Discriminative LLM Fine-Tuning by Deactivating Causal Attention for Structure Prediction

Anonymous ACL submission

Abstract

Decoder-based LLMs possess versatile capabilities that enable them to adapt to various natural language tasks, such as summarization, question answering, and logical reasoning, using zero- or few-shot in-context learning methods. Nonetheless, the versatile abilities of decoder-based LLMs struggle to achieve high performance on structure prediction tasks, such as named entity recognition (NER), relation extraction (RE), and event extraction, due to the constraint of autoregressive token generation during both training and inference. This paper aims to address the deficiencies of LLMs in structure prediction tasks by focusing on the attention mechanism using a discriminative approach with supervised labels. Specifically, we demonstrate that this deficiency can be overcome by deactivating their autoregressiveness during both training and inference, similar to encoder LLMs like BERT. After the attention mechanism is intervened in, we confirm that fine-tuned models based on decoder LLMs perform slightly better for NER tasks and competitively for RE tasks, compared to 'state-of-the-art' encoder-based methods such as BERT.

1 Introduction

Decoder-based LLMs trained with a large unstructured dataset in multiple domains have versatile capabilities that allow them to adapt to various tasks such as summarization, translation, question answering, logical reasoning, etc by using zero or few-shot in-context learning methods. Even though the performance achievable for these tasks by zero or few-shot in-context learning may not be on par with the existing 'state-of-the-art' methods, it can be comparable or outperform them if it employs supervised instruction-following fine-tuning (SFT). Moreover, previous literature shows that augmented data, such as knowledge data, can be further leveraged for in-context learning without relying on fine-tuning (Li et al., 2025). Due to their generative nature, decoder-based LLMs attempt to solve tasks by generating outputs in response to given prompts. However, these characteristics burden structural prediction tasks that require detailed chunking and typing information over the input sentence.

To tackle these challenges, unlike previous generation-based approaches by LLMs, we employ a label-supervised discriminative approach and propose a supervised fine-tuning method for LLMs targeted for structural prediction tasks such as NER and relation extraction. The proposed method is the same as encoder-based models such as BERT (Devlin et al., 2019), but it is fine-tuned on decoder-based LLMs. The core idea of the proposed method is that it intervenes in the original self-attention mechanism, transitioning from causal attention to full attention during both training and inference.

Our results show that discriminatively fine-tuned LLMs, initially trained as decoders, outperform the instruction-following fine-tuned counterparts, and compete competitively with or better than strong encoder-based models, such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019).

The contributions of this paper are

- We propose a discriminative fine-tuning approach of LLMs to tackle the structure prediction tasks such as NER and RE, and confirmed that it performs better for NER tasks and competitively for RE tasks, compared to 'state-of-the-art' encoder-based methods such as BERT and RoBERTa.

- The proposed method is straightforward, requiring minimal modifications to the original code to transition from causal masking to full-attention masking during both training and inference.

- Our analysis suggests that the proposed approach is beneficial not only for structural pre-

diction tasks but also for other tasks such as similarity or paraphrase classification.

2 Related Work

Structure prediction, such as NER and relation extraction, has been tackled using numerous methods. The major approaches using deep neural networks can be categorized as follows: (1) supervised fine-tuning using encoder-based models such as BERT (Devlin et al., 2019) and its variants (Liu et al., 2019), (2) supervised fine-tuning using encoder-decoder models such as T5 (Liu et al., 2022; Paolini et al., 2021a), (3) unsupervised in-context learning using LLMs (Ding et al., 2021; Monajatipoor et al., 2024), and (4) supervised fine-tuning using LLMs (Xu et al., 2025; Wang et al., 2025).

Additionally, a data augmentation method utilizing external resources, such as a knowledge base, has been proposed to complement these methods.

Another line of effort focuses on the structure prediction task for longer contexts, such as multi-document relation extraction (Tan et al., 2022; Xue et al., 2024; Chen et al., 2020; Amalvy et al., 2023). Even though LLMs have achieved SOTA performances on a variety of NLP tasks, their performance on NER is still significantly below supervised baselines (Yano and Miwa, 2025; Wang et al., 2025). This is due to the gap between the two tasks, the NER and LLMs: the former is a sequence labeling task in nature, while the latter is a text-generation model. Although the authors of (Wang et al., 2025) claim that GPT-NER performs on par with the encoder model, their method has some limitations, as it can extract only one entity type at a time. The technique cannot detect various types of entities concurrently. Moreover, it is unclear how the proposed method can be extended beyond NER, such as relation and event extraction tasks.

This paper is closely related to (Li et al., 2023), where the authors employ a similar discriminative approach to solve text classification and NER tasks. This work conducts a more in-depth analysis, comparing the instruction-following fine-tuning approach with concrete examples, as well as an ablation test and analyses of training and runtime.

3 Method

We aim to determine whether pre-trained decoder-based LLMs can perform comparably with encoder-based pre-trained models, such as BERT, for structural prediction tasks like NER and relation extraction by adopting a discriminative approach rather than a generative one.

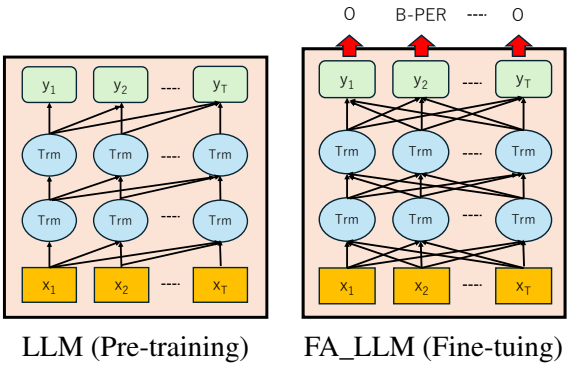


Figure 1: Left: Pretrained LLM with autoregressive causal attention, Right: Discriminatively fine-tuned LLMs for NER with full attention denoted as FA_LLM by deactivating causal attention during fine-tuning and inference. (x_i: input tokens, y_i: output token representations, Trm: transformer block)

We hypothesize that autoregressive causality attention used for decoder-based LLMs becomes a burden for structural prediction tasks. To verify our hypothesis, as shown in Fig. 1, our method intervenes in the mechanism of causal attention of pretrained LLMs to deactivate it during fine-tuning training and inference. The intervention is mathematically described as follows. We let $\{x_i | 1 \leq i \leq T\}$ be the input tokens. Then, the token represented at the top layer by the original pretrained LLM is defined as

$$y_t = LLM(x_t | x_{i<t}). \quad (1)$$

We define FA_LLM by replacing the causal attention of the original LLM with full attention for all self-attention layers. Hence, the output token representation y_t is conditioned on the whole input tokens and is represented as follows,

$$y_t = FA_LLM(x_t | x_{1 \leq i \leq T}). \quad (2)$$

We used named entity recognition (NER) and relation extraction (RE) tasks for evaluation, as we assumed that the results of NER and RE can be generalized to other structure prediction tasks, such as event extraction and coreference resolution.

We also added experimental conditions to prevent the use of auxiliary datasets that could boost LLM's performance. Hence, we train each model using only the corresponding training and validation datasets and evaluate its performance on the

corresponding testing dataset. The following sections describe the details of NER and RE adopting LLMs.

3.1 Named Entity Recognition

NER model is constructed by adding a linear layer to FA_LLM to transform the token output feature to the probability over named entity (NE) label k defined as $P_{NER}(k)$, namely implemented as

$$P_{NER}(k) = Softmax(Linear(y_t)), \quad (3)$$

where $Linear$ is a linear function defined as $Linear(y_t) = Wy_t + b$, and W and b are parameters. We use the cross entropy loss for training.

3.2 Relation Extraction

RE model is constructed by appending an additional linear layer to the previous NER model to determine the existence of any given relation label denoted as l between two named entities, the head and tail. Conceptually, the probability is defined as

$$P_{RE}(l) = Softmax(g(NE_{head}, NE_{tail})) \quad (4)$$

where g is a function to construct input features from the NE_{head} and NE_{tail}, which are the head and tail NEs detected by NER.

Although there are various ways to define the function g, we define it as follows.

$$\begin{aligned} g(NE_{head}, NE_{tail}) = Linear(\\ y_{t_{head}} \oplus emb(NE_{head}) \\ \oplus y_{t_{tail}} \oplus emb(NE_{tail}) \\ \oplus MaxPool(y_t | t_{head} < t < t_{tail})), \end{aligned} \quad (5)$$

where t_{head} and t_{tail} specify the first token positions of NE_{head} and NE_{tail}, respectively, and $y_{t_{head}}$ and $y_{t_{tail}}$ are the corresponding output token representations. $emb(NE_{head})$ and $emb(NE_{tail})$ are the embeddings of the NE label of NE_{head} and NE_{tail}, respectively. $MaxPool$ computes the context feature between t_{head} and t_{tail} by using the maximum pooling of the corresponding token features. Note that t_{tail} may be smaller than t_{head}. $\oplus$ represents the concatenation of features.

We added negative relation samples during training by randomly choosing distinct head and tail NEs and defining a negative class between them. We also add negative samples by exchanging the head and tail NEs for positive ones. The number of generated negative samples is 20 times that of positive samples.

In the NER task, the label "O" outside of any entity mention's tokens behaves as negative samples. In contrast, negative samples for the RE task are generated algorithmically to improve performance. The loss for RE is defined as the sum of the NER loss and the cross entory loss defined from $P_{RE}(l)$ and RE labels.

4 Experiments

We experimented using BC5CDR (Wei et al., 2016) and CoNLL03 (Tjong Kim Sang and De Meulder, 2003) for the NER task and CoNLL04 (Roth and Yih, 2004) for the NER and RE tasks. The statistics of each corpus is shown in Table 11.

We used BERT (bert-base-cased) (Devlin et al., 2019) and RoBERTa (roberta-base) (Liu et al., 2019) as the encoder-based baselines and evaluated Gemma-2B (Team et al., 2024), Llama-3.2-1B, and Llama-3.2-3B (Grattafiori et al., 2024), which are trained initially as decoder-based LLMs. The details of the models are shown in Table 12.

We evaluated the performance of the encoder-based models and the LLMs before and after the intervention of the attention mechanism for the NER and RE tasks. The pretrained model weights for each model are used to preload the parameter values of the experimental models, except for the weights of the token classifier defined for the NER task and the relation classifier for the RE task, which were initialized randomly.

To fine-tune the LLMs with full attention, we modified the source code by deactivating causal attention masking and explicitly forcing full attention for all layers during both training and inference. For the training parameters of all models, we set the learning rate to $1e-6$, device batch size to 8, the scheduler type to linear, warmup ratio to 0.1, and the weight decay to 0.001. We set the training epochs to 5 and 10 for NER and RE, respectively. We present details of the time required for fine-tuning the NER and RE models, as well as the runtime they took for testing, in Section 6.3.

We also evaluated the same tasks by fine-tuning the base LLMs using an instruction-following generative approach for comparison. In this setting, each LLM is fine-tuned with the corresponding training dataset in a generative manner. We use the same learning parameter as NER and ER tasks, except that the number of training epochs is set to 40 for both NER and RE tasks. The detailed instruction prompts we used for NER and RE tasks

BC5CDR			
	P	R	F1
INSTRUCTION FINE-TUNED DECODER			
Gemma-2B	0.761	0.625	0.687
Llama3.2-1B	0.788	0.712	0.748
Llama3.2-3B	0.821	0.779	0.799
BASELINE ENCODERS			
BERT	0.823	0.878	0.850
RoBERTa	0.854	0.885	0.864
CAUSAL ATTENTION			
Gemma-2B	0.741	0.754	0.748
Llama3.2-1B	0.687	0.689	0.688
Llama3.2-3B	0.635	0.624	0.630
FULL ATTENTION			
FA_Gemma-2B	0.858	0.894	0.876
FA_Llama3.2-1B	0.874	0.885	0.880
FA_Llama3.2-3B	0.867	0.874	0.870

Table 1: The results of NER for BC5CDR (P: Precision, R: Recall, F1: F1 score)

CoNLL03			
	P	R	F1
INSTRUCTION FINE-TUNED DECODER			
Gemma-2B	0.886	0.731	0.801
Llama3.2-1B	0.903	0.767	0.829
Llama3.2-3B	0.912	0.804	0.854
BASELINE ENCODERS			
BERT	0.904	0.920	0.912
RoBERTa	0.912	0.929	0.920
CAUSAL ATTENTION			
Gemma-2B	0.752	0.805	0.778
Llama3.2-1B	0.677	0.734	0.704
Llama3.2-3B	0.686	0.742	0.713
FULL ATTENTION			
FA_Gemma-2B	0.911	0.924	0.918
FA_Llama3.2-1B	0.910	0.919	0.915
FA_Llama3.2-3B	0.923	0.932	0.927

Table 2: The results of NER for CoNLL03 (P: Precision, R: Recall, F1: F1 score)

CoNLL04-NER			
	P	R	F1
INSTRUCTION FINE-TUNED DECODER			
Gemma-2B	0.810	0.764	0.786
Llama3.2-1B	0.852	0.820	0.836
Llama3.2-3B	0.865	0.832	0.848
BASELINE ENCODERS			
BERT	0.868	0.903	0.885
RoBERTa	0.865	0.893	0.879
CAUSAL ATTENTION			
Gemma-2B	0.766	0.817	0.791
Llama3.2-1B	0.707	0.775	0.739
Llama3.2-3B	0.723	0.792	0.756
FULL ATTENTION			
FA_Gemma-2B	0.858	0.883	0.870
FA_Llama3.2-1B	0.855	0.882	0.868
FA_Llama3.2-3B	0.872	0.885	0.879

Table 3: The results of NER for CoNLL04 (P: Precision, R: Recall, F1: F1 score)

are explained in Appendix C.

5 Results

5.1 Named Entity Recognition

Tables 1 and 2 show the results of NER for BC5CDR and CoNLL03 corpora. In both tables, the rows starting with INSTRUCTION-TUNED DECODER show the results of the generative approach. These results are followed by the results using discriminative approaches. The discriminative LLMs with causal attention, denoted by CAUSAL ATTENTION, Gemma-2B, Llama-3.2-1B, and Llama-3.23-B, lag far behind the encoder-based BERT and RoBERTa.

Moreover, the discriminative LLMs with causal attention mostly underperform compared to the corresponding results using a generative approach, except in the case of BC5CDR by Gemma-2B.

However, the results of a discriminative approach using full attention, denoted as FULL ATTENTION, outperform the generative LLMs and perform equally or better than BERT and RoBERTa. On average, discriminative LLMs with full attention improve the F1 score by 26 percentage points from corresponding LLMs with causal attention.

5.2 Relation Extraction

We first show the results of NER for CoNLL04 in Table 3.

For the NER results, we observe similar trends to the results of BC5CDR. The results of generative approaches outperform the discriminative LLMs with causal attention, except for Gemma-2B. However, the discriminative LLMs with full attention outperform the instruction-following decoder models by a large margin and perform equally well as encoder-based models. However, we do not admit a significant advantage of the discriminative LLMs over the base encoder models, similar to the previous NER results. This is likely because it is trained with both NER and RE losses.

Tables 4 and 5 show the results of RE for CoNLL04. The performance in Table 4 represents the RE performance when using gold NER annotations; in contrast, Table 5 shows the results when using predicted NER annotations. Hence, the results of Table 4 are the upper bound of RE performances of the proposed methods and models.

Unlike the trend in NER results, we could not

confirm clear advantages of the generative approaches' results over the corresponding results of discriminative approaches with causal attention. We assume that the performance of RE using a generative approach is highly dependent on the employed instruction-following format, suggesting that optimizing the instruction format may improve the results. Nonetheless, the proposed discriminative approach with full attention outperforms both the generative approach and the discriminative approach with causal attention.

However, unlike NER results, the discriminative approach with full attention does not perform competitively with the base encoder models. We have yet to analyze this behavior, and leave it to the feature work.

6 Analysis

6.1 Ablation analysis

The results in Section 5 support our hypothesis that token features generated by full self-attention are crucial to boosting the performance of structural prediction. Here, we conduct an ablation study to verify the effect of applying full attention to only some transformer layers. Specifically, we conduct two ablation scenarios: we apply full attention to (1) the bottom three transformer layers of the corresponding LLM, denoted as the prefix of B3_FA, or (2) the top three layers of the corresponding LLM, denoted as the prefix of T3_FA.

Table 6 shows the ablation test results of BC5CDR for NER. Compared with Table 6, in the case of B3_FA, the F1 score decreased by 12.2, 11.0, and 11.4 percentage points for Gemma-2B, Llama3.2-1B, and Llama3.2-3B, respectively, resulting in an average 11.6 percentage point reduction in the F1 score. On the other hand, in the case of T3_FA, the F1 score decreased by 10.8, 11.5, and 10.8 percentage points for Gemma-2B, Llama3.2-1B, and Llama3.2-3B, respectively, resulting in an average 11.0 percentage point reduction in the F1 score.

Similarly, Table 7 shows the ablation results of CoNLL03 for NER. Compared with Table 7, in the case of B3_FA, the F1 score decreased by 13.0, 13.1, and 14.8 percentage points for Gemma-2B, Llama3.2-1B, and Llama3.2-3B, respectively, resulting in an average 13.6 point reduction in the F1 score. On the other hand, in the case of T3_FA, the F1 score decreased by 10.8, 12.6, and 13.6 percentage points for Gemma-2B, Llama3.2-1B, and Llama3.2-3B, respectively, resulting in an average 12.3 point reduction in the F1 score.

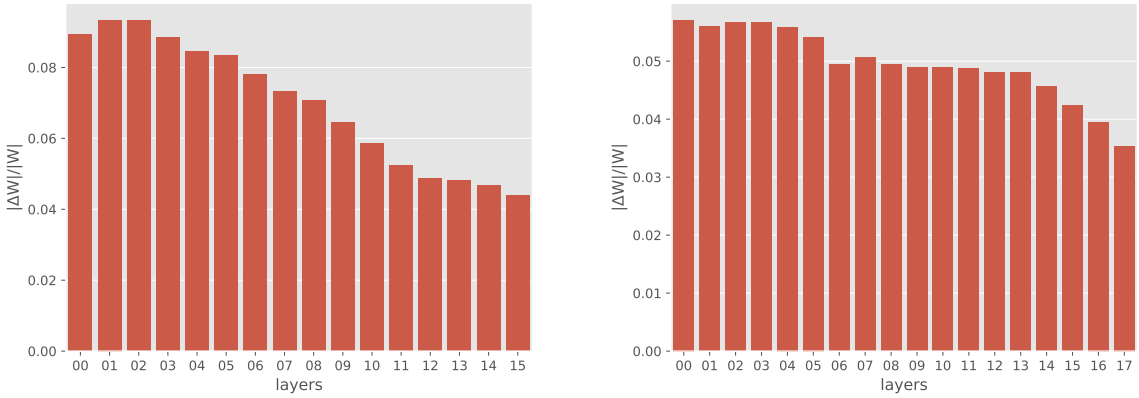
To conclude, the reduction of the F1 score from the corresponding FA_LLM is smaller for T3_FA_LLM than for B3_FA_LLM, suggesting that full attention on the top three layers is more effective than on the bottom three layers. Nonetheless, performance reduction is not ignorable in both cases, implying that the full attention mechanism must be used at all layers to achieve optimal performance.

	CoNLL4-Relations (True NER)		
	P	R	F1
INSTRUCTION FINE-TUNED DECODER			
Gemma-2B	0.495	0.765	0.601
Llama3.2-1B	0.530	0.810	0.641
Llama3.2-3B	0.763	0.850	0.804
BASELINE ENCODERS			
BERT	0.801	0.863	0.831
RoBERTa	0.805	0.889	0.845
CAUSAL ATTENTION			
Gemma-2B	0.681	0.684	0.683
Llama3.2-1B	0.697	0.721	0.709
Llama3.2-3B	0.706	0.706	0.706
FULL ATTENTION			
FA_Gemma-2B	0.794	0.806	0.800
FA_Llama3.2-1B	0.766	0.765	0.765
FA_Llama3.2-3B	0.800	0.807	0.803

Table 4: The results of RE for CoNLL04 when using true NER labels (P: Precision, R: Recall, F1: F1 score)

	CoNLL04-Relations (Pred NER)		
	P	R	F1
INSTRUCTION FINE-TUNED DECODER			
Gemma-2B	0.567	0.664	0.612
Llama3.2-1B	0.565	0.649	0.604
Llama3.2-3B	0.642	0.713	0.676
BASELINE ENCODERS			
BERT	0.765	0.850	0.805
RoBERTa	0.765	0.871	0.815
CAUSAL ATTENTION			
Gemma-2B	0.630	0.654	0.642
Llama3.2-1B	0.631	0.672	0.651
Llama3.2-3B	0.615	0.641	0.628
FULL ATTENTION			
FA_Gemma-2B	0.753	0.780	0.767
FA_Llama3.2-1B	0.740	0.754	0.747
FA_Llama3.2-3B	0.765	0.768	0.776

Table 5: The results of RE for CoNLL04 when using predicted NER labels (P: Precision, R: Recall, F1: F1 score)



(a) Llama3.2-1B

(b) Gemma-2B

Figure 2: Differential analysis of parameters of LLMs before and after intervention of attention mechanism

	BC5CDR		
	P	R	F1
FULL ATTENTION (bottom 3 layers)			
B3_FA_Gemma-2B	0.748	0.751	0.750
B3_FA_Llama3.2-1B	0.739	0.749	0.744
B3_FA_Llama3.2-3B	0.738	0.751	0.744
FULL ATTENTION (top 3 layers)			
T3_FA_Gemma-2B	0.762	0.766	0.764
T3_FA_Llama3.2-1B	0.736	0.741	0.739
T3_FA_Llama3.2-2B	0.763	0.739	0.751

Table 6: The ablation results of NER for BC5CDR (P: Precision, R: Recall, F1: F1 score)

	CoNLL03		
	P	R	F1
FULL ATTENTION (bottom 3 layers)			
B3_FA_Gemma-2B	0.757	0.805	0.781
B3_FA_Llama3.2-1B	0.730	0.775	0.752
B3_FA_Llama3.2-3B	0.726	0.754	0.740
FULL ATTENTION (top 3 layers)			
T3_FA_Gemma-2B	0.785	0.820	0.802
T3_FA_Llama3.2-1B	0.736	0.779	0.757
T3_FA_Llama3.2-3B	0.733	0.773	0.752

Table 7: The ablation results of NER for CoNLL03 (P: Precision, R: Recall, F1: F1 score)

6.2 Differential analysis

We conduct differential analysis by examining the fine-tuned original LLM with causal attention and the FA_LLM with full attention regarding the trained weights of layer-wise parameters.

Let W_l be the set of parameters of the original LLM defined for the l-th layer, and W'_l be the corresponding parameters of the FA_LLM defined for the l-th layer. Here, W_l includes projection parameters of q, k, and v defined for self-attention and *gate*, *up*, and *down* projection parameters defined for MLP, etc. We define the ratio of the difference of parameter weights between W_l and W'_l as follows:

$$\frac{|\Delta W_l|}{|W_l|} = \frac{|W_l - W'_l|}{|W_l|}, \quad (6)$$

where $|W_l|$ is the sum of the norms of each parameter defined in W_l.

Figure 2 shows the differential analysis of Llama3.2-1B (2a) and Gemma-2B (2b) before and after the intervention of the attention mechanism.

These results provide several insights. First, the intervention affects the parameters of all the layers, gradually increasing the strength toward the bottom layers. However, the pattern of the change in strength is not quite similar between Llama3.2-1B and Gemma-2B. The norm of the difference of the lawyer-wise parameters before and after the intervention, divided by the norm of the original parameter, ranges from 4 to 9%.

This suggests that the proposed intervention method needs to modify both the primitive feature representation at lower layers and the abstract feature representation for the targeted task at upper layers to optimize the structure prediction task by LLMs.

6.3 Training and runtime analysis

In this section, we present training time and running time measured on NVIDIA 1x H200 (8GPUs) with 141 GB of memory.

Table 8 shows the training time taken for fine-tuning models for the NER and RE tasks. The training time generally increases proportionally to the size of the models, except for the time taken for Gemmma-2B with the causal attention case, which is somewhat shorter than the time it took for Llama3.2-1B. Except for Gemma-2B, we do not observe a significant difference in time between

	NER		RE
	BC5CDR	CoNLL03	CoNLL04
BASELINE ENCODERS			
BERT	1156	1882	1042
RoBERTa	1133	1889	1597
CAUSAL ATTENTION			
Gemma-2B	1683	3039	8737
Llama3.2-1B	1429	2427	4322
Llama3.2-3B	1879	3289	10935
FULL ATTENTION			
FA_Gemma-2B	1693	3041	8766
FA_Llama3.2-1B	1441	2432	4186
FA_Llama3.2-3B	1875	3305	10997

Table 8: Training time for the NER and RE models in seconds

	NER		RE
	BC5CDR	CoNLL03	CoNLL04
INSTRUCTION FINE-TUNED DECODER			
Gemma-2B	152.33	104.67	110.33
Llama3.2-1B	120.67	91.00	68.67
Llama3.2-3B	184.33	117.00	263.33
BASELINE ENCODERS			
BERT	46.67	36.00	24.00
RoBERTa	47.00	36.33	24.33
CAUSAL ATTENTION			
Gemma-2B	99.33	70.67	37.33
Llama3.2-1B	110.00	57.33	73.00
Llama3.2-3B	122.67	85.67	42.67
FULL ATTENTION			
FA_Gemma-2B	98.00	70.67	30.00
FA_Llama3.2-1B	109.67	60.00	48.40
FA_Llama3.2-3B	122.67	84.00	35.40

Table 9: Average runtime of the NER and RE models for testing in seconds

causal attention and full attention, so one can conclude that full attention mechanisms do not significantly impact training time.

Table 9 shows the average running time in seconds measured on three trials by testing each model for the NER and RE tasks. Again, the running time generally increases proportionally to the size of the models. Similarly, one cannot observe a significant difference in runtime between causal and full attention.

These results suggest that enforcing the change from casual attention to full attention in LLMs will not have a significant impact on either training or running time.

6.4 Effects of LLMs fine-tuned on NER or RE on other NLP tasks

In this section, we conducted a comparative analysis of the effects of the LLMs, which are either instruction fine-tuned or discriminatively fine-tuned using the NER and RE tasks, on other NLP tasks.

For evaluation, we used HellaSwag (Zellers et al., 2019) and GLUE (Wang et al., 2018), which comprise a collection of evaluation tools for assessing the model across various NLU tasks (cola, mnli, mnli_mismatch, mrpc, qnli, qqp, rte, sst2, and wnli). All the scores are obtained by using lm-eval (Gao et al., 2024) using five few-shot settings.

Table 10 shows the comparative results between three baseline LLMs (Gemma-2B, LLama3.2-1B, and Llama3.2-3B), their instruction fine-tuned models denoted as FINE-TUNED BY INSTRUCTION-FOLLOWING, and discriminatively fine-tuned models denoted as DESCRIMINATIVELY TRAINED, all targeted for BC5CDR and CoNLL02 NER tasks, and CoNLL04 RE task. Rows labeled as '+fullattn' are the full-attention (FA) models described in the Results section.

For each benchmark, among the models derived from the same baseline, the best scores are shown in bold face. Note that all experiments are conducted using a causal attention mask.

From these benchmark performance results, several insights can be obtained. Firstly, fine-tuned models (either instruction-following or discriminatively trained) on each task of NE or RE do not always underperform the corresponding base pre-trained model. This is somewhat counterintuitive, as the models are fine-tuned on specific NER and RE tasks, rather than being specifically designed for these NLP benchmarks.

Secondly, we observe that instruction-following models tend to perform better than discriminatively trained models on specific tasks, such as HellaSwag, QNLI, RTE, and SST-2. These tasks are considered to measure the model's ability in inference, sentiment, and acceptability. We assume that these abilities are implicitly enhanced positively by instruction following training on NER or RE tasks.

Lastly, we also notice that discriminatively trained models, especially trained on the RE task, perform better than instruction-following models and the base pre-trained model on some benchmarks, such as MRPC and QQP. These benchmarks are designed to measure the models' ability to determine the similarity of two sentences or to distinguish between a paraphrase and its original. Obviously, these benchmarks require models to analyze the semantic similarity between two sentences, not only in the forward but also in the reverse direction of the input. A closer look at the results of

	HELLASWAG		COLA	MNLI	MNLI_MISMATCH	MRPC		QNLI	QQP		RTE	SST2	WNLI
	ACC	ACC_NORM	MCC	ACC	ACC	ACC	F1	ACC	ACC	F1	ACC	ACC	ACC
Gemma-2B	.529	.710	.103	.383	.391	.434	.449	.515	.615	.255	**.682**	.912	.479
FINE-TUNED BY INSTRUCTION-FOLLOWING													
BC5CDR	.533	.713	**.136**	.386	.386	.422	.395	.518	.624	.176	.657	.899	.437
CoNLL03	.530	.713	.107	.392	.396	.397	.335	.511	.626	.130	.661	.888	.451
CoNLL04	**.536**	**.715**	.101	.382	.382	.449	.455	.525	.623	.248	.639	**.914**	.451
DESCRINATIVELY TRAINED													
BC5CDR	.531	.711	.022	**.453**	**.467**	.583	.694	**.533**	.539	.436	.672	.884	.465
+fullattn	.438	.592	.000	.319	.318	.628	.759	.507	.488	.436	.599	.678	.521
CoNLL03	.519	.703	.049	.443	.444	.358	.211	.508	**.631**	.017	.603	.905	.437
+fullattn	.381	.526	.040	.422	.428	.365	.167	.507	.630	.020	.538	.540	.437
CoNLL04	.428	.562	.004	.313	.317	.346	.107	.509	.630	.016	.527	.569	.521
+fullattn	.402	.533	.046	.361	.373	**.684**	**.812**	.497	.397	**.523**	.570	.732	**.578**
Llama3.2-1B	.482	.650	.003	**.364**	**.375**	.601	.730	.512	.567	.358	.560	**.702**	.535
FINE-TUNED BY INSTRUCTION-FOLLOWING													
BC5CDR	.483	.648	.002	.356	.366	.574	.702	**.518**	.589	.310	**.578**	.671	.549
CoNLL03	.482	.647	.005	.358	.370	.630	.761	.516	.571	.350	.552	.697	.549
CoNLL04	**.485**	**.651**	.005	.342	.355	.608	.737	.520	.573	.358	.570	.663	**.563**
DESCRINATIVELY TRAINED													
BC5CDR	.272	.306	.006	.356	.350	.679	.808	.495	.369	.537	.531	.649	.423
+fullattn	.263	.278	**.020**	.353	.352	.677	.807	.492	.379	.520	.534	.509	.437
CoNLL03	.258	.265	.000	.355	.352	.316	.000	.505	**.632**	.000	.527	.509	.437
+fullattn	.254	.264	.000	.318	.318	.316	.000	.505	**.632**	.000	.527	.491	.437
CoNLL04	.275	.305	.000	.318	.318	**.684**	**.812**	.494	.368	**.538**	.527	.509	.451
+fullattn	.265	.288	.053	.347	.338	**.684**	**.812**	.495	.368	**.538**	.527	.516	.437
Llama3.2-3B	.562	.756	.231	.470	**.479**	.537	.609	.522	.647	.178	.672	**.903**	.535
FINE-TUNED BY INSTRUCTION-FOLLOWING													
BC5CDR	**.568**	.759	.274	.472	.478	.512	.585	**.529**	.647	.167	.679	.896	**.578**
CoNLL03	**.568**	**.760**	**.282**	**.476**	.477	.559	.646	.528	.652	.221	**.693**	.899	.535
CoNLL04	**.568**	**.760**	.227	.461	.467	.544	.627	.527	**.660**	.270	.671	.891	.549
DESCRINATIVELY TRAINED													
BC5CDR	.258	.265	.000	.355	.353	.316	.000	.506	.632	.000	.527	.508	.437
+fullattn	.258	.265	.000	.327	.330	**.684**	**.812**	.495	.368	**.538**	.473	.491	.563
CoNLL03	.261	.273	.048	.355	.352	.679	.808	.494	.382	.499	.527	.509	.437
+fullattn	.259	.265	.000	.328	.325	.316	.000	.505	.632	.000	.527	.491	.437
CoNLL04	.310	.390	.010	.329	.332	.316	.000	.504	.628	.028	.480	.538	.563
+fullattn	.304	.369	.008	.348	.346	.677	.806	.495	.371	.534	.530	.538	.423

Table 10: NLP benchmarks (HellaSwag (Zellers et al., 2019) and GLUE (Wang et al., 2018)) performance comparison between three base pretrained LLMs (Gemma-2B, Llama3.2-1B, and Llama3.2-3), fine-tuned by instruction-following models, and discriminatively trained models with causal or full attention, each targeted for BC5CDR and CoNLL03 NER tasks or CoNLL04 RE task. For each benchmark and the models derived from the same pre-trained LLM, the best scores are shown in boldface. The number in blue indicates a minus value.

MRPC and QQP reveals that full-attention models sometimes outperform the corresponding causal-attention models in the decrinative training setting, such as CoNLL04+fullatten of Gemma-2B and BC5CDR+fullattn of Llama3.2-3B.

This observation clearly indicates that instruction-following generative fine-tuning of LLMs is not an optimal solution for these tasks, including NER and RE tasks. It suggests that discriminative training with full attention is a promising method for those tasks.

7 Conclusion

This paper proposes an efficient discriminative fine-tuning method for LLMs for structured prediction tasks such as NER and RE instead of usual instruction-following fine-tuning approaches. The proposed method intervenes in the original attention mechanism to transition from causal attention to full attention during both training and inference times. Our results show that it performs significantly better than a naive discriminative training approach with a causal attention mechanism inherited from the base pre-trained model. Furthermore, we verified that the proposed model outperforms the instruction-following fine-tuning approaches.

Moreover, it outperforms and competes with state-of-the-art encoder-based methods, such as BERT and RoBERTa, for NER and RE tasks, respectively. Further analysis reveals that the proposed method will be effective not only for structure prediction tasks such as NER and RE but also for text similarity and paraphrase inference tasks. The inherent longer context size provided by the base LLMs provides an additional benefit to the proposed method compared to BERT-like models with a shorter context size of 512.

Limitations

We evaluated our proposed method using a limited benchmark dataset of NER and RE tasks. However, we assume that the results obtained in our experiments can similarly be applied to other structure prediction tasks.

We have tested only a small-sized LLM, with the intention that using a much larger LLM would be unfair for the targeted tasks, as we intend to compare it with a much smaller encoder-based model, such as BERT.

Employed generative instruction formats for solving NER and RE tasks were not optimally configured, so we assume that an optimally designed instruction format can increase the scores by this generative approach, especially for RE tasks. We did not show the results of the few-shot in-context learning of the instruction-following method because it underperforms the fine-tuned counterparts.

References

Arthur Amalvy, Vincent Labatut, and Richard Dufour. 2023. The role of global and local context in named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 714–722, Toronto, Canada. Association for Computational Linguistics.

Yubo Chen, Chuhan Wu, Tao Qi, Zhigang Yuan, and Yongfeng Huang. 2020. Named entity recognition in multi-level contexts. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 181–190, Suzhou, China. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ning Ding, Guangwei Xu, Yulin Chen, Xiaobin Wang, Xu Han, Pengjun Xie, Haitao Zheng, and Zhiyuan Liu. 2021. Few-NERD: A few-shot named entity recognition dataset. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3198–3213, Online. Association for Computational Linguistics.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2024. The language model evaluation harness.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Qi Li, Tingyu Xie, Jian Zhang, Ke Ma, Jiayuan Su, Kaixiang Yang, and Hongwei Wang. 2025. Enhancing named entity recognition with external knowledge from large language model. *Knowledge-Based Systems*, 318:113471.

Zongxi Li, Xianming Li, Yuzhang Liu, Haoran Xie, Jing Li, Fu lee Wang, Qing Li, and Xiaoqin Zhong. 2023. Label supervised llama finetuning. *Preprint*, arXiv:2310.01208.

Tianyu Liu, Yuchen Eleanor Jiang, Nicholas Monath, Ryan Cotterell, and Mrinmaya Sachan. 2022. Autoregressive structured prediction with language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 993–1005, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *Preprint*, arXiv:1907.11692.

Masoud Monajatipoor, Jiaxin Yang, Joel Stremmel, Melika Emami, Fazlolah Mohaghegh, Mozhdeh Rouhsedaghat, and Kai-Wei Chang. 2024. Llms in biomedicine: A study on clinical named entity recognition. *Preprint*, arXiv:2404.07376.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cícero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021a. Structured prediction as translation between augmented natural languages. *ArXiv*, abs/2101.05779.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cicero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021b. Structured prediction as translation between augmented natural languages. *Preprint*, arXiv:2101.05779.

Dan Roth and Wen-tau Yih. 2004. A Linear Programming Formulation for Global Inference in Natural

Language Tasks. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, pages 1–8, Boston, Massachusetts, USA. Association for Computational Linguistics.

Qingyu Tan, Lu Xu, Lidong Bing, Hwee Tou Ng, and Sharifah Mahani Aljunied. 2022. Revisiting DocRED - addressing the false negative problem in relation extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8472–8487, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, and 89 others. 2024. Gemma: Open models based on gemini research and technology. *Preprint*, arXiv:2403.08295.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, Guoyin Wang, and Chen Guo. 2025. GPT-NER: Named entity recognition via large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4257–4275, Albuquerque, New Mexico. Association for Computational Linguistics.

Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Jiao Li, Thomas C. Wiegers, and Zhiyong Lu. 2016. Assessing the state of the art in biomedical relation extraction: overview of the biocreative v chemical-disease relation (cdr) task. *Database: The Journal of Biological Databases and Curation*, 2016.

Weilu Xu, Renfei Dang, and Shujian Huang. 2025. LLM's weakness in NER doesn't stop it from enhancing a stronger SLM. In *Proceedings of the Second Workshop on Ancient Language Processing*, pages 170–175, The Albuquerque Convention Center, Laguna. Association for Computational Linguistics.

Lilong Xue, Dan Zhang, Yuxiao Dong, and Jie Tang. 2024. AutoRE: Document-level relation extraction with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 211–220, Bangkok, Thailand. Association for Computational Linguistics.

Ken Yano and Makoto Miwa. 2025. Effect of multilingual and domain-adapted continual pre-training on few-shot promptability. In *Proceedings of the 24th Workshop on Biomedical Language Processing*, pages 18–26, Viena, Austria. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

A Details of NER and RE corpora

Name	Train	Validation	Test	NE classes	Relation classes
BC5CDR	5,228	5,330	5,865	Chemical, Disease	NA
CoNLL03	14,041	3,250	3,453	NER,PER,ORG,LOC,MISC	NA
CoNLL04	922	231	288	Peop, Loc, Org, Other	Located_In, Work_for, OrgBased_In, Live_In, Kill

Table 11: The statistics for the number of sample for BC5CDR, CoNLL03, and CoNLL04. BC5CDR and CoNLL03 are used for the NER task, and CoNLL04 is used for the NER and RE tasks. The labels for NE and Relation classes are copied from the original corpora.

B Details of models used for experiments

	BERT	RoBERTa	Gemma-2B	Llama3.2-1B	Llama3.2-3B
model type	encoder	encoder	decoder	decoder	decoder
vocab size	28,996	50,265	256,000	128,256	128,256
#layers	12	12	18	16	28
hidden_dim	768	768	2,048	2,048	3,072
#heads	12	12	8	32	24
max_token_len	512	514	8,192	131,072	131,072
#params	108,310,272	124,058,885	2,506,182,661	1,235,824,645	3,212,765,189

Table 12: The details of base models used for experiments. BERT and RoBERTa are the baseline encoders, and Gemma-2B, Llama3.2-1B, and Llama3.2-3B are the LLMs targeted for intervention of the attention mechanism.

C Sample generative Instruction format of LLMs to solve NER and RE tasks

C.1 Name entity recognition (NER)

We adopt the TANL format (Paolini et al., 2021b) to solve NER by LLM. In this format, the LLM is expected to copy the source context to the target by annotating detected entity mentions, enclosing them in brackets, and suffixing them with the detected entity type. The descriptions of valid entity types and their labels are specified after the task description. Listings 1 and 2 illustrate a sample instruction format for the NER task of BC5CDR (Wei et al., 2016) and CoNLL03 (Tjong Kim Sang and De Meulder, 2003), respectively.

C.2 Relation Extraction (RE)

To solve the RE task, there are two approaches. One approach is first to use the NER method shown in Appendix C.1 to extract entities, and then extract relations using the NER results. The other is to solve NER and RE simultaneously.

Similar to NER, we adopt the TANL format (Paolini et al., 2021b) to solve RE by LLM. The descriptions of valid entity types and their labels, as well as relation types and their labels, are specified after the task description. Listing 3 illustrates a sample instruction format for relation extraction given the NER results, and listing 4 shows the instruction format to solve the NER and relation extraction simultaneously in the case of CoNLL04 (Roth and Yih, 2004).

```
{
  "conversations": [
    {
      "from": "system",
      "value": "You are an AI language model trained to extract entities from
      ↪  sentences. See all possible labels and their descriptions below.
      ↪  \n###\nDescription: Disease name\nLabel: Disease\n###\nDescription:
      ↪  Chemical name\nLabel: Chemical\n###\n"
    },
    {
      "from": "user",
      "value": "Galanthamine hydrobromide, a longer acting anticholinesterase
      ↪  drug, in the treatment of the central effects of scopolamine
      ↪  (Hyoscine).\nGalanthamine hydrobromide, an anticholinesterase drug
      ↪  capable of penetrating the blood-brain barrier, was used in a patient
      ↪  demonstrating central effects of scopolamine (hyoscine) overdosage.
      ↪  It is longer acting than physostigmine and is used in anaesthesia to
      ↪  reverse the non-depolarizing neuromuscular block. However, studies
      ↪  into the dose necessary to combating scopolamine intoxication are
      ↪  indicated."
    },
    {
      "from": "assistant",
      "value": "[Galanthamine hydrobromide|Chemical], a longer acting
      ↪  anticholinesterase drug, in the treatment of the central effects of
      ↪  [scopolamine|Chemical] ([Hyoscine|Chemical]).\n[Galanthamine
      ↪  hydrobromide|Chemical], an anticholinesterase drug capable of
      ↪  penetrating the blood-brain barrier, was used in a patient
      ↪  demonstrating central effects of [scopolamine|Chemical]
      ↪  ([hyoscine|Chemical]) [overdosage|Disease]. It is longer acting than
      ↪  [physostigmine|Chemical] and is used in anaesthesia to reverse the
      ↪  non-depolarizing neuromuscular block. However, studies into the dose
      ↪  necessary to combating [scopolamine|Chemical] intoxication are
      ↪  indicated."
    }
  ]
}
```

Listing 1: Instruction format for NER (BC5CDR)

```
{
  "conversations": [
    {
      "from": "system",
      "value": "You are an AI language model trained to extract entities from
      ↪  sentences. See all possible labels and their descriptions below.
      ↪  \n###\nDescription: Location name\nLabel: LOC\n###\nDescription:
      ↪  Person name\nLabel: PER\n###\nDescription: Organization name\nLabel:
      ↪  ORG\n###\nDescription: Miscellaneous name\nLabel: MISC\n"
    },
    {
      "from": "user",
      "value": "China says time right for Taiwan talks . BEIJING 1996-08-22
      ↪  China has said it was time for political talks with Taiwan and that
      ↪  the rival island should take practical steps towards that goal .
      ↪  Consultations should be held to set the time and format of the talks ,
      ↪  the official Xinhua news agency quoted Tang Shubei , executive vice
      ↪  chairman of the Association for Relations Across the Taiwan Straits ,
      ↪  as saying late on Wednesday ."
    },
    {
      "from": "assistant",
      "value": "[China|LOC] says time right for [Taiwan|LOC] talks .
      ↪  [BEIJING|LOC] 1996-08-22 [China|LOC] has said it was time for
      ↪  political talks with [Taiwan|LOC] and that the rival island should
      ↪  take practical steps towards that goal . Consultations should be held
      ↪  to set the time and format of the talks , the official [Xinhua|ORG]
      ↪  news agency quoted [Tang Shubei|PER] , executive vice chairman of the
      ↪  [Association for Relations Across the Taiwan Straits|ORG] , as saying
      ↪  late on Wednesday ."
    }
  ]
}
```

Listing 2: Instruction format for NER (CoNLL03)

```
{
  "conversations": [
    {
      "from": "system",
      "value": "You are an AI language model trained to extract all relations of
       ↪ entities from sentences. See all possible labels and their
       ↪ descriptions below. \n###\nDescription: \"Located in\" defined for
       ↪ [Loc|Loc]\nLabel: Located_In\n###\nDescription: \"Work for\" defined
       ↪ for [Peop|Org]\nLabel: Work_for\n###\nDescription: \"Organization
       ↪ based in\" defined for [Org|Loc]\nLabel:
       ↪ OrgBased_In\n###\nDescription: \"Live in\" defined for
       ↪ [Peop|Loc]\nLabel: Live_In\n###\nDescription: \"Kill\" defined for
       ↪ [Peop|Peop]\nLabel: Kill"
    },
    {
      "from": "user",
      "value": "[Annie Oakley|Peop] , also known as [Little Miss Sure
       ↪ Shot|Peop] , was born [Phoebe Ann Moses|Peop] in [Willowdell , Darke
       ↪ County|Loc] , in 1860 ."
    },
    {
      "from": "assistant",
      "value": "\nRelations:\nLive_In: [Annie Oakley|Peop] [Willowdell , Darke
       ↪ County|Loc]\nLive_In: [Little Miss Sure Shot|Peop] [Willowdell ,
       ↪ Darke County|Loc]\nLive_In: [Phoebe Ann Moses|Peop] [Willowdell ,
       ↪ Darke County|Loc]\n"
    }
  ]
}
```

Listing 3: Instruction format for RE (CoNLL04) with NER results (ground truth or inferred).

```
{
  "conversations": [
    {
      "from": "system",
      "value": "You are an AI language model trained to extract all entities and
       ↪ relations of entities from sentences. See all possible labels and
       ↪ their descriptions below.\n####\n[Entities]\n####\nDescription:
       ↪ Location\nLabel: Loc\n###\nDescription: Organization\nLabel:
       ↪ Org\n###\nDescription: People\nLabel: Peop\n###\nDescription:
       ↪ Other\nLabel: Other\n####\n[Relations]\n####\nDescription: \"Located
       ↪ in\" defined for [Loc|Loc]\nLabel: Located_In\n###\nDescription:
       ↪ \"Work for\" defined for [Peop|Org]\nLabel:
       ↪ Work_for\n###\nDescription: \"Organization based in\" defined for
       ↪ [Org|Loc]\nLabel: OrgBased_In\n###\nDescription: \"Live in\" defined
       ↪ for [Peop|Loc]\nLabel: Live_In\n###\nDescription: \"Kill\" defined
       ↪ for [Peop|Peop]\nLabel: Kill"
    },
    {
      "from": "user",
      "value": "` ` If it does not snow , and a lot , within this month we will
       ↪ have no water to submerge 150 , 000 hectares ( 370 , 500 acres ) of
       ↪ rice , ' ' said Bruno Pusterla , a top official of the Italian
       ↪ Agricultural Confederation ."
    },
    {
      "from": "assistant",
      "value": "` ` If it does not snow , and a lot , within this month we will
       ↪ have no water to submerge [150 , 000 hectares|Other] ( [370 , 500
       ↪ acres|Other] ) of [rice|Other] , ' ' said [Bruno Pusterla|Peop] , a
       ↪ top official of the [Italian Agricultural Confederation|Org]
       ↪ .\nRelations:\nWork_For: [Bruno Pusterla|Peop] [Italian Agricultural
       ↪ Confederation|Org]\n"
    }
  ]
}
```


Listing 4: Instruction format for RE (CoNLL04) to solve NER and RE simultaneouly