are respectively 70.48, 63.22, 62.98, and 67.26. We omit to augment SAOKE$^{\text{Zh}}$ for the complexity of permutations of multiple arguments and the ambiguity of Chinese that an argument tuple can entail a certain relation when containing some unnecessary argument or not.

[2]https://github.com/zhanjunlang/Span_OIE
[3]https://github.com/Jacobsolawetz/large-scale-oie
[4]https://github.com/TJUNLP/COER
[5]https://ai.baidu.com/broad/introduction?dataset=saoke

Algorithm 1 Augmentation for Non-existent Open Relations

Input: All tuples $Tps = [tps_1, tps_2, ..., tps_k]$ of existent relations in the corresponding k contexts, where $tps_i = [(A_{i,j,1}, R_{i,j}, A_{i,j,2})_{j=1}^{l_i}]$, $i \in [1, k]$ and l_i is the number of tuples in tps_i

Output: Augmented tuples $Tps^{aug} = [tps_1^{aug}, tps_2^{aug}, ..., tps_k^{aug}]$ with non-existent relations

Let Tps^{aug} be an empty list
for tuples tps in Tps **do**
  Let S_{arg} be an empty set and tps^{aug} be a deep copy of tps
  for tuple tp in tps **do**
    Find the arguments A_1, A_2 in tp
    S_{arg}.*ADD*(A_1) and S_{arg}.*ADD*(A_2)
  end for
  Construct the argument-pair permutations P_{arg} of length 2 from S_{arg}
  for tuple tp in tps **do**
    Remove the pair (A_1, A_2) of tp from P_{arg}
  end for
  Add a special empty pair (A_0, A_0) to P_{arg}
  Randomly select a pair $p_{non} = (A_x, A_y)$ from P_{arg}
  if $p_{non} \neq (A_0, A_0)$ **then**
    Create a non-existent relation tuple $tp_{non} = (A_x, empty\_str, A_y)$
    tps^{aug}.*ADD*(tp_{non}) then Tps^{aug}.*ADD*(tps^{aug})
  else
    Tps^{aug}.*ADD*(tps^{aug})
  end if
end for
return Tps^{aug}

3.2 Implementations

Encoder We utilize the *bert-base-cased* language model as the encoder on English datasets and *bert-base-chinese* on Chinese datasets.

Model Training BertAdam optimizer is used with default parameters and a learning rate of 3e-5. We train on a single NVIDIA Tesla P100 GPU with a batch size of 12 for 15 epochs with an early-stopping patience of 5. Evaluation is performed with our token-level evaluation script.

Sub-modules In some experiments, we implement the multi-head QuORE model along with its individual sub-modules (i.e., SSE and QASL models) which work solely controlled by our *not-to-integrate* option.

	OpenIE4^{En}				LSOIE-wiki^{En}			
	P	R	F1	EM	P	R	F1	EM
SSE	**.9911**	**.9934**	**.9910**	**.9817**	.9639	**.9703**	**.9650**	.9572
QASL	.9820	.9861	.9821	.9677	.9587	.9671	.9611	.9472
Multi-H	.9897	.9913	.9891	.9790	**.9653**	.9666	**.9650**	**.9600**
LOREM	.9287	.9441	.9216	.8098	.7199	.7108	.7138	.7017
	LSOIE-sci^{En}				COER^{Zh}			
	P	R	F1	EM	P	R	F1	EM
SSE	.9678	.9746	.9690	.9566	.9435	**.9585**	**.9377**	**.8837**
QASL	.9640	.9730	.9664	.9500	.9396	.9504	.9318	.8758
Multi-H	**.9730**	**.9793**	**.9740**	**.9620**	**.9475**	.9529	.9356	.8826
LOREM	.7812	.7698	.7736	.7585	.4537	.4535	.4228	.2693

Table 2: Evaluation on four datasets with all existing open relations. *(Multi-H: Multi-head.)*

3.3 Evaluation Metrics

We keep track of the metrics of F1 score, precision, recall, and Exact Match (EM). (1) For existing relations, the metrics are token-level. The F1 score measures the average overlap between a model's prediction and the ground-truth relation. Formally, F1 denotes the harmonic mean of precision and recall, where precision is defined as the ratio of correctly predicted tokens to the total number of predicted relation tokens. Recall, meanwhile, is the ratio of correctly predicted tokens to the total number of tokens in the ground-truth relation. Exact Match considers if a prediction exactly matches the correct relation, which means every token is the same. (2) For non-existent relations, we regard the values of F1, precision, and recall as the same as Exact Match, indicating if the prediction suggests non-existent, then the four metrics are all 1; if existing, then 0.

4 Experimental Results

4.1 $\mathbf{H_1}$: QuORE for Existing Single-span Relations

In $\mathbf{H_1}$, we train three individual models of SSE, QASL and multi-head to compare with the SOTA method LOREM. From Table 2, we observe that all the models of QuORE outperform LOREM on four existing single-span relation datasets of English and Chinese, with notable performances from the multi-head and SSE models. We next explain the advantages of QuORE over LOREM by comparing the two architectures.

LOREM encodes an input sequence using pre-trained word embeddings and adds argument tag vectors to the word embeddings. The argument tag vectors are simple one-hot encoded vectors indicating if a word is part of an argument. Then LOREM utilizes CNN and BiLSTM layers to form a representation of each word. The CNN is used to capture the local feature information, as LOREM considers that certain parts of the context might have higher chances of containing relation words than others. Meanwhile, the BiLSTM captures the forward and backward context of each word. Next, a CRF layer tags each word using the NST tagging scheme (Jia, Xiang, and Chen 2018): `S` (Single-word relation), `B` (Beginning of a relation), `I` (Inside a relation), `E` (Ending of a relation), `O` (Outside a relation).

Our QuORE framework generates the initial representations with word embeddings and position embeddings from BERT (Devlin et al. 2019). Unlike the simple one-hot argument vectors of LOREM, QuORE derives the argument information by creating a query template of arguments. We combine the query with the context to form the input of encoder, and the encoder outputs a contextual representation that we utilize to compute the relevance of each token to a gold relation (Equation 4, 5 for SSE and Equation 9 for QASL). Moreover, by employing the self-attention mechanism of the Transformers-based encoder, QuORE has the benefit of learning long-range dependencies easier and deriving a better representation for computing relevances, which we interpret in the following. Learning long-range dependencies is a key challenge in encoding sequences and solving related tasks (Vaswani et al. 2017). One key factor affecting the ability to learn such dependencies is the length of the paths forward and backward signals have to traverse between any two input and output positions in the network. The shorter these paths between any combination of positions in the input and output sequences, the easier it is to learn long-range dependencies. Vaswani et al. also provide the maximum path length between any two input and output positions in self-attention, recurrent, and convolutional layers, which are $O(1)$, $O(n)$, and $O(\log_k(n))$, respectively. (k is the kernel width of a convolutional layer.) The constant path length of self-attention makes it easier to learn long-range dependencies than CNN and BiLSTM layers. Despite the gating in LSTMs and gradient clipping (Graves 2013), recurrent layers are difficult to optimize for such long dependencies due to gradient vanishing and explosion problems. As for CNNs, the way they derive dependencies is by applying different kernels to a sequence. For example, a kernel of size 2 learns connections between pairs of words, a kernel of size 3 captures connections between triplets of words. The evident problem here is that the number of kernels required to capture dependencies among all combina-

	$\text{OpenIE4}^{\text{En}}_{\text{Aug}}$				$\text{LSOIE-wiki}^{\text{En}}_{\text{Aug}}$			
	P	R	F1	EM	P	R	F1	EM
Multi-H [All]	.9602	.9624	**.9605**	.9544	.9165	.9170	**.9159**	.9102
[Exg.]	.9744	.9774	**.9748**	.9661	.9582	.9589	**.9572**	.9482
[Non.]	.9265	.9265	**.9265**	.9265	.8449	.8449	.8449	.8449
LOREM [All]	.8971	.8855	.8816	.8108	.7811	.7751	.7771	.7692
[Exg.]	.9024	.8859	.8803	.7798	.7205	.7111	.7142	.7017
[Non.]	.8847	.8847	.8847	.8847	.8857	.8857	**.8857**	.8857
	$\text{LSOIE-sci}^{\text{En}}_{\text{Aug}}$				$\text{COER}^{\text{Zh}}_{\text{Aug}}$			
	P	R	F1	EM	P	R	F1	EM
Multi-H [All]	.9402	.9450	**.9411**	.9345	.9415	.9397	**.9307**	.8959
[Exg.]	.9605	.9680	**.9619**	.9513	.9396	.9369	**.9235**	.8718
[Non.]	.9058	.9058	.9058	.9058	.9455	.9455	**.9455**	.9455
LOREM [All]	.7983	.7912	.7935	.7840	.6053	.5888	.5784	.4839
[Exg.]	.7225	.7111	.7149	.6998	.4420	.4175	.4020	.2614
[Non.]	.9281	.9281	**.9281**	.9281	.9408	.9408	.9408	.9408

Table 3: Evaluation on four augmented datasets with non-existent open relations. *(Multi-H: Multi-head; Exg.: Existing; Non.: Non-existent; Aug: Augmented.)* The measures of [All], [Exg.], and [Non.] are the performances on a whole dataset and the subsets of existing and non-existent relationships, respectively.

tions of words would be enormous and unpractical, because of the exponentially growing combinations when increasing the maximum length of sequences. Overall, QuORE achieves substantial improvements over LOREM in extracting existing relations due to the better sequence representations with long-term dependencies, a basis of computing token relevances to gold relations.

If we focus on the performances of SSE, QASL and multi-head models, we find that the results from multi-head and SSE are relatively more significant than QASL. The SSE model exceeds QASL with marginal improvements in F1 scores on the four datasets of single-span relations, which is in line with the advantage of SSE on the single-span extraction task. The results also suggest that the single/multi-span QASL may be used by itself as a more general extraction method on these single-span relation datasets. Meanwhile, we observe that the multi-head models maintain high precisions due to integration.

4.2 $\mathbf{H_2}$: QuORE with Non-existent Relations

For open relation extraction with non-existent relationships, we compare our multi-head QuORE model and LOREM on four augmented datasets, as shown in Table 3. We also provide an in-depth analysis of the functions of each sub-module (i.e., head) in our multi-head framework via the visualization in Figure 3.

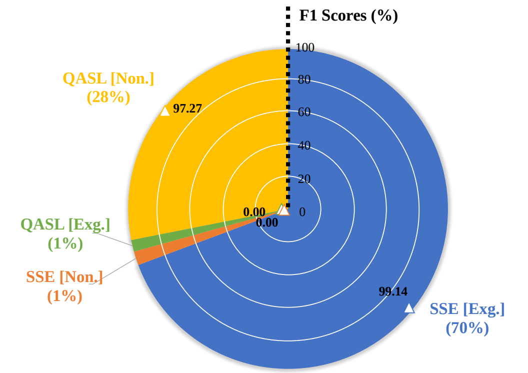
Table 3 presents the evaluation results of multi-head QuORE and LOREM on the augmented datasets and their subsets of existing and non-existent relationships. Our multi-head model achieves more precise outcomes on the overall performances and the parts of existing relations than LOREM on all four datasets. LOREM, meanwhile, outperforms on the non-existent relation subsets of $\text{LSOIE-wiki}^{\text{En}}_{\text{Aug}}$ and $\text{LSOIE-sci}^{\text{En}}_{\text{Aug}}$. If we focus on contrasting the partial measures between the [Exg.] and [Non.] parts of multi-head QuORE itself, we observe that the performances in [Non.] part are not compatible with the ones in [Exg.] on the datasets except $\text{COER}^{\text{Zh}}_{\text{Aug}}$, implying there exists improvement space for the task of verifying non-existent relations.

We further probe into the functions of SSE and QASL heads in our multi-head framework by studying the data percents processed by each head and the corresponding F1 scores in Figure 3. By observation, the Subfigures 3a, 3c, and 3d present a similar trend in the data distribution while the Subfigure 3b is different. In the following, we first analyze the performances on the three datasets of $\text{OpenIE4}^{\text{En}}_{\text{Aug}}$, $\text{LSOIE-sci}^{\text{En}}_{\text{Aug}}$, and $\text{COER}^{\text{Zh}}_{\text{Aug}}$.

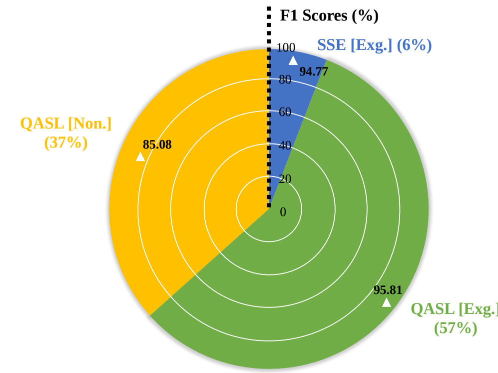
In the Subfigures 3a, 3c, and 3d, the vast majority of examples with existing relations are distributed to the SSE head due to the observed high precision of SSE in extracting single spans from Table 2. Meanwhile, the QASL head is responsible for most non-existent relationships, obtaining notable results with F1 scores (%) of 97.27, 94.10, and 95.88 on its processed [Non.] parts of $\text{OpenIE4}^{\text{En}}_{\text{Aug}}$, $\text{LSOIE-sci}^{\text{En}}_{\text{Aug}}$, and $\text{COER}^{\text{Zh}}_{\text{Aug}}$, respectively. We also observe that SSE is not available to detect non-existent relations from 3a and 3c, and QASL cannot extract effectively if given only a small portion of existing relation examples.

The data distribution in $\text{LSOIE-wiki}^{\text{En}}_{\text{Aug}}$ (Subfigure 3b) is different from the other three datasets, suggesting that our multi-head framework can dynamically decide which head to use if the overall data feature of a dataset changes. In Subfigure 3b, the non-existent relations are allocated to the QASL head similarly to the other three charts, but QASL also processes the large part of existing relations, illustrating that QASL has the general ability to extract existing relations and detect non-existent relationships.

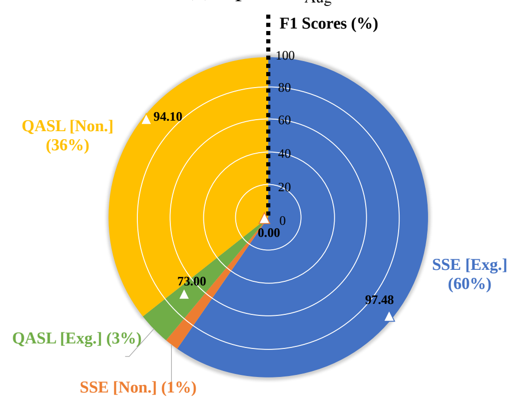
On the whole, the overall performances from Table 3 and



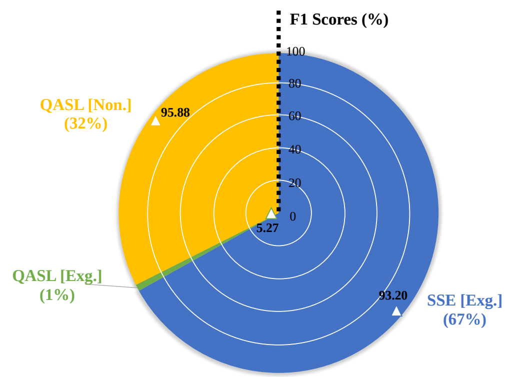
(a) OpenIE4$^{\text{En}}_{\text{Aug}}$



(b) LSOIE-wiki$^{\text{En}}_{\text{Aug}}$



(c) LSOIE-sci$^{\text{En}}_{\text{Aug}}$



(d) COER$^{\text{Zh}}_{\text{Aug}}$

Figure 3: Performances of the SSE and QASL sub-modules in multi-head QuORE on existing and non-existent relations from the four augmented datasets. *(Exg.: Existing; Non.: Non-existent.)* The percents in the parentheses are the data percentage processed by a sub-module, and [Exg.] or [Non.] denotes the relation type. The F1 scores are beside the triangles of each part in a pie chart.

the detailed measures of each head in Figure 3 show that our multi-head model can determine a suitable sub-module to work dynamically based on the input data mixed with both existing and non-existent relationships.

4.3 $\mathrm{H_3}$: QuORE with Multi-span Relations

In the challenging Chinese dataset SAOKE$^{\text{Zh}}$, the relation tuples are not necessarily in the form of single-span relations and binary arguments; some have multi-span relations along with polyadic arguments, as shown in Figure 1. We evaluate the multi-span extraction abilities of QuORE and LOREM on SAOKE$^{\text{Zh}}$ and present the results in Table 4 and Figure 4.

Table 4 shows that our individual SSE, QASL and multi-head models all outperform LOREM. We also investigate from the results on SAOKE$^{\text{Zh}}$ and COER$^{\text{Zh}}$ from Tables 2 and 4 that the performances of LOREM on the Chinese datasets are dissatisfactory. If we focus on the extracted number of spans by the multi-head QuORE and LOREM in Figure 4, we discover that LOREM tends to output single-span relations while QuORE is able to generate a various number of spans dynamically depending on the input and achieve relatively high precisions. The sub-module SSE accounts for the effective single-span extraction, and QASL contributes to the diverse outputs of multi-span relationships. Thus, the overall performance of multi-head model is influenced by the above specialties of its two sub-modules.

4.4 Case Study

We conduct case studies of the above three hypotheses to compare the predictions of the multi-head QuORE and LOREM. We present context words in double-quotes.

- Cases in $\mathbf{H_1}$: (1) We notice that QuORE is better at handling cases where arguments separately locate in main and subordinate clauses than LOREM. For instance, given a sentence "*Different enzymes that catalyze the same chemical reaction are called isozymes.*" and the arguments ("*Different enzymes*", "*the same chemical reaction*"), QuORE gives out the gold relation "*catalyze*" whereas LOREM predicts "*called*" incorrectly. (2) We summarize that the major errors of QuORE occur in cases where the relation concerns modal verbs or auxiliary verbs, such as "*can*" and "*have been*". QuORE may extract either fewer or more words than a gold relation, e.g., provided a sentence "*Algae had covered moist land*

	SAOKE^{Zh}			
	P	R	F1	EM
SSE	**.9123**	.8015	.8114	.6045
QASL	.8937	.9162	.8867	**.7389**
Multi-H	.8918	**.9220**	**.8887**	.7347
LOREM	.6904	.5349	.5657	.3544

Table 4: Evaluation on SAOKE^{Zh} with multi-span relations

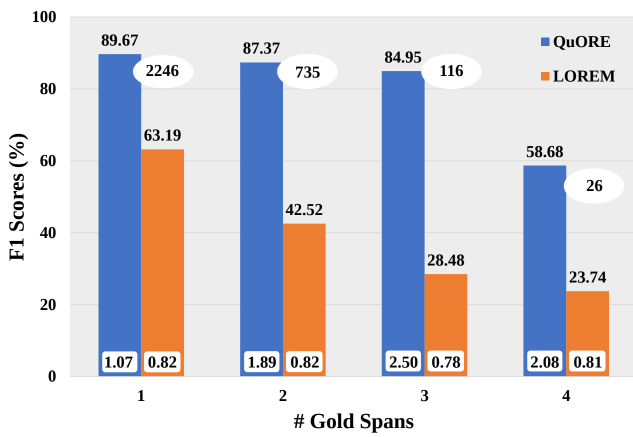


Figure 4: Comparison of QuORE (multi-head) and LOREM on the test set of SAOKE^{Zh} by the number of spans in gold relations. Labels at the bottom indicate the average number of predicted spans. Ovals at the top denote the number of examples.

areas for millions of years." and the arguments ("*Algae*", "*moist land areas*"), QuORE predicts "*covered*" as the relation, fewer than the ground truth "*had covered*". Meanwhile, we observe that LOREM also outputs "*covered*".

- Case in $\mathbf{H_2}$: QuORE can take the argument order into account and detect a non-existent relation. Given a context "*The bus was out-of-control and going downhill when it struck a truck from behind.*" and the arguments ("*downhill*", "*The bus*"), our model correctly predicts a non-existent relationship considering the reverse order of arguments.
- Case in $\mathbf{H_3}$: Provided a multi-span case of a context "“远望5号”测量船给“嫦娥一号”下达指令，指示“嫦娥”不断变轨，使其按照固定轨道顺利运行。" and the arguments ("“远望5号”", "“嫦娥”"), QuORE outputs the gold multi-span relation ("指示", "不断变轨").

5 Related Work

We have reviewed the previous open relation extraction systems and datasets in the Introduction. In this section, we focus on the related works of our model framework.

Multi-span Extraction Some recent studies on machine reading comprehension have designed different models to extract multi-span answers. Segal et al. cast the multi-span question answering task as a sequence tagging problem, predicting for each token whether it is part of the answer. Yang, Zhang, and Zhao modify the single-span extraction method by adding a special virtual span to generate multi-span answers. Hu et al. propose to predict the number of output spans for each question and use a non-differentiable inference procedure to find them in the text.

Multi-head Models There has been substantial interest in training a multi-head model for multiple tasks, including the fields of languages (Collobert and Weston 2008; Liu et al. 2019; McCann et al. 2018; Dua et al. 2019; Hu et al. 2019; Segal et al. 2020), computer vision (Lu et al. 2020), and robotics (Teh et al. 2017).

6 Conclusion

Our work targets the practical demands of open relation extraction with non-existent and multi-span relationships using a novel query-based multi-head framework QuORE. The experiments and analyses present the effective performances in the multiple tasks and corresponding explanations. In the future, we will enhance the model architecture, transfer our model to other languages, and explore the few/zero-shot learning abilities in low-resource languages and domains.

A Different Query Templates

		QM	w/o QM	Comma	NL^{En}
SSE	F1	0.9650	0.9654	0.9660	**0.9663**
	EM	0.9572	0.9594	0.9588	**0.9597**
QASL	F1	**0.9611**	0.9548	0.9537	0.9585
	EM	**0.9472**	0.9422	0.9419	0.9403
Multi-H	F1	0.9650	0.9664	**0.9674**	0.9640
	EM	**0.9600**	0.9582	**0.9600**	0.9572

Table 5: Evaluation of different queries on LSOIE-wiki^{En}

We analyze the effects of different query templates by experimenting with the question-mark (QM) style $\boldsymbol{T}_{QM}$, w/o QM style $\boldsymbol{T}_{w/oQM}$, comma style $\boldsymbol{T}_{comma}$, and language-specific natural-language (NL) style $\boldsymbol{T}_{NL}$ in English (*En*) or Chinese (*Zh*), as listed below. However, the evaluation results of these templates have marginal differences, which is observed on all datasets used in this paper. We present the evaluation of different queries on LSOIE-wiki^{En} in Table 5. The possible reason for the marginal results is that all the templates consist of the necessary argument information, and the representations learned via a pre-trained Transformers encoder are similar due to the robust expression ability of the encoder.

- $\boldsymbol{T}_{w/oQM}$ = "A_1" "A_2" ... "A_n"
- $\boldsymbol{T}_{comma}$ = "A_1", "A_2", ..., "A_n"
- $\boldsymbol{T}_{NL^{En}}$ = What is the relation among "A_1", ..., and "A_n"?
- $\boldsymbol{T}_{NL^{Zh}}$ = "A_1", "A_2", ..., "A_n"之间的关系是?

Acknowledgments

This work is supported by the NSFC-General Technology Basic Research Joint Funds under Grant (U1936220), the National Natural Science Foundation of China under Grant (61972047) and the National Key Research and Development Program of China (2018YFC0831500).

References

Banko, M.; Cafarella, M. J.; Soderland, S.; Broadhead, M.; and Etzioni, O. 2007. Open information extraction from the web. In Veloso, M. M., ed., *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, 2670–2676.

Collobert, R., and Weston, J. 2008. A unified architecture for natural language processing: deep neural networks with multitask learning. In Cohen, W. W.; McCallum, A.; and Roweis, S. T., eds., *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, volume 307 of *ACM International Conference Proceeding Series*, 160–167. ACM.

Corro, L. D., and Gemulla, R. 2013. Clausie: clause-based open information extraction. In Schwabe, D.; Almeida, V. A. F.; Glaser, H.; Baeza-Yates, R.; and Moon, S. B., eds., *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, 355–366. International World Wide Web Conferences Steering Committee / ACM.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, 4171–4186. Association for Computational Linguistics.

Dua, D.; Wang, Y.; Dasigi, P.; Stanovsky, G.; Singh, S.; and Gardner, M. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, 2368–2378. Association for Computational Linguistics.

Fader, A.; Soderland, S.; and Etzioni, O. 2011. Identifying relations for open information extraction. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL*, 1535–1545. ACL.

FitzGerald, N.; Michael, J.; He, L.; and Zettlemoyer, L. 2018. Large-scale QA-SRL parsing. In Gurevych, I., and Miyao, Y., eds., *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, 2051–2060. Association for Computational Linguistics.

Graves, A. 2013. Generating sequences with recurrent neural networks. *CoRR* abs/1308.0850.

Harting, T.; Mesbah, S.; and Lofi, C. 2020. LOREM: language-consistent open relation extraction from unstructured text. In Huang, Y.; King, I.; Liu, T.; and van Steen, M., eds., *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, 1830–1838. ACM / IW3C2.

Hu, M.; Peng, Y.; Huang, Z.; and Li, D. 2019. A multi-type multi-span network for reading comprehension that requires discrete reasoning. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, 1596–1606. Association for Computational Linguistics.

Huang, Z.; Xu, W.; and Yu, K. 2015. Bidirectional LSTM-CRF models for sequence tagging. *CoRR* abs/1508.01991.

Jia, S.; Xiang, Y.; and Chen, X. 2018. Supervised neural models revitalize the open relation extraction. *CoRR* abs/1809.09408.

Liu, X.; He, P.; Chen, W.; and Gao, J. 2019. Multi-task deep neural networks for natural language understanding. In Korhonen, A.; Traum, D. R.; and Màrquez, L., eds., *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28-August 2, 2019, Volume 1: Long Papers*, 4487–4496. Association for Computational Linguistics.

Lu, J.; Goswami, V.; Rohrbach, M.; Parikh, D.; and Lee, S. 2020. 12-in-1: Multi-task vision and language representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 10434–10443. Computer Vision Foundation / IEEE.

Lyu, Z.; Shi, K.; Li, X.; Hou, L.; Li, J.; and Song, B. 2021. Multi-grained dependency graph neural network for chinese open information extraction. In Karlapalem, K.; Cheng, H.; Ramakrishnan, N.; Agrawal, R. K.; Reddy, P. K.; Srivastava, J.; and Chakraborty, T., eds., *Advances in Knowledge Discovery and Data Mining - 25th Pacific-Asia Conference, PAKDD 2021, Virtual Event, May 11-14, 2021, Proceedings, Part III*, volume 12714 of *Lecture Notes in Computer Science*, 155–167. Springer.

Martínez-Rodríguez, J.; López-Arévalo, I.; and Ríos-Alvarado, A. B. 2018. Openie-based approach for knowledge graph construction from text. *Expert Syst. Appl.* 113:339–355.

Mausam. 2016. Open information extraction systems and downstream applications. In Kambhampati, S., ed., *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, 4074–4077. IJCAI/AAAI Press.

McCann, B.; Keskar, N. S.; Xiong, C.; and Socher, R. 2018.

The natural language decathlon: Multitask learning as question answering. *CoRR* abs/1806.08730.

Qiu, L., and Zhang, Y. 2014. ZORE: A syntax-based system for chinese open relation extraction. In Moschitti, A.; Pang, B.; and Daelemans, W., eds., *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, 1870–1880. ACL.

Sang, E. F. T. K. 2000. Transforming a chunker to a parser. In Daelemans, W.; Sima'an, K.; Veenstra, J.; and Zavrel, J., eds., *Computational Linguistics in the Netherlands 2000, Selected Papers from the Eleventh CLIN Meeting, Tilburg, November 3, 2000*, volume 37 of *Language and Computers - Studies in Practical Linguistics*, 177–188. Rodopi.

Segal, E.; Efrat, A.; Shoham, M.; Globerson, A.; and Berant, J. 2020. A simple and effective model for answering multi-span questions. In Webber, B.; Cohn, T.; He, Y.; and Liu, Y., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, 3074–3080. Association for Computational Linguistics.

Solawetz, J., and Larson, S. 2021. LSOIE: A large-scale dataset for supervised open information extraction. In Merlo, P.; Tiedemann, J.; and Tsarfaty, R., eds., *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, 2595–2600. Association for Computational Linguistics.

Sun, M.; Li, X.; Wang, X.; Fan, M.; Feng, Y.; and Li, P. 2018. Logician: A unified end-to-end neural approach for open-domain information extraction. In Chang, Y.; Zhai, C.; Liu, Y.; and Maarek, Y., eds., *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018*, 556–564. ACM.

Teh, Y. W.; Bapst, V.; Czarnecki, W. M.; Quan, J.; Kirkpatrick, J.; Hadsell, R.; Heess, N.; and Pascanu, R. 2017. Distral: Robust multitask reinforcement learning. In Guyon, I.; von Luxburg, U.; Bengio, S.; Wallach, H. M.; Fergus, R.; Vishwanathan, S. V. N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 4496–4506.

Tseng, Y.; Lee, L.; Lin, S.; Liao, B.; Liu, M.; Chen, H.; Etzioni, O.; and Fader, A. 2014. Chinese open relation extraction for knowledge acquisition. In Bouma, G., and Parmentier, Y., eds., *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014, April 26-30, 2014, Gothenburg, Sweden*, 12–16. The Association for Computer Linguistics.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need. In Guyon, I.; von Luxburg, U.; Bengio, S.; Wallach, H. M.; Fergus, R.; Vishwanathan, S. V. N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 5998–6008.

Viterbi, A. J. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. Inf. Theory* 13(2):260–269.

Yang, J.; Zhang, Z.; and Zhao, H. 2021. Multi-span style extraction for generative reading comprehension. In Veyseh, A. P. B.; Dernoncourt, F.; Nguyen, T. H.; Chang, W.; and Celi, L. A., eds., *Proceedings of the Workshop on Scientific Document Understanding co-located with 35th AAAI Conference on Artificial Inteligence, SDU@AAAI 2021, Virtual Event, February 9, 2021*, volume 2831 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Zhan, J., and Zhao, H. 2020. Span model for open information extraction on accurate corpus. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, 9523–9530. AAAI Press.