DORE: Document Ordered Relation Extraction based on Generative Framework

Qipeng Guo[2]*, Yuqing Yang[1]*†, Hang Yan[1], Xipeng Qiu[1]‡, Zheng Zhang[2]
[1]School of Computer Science, Fudan University
[2]Amazon AWS AI
{gqipeng, zhaz}@amazon.com
yuqingyang21@m.fudan.edu.cn, {hyan19, xpqiu}@fudan.edu.cn

Abstract

In recent years, there is a surge of generation-based information extraction work, which allows a more direct use of pre-trained language models and efficiently captures output dependencies. However, previous generative methods using lexical representation do not naturally fit document-level relation extraction (DocRE) where there are multiple entities and relational facts. In this paper, we investigate the root cause of the underwhelming performance of the existing generative DocRE models and discover that the culprit is the inadequacy of the training paradigm, instead of the capacities of the models. We propose to generate a symbolic and ordered sequence from the relation matrix which is deterministic and easier for model to learn. Moreover, we design a parallel row generation method to process overlong target sequences. Besides, we introduce several negative sampling strategies to improve the performance with balanced signals. Experimental results on four datasets show that our proposed method can improve the performance of the generative DocRE models. We have released our code at https://github.com/ayyyq/DORE.

1 Introduction

Document-level relation extraction (DocRE) is a fundamental information extraction (IE) task which aims to extract relational facts among entities across multiple sentences. For IE, most previous approaches are classification-based, which first extract features of certain objects using pre-trained language models and then classify according to the merged features. Recent years have witnessed a rising trend of regarding the task of IE as a sequence generation problem, linearizing the extracted structures as a sequence. Compared to classification-based methods, generative framework extracts features and classifies simultaneously, allowing a more direct use of latent knowledge in pre-trained language models without an untrained classification module on the top. Besides, the generation process can naturally recover high-order dependencies when generating the output step by step. Generation-based methods have been successfully adapted to many settings including universal IE which intends to solve several IE tasks in a unified way (Paolini et al., 2021; Lu et al., 2022), low-resource (Hsu et al., 2021), and transfer learning (Liu et al., 2021, 2022), and have achieved competitive results on most sentence-level benchmarks (Cui et al., 2021; Liu et al., 2021) and document-level event extraction task (Li et al., 2021; Zhang et al., 2022).

Considering that generative framework is simple and effective, prior work adopts it for DocRE (Huang et al., 2021; Giorgi et al., 2022). This line of work features *lexical generation* since they use natural language to represent entities and relations, which directly borrows from text generation tasks (Lewis et al., 2020; Raffel et al., 2020). Also, they need special separator tokens to distinguish token spans. However, the lexical generation paradigm dose not perfectly fit the more complex DocRE task, where the source sequence contains numerous entities and relations (e.g., a document can contain up to about 40 relation instances on DocRED), leading to a performance gap between generation-based and classification-based methods. Our experiment verifies that the generative baseline performs 6.00 points worse than classification-based methods on DocRED dataset (Yao et al., 2019).

The lexical generation paradigm faces two significant challenges, which impede its performance. (1) **non-unique target sequence**: for the example shown in Figure 1, a document often needs to mention the same knowledge many times, each of which could be represented in multiple ways (i.e.,

*Equal contribution.
† Work done during internship at Amazon Shanghai AI Lab.
‡Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 3463–3474
December 7-11, 2022

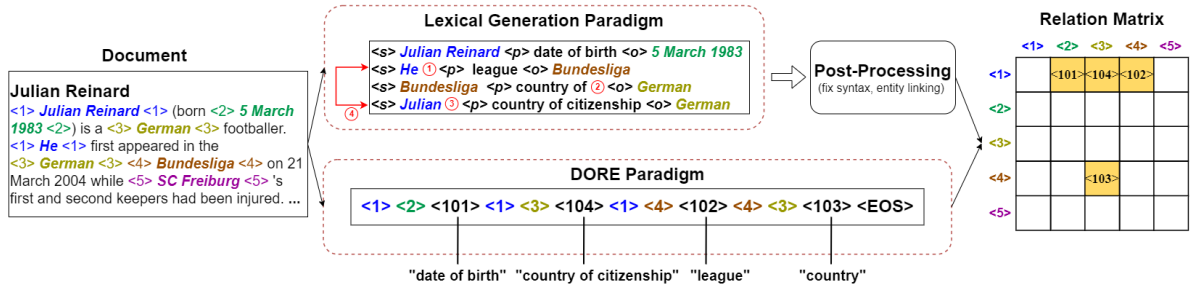


Figure 1: An example from DocRED dataset, and we highlight entities with different colors. The left side is the input document and the right side is the target relation matrix. Each cell in this matrix is filled with the relation between two entities. We compare two paradigms in the middle, and the lexical sequence is much longer than our DORE. Meanwhile, there are four weaknesses in lexical generation paradigm. ① shows the case of using different mentions. Both "Julian Reinard" and "He" point to the same entity, but only one of them are used in the annotation. And this leads to an incorrect training signal if the model use a different mention. In ②, the model struggles to choose from two similar relations "country" and "country of citizenship". The "country of" is a meaningful lexicon but it is not valid in the relation vocabulary. In ③, the model outputs a new mention "Julian" that does not appear in the text. ④ shows that the prediction order does not follow the human reading order, as the knowledge of citizenship appears before the league information according to the document. Lexical generation paradigm adopts a post-processing step to address above issues. In contrast, our DORE paradigm directly predicts elements in the relation matrix.

diverse lexical forms for an entity). Pre-defining a certain lexical form of an entity will introduce incorrect bias, and a complicated post-processing step is needed to align the generated sequence and relational facts. (2) **overlong generated sequence**: DocRE requires the model to extract more facts, leading to much longer output sequences, and thus causes difficulties to efficiency and memory support. However, it is hard for lexical generation approaches using natural language representation and extra separator tokens to cope with such dilemma in a concise way.

To alleviate issues in the lexical generation paradigm, we treat generative DocRE as deterministically generating a relation matrix where each cell corresponds to an entity pair with predefined relation or no relation. The paradigm, which we call DORE (**D**ocument **O**rdered **R**elation **E**xtraction), assigns each entity and relation a special id and linearizes the relation matrix in the row-column order, resulting a symbolic ordered sequence. It is much easier to learn and control generation. In addition, the paradigm is able to resolve overlong output sequences in a concise way when generating rows of the relation matrix in parallel. Besides, we show that the loss function taken from previous work is imbalanced for the complicated DocRE, and we introduce several negative sampling strategies to mitigate it. Taken together, we find that the underwhelming performance of generative framework for DocRE comes from the improper training and generation ways instead of the model architecture.

We conduct experiments on four popular DocRE benchmarks. We improve the generative model's F_1 score from 51.36 to 60.67 for DocRED by changing training paradigm only, and further improve it to 65.26 with distantly supervised training data (officially collected by DocRED). Besides, we bridge the performance gap between generation-based and classification-based methods on CDR (Li et al., 2016) and GDA (Wu et al., 2019) by obtaining 72.6 and 85.3 F_1 score, individually. We also achieve new state-of-the-art results on SciREX (Jain et al., 2020) both with gold inputs and end-to-end for binary and 4-ary relation extraction. Our work brings generative framework to DocRE into a performance region that matches classification-based approaches, with the added advantage of supporting high-order relation discovery afforded by the nature of sequence-to-sequence models.

2 Related Work

2.1 Generation-based Information Extraction

In recent years, more and more work seeks to use a new generative paradigm to solve information extraction tasks. Paolini et al. (2021); Zhang et al. (2021b) transform IE tasks into translation between label-augmented texts, Yan et al. (2021); Lu et al. (2021); Huang et al. (2021); Zhang et al. (2022) design a linearization schema with constrained decoding strategies, and Li et al. (2021); Hsu et al. (2021); Liu et al. (2022) adopt template-based conditional generation. Though simple the paradigm seems, generation-based methods report compet-

itive results especially on sentence-level benchmarks. However, previous methods can not scale to the document-level relation extraction task which requires to extract multiple facts, or perform worse than most classification-based methods.

2.2 Classification-based Document-Level Relation Extraction

Most previous work treats DocRE as a classification task, which typically breaks down the task into two stages, extracting the feature of entities followed by classifying the relation of every entity pair according to their features. More specifically, a stream of classification-based work introduces the graph structure on top of pre-trained representations to address long-term dependencies and multi-hop reasoning (Nan et al., 2020; Wang et al., 2020; Zeng et al., 2020, 2021; Xu et al., 2021b). However, for long document, compared with keeping a graph representation and merging new relations parsed from new paragraphs, using, for example, a seq2seq model, is a more scalable approach, and a direction worth exploring.

Recent work enhances classification-based methods in different aspects. Zhang et al. (2021a) tackles the problem of lacking high-order dependencies by introducing convolution on relation matrices to encourage interaction among relations. On the other hand, Xu et al. (2021a); Xiao et al. (2021) enrich the features by introducing linguistic knowledge or statistic information of entities. Another popular idea (Huang et al., 2020; Xie et al., 2021) is to detect the evidence sentences before relation extraction. This line of work provides a strong guideline for relation extraction and reduces irrelevant contexts. Some of these ideas are complementary to DORE; our core idea is to understand how generative framework can regain its advantages in dealing with document-level relation extraction and high-order relation discovery.

3 Method

3.1 Task Formulation

Document-level relation extraction task aims to extract relational facts given a document $\boldsymbol{D}$ and a set of entities $\boldsymbol{E}$. Each entity e_i is represented as the set of its coreferent mentions $\{e_i^j\}$ in the document, some of which have different natural language forms. Each of the extracted instances can be expressed as a tuple $(e_1, \ldots, e_k, r)$, where k is the number of participating entities, and r is from a pre-defined set of relations. We focus on binary and 4-ary relation extraction, that is, $k = 2$ or 4.

Since relation instances in the document can naturally formulate a matrix, we frame the generative DocRE as generating a relation matrix. Take binary relation extraction as an example. As shown in Figure 1, each cell (i, j) in the relation matrix corresponds to an entity pair with head entity e_i and tail entity e_j, and can be filled with a relation. Then the goal of DocRE changes to estimate a conditional probability $P(\boldsymbol{R}|\boldsymbol{D}, \boldsymbol{E})$, where $\boldsymbol{R} \in \mathbb{R} = [0, 1]^{|\boldsymbol{E}|\times|\boldsymbol{E}|\times C}$ is a 3D-matrix, and C is the number of relation categories. In practice, the goal is to find the most possible relation matrix.

$$\boldsymbol{R}^* = \arg\max_{\boldsymbol{R}\in\mathbb{R}} P(\boldsymbol{R}|\boldsymbol{D}, \boldsymbol{E}). \quad (1)$$

To further model DocRE as a sequence generation problem, we introduce a variable $\boldsymbol{S} \in \mathbb{S}$ to represent a sequence. We will discuss the choice of how to represent this sequence space $\mathbb{S}$ shortly.

$$P(\boldsymbol{R}|\boldsymbol{D}, \boldsymbol{E}) = \frac{\sum_{\boldsymbol{S}\in\mathbb{S}} P(\boldsymbol{R}, \boldsymbol{S}, \boldsymbol{D}, \boldsymbol{E})}{P(\boldsymbol{D}, \boldsymbol{E})}, \quad (2)$$

$$= \sum_{\boldsymbol{S}\in\mathbb{S}} P(\boldsymbol{R}|\boldsymbol{S}, \boldsymbol{D}, \boldsymbol{E}) P(\boldsymbol{S}|\boldsymbol{D}, \boldsymbol{E}). \quad (3)$$

Clearly, this computation is intractable, unless it is costly to enumerate the sequence space.

3.2 Symbolic and Ordered Sequence Representation of Relation Matrix

In our context, all we need to do is to represent the relation matrix and linearize it as a sequence.

To represent the relation matrix, we assign each entity and relation a special symbol, or, id, at first. In a real scenario, an entity can occur multiple times in the document by mentions, and expressions may be a little different in natural language, such as aliases, abbreviations or acronyms. A special id assures a unique and unambiguous entity. Besides, there is no need to use separators to distinguish entities and relations which contain more than one tokens. As shown in Figure 1, we use different ranges of "<i>" ("<extra_id_i>" in implementation) to represent entities ($i \in [1, 100]$) and relations ($i \in [101, 200]$). Entities are arranged according to their first appearance in the document. The embeddings of entity ids are initialized with those of corresponding sequential numbers. For example, we use the embedding of "1" to initialize the embedding of "<1>". Similarly, the embeddings of

relation ids are initialized with the meaning pooling of the embeddings of the corresponding natural languages. The initialization benefits the pre-trained generative models to learn the meaning of the special tokens, as shown in Appendix-A.1. In this way, a relation tuple of *(Julian Reinard, 5 March 1983, date of birth)* can be represented as "<1> <2> <101>" in our paradigm.

For linearization of the relation matrix, we simply organize relation tuples in the row-column order. The result is that the relation instances whose head entity appears earlier in the document proceed those appear later in the output sequence, and the order of relations sharing the same head entity is decided by their tail entities. The optimal order that complies with the logical reasoning is hard to define in advance, unless the model bears heavy computation to enumerate the sequence space. On the contrary, the row-column order is deterministic and easy for the model to understand.

More formally, let $\hat{\boldsymbol{S}}$ be the corresponding sequence of the relation matrix $\boldsymbol{R}$, i.e., $\sum_{\boldsymbol{S}\in\mathbb{S}} P(\boldsymbol{R},\boldsymbol{S}) = P(\boldsymbol{R},\hat{\boldsymbol{S}})$, and $P(\boldsymbol{R}|\hat{\boldsymbol{S}},\boldsymbol{D},\boldsymbol{E}) = P(\boldsymbol{R}|\hat{\boldsymbol{S}}) = 1$. Let $\tau(\cdot)$ be the linearization function that converts a relation matrix to a sequence following the symbolic format and row-column order we described above, i.e., $\hat{\boldsymbol{S}} = \tau(\boldsymbol{R})$. We have:

$$
\begin{aligned}
& P(\boldsymbol{R}|\boldsymbol{D},\boldsymbol{E}) && (4)\\
=& \sum_{\boldsymbol{S}\in\mathbb{S}} P(\boldsymbol{R}|\boldsymbol{S},\boldsymbol{D},\boldsymbol{E})P(\boldsymbol{S}|\boldsymbol{D},\boldsymbol{E}), && (5)\\
=& P(\boldsymbol{R}|\hat{\boldsymbol{S}},\boldsymbol{D},\boldsymbol{E})P(\hat{\boldsymbol{S}}|\boldsymbol{D},\boldsymbol{E}), && (6)\\
=& P(\hat{\boldsymbol{S}}|\boldsymbol{D},\boldsymbol{E}). && (7)
\end{aligned}
$$

4-ary Relation Extraction The symbolic relation matrix can be easily extended to 4-ary relation extraction and the setting where entity type information is provided. For instance, a 4-ary relation instance $(e_{\text{Task}}, e_{\text{Method}}, e_{\text{Material}}, e_{\text{Metric}}, r)$ in SciREX is composed of four types of entities and a binary relation. Each type of entities can be further divided into different ranges of "<i>", and a relation tuple can be transformed to a similar sequence like "<1> <26> <51> <76> <101>".

Constrained Decoding Considering that the reference sequence is completely a series of triples for binary relation extraction or 5-ary tuples for 4-ary relation extraction, we utilize a relatively simple constrained decoding method to control generation compared to lexical generation paradigm. The vocabulary is confined to a certain range of special tokens barely depending on the current step, so that the decoder's vocabulary size is small. On the contrary, lexical generation methods requires a full vocabulary because they need to predict entities' text forms.

3.3 Parallel Row Generation

Due to the autoregressive nature of generative models, longer the output, slower the decoding process. Besides, when the output is too long, the memory is not supported. Fortunately, our method can easily accommodate to such situation. Instead of generating the whole relation matrix in one pass, we can choose to generate one row of the relation matrix each time. For example, in Figure 1, the model first only generates relation triples started with "*Julian Reinard*" which is denotes as "<1>" in the output sequence, and then restarts to generate other rows of the relation matrix in turn in the same way. Since the input is the same, the procedure can be parallel, thus saving time and generated length. All we need is different decoder start tokens. The parallel row generation sacrifices some relation dependencies, but saves length. And experiments in Sec-4.3.1 show that the trade-off is positive.

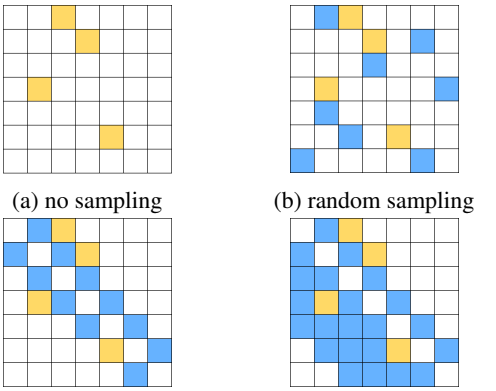
3.4 Loss Function Design

The relation matrix is typically sparse for DocRE task. For example, there are only approximately 3% entity pairs having relations on DocRED. Some previous work has found that sampling negative training examples, that is, entity pairs having no relation, during training is effective to improve the model performance. We propose several negative sampling strategies for our method and explain the reason in terms of loss function.

Our training target is the ground truth sequence $\boldsymbol{S}^* = (i, j, \boldsymbol{R}_{ij})_{i,j\in\boldsymbol{R}^+}$. Here, $\boldsymbol{R}^+$ is a set of all nonzero elements in the relation matrix, i is the row index and j is the column index according to the row-column order. Similarly, we denote $\boldsymbol{R}^-$ as the set of all zero elements. This produces the following generative loss function:

$$
\begin{aligned}
\mathcal{L}_{\text{seq}} = & \sum_{t=1}^{T-1} \text{CE}(\boldsymbol{S}^*{}_t, P(\boldsymbol{S}_t|\boldsymbol{S}_{<t},\boldsymbol{D},\boldsymbol{E})) \\
& + \text{CE}(\text{<EOS>}, \boldsymbol{S}_T),
\end{aligned} \tag{8}
$$

where T is the sequence length, and the last token is "<EOS>", which means the end of sequence.



(a) no sampling (b) random sampling

(c) diagonal sampling (d) asymmetric sampling

Figure 2: Different strategies of negative sampling. Light yellow elements are annotated relations, and blue elements are negative samples. There is a flipped version of (d) used in the dynamic sampling, which keeps one element in the left side of the diagonal and multiple elements in the right side.

However, the loss function to generate the relation matrix is applying cross entropy (Cox, 1958) to each element in the relation matrix:

$$\mathcal{L} = \sum_{i,j} \text{CE}(\boldsymbol{R}_{ij}, P(\boldsymbol{R}_{ij}|\boldsymbol{D}, \boldsymbol{E})), \tag{9}$$

$$\begin{aligned} &= \sum_{i,j \in \boldsymbol{R}^+} \text{CE}(\boldsymbol{R}_{ij}, P(\boldsymbol{R}_{ij}|\boldsymbol{D}, \boldsymbol{E})) \\ &+ \sum_{i,j \in \boldsymbol{R}^-} \text{CE}(\boldsymbol{R}_{ij}, P(\boldsymbol{R}_{ij}|\boldsymbol{D}, \boldsymbol{E})). \end{aligned} \tag{10}$$

By comparison, the sequence generation loss which we use in practice purges all the zero elements from the $\boldsymbol{R}^-$ set and lumps all of their effect into predicting the end of sequence. While it certainly shortens the target sequence, it also leads to severe imbalance loss terms. On the other hand, having the generative model produces all of zero elements defeats the purpose of symbolic format and can easily exceed the maximum length supported by most pre-trained seq2seq models. What is needed here is to balance the population of negative samples without overwhelming the generative model.

Negative Sampling Strategies Straightforwardly, we can add random zero elements in the relation matrix as negative samples, as shown in Figure 2b. For example, we randomly pick 10% zero elements and add them to the target sequence. However, this raises the difficulty of sequence prediction since the model might struggle to remember the order for each training sample, ignoring the contextual information. A more effective way is to regularly preserve elements in the diagonal band with a constant (and therefore balanced) budget of the relation matrix, namely, *diagonal negative sampling*. Further to alleviate the bias brought by the limited nonzero element space, we introduce a dynamic strategy to provide negative samples by randomly picking from: no sampling (Figure 2a), diagonal sampling (Figure 2c), and asymmetric sampling (Figure 2d). We call it *dynamic negative sampling*. In evaluation, we remove the zero elements the model generates.

4 Experiments

4.1 Datasets and Evaluation Metrics

Dataset	# Train	# Dev	# Test
DocRED	3053	1000	1000
CDR	500	500	500
GDA	23353	5839	1000
SciREX	306	66	66

Table 1: Statistics of the datasets in experiments.

We evaluate our model on four commonly used DocRE datasets. **DocRED** (Yao et al., 2019) is a human-annotated DocRE dataset with 96 relation types between two entities. Articles and their relation sets are mined from Wikipedia. **CDR** (Li et al., 2016) is a manually annotated dataset for DocRE in the biochemical domain. The aim is to predict whether there is a chemical-induced disease (CID) relation between Chemical and Disease. **GDA** (Wu et al., 2019) is also a biochemical dataset annotated with binary interactions between Gene and Disease concepts at the document-level via distant supervision. **SciREX** (Jain et al., 2020) is a document-level information extraction dataset, including binary and 4-ary relation extraction from scientific articles. It contains four types of entities, Task, Method, Material, and Metric, and coreference is annotated. The dataset statistics are listed in Table 1.

For DocRED, we use F_1 and Ign F_1 in evaluation following Yao et al. (2019), where Ign F_1 denotes F_1 removing triples having appeared in both the training and development/testing set. For other datasets, we use F_1 in evaluation.

4.2 Implementation details

We implement our model in PyTorch, and use pre-trained generative models provided by hugging-

Model	Dev		Test	
	Ign F_1	F_1	Ign F_1	F_1
classification-based				
$\text{BERT}_{\text{base}}$ (Wang et al., 2019)	-	54.16	-	53.20
T5_{large}	56.20	57.99	55.44	57.36
$\text{RoBERTa}_{\text{large}}$ (Ye et al., 2020)	57.19	59.40	57.74	60.06
strong $\text{RoBERTa}_{\text{large}}$ (Xu et al., 2021a)	58.45	60.58	58.43	60.54
$\text{SAIS}^{\text{B}}_{\text{All}}$-$\text{RoBERTa}_{\text{large}}$ (Xiao et al., 2021)	62.23 ± 0.15	65.17 ± 0.08	63.44	65.11
NCRL+ATLOP-$\text{DeBERTa}_{\text{large}}$ + distant † (Zhou and Lee, 2022)	66.11 ± 0.14	67.92 ± 0.14	65.81	67.53
generation-based				
lexical generation	48.43	50.34	49.32	51.36
DORE	52.79	55.12	52.53	55.10
DORE + negative $\text{sampling}_{\text{dynamic}}$	58.43	60.42	57.58	59.88
DORE + negative $\text{sampling}_{\text{all}}$ + parallel row generation	58.55 ± 0.11	60.61 ± 0.10	58.44	60.67
DORE + negative $\text{sampling}_{\text{diagonal}}$ + distant †	$\mathbf{62.91 \pm 0.13}$	$\mathbf{64.70 \pm 0.12}$	**63.26**	**65.26**

Table 2: Main results on DocRED. Results with † mean the models are pre-trained on the distantly supervised dataset provided by DocRED. Rows in gray denote the models are implemented by ourselves. The best results are underlined and the best results of the generation-based models are in bold.

face [1] as the backbone. For DocRED, we choose T5 (Raffel et al., 2020), which is pre-trained on a multi-task mixture of unsupervised and supervised tasks and shows power in a variety of NLP tasks. In fact, any pre-trained generative models can be used, and we show the experiments in Appendix-A.2. For the two biochemical datasets, we use BioBART (Yuan et al., 2022), which adapts BART (Lewis et al., 2020) to the biochemical domain and benefits the domain-specific sequence generation tasks. For SciREX, we choose Longformer-Encoder-Decoder (LED) (Beltagy et al., 2020) as the backbone, a Longformer variant for supporting long document seq2seq tasks, since documents in SciREX are much longer than 1024. The models are trained using the AdamW (Loshchilov and Hutter, 2019) optimizer with weight decay coefficient of 0.01, and a linearly decaying scheduler (Goyal et al., 2017). Other hyperparameters are listed in Appendix-A.3. All experiments are conducted with Tesla T4 GPUs.

4.3 Main Results

4.3.1 Comparison on DocRED

In Table 2, "T5_{large}" is a classification baseline that we replace the "Enhanced BERT Baseline" using a bilinear classifier on top of the pre-trained language model provided by Zhou et al. (2021) with T5_{large}'s encoder. All generation-based methods are implemented by ourselves using T5_{large}. "lexical generation" means that we directly adopt T5 under lexical generation paradigm. "DORE" refers to the symbolic and ordered sequence representation for relation matrix introduced in Sec-3.2. "+ negative sampling" and "+ parallel row generation" add negative sampling and parallel row generation, respectively. Finally, "distant" means that the model is first trained on the additional noisy distant training corpus (provided by DocRED) and then fine-tuned on the human-annotated training set.

Experiments using T5 demonstrate that our proposed framework is effective. Results of "DORE" show that adopting the symbolic and ordered sequence format improves the test F_1 score against the lexical generation by 3.74 points. Adding dynamic negative sampling further brings 4.78 points improvement, which intuitively verifies the imbalance training signal harms the classification task and mitigating it brings significant benefit. When using all nonzero elements in the relation matrix and utilizing parallel row generation to support it with limited resources, there is still a minor improvement. It proves that the proposed parallel row generation method is practical, and considering more negative samples is valuable with less data. Pre-training on the distant corpus, we adopt diagonal negative sampling to reduce the computational cost, and can achieve 65.26 test F_1 on DocRED [2].

Besides generation-based methods, we also compare DORE with classification-based methods. The most effective setting "DORE + negative $\text{sampling}_{\text{all}}$ + parallel row generation", abbreviated as "$\text{DORE}_{\text{NS + PRG}}$", improves "$\text{T5}_{\text{large}}$" by 3.31 points, demonstrating that DORE can beat the classification-based method with the same feature

[1] https://huggingface.co/

[2] Experiment results show that adopting diagonal negative sampling is enough when the model is pre-trained on the large-scale distantly supervised dataset.

extraction flow (using T5) since the only difference between these two methods is how they compute the relation matrix. Also, "DORE$_{\text{NS + PRG}}$" outperforms BERT$_{\text{base}}$ and RoBERTa$_{\text{large}}$ baselines without changing the model architecture or using extra training data. Admittedly, our proposed method still has a performance gap with SOTA methods on DocRED, which employ a series of advanced techniques. For example, **SAIS$^{\text{B}}_{\text{All}}$-RoBERTa$_{\text{large}}$** designed complicated pipeline multi-task learning and data augmentation, and **NCRL+ATLOP-DeBERTa$_{\text{large}}$ + distant** utilized DeBERTa$_{\text{large}}$ which is proved to be more powerful than RoBERTa$_{\text{large}}$ on DocRED (Zhou and Lee, 2022). In contrast, our generative framework is more concise and potential.

4.3.2 Comparison on CDR and GDA

Model	CDR	GDA
classification-based		
EoG (Christopoulou et al., 2019)	63.6	81.5
BioBART$_{\text{base}}$	64.1	81.6
SciBERT (Zhou et al., 2021)	65.1	82.5
BioBART$_{\text{large}}$	67.3	82.3
SSAN-SciBERT (Xu et al., 2021a)	68.7	83.7
ATLOP-SciBERT (Zhou et al., 2021)	69.4	83.9
DocuNet-SciBERT (Zhang et al., 2021a)	76.3	85.3
SAIS$^{\text{O}}_{\text{RE+CR+ET}}$-SciBERT (Xiao et al., 2021)	79.0	87.1
generation-based		
seq2rel (Giorgi et al., 2022)	67.2	84.9
DORE-BioBART$_{\text{base}}$	69.0	84.7
DORE-BioBART$_{\text{large}}$	**72.6**	**85.3**

Table 3: Test F_1 scores on CDR and GDA. Row in gray denote the models are implemented by ourselves.

For a fair comparison, we leverage entity type information when evaluating on the two biochemical datasets. "BioBART$_{\text{base}}$" and "BioBART$_{\text{large}}$" are classification baselines by replacing "SciBERT" implemented by Zhou et al. (2021) with corresponding BioBART's encoder. **Seq2rel** is a lexical generation method and employs copy mechanism and entity hinting to control generation.

As shown in Table 3, our method improves the performance of the previous generative DocRED method on CDR using BioBART$_{\text{base}}$ and BioBART$_{\text{large}}$, and further bridges the gap between classification-based and generation-based methods. It illustrates the advantage of symbolic sequence representation. DORE-BioBART$_{\text{base}}$ performs slightly worse than seq2rel on GDA, and we owe it to the weakness of BioBART on this dataset, given the comparison between classification-based methods using SciBERT and BioBART. Besides, DORE outperforms corresponding classification-based methods by 4.9/3.1 and 5.3/3.0 points on CDR/GDA using BioBART$_{\text{base}}$ and BioBART$_{\text{large}}$, respectively, which verifies the strength of our generative framework.

4.3.3 Comparison on SciREX

Model	Binary RE P	Binary RE R	Binary RE F_1	4-ary RE P	4-ary RE R	4-ary RE F_1
Component-wise (gold input)						
SciREX-P	82.0	44.0	57.0	53.1	71.8	61.1
DORE-LED$_{\text{base}}$	88.7	77.8	**82.9**	79.5	55.5	**65.4**
End-to-end						
TANL-BART$_{\text{base}}$	0.74	0.67	0.62	0.00	0.00	0.00
DYGIE++	2.9	12.8	3.8	-	-	-
SciREX-P	6.5	44.1	9.6	0.7	17.3	0.8
TempGen-BART$_{\text{base}}$	17.11	13.56	14.47	3.19	4.26	3.55
TempGen-LED$_{\text{base}}$ *	18.75	14.48	15.59	0.00	0.00	0.00
DORE-LED$_{\text{base}}$	30.45	23.93	**26.80**	9.52	5.41	**6.90**

Table 4: Main results on SciREX. Results with $*$ denote the models are implemented by ourselves.

In Table 4, **DYGIE++** (Wadden et al., 2019) and **SciREX-P** (Jain et al., 2020) are classification-based methods, while **TANL** (Paolini et al., 2021) and **TempGen-BART$_{\text{base}}$** (Huang et al., 2021) are lexical generation-based methods in general. We replace BART$_{\text{base}}$ with LED$_{\text{base}}$ for TempGen, namely, TempGen-LED$_{\text{base}}$. There is a slight improvement using TempGen-LED$_{\text{base}}$ on the binary relation extraction, mainly because encoding longer documents (4096 vs. 1024) provides more useful contextual information and relational facts. However, it cannot extract valid entities for 4-ary relation extraction. We attribute the bad performance to lexical generation paradigm making the model confused to represent entities.

To fairly compare when evaluating using gold inputs, we add entity type information. To compare with end-to-end relation extraction methods, we adopt *fast-coref* [3] (Toshniwal et al., 2020) to resolve coreference resolution using Longformer$_{\text{base}}$, which achieves 34.5 F_1 score while DYGIE++ 47.6 and SciREX-P 25.5. Experiments show that our proposed method achieves new SOTA results on both binary and 4-ary relation extraction tasks in both settings, which demonstrates the effectiveness and generalization of DORE.

4.4 Ablation Study

Symbolic vs. Lexical In this section, we compare the lexical representation and our symbolic

[3] https://github.com/shtoshni/fast-coref

Method	Ign F_1	F_1
lexical	48.43	50.34
symbolic	**52.02**	**54.10**
random order	51.40	53.39
annotation order	52.02	54.10
row-column order	**52.79**	**55.12**
10% random	52.89	55.45
diagonal	56.75	58.81
dynamic	**58.55**	**60.61**

Table 5: Ablation studies of symbolic representation, sequence order, and negative sampling. All results are from DocRED dev set. The best results in each block are in bold.

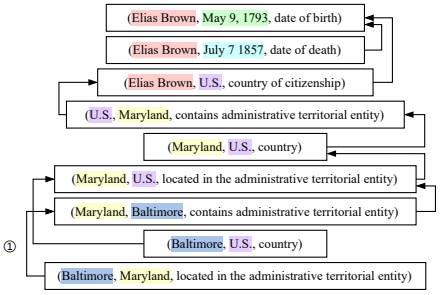


Figure 3: Case study of how DORE attends to previously generated triples from the document "Elias Brown (May 9,1793 – July 7, 1857) was a U.S. Representative from Maryland. Born near Baltimore, Maryland, Brown attended the common schools . . . "

representation. We choose $T5_{large}$ as the testbed, which can learn the symbolic representation without constrained decoding.

The upper part of Table 5 shows that the symbolic representation betters the performance by 3.76 points, which is a substantial improvement. We believe the improvement comes from two aspects: the symbolic representation largely reduces the sequence length, alleviating the accumulation decoding error; and it simplifies the copy mechanism since one symbol represents a long text phrase.

Sequence Order The sequence order plays an essential role in DORE. To understand its effect, we compare the annotation order, i.e., the order of how annotators annotate a document, and the row-column order, against a reference baseline using random order, where each sample is associated with a random sequence to be generated. Experiments are conducted with $T5_{large}$ and symbolic formatted sequences.

The middle part of Table 5 gives a comparison between three orders. The annotation order does outperform the random order since it is more predictable. However, we can not assume that annotators' behaviors are consistent. As we expected, the row-column order, which is not only stable but also deterministic, further outperforms the annotation order by 1.02 points. Still, we do not believe it is necessarily the best order. In general, a better order should reflect high-order dependencies' topology, and we leave this as a future direction.

Negative Sampling We also test different negative sampling strategies introduced in Sec-3.4. There are three settings. "10% random" uniformly picks 10% negative samples from the relation matrix. "diagonal" means we select negative samples with window size of 1 around the diagonal. And "dynamic" uniformly selects the different strategies we introduced before. We test these settings with $T5_{large}$ plus symbolic representation and row-column order. We found the "10% random" option contributes little, the "diagonal" outperforms it since it is consistent across entities. Finally, the "dynamic" option performs the best because it provides the model a chance to see all negative samples in different passes.

4.5 High-order Dependencies

To verify whether the proposed model captures high-order dependencies, we provide a case study in Figure 3 by probing into decoder attention scores. For each triple, we draw an edge to the generated triples in previous steps that receive highest attention. And we explain how to compute this score in Appendix-A.4.

The decoder would always attend to the last generated triples if it could not recover output dependencies. In contrast, we can find that the decoder of DORE tends to attend to the previous triples with the same head or tail entity according to Figure 3, which is more likely to bed latent associations. As the example ① shows, the latter triple accurately predicts the symmetric relation based on the former one. Conventional classification-based methods can not do this without additional modules.

5 Conclusion

We propose a new generative paradigm DORE for DocRE. DORE adopts a symbolic and ordered sequence representation, establishing a clean connection between the sequence generation and DocRE. We also introduce parallel row generation and several negative sampling methods to improve the effectiveness and efficiency. Experiments on four

DocRE datasets demonstrate that our method can substantially improve generative models without changing their designs.

Limitations

As shown in Table 2 and Table 3, although our proposed method without extra modules has outperformed classification baselines which have a simple classifier on top of pre-trained language models, there still exists a performance gap on the relatively complicated DocRED and domain-specific CDR and GDA. For one thing, we assume that some techniques proposed by SOTA work are complementary for DORE, and experiments are needed to verify whether DORE faces same issues. For another, experiments show that encoders of pre-trained generative models including T5 and BioBART are weaker to extract features compared to popular non-generative models like RoBERTa and DeBERTa, which impedes the model performance to some extent. Therefore, replacing the generative model's encoder with pre-trained language models used for classification for a fairer comparison leaves for future work.

Ethics Statement

Our work complies with the ACL Ethics Policy. As document-level relation extraction is a standard task in NLP, and all datasets we used are public, we do not see any critical ethical considerations.

Acknowledgement

We would like to express gratitude to the anonymous reviewers for their kind and insightful comments. This work was supported by the National Key Research and Development Program of China (No.2020AAA0108700) and National Natural Science Foundation of China (No.62022027).

References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *CoRR*, abs/2004.05150.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 4924–4935. Association for Computational Linguistics.

David R Cox. 1958. The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2):215–232.

Leyang Cui, Yu Wu, Jian Liu, Sen Yang, and Yue Zhang. 2021. Template-based named entity recognition using BART. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 1835–1845. Association for Computational Linguistics.

John M. Giorgi, Gary D. Bader, and Bo Wang. 2022. A sequence-to-sequence approach for document-level relation extraction. In *Proceedings of the 21st Workshop on Biomedical Language Processing, BioNLP@ACL 2022, Dublin, Ireland, May 26, 2022*, pages 10–25. Association for Computational Linguistics.

Priya Goyal, Piotr Dollár, Ross B. Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, large minibatch SGD: training imagenet in 1 hour. *CoRR*, abs/1706.02677.

I-Hung Hsu, Kuan-Hao Huang, Elizabeth Boschee, Scott Miller, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2021. Event extraction as natural language generation. *CoRR*, abs/2108.12724.

Kevin Huang, Guangtao Wang, Tengyu Ma, and Jing Huang. 2020. Entity and evidence guided relation extraction for docred. *CoRR*, abs/2008.12283.

Kung-Hsiang Huang, Sam Tang, and Nanyun Peng. 2021. Document-level entity-based extraction as template generation. *CoRR*, abs/2109.04901.

Sarthak Jain, Madeleine van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy. 2020. Scirex: A challenge dataset for document-level information extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7506–7516. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL*, pages 7871–7880. Association for Computational Linguistics.

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

Sha Li, Heng Ji, and Jiawei Han. 2021. Document-level event argument extraction by conditional generation. In *Proceedings of the 2021 Conference of the North*

American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021, pages 894–908. Association for Computational Linguistics.

Jian Liu, Zhiyang Teng, Leyang Cui, Hanmeng Liu, and Yue Zhang. 2021. Solving aspect category sentiment analysis as a text generation task. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 4406–4416. Association for Computational Linguistics.

Xiao Liu, Heyan Huang, Ge Shi, and Bo Wang. 2022. Dynamic prefix-tuning for generative template-based event extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 5216–5228. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *ICLR (Poster)*. OpenReview.net.

Yaojie Lu, Hongyu Lin, Jin Xu, Xianpei Han, Jialong Tang, Annan Li, Le Sun, Meng Liao, and Shaoyi Chen. 2021. Text2event: Controllable sequence-to-structure generation for end-to-end event extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 2795–2806. Association for Computational Linguistics.

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. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 5755–5772. Association for Computational Linguistics.

Guoshun Nan, Zhijiang Guo, Ivan Sekulic, and Wei Lu. 2020. Reasoning with latent structure refinement for document-level relation extraction. In *ACL*, pages 1546–1557. Association for Computational Linguistics.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cícero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages. In *ICLR*. OpenReview.net.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Shubham Toshniwal, Sam Wiseman, Allyson Ettinger, Karen Livescu, and Kevin Gimpel. 2020. Learning to ignore: Long document coreference with bounded memory neural networks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 8519–8526. Association for Computational Linguistics.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 5783–5788. Association for Computational Linguistics.

Difeng Wang, Wei Hu, Ermei Cao, and Weijian Sun. 2020. Global-to-local neural networks for document-level relation extraction. In *EMNLP (1)*, pages 3711–3721. Association for Computational Linguistics.

Hong Wang, Christfried Focke, Rob Sylvester, Nilesh Mishra, and William Yang Wang. 2019. Fine-tune bert for docred with two-step process. *CoRR*, abs/1909.11898.

Ye Wu, Ruibang Luo, Henry C. M. Leung, Hing-Fung Ting, and Tak Wah Lam. 2019. RENET: A deep learning approach for extracting gene-disease associations from literature. In *Research in Computational Molecular Biology - 23rd Annual International Conference, RECOMB 2019, Washington, DC, USA, May 5-8, 2019, Proceedings*, volume 11467 of *Lecture Notes in Computer Science*, pages 272–284. Springer.

Yuxin Xiao, Zecheng Zhang, Yuning Mao, Carl Yang, and Jiawei Han. 2021. SAIS: supervising and augmenting intermediate steps for document-level relation extraction. *CoRR*, abs/2109.12093.

Yiqing Xie, Jiaming Shen, Sha Li, Yuning Mao, and Jiawei Han. 2021. Eider: Evidence-enhanced document-level relation extraction. *CoRR*, abs/2106.08657.

Benfeng Xu, Quan Wang, Yajuan Lyu, Yong Zhu, and Zhendong Mao. 2021a. Entity structure within and throughout: Modeling mention dependencies for document-level relation extraction. In *AAAI*, pages 14149–14157. AAAI Press.

Wang Xu, Kehai Chen, and Tiejun Zhao. 2021b. Discriminative reasoning for document-level relation extraction. In *ACL/IJCNLP (Findings)*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 1653–1663. Association for Computational Linguistics.

Hang Yan, Tao Gui, Junqi Dai, Qipeng Guo, Zheng Zhang, and Xipeng Qiu. 2021. A unified generative framework for various NER subtasks. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International*

Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021, pages 5808–5822. Association for Computational Linguistics.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. Docred: A large-scale document-level relation extraction dataset. In *ACL (1)*, pages 764–777. Association for Computational Linguistics.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Peng Li, Maosong Sun, and Zhiyuan Liu. 2020. Coreferential reasoning learning for language representation. In *EMNLP (1)*, pages 7170–7186. Association for Computational Linguistics.

Hongyi Yuan, Zheng Yuan, Ruyi Gan, Jiaxing Zhang, Yutao Xie, and Sheng Yu. 2022. Biobart: Pretraining and evaluation of A biomedical generative language model. In *Proceedings of the 21st Workshop on Biomedical Language Processing, BioNLP@ACL 2022, Dublin, Ireland, May 26, 2022*, pages 97–109. Association for Computational Linguistics.

Shuang Zeng, Yuting Wu, and Baobao Chang. 2021. SIRE: separate intra- and inter-sentential reasoning for document-level relation extraction. In *ACL/IJCNLP (Findings)*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 524–534. Association for Computational Linguistics.

Shuang Zeng, Runxin Xu, Baobao Chang, and Lei Li. 2020. Double graph based reasoning for document-level relation extraction. In *EMNLP (1)*, pages 1630–1640. Association for Computational Linguistics.

Ningyu Zhang, Xiang Chen, Xin Xie, Shumin Deng, Chuanqi Tan, Mosha Chen, Fei Huang, Luo Si, and Huajun Chen. 2021a. Document-level relation extraction as semantic segmentation. In *IJCAI*, pages 3999–4006. ijcai.org.

Shuai Zhang, Yongliang Shen, Zeqi Tan, Yiquan Wu, and Weiming Lu. 2022. De-bias for generative extraction in unified NER task. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 808–818. Association for Computational Linguistics.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2021b. Towards generative aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021*, pages 504–510. Association for Computational Linguistics.

Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. 2021. Document-level relation extraction with adaptive thresholding and localized context pooling. In *AAAI*, pages 14612–14620. AAAI Press.

Yang Zhou and Wee Sun Lee. 2022. None class ranking loss for document-level relation extraction. *CoRR*, abs/2205.00476.

A Appendix

A.1 Initialization of Entity ID

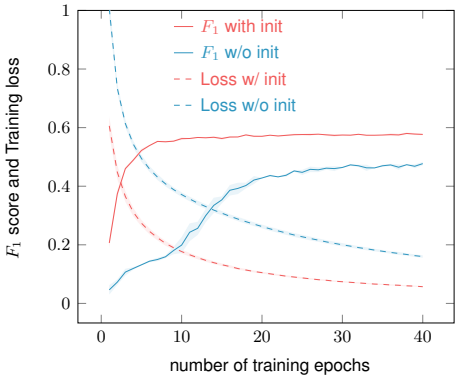


Figure 4: Comparison between random initialization and taking numbers for initialization on entity id tokens. Solid lines are F_1 scores on DocRED Dev set, and dashed lines are training loss on DocRED Train set. We omit the cross entropy loss higher than 1.0 for better visualization.

As we mentioned before, we initialize symbols that represent entity ids by numbers, since these symbols are not trained in the pre-training phase, so the model can not recognize their meaning. Alternatively, the model can learn from scratch during fine-tuning. However, we find the cold-start costs time and makes the training unstable. Note this strategy is very similar to position embedding used in standard Transformer-based models. We also try to initialize the relation embedding by averaging word embeddings of their lexical forms, whereas we find it does not influence the performance significantly.

Figure 4 shows that the initialized method converges fast and achieves a higher performance. The training loss of the first ten epochs illustrates a big gap between the cold-start and warm-start methods. That demonstrates the effectiveness of our warm start strategy.

A.2 Backbones

Backbone	Ign F_1	F_1
$T5_{large}$	56.94	58.95
$BART_{large}$	56.96	59.22
LED_{large}	57.04	59.10

Table 6: Results of DORE using different backbones on the development set of DocRED.

Hyperparameter	DocRED		CDR	GDA	SciREX
Backbone	T5	BART/LED	BioBART	BioBART	LED
Batch size	4		4	4	32
Training epochs	40		40	10	40
Learning rate	1e-4	3e-5	2e-5	2e-5	5e-5
Warmup ratio	0.06		0.1	0.15	0.06
Max input length	1024		1024	1024	4096
Beam size	4		1	1	1

Table 7: Hyperparameters used for each dataset.

In principle, the method we proposed can be adapted to any pre-trained generative language models. We verify the supposition by changing the backbone with the same symbolic and ordered sequence representation and diagonal negative sampling. From Table 6 we can see that T5, BART, or LED can achieve comparable results with our simple constrained decoding strategy, which proves the generalization ability of DORE.

A.3 Hyperparameters

In Table 7, we list the hyperparameters used when training the model for each dataset. When beam size = 1, we use greedy search decoding. When beam size = 4, we use beam search decoding, and tune the length penalty $\alpha = \{0.2, \ldots, 2.0\}$ with a step size of 0.2.

A.4 Visualization details

In detail, we use the attention scores from the last decoder layer of T5_{large}, and then we sum all attention heads. We conduct this visualization with our "DORE + negative sampling$_{\text{diagonal}}$ + distant" model, and the attention score of a triple is computed by adding up the attention scores of all its member tokens. Also, we do not consider the decoder start token "<BOS>".

In this way, we compute the attention score of previously generated triples for each time that the model predict the relation type. As a result, each triple will point to a triple before it as we shown in Figure 3.

A.5 Consistent Optimum

Theorem A.1 *Let* $\boldsymbol{S}^* = \tau(\boldsymbol{R}^*) = \arg\max_{\boldsymbol{S}\in\mathbb{S}} P(\boldsymbol{S}|\boldsymbol{D},\boldsymbol{E})$*, then we have* $\boldsymbol{R}^* = \arg\max_{\boldsymbol{R}\in\mathbb{R}} P(\boldsymbol{R}|\boldsymbol{D},\boldsymbol{E})$.

Proof Since $\boldsymbol{S}^* = \arg\max_{\boldsymbol{S}\in\mathbb{S}} P(\boldsymbol{S}|\boldsymbol{D},\boldsymbol{E})$, so for any $\boldsymbol{S} \in \mathbb{S}$, we have $P(\boldsymbol{S}^*|\boldsymbol{D},\boldsymbol{E}) - P(\boldsymbol{S}|\boldsymbol{D},\boldsymbol{E}) \geq 0$. According to the eq. (7), we can rewrite the formulation.

$$P(\boldsymbol{R}^*|\boldsymbol{D},\boldsymbol{E}) - P(\boldsymbol{R}|\boldsymbol{D},\boldsymbol{E}), \tag{11}$$

$$= P(\boldsymbol{S}^*|\boldsymbol{D},\boldsymbol{E}) - P(\tau(\boldsymbol{R})|\boldsymbol{D},\boldsymbol{E}), \tag{12}$$

$$\geq 0. \tag{13}$$