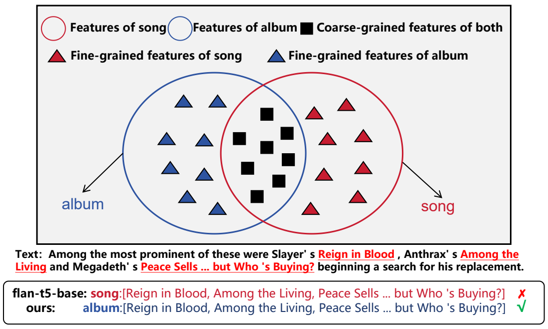
Confusion Quantification and Reasoning Enhancement for Cross-domain Named Entity Recognition

Anonymous ACL submission

Abstract

Cross-domain Named Entity Recognition (CD-NER) aims to transfer the rich knowledge in the source domain to the target domain. Recent studies adopting decomposition or generation paradigms have achieved significant performance improvements, demonstrating high accuracy in entity span detection. However, during entity type classification, models severely suffer from entity type confusion, the erroneous tendency that models classify entities of one type in the text as another similar but incorrect type. To address this issue, we first propose a Multidimensional Confusion Quantification Model (MCQM) that quantifies a model's confusion extent between entity types from three dimensions: source-target hierarchy analysis, semantic similarity analysis, and explicit data evaluation. Moreover, we propose the Progressive Bidirectional Reasoning Chain (PBRC). PBRC leverages the source-target hierarchy and confusion analysis from the MCQM to prompt the LLM to generate two-stage reasoning information. The two-stage reasoning information is utilized to augment the knowledge of the model, significantly mitigating entity type confusion and improving the model's generalization performance. Experimental results demonstrate that our method achieves new state-of-the-art results on all domains of the CrossNER dataset. [1]


Features of song
Features of album
Coarse-grained features of both
Fine-grained features of song
Fine-grained features of album
album
song

Text: Among the most prominent of these were Slayer' s Reign in Blood , Anthrax' s Among the Living and Megadeth' s Peace Sells ... but Who 's Buying? beginning a search for his replacement.

flan-t5-base: song:[Reign in Blood, Among the Living, Peace Sells ... but Who 's Buying?] ✗
ours: album:[Reign in Blood, Among the Living, Peace Sells ... but Who 's Buying?] √

Figure 1: *song* and *album* share similar contextual features and entity forms, which can lead to confusion.

1 Introduction

Named Entity Recognition (NER) is a core task in information extraction, which aims to identify specific entities in texts that belong to predefined types, such as person, location, and organization (Li et al., 2022; Yadav and Bethard, 2018; Hu et al., 2024; Esmaail et al., 2024). NER plays a critical role in various tasks, including information retrieval (Long et al., 2024; Cong et al., 2023; Nguyen et al., 2024), knowledge graphs (Chen et al., 2024; Wang et al., 2020; Jin et al., 2022), and recommendation (Li et al., 2024; Jacucci et al., 2021). With large-scale annotated data, neural network-based methods have performed remarkably in traditional NER (Shen et al., 2022; Zhu and Li, 2022; Shen et al., 2023; Yamada et al., 2020). However, in cross-domain NER, several challenges persist, especially in low-resource scenarios where model performance experiences a significant decline.

Recent studies (Xu and Cai, 2023; Zhang et al., 2024a; Xu et al., 2024) decompose cross-domain NER into subtasks to capture transferable patterns across domains. These methods focus on identifying shared feature patterns between domains through specialized components to strengthen source-domain knowledge transfer. Besides, some studies (Chen et al., 2023; Zhang et al., 2024b; Nandi and Agrawal, 2024), based on pre-trained language models designed for text generation, transform the NER into a generation task, incorporating a task-specific prompt into the input. These methods can effectively utilize large-scale knowledge from pretraining and exhibit high flexibility. Although both paradigms have demonstrated effectiveness in improving cross-domain NER performance, they remain susceptible to entity type confusion, leading to significant performance degradation in entity type classification.

[1] Code: `https://anonymous.4open.science/r/NLP/`

We observe that cross-domain NER currently faces two major challenges. (1) Granularity differences between coarse-grained entity types in the source domain and fine-grained entity types in the target domain. Entity types in the source domain are coarse-grained, such as PER and LOC, whereas entity types in the target domain are more specific and domain-specialized, such as scientist and country. Compared to coarse-grained types, distinguishing fine-grained types often requires more detailed and nuanced features. Relying solely on the coarse-grained features learned from the source domain makes it difficult for the model to accurately classify fine-grained types in the target domain, such as song and album (as shown in Figure 1). (2) The lack of contextual information in the text further limits the model's ability to classify fine-grained entity types accurately. In the target domain, some samples lack the necessary contextual cues, making it difficult for the model to effectively learn and capture the required fine-grained features for type classification. As shown in Figure 1, we selected a sample from CrossNER (Liu et al., 2021), where the song and the album are two entity types that are difficult to distinguish. Entities of both types share similar forms and contextual environments. Due to insufficient information in the text, the model misclassifies these entities as the song. The lack of adequate contextual information prevents the model from effectively learning the feature differences between similar fine-grained entity types, which reduces the precision of type classification.

To address the challenges in cross-domain NER, we propose the Multidimensional Confusion Quantification Model (MCQM) and the Progressive Bidirectional Reasoning Chain (PBRC). (1) We propose the MCQM that quantifies the model's confusion extent from three dimensions: source-target hierarchy analysis, semantic similarity analysis, and explicit data evaluation. MCQM enables in-depth analysis and quantification of the model's confusion extent between fine-grained entity types in the target domain, providing a foundation for addressing the entity type confusion. (2) Furthermore, we innovatively leverage the source-target hierarchy, along with the confusion analysis from the MCQM, to propose the PBRC, which employs a two-stage reasoning strategy. Specifically, PBRC first instructs the LLM to perform initial reasoning based on contextual information using coarse-grained entity types from the source domain. Then, it instructs the LLM to perform bidirectional reasoning using fine-grained entity types from the target domain and confusion analysis from the MCQM. Finally, the LLM generates two-stage reasoning information for knowledge augmentation. The two-stage reasoning strengthens the fine-grained features and alleviates the granularity gap between the source and target domains, while external knowledge provided by the LLM solves the issue of insufficient contextual information. Our method significantly mitigates entity type confusion. Experimental results demonstrate that our method effectively improves the model's generalization ability, with the average F1 score increasing by more than 10.00%. In summary, our contributions are as follows:

- We systematically analyze the entity type confusion in cross-domain NER. We propose a Multidimensional Confusion Quantification Model (MCQM), which can effectively quantify the model's confusion extent between fine-grained entity types in the target domain.
- We propose the Progressive Bidirectional Reasoning Chain (PBRC) based on the source-target hierarchy and confusion analysis from the MCQM, which can leverage the LLM to mitigate entity type confusion significantly.
- We conduct extensive experiments to evaluate our method. The results show that our method significantly improves the model's generalization ability, with the average F1 score increasing by more than 10.00%. Our method achieves new state-of-the-art results on all domains of the CrossNER dataset.

2 Related Work

Named Entity Recognition (NER). Traditional methods mainly rely on handcrafted features or rules combined with machine learning models like CRF (Lafferty et al., 2001) for entity recognition, which are labor-intensive and lack adaptability to complex contexts. With the development of deep learning, neural network-based methods, such as LSTM (zhiheng huang et al., 2015) and transformer (Vaswani et al., 2017) models, have become mainstream. Models (Raffel et al., 2020; Radford et al., 2019; Devlin et al., 2019) based on these architectures capture sequential relationships and contextual dependencies, offering improved performance and flexibility. However, existing methods still face challenges in cross-domain NER due to the variations across different domains.

Cross-Domain NER. Current studies can be broadly categorized into two paradigms: decomposition-based and generation-based. Decomposition-based methods (Xu and Cai, 2023; Xu et al., 2024; Zhang et al., 2024a, 2022) reformulate cross-domain NER into multiple subtasks. These methods improve knowledge transfer by learning shared cross-domain patterns through specialized subtask modules. Generation-based methods (Bao and Yang, 2024; Chen et al., 2023; Zhang et al., 2024b; Nandi and Agrawal, 2024) transform the NER into a generation task. These methods employ prompts containing task descriptions and auxiliary information, achieving robust performance through instruction fine-tuning. Although both paradigms effectively improve cross-domain performance, particularly in entity span detection, they still suffer from significant accuracy degradation in entity type classification.

LLMs for NER. With the emergence of LLMs, their reasoning capabilities have been leveraged to improve NER performance. Recent studies (Ashok and Lipton, 2023; Wang et al., 2023; Zhu et al., 2024; Ji, 2023) employ prompt learning for NER, where task-specific templates are augmented with demonstration examples and knowledge. LLMs are also used for cross-domain NER. Nandi and Agrawal (Nandi and Agrawal, 2024) perform instruction fine-tuning of LLMs by retrieving and leveraging similar examples. Zhang et al. (Zhang et al., 2024b) utilize the LLM to generate task-oriented knowledge, used to conduct additional task-oriented pre-training of the backbone model for domain adaptation. These methods demonstrate strong few-shot performance.

In this paper, we focus on entity type confusion and adopt the generation-based paradigm. We leverage the reasoning information generated by LLMs as input to a smaller text-to-text backbone model, which is fine-tuned through supervised learning to optimize the task. Compared to fine-tuning LLMs (Nandi and Agrawal, 2024) and additional pre-training corpora (Zhang et al., 2024b), our method reduces training time for domain adaptation. Experiments demonstrate our method significantly mitigates entity type confusion.

3 Methodology

3.1 Task Description

Given a sentence $X = \{x_1, x_2, \dots, x_N\}$, where w_i represents the i-th token in the sentence and N is the sentence length, the goal of NER is to identify all entities E within the sentence and classify each entity into a specific type. The set of entity types is denoted as L. Each entity $e_i \in E$ can be represented as $e_i = (y, x_{l:r})$, where $y \in L$ indicates the entity type, and l and r represent the start and end boundary indexes of the entity within the sentence, respectively. In the cross-domain NER, two distinct datasets are considered: the source domain dataset $\mathcal{D}_{src}$ and the target domain dataset $\mathcal{D}_{tgt}$. The set of entity types in the source domain is denoted as $L_s = \{s_1, s_2, \dots, s_n\}$, and the set of entity types in the target domain is denoted as $L_t = \{t_1, t_2, \dots, t_m\}$. The goal is to leverage the knowledge learned from $\mathcal{D}_{src}$ to improve recognition performance on $\mathcal{D}_{tgt}$. Specifically, we focus on low-resource scenarios, where the training data in the target domain is significantly smaller than that in the source domain, i.e., $|\mathcal{D}_{tgt}| \ll |\mathcal{D}_{src}|$.

3.2 Multidimensional Confusion Quantification Model

In this section, we introduce the MCQM. Entities of the same type share certain common features, referred to as entity type features, and entity types with similar features are more prone to confusion. Based on this, we quantify entity type confusion from the following dimensions. For implicit factors: (1) In 3.2.1, we analyze confusion arising from the source-target hierarchy, proposing our corresponding quantification method (Figure 2 (a)); (2) In 3.2.2, we analyze confusion arising from the semantic similarity, proposing our corresponding quantification method (Figure 2 (b)). For explicit data evaluation: (3) In 3.2.4, we evaluate the fine-tuned model on the target domain to explicitly analyze the confusion extent (Figure 2 (c)).

3.2.1 Source-Target Hierarchy Analysis

Entity types in the source domain are conceptually coarse-grained, whereas those in the target domain are fine-grained. In the feature space, the fine-grained entity types in the target domain can be viewed as extensions of the entity types in the source domain. For example, PER in the source domain can be expanded into fine-grained entity types in the target domain, such as politician and scientist. The expanded entity types retain the coarse-grained features in the source domain. Therefore, fine-grained entity types derived from the same entity type in the source domain exhibit similar features, meaning closer in the feature space and

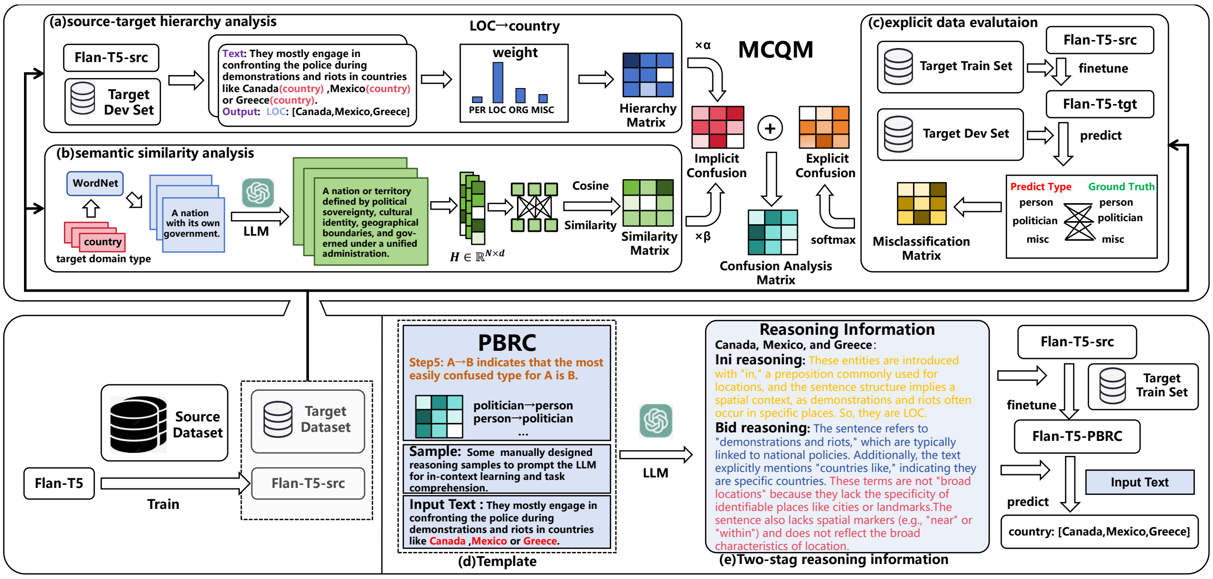

(a)source-target hierarchy analysis
Flan-T5-src
Target Dev Set
Text: They mostly engage in confronting the police during demonstrations and riots in countries like Canada(country) ,Mexico(country) or Greece(country).
Output: LOC: [Canada,Mexico,Greece]
LOC→country
weight
PER LOC ORG MISC
Hierarchy Matrix
(b)semantic similarity analysis
WordNet
country
target domain type
A nation with its own government.
LLM
A nation or territory defined by political sovereignty, cultural identity, geographical boundaries, and governed under a unified administration.
$H \in \mathbb{R}^{N \times d}$
Cosine Similarity
Similarity Matrix
×α
×β
MCQM
Implicit Confusion
Explicit Confusion
softmax
Confusion Analysis Matrix
(c)explicit data evalutaion
Target Train Set
Flan-T5-src
finetune
Flan-T5-tgt
Target Dev Set
predict
Predict Type
Ground Truth
person
politician
misc
Misclassification Matrix
Source Dataset
Target Dataset
Flan-T5
Train
Flan-T5-src
PBRC
Step5: A→B indicates that the most easily confused type for A is B.
politician→person
person→politician
...
Sample: Some manually designed reasoning samples to prompt the LLM for in-context learning and task comprehension.
Input Text : They mostly engage in confronting the police during demonstrations and riots in countries like Canada ,Mexico or Greece.
(d)Template
LLM
Reasoning Information
Canada, Mexico, and Greece:
Ini reasoning: These entities are introduced with "in," a preposition commonly used for locations, and the sentence structure implies a spatial context, as demonstrations and riots often occur in specific places. So, they are LOC.
Bid reasoning: The sentence refers to "demonstrations and riots," which are typically linked to national policies. Additionally, the text explicitly mentions "countries like," indicating they are specific countries. These terms are not "broad locations" because they lack the specificity of identifiable places like cities or landmarks.The sentence also lacks spatial markers (e.g., "near" or "within") and does not reflect the broad characteristics of location.
(e)Two-stag reasoning information
Flan-T5-src
Target Train Set
finetune
Flan-T5-PBRC
Input Text
predict
country: [Canada,Mexico,Greece]

Figure 2: Overview of our proposed method, including MCQM and PBRC. MCQM consists of three components: (a) shows source-target domain hierarchy analysis; (b) shows semantic similarity analysis; (c) shows explicit data evaluation. (d) shows the template, which includes PBRC, some manually designed reasoning samples for in-context learning, and the input text to be reasoned by the LLM. The specific contents of PBRC are shown in Figure 3.

more prone to confusion among these entity types.

To quantify the model's confusion arising from the source-target hierarchy between the source and target domains, inspired by (Bao and Yang, 2024), we propose a reliable quantification method. For each entity in the target domain, its feature vector $\boldsymbol{H}_e$ can be considered as a combination of coarse-grained and fine-grained features. Therefore, it can be represented as follows:

$$\boldsymbol{H}_e = \psi\left(\boldsymbol{H}_{coarse}, \boldsymbol{H}_{fine}\right) \tag{1}$$

where $\boldsymbol{H}_{coarse}$ is the coarse-grained features, and $\boldsymbol{H}_{fine}$ is the fine-grained features unique to the target domain entity type. $\psi(\cdot)$ denotes the feature fusion.

We utilize the model $\mathcal{M}_{src}$, trained on $\mathcal{D}_{src}$, to extract the coarse-grained features $\boldsymbol{H}_{coarse}$ of entities in the target domain and perform predictions based on the source domain type set L_s. We calculate the proportion of entities corresponding to each fine-grained entity type t_i in the target domain $\mathcal{D}_{tgt}$ that are classified as the coarse-grained entity type s_j in the source domain $\mathcal{D}_{src}$:

$$R_H\left(t_i \rightarrow s_j\right) = \frac{P\left(l_{tgt} = t_i \wedge l_{pred} = s_j\right)}{P\left(l_{tgt} = t_i\right)} \tag{2}$$

where $P(l_{tgt} = t_i)$ is the proportion of entities in $\mathcal{D}_{tgt}$ that belong to type t_i, $P(l_{tgt} = t_i \wedge l_{pred} = s_j)$ is the proportion of entities in $\mathcal{D}_{tgt}$ that belong to type t_i and are classified as s_j, and the proportion of entities of type t_i classified as s_j is denoted as $R_H(t_i \rightarrow s_j)$.

We use the source domain entity type l_{src} with the highest proportion as the prefix for the target domain entity type l_{tgt}, denoted as $l_{src} \rightarrow l_{tgt}$. The target domain entity type l_{tgt} is considered a fine-grained extension of the source domain entity type l_{src}:

$$prefix\left(t_i\right) = \underset{j \in \{1,2,\ldots,n\}}{\arg\max} \left(R_H\left(t_i \rightarrow s_j\right)\right) \tag{3}$$

where $prefix(t_i)$ is the prefix of t_i, and n is the number of entity types in the source domain.

$R_H(t_i \rightarrow s_j)$ measures how much of the coarse-grained features of s_j are contained in t_i. The more entities corresponding to t_i are classified as s_j, the more coarse-grained features of s_j are contained in t_i. Therefore, for two fine-grained entity types, t_a and t_b, if they share the same prefix, it implies that t_a and t_b exhibit a significant overlap in their coarse-grained features, which originate from the same source domain entity type. This overlap increases the likelihood of confusion between t_a and t_b. We quantify the confusion arising from the source-target hierarchy using $R_H(t_i \rightarrow s_j)$, based on the prefixes of fine-grained entity types:

$$u_{hie}^{(t_i,t_j)} = \delta_{hie} \cdot R_H\left(t_i \rightarrow prefix\left(t_j\right)\right) \tag{4}$$

where $u_{hie}^{(t_i,t_j)}$ is the extent of hierarchical confusion from the fine-grained entity type t_i to t_j, and δ_{hie} is the scaling factor. This method quantifies the model's confusion arising from the source-target hierarchy, serving as an important metric in the MCQM for quantifying entity type confusion.

3.2.2 Semantic Similarity Analysis

The semantics of entity types reflect the common features of their entities. When the model maps the embedding vectors of semantics of two entity types to closer positions in the semantic space, the model considers the entity features of the two types as more similar, which increases the probability of confusing the two types in classification.

Since entity type labels are short words, they cannot sufficiently reflect the features of the entity types. To enrich the semantics of each target domain entity type $t_{tgt} \in L_t$, we utilize the WordNet (Fellbaum, 2010) to obtain a brief description C_s of t_{tgt}. Then, we leverage the LLM to enhance the semantics of C_s, obtaining a final description C_e of t_{tgt}:

$$\begin{aligned} C_s &= \mathcal{F}_{\text{WordNet}}\left(t_{tgt}\right) \\ C_e &= \mathcal{F}_{\text{LLM}}\left(C_s, \varphi_0\right) \end{aligned} \tag{5}$$

where φ_0 is a prompt used to guide the LLM to enhance the semantics.

C_e is fed into the encoder of the model, from which the hidden layer feature sequence $\boldsymbol{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_Z] \in \mathbb{R}^{Z \times d}$ can be extracted:

$$\boldsymbol{H} = \text{Encoder}\left(C_e\right) \tag{6}$$

where $\boldsymbol{h}_i$ denotes the feature vector from the final hidden layer for the i-th token, Z is the number of tokens, and d represents the dimension of the final hidden layer of the encoder.

This study (Reimers and Gurevych, 2019) demonstrates that Mean Pooling outperforms both [CLS] token and Max Pooling strategies in semantic textual similarity tasks. Therefore, we use Mean Pooling to calculate the global feature vector $\overline{\boldsymbol{H}}$, which better integrates the semantic information of all tokens and provides a more comprehensive representation of the entity type features:

$$\overline{\boldsymbol{H}} = \frac{\sum_{i=1}^{Z} m_i \cdot \boldsymbol{h}_i}{\sum_{i=1}^{Z} m_i} \tag{7}$$

where m_i is the mask value of the i-th token, used to ignore the influence of padding.

In high-dimensional space, the features of entity types are amplified, and the mapping rules for similar features are also similar. Therefore, we calculate the semantic similarity between entity types using $\overline{\boldsymbol{H}}$. We quantify the confusion arising from the semantic similarity by computing the cosine similarity between the global feature vectors $\overline{\boldsymbol{H}}$ of entity types:

$$u_{\text{sim}}^{(t_i,t_j)} = \frac{\overline{\boldsymbol{H}}_i \cdot \overline{\boldsymbol{H}}_j}{\|\overline{\boldsymbol{H}}_i\| \cdot \|\overline{\boldsymbol{H}}_j\|} \tag{8}$$

where $u_{sim}^{(t_i,t_j)}$ is the extent of semantic confusion from the fine-grained entity type t_i to t_j. This method effectively quantifies the model's confusion arising from the semantic similarity between entity types, serving as an important metric in the MCQM for quantifying entity type confusion.

3.2.3 Fusion of Implicit Confusion Factors

The quantification of confusion arising from the source-target hierarchy and the semantic similarity of entity types is considered implicit confusion analysis. We use two fixed mixing ratios, α and β, to balance the weights of these two factors:

$$u_{iml}^{(t_i,t_j)} = \alpha \cdot u_{hie}^{(t_i,t_j)} + \beta \cdot u_{sim}^{(t_i,t_j)} \tag{9}$$

Then, we use the softmax function to model the implicit confusion and define the implicit confusion distribution among fine-grained entity types in the target domain:

$$v_{iml}(t_i \to t_j) = \frac{\exp(w_0 \cdot u_{iml}^{(t_i,t_j)})}{\sum\limits_{\substack{k=1 \\ k \neq i}}^{m} \exp(w_0 \cdot u_{iml}^{(t_i,t_k)})} \tag{10}$$

where $w_0 \in \mathbb{R}^+$ is a temperature coefficient.

3.2.4 Explicit Data Evaluation

We denote the model obtained by fine-tuning $\mathcal{M}_{src}$ on the target domain dataset as $\mathcal{M}_{tgt}$. We use the dev set of the target domain to perform a statistical analysis of the misclassification proportions for all entity types. Evaluating $\mathcal{M}_{tgt}$ reflects the model's ability to distinguish between different fine-grained entity types and explicitly reveals the model's confusion extent between entity types:

$$R_M^{(t_i,t_j)} = \frac{P(l_{true} = t_i \land l_{pred} = t_j)}{P(l_{true} = t_i)} \tag{11}$$

where $P(l_{true} = t_i)$ is the proportion of entities that truly belong to type t_i, $P(l_{true} = t_i \land l_{pred} = t_j)$ is the proportion of entities that truly belong to type t_i but are misclassified as t_j, and the proportion of entities of type t_i misclassified as t_j is denoted as $R_M^{(t_i,t_j)}$.

The value of $R_M^{(t_i,t_j)}$ explicitly reflects the model's learning effectiveness in distinguishing the fine-grained features of t_i and t_j. A larger value indicates that t_i and t_j are more prone to confusion. We use $R_M^{(t_i,t_j)}$ as an important metric for confusion analysis in the MCQM. Then, we use the softmax function to model explicit confusion and define the explicit confusion distribution among fine-grained entity types in the target domain:

$$v_{exl}(t_i \to t_j) = \frac{\exp(w_0 \cdot R_M^{(t_i,t_j)})}{\sum_{\substack{k=1 \\ k \neq i}}^{m} \exp(w_0 \cdot R_M^{(t_i,t_k)})} \tag{12}$$

where $w_0 \in \mathbb{R}^+$ is a temperature coefficient.

3.2.5 Confusion Analysis Results

We integrate the implicit and explicit evaluations to obtain the final confusion analysis results for fine-grained entity types in the target domain:

$$\begin{aligned} v_{conf}(t_i \to t_j) = v_{iml}(t_i \to t_j) \\ + v_{exl}(t_i \to t_j) \end{aligned} \tag{13}$$

where $v_{conf}(t_i \to t_j)$ is the model's confusion extent from t_i to t_j. For each entity type t_i, we select the entity type with the highest confusion extent as its most easily confused entity type:

$$T_{conf}^{(t_i)} = \underset{j \in \{1,2,\ldots,m\}, j \neq i}{\arg\max} (v_{conf}(t_i \to t_j)) \tag{14}$$

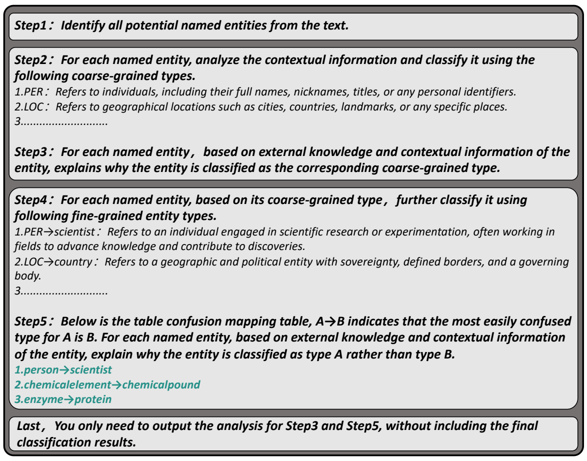
where $T_{conf}^{(t_i)}$ is the most easily confused entity type for the entity type t_i. MCQM provides a comprehensive analysis of the model's confusion extent between fine-grained entity types, playing a critical role in the bidirectional reasoning of PBRC.

3.3 Progressive Bidirectional Reasoning Chain

3.3.1 Contents of PBRC

We propose PBRC based on the source-target hierarchy and the confusion analysis from MCQM. As shown in Figure 2, PBRC prompts the LLM to output two-stage reasoning information. Figure 3 presents the contents of PBRC, and Figure 8, 9, 10, and 11 provide some specific examples.

Step1: Identifying Named Entities. This step instructs the LLM to identify all potential named entities in the text. With pre-trained multi-domain knowledge and contextual understanding, the LLM is capable of recognizing potential entities in texts across various domains.



Step1：Identify all potential named entities from the text.

Step2：For each named entity, analyze the contextual information and classify it using the following coarse-grained types.

1.PER：Refers to individuals, including their full names, nicknames, titles, or any personal identifiers.

2.LOC：Refers to geographical locations such as cities, countries, landmarks, or any specific places.

3..........................

Step3：For each named entity，based on external knowledge and contextual information of the entity, explains why the entity is classified as the corresponding coarse-grained type.

Step4：For each named entity, based on its coarse-grained type，further classify it using following fine-grained entity types.

1.PER→scientist：Refers to an individual engaged in scientific research or experimentation, often working in fields to advance knowledge and contribute to discoveries.

2.LOC→country：Refers to a geographic and political entity with sovereignty, defined borders, and a governing body.

3..........................

Step5：Below is the table confusion mapping table, A→B indicates that the most easily confused type for A is B. For each named entity, based on external knowledge and contextual information of the entity, explain why the entity is classified as type A rather than type B.

1.person→scientist

2.chemicalelement→chemicalpound

3.enzyme→protein

Last，You only need to output the analysis for Step3 and Step5, without including the final classification results.

Figure 3: Contents of PBRC.

Step2 and Step3: Coarse-grained Classification and Initial Reasoning. We provide all source domain coarse-grained entity types and explanations. For each entity in Step1, we instruct the LLM to perform coarse-grained classification and make explanations based on the LLM's rich external knowledge and the initial analysis of contextual information. This step unifies entity types across different domains, serving as an initial classification and reasoning process for the entities.

Step4 and Step5: Fine-grained Classification and Bidirectional Reasoning. We provide all fine-grained entity types of the target domain along with their corresponding source domain prefixes and explanations. For each entity, we instruct the LLM to perform fine-grained classification based on the initial classification and reasoning information from Step2 and Step3, and then conduct bidirectional reasoning through a deep analysis of contextual information and the LLM's rich external knowledge. As shown in Step5 of Figure 3, we present the confusion analysis result of the MCQM, where $A \to B$ indicates that A's most easily confused type is B. Bidirectional reasoning consists of two parts: (1) **forward reasoning**, which infers that the entity type is A; and (2) **backward reasoning**, which infers that the entity type is not B.

Two-stage Reasoning Information. The LLM only needs to output the initial reasoning from Step3 and the bidirectional reasoning from Step5, without including the final fine-grained classification results for entities. The two-stage reasoning in-

	CoNLL2003						Twitter					
Method	**sci.**	**pol.**	**mus.**	**lit.**	**AI**	**Avg.**	**sci.**	**pol.**	**mus.**	**lit.**	**AI**	**Avg.**
BiLSTM-CRF [NAACL2016]	49.97	56.60	44.79	43.03	43.56	47.59	47.33	53.64	48.85	45.23	44.08	47.83
BERT-JF [AAAI2021]	65.03	68.85	67.59	62.57	58.57	64.52	64.51	67.52	67.74	61.38	57.05	63.34
LST-NER [ACL2022]	70.07	73.25	76.83	70.76	63.28	70.84	-	-	-	-	-	-
MTD [SIGIR2022]	72.35	76.70	76.10	69.22	68.93	72.66	71.37	74.62	74.41	69.67	64.55	70.92
CP-NER [IJCAI2023]	75.82	74.25	79.10	72.17	67.95	73.86	-	-	-	-	-	-
MTD-MoCL [ACL2023]	-	-	-	-	-	-	72.83	75.13	77.15	70.71	67.87	72.74
DH-GAT [SIGIR2023]	74.21	77.06	78.77	72.51	69.30	74.37	74.55	76.46	77.33	71.52	67.65	73.50
GPDA [ACL2023]	75.55	75.95	80.16	72.34	70.05	74.81	-	-	-	-	-	-
PromptNER [arXiv2023]	72.59	78.61	84.26	74.44	64.83	74.95	-	-	-	-	-	-
Dual-CL [INFORM SYST2024]	74.17	77.56	78.57	72.43	69.49	74.44	74.07	77.53	77.21	71.72	68.71	73.85
DT-MPrompt [INFORM SYST2024]	73.06	80.54	79.54	73.51	70.13	75.36	73.18	79.86	77.93	72.74	69.13	74.57
IF-WRANER-7B [EMNLP2024]	75.31	79.8	85.43	75.52	68.81	76.97	-	-	-	-	-	-
TOPT [EMNLP2024]	80.16	81.55	82.03	77.85	72.34	78.78	-	-	-	-	-	-
Flan-T5-base-250M(ours)	81.43	83.53	83.69	79.03	74.29	80.39	80.32	82.43	81.93	77.62	73.97	79.25
Flan-T5-large-780M(ours)	**82.34**	**85.03**	**85.62**	**79.96**	**75.99**	**81.79**	**81.62**	**84.64**	**84.02**	**78.61**	**75.50**	**80.88**
Improve	**↑2.18**	**↑3.48**	**↑0.19**	**↑2.11**	**↑3.65**	**↑3.01**	**↑7.07**	**↑4.78**	**↑6.09**	**↑5.87**	**↑6.37**	**↑6.31**

Table 1: F1 scores on CrossNER: CoNLL2003 and Twitter as source domains, respectively. Bold marks the highest, and blue marks the absolute increase compared with prior SOTA.

formation is fed into the model for knowledge augmentation: the initial reasoning guides the model to leverage the knowledge from the source domain, strengthening knowledge transfer; the bidirectional reasoning guides the model to differentiate easily confused entity types, mitigating the entity type confusion. Besides, external knowledge mitigates the insufficiency of contextual information.

3.3.2 Knowledge Augmentation

The two-stage reasoning information generated by the LLM is utilized for knowledge augmentation in the backbone model, which learns to leverage reasoning information through supervised fine-tuning.

- **Input:** Find all entities of types {politician, person, country ...} in {text}.
  Reasoning information: {...}.
  Output format: {"type 1": ["entity 1", "entity 2"], "type 2": ["entity 3"], ...}.

- **Gold Sequence:** {"country": ["Afghanistan"], ”politician": ["Barack Obama"], ...}.

More details are provided in Appendix A.

4 Experiments

4.1 Experimental Setup

We evaluate on three datasets: two source domains (CoNLL2003 (Tjong Kim Sang and De Meulder, 2003), Twitter (Lu et al., 2018)) and one target domain (CrossNER (Liu et al., 2021) with five sub-domains: politics, science, music, literature, and AI). We employ Flan-T5-base and Flan-T5-large (Chung et al., 2024) as backbone models. we compare our method with prior SOTA baselines. More details, including datasets, implementation, and baselines, are provided in Appendix B.

4.2 Main Results

The main results are shown in Table 1.

CoNLL2003 as the Source Domain. It is observed that the F1 score improves as the parameter scale of the model increases. On average, our method achieves consistent F1 score improvements of +1.61% (Flan-T5-base) and +3.01% (Flan-T5-large). Our method surpasses prior SOTA results in science, politics, literature, and AI domains (effective for both Flan-T5-base and Flan-T5-large), achieving improvements exceeding 2.10% in these domains. These results demonstrate the strong effectiveness of two-stage information generated by the LLM in cross-domain NER. We observe that **IF-WARNER-7B** (85.43%), which employs a fine-tuned 7B LLaMA, and **PromptNER** (84.26%), which utilizes GPT-4, both achieve significantly higher F1 scores in the music domain compared to other baselines using smaller models. This proves the advantage of LLMs in the music domain, ensuring the quality of our two-stage information generated by the LLM. However, Flan-T5-large achieves only a marginal improvement of 0.19% over prior SOTA in the music domain, while Flan-T5-base fails to surpass the prior SOTA. Through careful analysis, we have identified the following key factors:

- The high overlap between the entity knowledge of the LLM and Flan-T5 in the music domain diminished their complementary effects, resulting in limited benefits from two-stage reasoning (as shown in Table 2).

- Although the two-stage information significantly improves F1 scores for certain entity types, such as musicalinstrument (Table 6),

Method	science	politics	music	literature	AI	Avg.
w/o all	69.96	70.28	75.32	67.87	63.56	69.40
w/o initial and backward reasoning	76.58	78.26	80.09	74.77	70.03	75.95
w/o backward reasoning	78.06	79.47	81.44	76.12	71.42	77.30
w/o initial reasoning	80.62	82.76	82.49	78.06	73.52	79.49
Flan-T5-base-250M(ours)	81.43	83.53	83.69	79.03	74.29	80.39
Improve	**↑11.47**	**↑13.25**	**↑8.37**	**↑11.16**	**↑10.73**	**↑10.99**

Table 2: Ablation study (F1 scores). Blue marks the absolute increase.

their limited entity count results in a negligible impact on the overall F1 metric.

Twitter as the Source Domain. When Twitter is used as the source domain, our method consistently achieves significant improvements across all five target domains, with an average F1 score increase of 6.31%. Since **TOPT** is not tested with Twitter as the source domain, we cannot compare with it directly. Compared to CoNLL2003, the performance of both the Flan-T5-base and Flan-T5-large shows a slight decline. We attribute this to the smaller dataset size of Twitter, which limits the model's ability to fully learn the coarse-grained features of entities. This highlights the importance of learning coarse-grained features of entities and further demonstrates the capability of our method to effectively transfer knowledge from the source domain to the target domain.

4.3 Ablation Study

To evaluate the effectiveness of each component in our proposed method, we conduct ablation studies, as shown in Table 2. Our ablation study incorporates the following configurations: (1) w/o initial reasoning: remove Step2 and Step3 in PBRC. (2) w/o backward reasoning: remove the confusion analysis results from MCQM at Step5 in PBRC. (3) w/o initial and backward reasoning: only the forward reasoning at Step5. (4) w/o all: remove all components. Through additive ablation analysis, we can obtain the following results:

The contribution of forward reasoning. Based on w/o all (69.40%) and w/o initial and backward reasoning (75.95%), we observe a +6.55% absolute improvement. This demonstrates that the reasoning information provided by the LLM can effectively support the backbone model in performing NER.

The contribution of initial reasoning. Based on w/o initial and backward reasoning (75.95%) and w/o backward reasoning (77.30%), we observe a +1.35% absolute improvement, demonstrating that the two-stage reasoning from source to target domain can enhance the backbone model's cross-domain transferability.

The contribution of backward reasoning. Based on w/o initial and backward reasoning (75.95%) and w/o initial reasoning (79.49%), we observe a +3.54% absolute improvement. This indicates that MCQM effectively analyzes and quantifies the model's confusion among entity types, while backward reasoning helps alleviate entity type confusion and improves the model's generalization ability.

The collective contribution of all components. Based on w/o all (69.40%) and ours (80.39%), we observe a +10.99% absolute improvement, demonstrating that the two-stage reasoning information generated by the LLM significantly mitigates the model's entity type confusion and improves the model's generalization ability.

In the music domain, all components contribute smaller improvements than in other domains. The reasons have been analyzed in Section 4.2. To further evaluate our method, **additional experiments** are provided in Appendix C.

5 Conclusion and Future Work

In this paper, we propose the MCQM, which effectively quantifies the model's confusion extent among fine-grained entity types from three dimensions: source-target hierarchy analysis, semantic similarity analysis, and explicit data evaluation. Furthermore, we propose the PBRC based on the source-target hierarchy and the MCQM, which can significantly mitigate entity type confusion and improve the model's generalization ability. Our method achieves SOTA results on all domains of the CrossNER dataset. Future research can focus on fully utilizing the MCQM. Confusion analysis of the MCQM can help optimize data labeling strategies and guide data augmentation. We anticipate that these findings will inspire further research.

Limitations

Although our method achieves new state-of-the-art performance, there are two major limitations.

Visualizations on MCQM show that some entity types do not have a single, clearly most confused entity type. Through experiments, we find that for these entity types, the improvement brought by the backward reasoning in the two-stage reasoning information generated by the LLM is limited. Moreover, PBRC does not fully leverage MCQM's confusion analysis results, which restricts MCQM from reaching its full potential. In the future, we need to explore the more effective utilization of MCQM further.

Our method adopts an LLM with frozen parameters combined with a fine-tuned domain-specific small model. The experimental results demonstrate that our method outperforms the individual use of the LLM or the fine-tuned small model. For a new domain, our method only requires a small amount of labeled data to fine-tune the small model, achieving strong performance while significantly reducing training time and cost. However, the model's performance is influenced by the quality of the information generated by the LLM. Besides, during inference, the model still relies on the LLM to generate two-stage reasoning information, which slows down inference speed.

Ethics Statement

Our study does not involve human subjects, personal data, or sensitive content. All experiments use publicly available datasets in compliance with their licenses. We employ language models such as GPT (OpenAI API) and Flan-T5, following their respective usage policies. These models are used in controlled settings and do not produce harmful or biased content in the context of our study.

References

Dhananjay Ashok and Zachary C. Lipton. 2023. Promptner: Prompting for named entity recognition. *Preprint*, arXiv:2305.15444.

Ke Bao and Chonghuan Yang. 2024. Label alignment and reassignment with generalist large language model for enhanced cross-domain named entity recognition. *Preprint*, arXiv:2407.17344.

Jiong Cai, Shen Huang, Yong Jiang, Zeqi Tan, Pengjun Xie, and Kewei Tu. 2023. Improving low-resource named entity recognition with graph propagated data augmentation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 110–118, Toronto, Canada. Association for Computational Linguistics.

Xiang Chen, Lei Li, Shuofei Qiao, Ningyu Zhang, Chuanqi Tan, Yong Jiang, Fei Huang, and Huajun Chen. 2023. One model for all domains: collaborative domain-prefix tuning for cross-domain ner. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, IJCAI '23.

Zhongwu Chen, Long Bai, Zixuan Li, Zhen Huang, Xiaolong Jin, and Yong Dou. 2024. A new pipeline for knowledge graph reasoning enhanced by large language models without fine-tuning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1366–1381, Miami, Florida, USA. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Xin Cong, Bowen Yu, Mengcheng Fang, Tingwen Liu, Haiyang Yu, Zhongkai Hu, Fei Huang, Yongbin Li, and Bin Wang. 2023. Universal information extraction with meta-pretrained self-retrieval. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4084–4100, Toronto, Canada. Association for Computational Linguistics.

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*, pages 1835–1845, Online. 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.

Naji Esmaail, Nazlia Omar, Masnizah Mohd, Fariza Fauzi, and Zainab Mansur. 2024. Named entity recognition in user-generated text: A systematic literature review. *IEEE Access*, 12:136330–136353.

Christiane Fellbaum. 2010. Wordnet. In Roberto Poli, Michael Healy, and Achilles Kameas, editors, *Theory and Applications of Ontology: Computer Applications*, pages 231–243. Springer.

Zhentao Hu, Wei Hou, and Xianxing Liu. 2024. Deep learning for named entity recognition: a survey. *Neural Comput. Appl.*, 36(16):8995–9022.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*, 43(2).

Giulio Jacucci, Pedram Daee, Tung Vuong, Salvatore Andolina, Khalil Klouche, Mats Sjöberg, Tuukka Ruotsalo, and Samuel Kaski. 2021. Entity recommendation for everyday digital tasks. *ACM Trans. Comput.-Hum. Interact.*, 28(5).

Bin Ji. 2023. Vicunaner: Zero/few-shot named entity recognition using vicuna.

Chen Jia and Yue Zhang. 2020. Multi-cell compositional LSTM for NER domain adaptation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5906–5917, Online. Association for Computational Linguistics.

Zhuoran Jin, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. 2022. A good neighbor, a found treasure: Mining treasured neighbors for knowledge graph entity typing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 480–490, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning*.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2022. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):50–70.

Xinhang Li, Xiangyu Zhao, Yejing Wang, Yu Liu, Chong Chen, Cheng Long, Yong Zhang, and Chunxiao Xing. 2024. Opensiterec: An open dataset for site recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 1483–1493, New York, NY, USA. Association for Computing Machinery.

Zihan Liu, Genta Indra Winata, Peng Xu, and Pascale Fung. 2020. Coach: A coarse-to-fine approach for cross-domain slot filling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 19–25, Online. Association for Computational Linguistics.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2021. Crossner: Evaluating cross-domain named entity recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(15):13452–13460.

Zijun Long, Xuri Ge, Richard McCreadie, and Joemon M. Jose. 2024. Cfir: Fast and effective long-text to image retrieval for large corpora. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 2188–2198, New York, NY, USA. Association for Computing Machinery.

Di Lu, Leonardo Neves, Vitor Carvalho, Ning Zhang, and Heng Ji. 2018. Visual attention model for name tagging in multimodal social media. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1990–1999, Melbourne, Australia. Association for Computational Linguistics.

Subhadip Nandi and Neeraj Agrawal. 2024. Improving few-shot cross-domain named entity recognition by instruction tuning a word-embedding based retrieval augmented large language model. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 686–696, Miami, Florida, US. Association for Computational Linguistics.

Thong Nguyen, Shubham Chatterjee, Sean MacAvaney, Iain Mackie, Jeff Dalton, and Andrew Yates. 2024. DyVo: Dynamic vocabularies for learned sparse retrieval with entities. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 767–783, Miami, Florida, USA. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

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(1).

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Yongliang Shen, Zeqi Tan, Shuhui Wu, Wenqi Zhang, Rongsheng Zhang, Yadong Xi, Weiming Lu, and Yueting Zhuang. 2023. PromptNER: Prompt locating and typing for named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12492–12507, Toronto, Canada. Association for Computational Linguistics.

Yongliang Shen, Xiaobin Wang, Zeqi Tan, Guangwei Xu, Pengjun Xie, Fei Huang, Weiming Lu, and Yueting Zhuang. 2022. Parallel instance query network for named entity recognition. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 947–961, Dublin, Ireland. Association for Computational Linguistics.

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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17. Curran Associates Inc.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023. Gpt-ner: Named entity recognition via large language models. *Preprint*, arXiv:2304.10428.

Zhichun Wang, Jinjian Yang, and Xiaoju Ye. 2020. Knowledge graph alignment with entity-pair embedding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1672–1680, Online. Association for Computational Linguistics.

Jingyun Xu and Yi Cai. 2023. Decoupled hyperbolic graph attention network for cross-domain named entity recognition. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, New York, NY, USA. Association for Computing Machinery.

Jingyun Xu, Junnan Yu, Yi Cai, and Tat seng Chua. 2024. Dual contrastive learning for cross-domain named entity recognition. *ACM Trans. Inf. Syst.*, 42:163:1–163:33.

Jingyun Xu, Changmeng Zheng, Yi Cai, and Tat-Seng Chua. 2023. Improving named entity recognition via bridge-based domain adaptation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3869–3882, Toronto, Canada. Association for Computational Linguistics.

Vikas Yadav and Steven Bethard. 2018. A survey on recent advances in named entity recognition from deep learning models. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2145–2158, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454, Online. Association for Computational Linguistics.

Yi Yang and Arzoo Katiyar. 2020. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6365–6375, Online. Association for Computational Linguistics.

Xinghua Zhang, Bowen Yu, Xin Cong, Taoyu Su, Quangang Li, Tingwen Liu, and Hongbo Xu. 2024a. Cross-domain ner under a divide-and-transfer paradigm. *ACM TRANSACTIONS ON INFORMATION SYSTEMS*, 42(5).

Xinghua Zhang, Bowen Yu, Yubin Wang, Tingwen Liu, Taoyu Su, and Hongbo Xu. 2022. Exploring modular task decomposition in cross-domain named entity recognition. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, New York, NY, USA. Association for Computing Machinery.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models.

Zhihao Zhang, Sophia Yat Mei Lee, Junshuang Wu, Dong Zhang, Shoushan Li, Erik Cambria, and Guodong Zhou. 2024b. Cross-domain NER with generated task-oriented knowledge: An empirical study from information density perspective. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1595–1609, Miami, Florida, USA. Association for Computational Linguistics.

Junhao Zheng, Haibin Chen, and Qianli Ma. 2022. Cross-domain named entity recognition via graph matching. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2670–2680, Dublin, Ireland. Association for Computational Linguistics.

zhiheng huang, wei xu, and kai yu. 2015. Bidirectional lstm-crf models for sequence tagging. *Computing Research Repository*, abs/1508.01991.

Enwei Zhu and Jinpeng Li. 2022. Boundary smoothing for named entity recognition. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7096–7108, Dublin, Ireland. Association for Computational Linguistics.

Xingyu Zhu, Feifei Dai, Xiaoyan Gu, Bo Li, Meiou Zhang, and Weiping Wang. 2024. Glner: Generation-aware large language models fornbsp;few-shot named entity recognition. In *Artificial Neural Networks and Machine Learning – ICANN 2024*.

A Knowledge Augmentation for Language Models

As shown in Figure 2 (d), we design a template $\tau(\cdot)$, which consists of three parts: some manually designed examples S, PBRC $\mathbf{P}$, and the text to be reasoned $X \in \mathcal{D}_{tgt}$. The examples help the LLM perform in-context learning, while PBRC includes the confusion analysis results from MCQM, which are transformed into a mapping table T_{map} (Step5). By feeding the template $\tau(\cdot)$ into the LLM, we obtain the two-stage reasoning information I_R of the text:

$$I_R = \mathcal{F}_{\text{LLM}}(\tau(P[T_{map}], S, X)) \tag{15}$$

The original text $X = \{x_1, x_2, \ldots, x_N\}$ is concatenated with the two-stage reasoning information $I_R = \{i_1, i_2, \ldots, i_M\}$ to form $[X, I_R] = \{x_1, x_2, \ldots, x_N, i_1, i_2, \ldots, i_M\}$. $[X, I_R]$ is fed into the model's encoder, resulting in the feature sequence of the final hidden layer $\boldsymbol{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_{N+M}] \in \mathbb{R}^{(N+M)\times d}$:

$$\boldsymbol{H} = \text{Encoder}([X, I_R]) \tag{16}$$

where d is the dimension of the encoder's final hidden layer. The model's encoder dynamically attends to the reasoning information through the attention mechanism, thereby enriching the original text representations and achieving effective knowledge augmentation.

The output sequence generated by the model before the timestep t is $Y_{1:t-1} = \{y_1, y_2, \ldots, y_{t-1}\}$ and its feature sequence is $\boldsymbol{H}_{Y[1:t-1]} \in \mathbb{R}^{(t-1)\times d}$. The decoder of the model calculates the hidden state z_t at the timestep t based on $\boldsymbol{H}$ and $\boldsymbol{H}_{Y[1:t-1]}$:

$$z_t = f(\boldsymbol{H}, \boldsymbol{H}_{Y[1:t-1]}) \tag{17}$$

The two-stage reasoning information further influences the calculation of the hidden state z_t. Then, the decoder uses the softmax to calculate the probability distribution over the vocabulary $V = \{v_1, v_2, \ldots, v_K\}$:

$$p(V|\boldsymbol{H}, \boldsymbol{H}_{Y[1:t-1]}) = \text{softmax}(Wz_t + b) \tag{18}$$

where the size of the vocabulary is K, $W \in \mathbb{R}^{K\times d}$ and $b \in \mathbb{R}^K$ are the output projection parameters.

The decoder selects the token with the highest probability as the output at the timestep t:

$$y_t = \underset{i\in\{1,2,\ldots,K\}}{\arg\max} \left(p(v_i|\boldsymbol{H}, \boldsymbol{H}_{Y[1:t-1]})\right) \tag{19}$$

where K is the size of the vocabulary.

Thus, after T time steps, the language model generates the final result. The two-stage reasoning information can effectively guide the language model to recognize named entities. We use the cross-entropy loss function to optimize the language model. Since our gold sequence is deterministic (i.e., the output token at each timestep is unique), we derive the simplified form:

$$\begin{aligned}\mathcal{L}_{\text{cross}} &= -\sum_{t=1}^{T}\sum_{i=1}^{K} P(y_t = v_i) \\ &\quad \cdot \log p(y_t = v_i \mid \boldsymbol{H}, \boldsymbol{H}_{Y[1:t-1]}) \\ &= -\sum_{t=1}^{T} \log p(y_t^* \mid \boldsymbol{H}, \boldsymbol{H}_{Y[1:t-1]})\end{aligned} \tag{20}$$

where y_t^* is the ground-truth token at timestep t.

Domain	Dataset	Category	Train	Dev	Test
Source	CoNLL2003	4	14987	-	-
	Twitter	4	4290	-	-
Target	politics	9	200	541	651
	science	17	200	450	543
	music	13	100	380	465
	literature	12	100	400	416
	AI	14	100	350	431

Table 3: The statistics of all datasets.

B Experimental Setup

B.1 Datasets & Evaluation Metrics

We conduct extensive experiments on classic datasets, including two source domain datasets, CoNLL2003 (Tjong Kim Sang and De Meulder, 2003) and Twitter (Lu et al., 2018), and one target domain dataset, CrossNER (Liu et al., 2021), including five domains: politics, science, music, literature, and AI. Table 3 shows the statistics of these datasets. We adhere to the official split, ensuring consistency with baselines. Specifically, the entity types in the source domains are coarse-grained, while the entity types in the target domains are fine-grained, indicating that the cross-domain setting is more challenging and closer to the real world. Besides, the training set in the target domain is smaller in scale.

We use the F1-score (F) as the primary metric to evaluate the performance of our method. An entity is considered correct only if both its span and type are correct. Moreover, we use precision (P) and recall (R) for a more detailed analysis.

B.2 Implementation Details

We conduct our experiments on a single NVIDIA A6000 GPU with 48GB, using the PyTorch framework. We employ Flan-T5-base (250M) and Flan-T5-large (780M) (Chung et al., 2024) as backbone models[2]. Flan-T5 has undergone instruction fine-tuning across multiple NLP tasks, enabling it to better follow instructions, generate outputs in specified formats, and demonstrate improved cross-domain adaptation and few-shot learning capabilities. We employ GPT-4o-mini (API)[3] as the LLM.

We utilize the AdamW optimizer with a learning rate of 1×10^{-5} and a batch size of 10. For MCQM, the mixing ratios α and β are both set to 0.5 to balance the weights of the two implicit factors. The temperature coefficient w_0 of the Softmax function is set to 2.5. The scaling factor δ_{hie} is set to 1 by default. For PBRC, the number of samples in the template for context learning is set to 10.

Following prior studies, we first train the model in the source domain and fine-tune it in the target domain training set.

B.3 Baselines

To evaluate the performance of our proposed method, we compare it with the following SOTA baselines:

- **BiLSTM-CRF** (Yamada et al., 2020): Combines BiLSTM and CRF to train the model.
- **BERT-JF** (Liu et al., 2021): Jointly fine-tunes BERT in source and target domains.
- **LST-NER** (Zheng et al., 2022): Adopts graph matching to address cross-domain data scarcity and label mismatches.
- **MTD** (Zhang et al., 2022): A modular learning-based method that decomposes NER into span detection and type classification.
- **CP-NER** (Chen et al., 2023): Transforms NER as text-to-text task and introduces collaborative domain-prefix tuning based T5 as well.
- **MTD-MoCL** (Xu et al., 2023): Use contrastive learning to refine representations, generate positive and negative samples, and optimize their distances to enhance distinguishability.
- **DH-GAT** (Xu and Cai, 2023): Applies Graph Attention Networks to encode syntactic and semantic information while embedding words into hyperbolic space.
- **GPDA** (Cai et al., 2023): constructs a text similarity graph between labeled data and unlabeled text and propagates entity annotations from labeled data to unlabeled text through graph propagation.
- **PromptNER** (Ashok and Lipton, 2023): Uses GPT4 for NER through prompt templates.
- **Dual-CL** (Xu et al., 2024): Uses dual contrastive learning to refine ambiguous representations and learn generalizable features.
- **DT-MPrompt** (Zhang et al., 2024a): Splits the cross-domain NER task into subtasks and uses separate functional modules for learning and knowledge transfer.
- **IF-WRANER** (Nandi and Agrawal, 2024): Fine-tunes 7B LLaMA with instruction fine-tuning and employs word embeddings to retrieve examples for in-context learning.
- **TOPT** (Zhang et al., 2024b): Utilizes LLaMA to generate task-oriented knowledge for flan-T5 and adopts task-oriented pre-training for domain adaptation (SOTA).

To evaluate the performance of our method in low-resource scenarios, we follow the study (Zheng et al., 2022) and conduct few-shot experiments, comparing our method with the following baselines:

- **BiLSTM-CRF** (Yamada et al., 2020): Combines BiLSTM and CRF to train the model.
- **Coach** (Liu et al., 2020): Initially learns a general slot-entity pattern, then predicts specific types, and enhances adaptability and robustness through the integration of template regularization.
- **Multi-Cell LSTM** (Jia and Zhang, 2020): A multi-task learning framework using LSTMs, where separate cell states are utilized to model each entity type.

[2] https://huggingface.co/docs/transformers/main/en/model_doc/flan-t5

[3] https://platform.openai.com/docs/models/gpt-4o-mini#gpt-4o-mini

Method	20-shot science	politics	music	litera.	AI	50-shot science	politics	music	litera.	AI
BiLSTM-CRF	42.54	41.75	37.96	35.78	37.59	48.89	53.46	43.65	41.54	44.73
BiLSTM-CRF-joint †	44.91	44.62	42.28	39.54	41.23	49.68	55.17	44.58	43.14	46.35
Coach †	48.71	46.15	43.37	41.64	41.55	52.03	60.97	51.56	48.73	51.15
Multi-Cell LSTM †	60.55	59.58	67.12	63.92	55.39	65.78	68.21	70.47	66.85	58.67
BERT-tagger	60.34	61.01	64.73	61.79	53.78	63.93	66.13	68.41	63.44	58.93
BERT-tagger-joint †	60.58	61.61	64.16	60.36	53.18	64.04	66.30	67.71	62.58	58.52
NNShot	60.67	60.93	64.21	61.64	54.27	63.78	66.33	67.94	63.19	59.17
StructShot	62.95	63.31	67.27	63.48	55.16	64.52	67.16	70.21	65.33	59.73
TemplateNER	62.64	63.39	62.00	61.84	56.34	62.84	65.23	64.57	64.49	56.58
LST-NER	64.03	64.06	68.83	64.94	57.78	66.48	68.51	72.04	66.73	60.69
Flan-T5-base(ours)	75.97	77.93	76.86	72.35	69.20	77.43	80.34	78.91	74.55	72.36
Flan-T5-large(ours)	**77.57**	**79.31**	**78.82**	**74.09**	**70.97**	**79.29**	**82.20**	**80.94**	**75.87**	**73.48**

Table 4: The results (F1 score) in few-shot settings (CoNLL2003 as the source domain). Bold marks the highest. † indicates both source and target labeled samples are used when training.

- **BERT-tagger** (Devlin et al., 2019): Fine-tunes the BERT-based model with a label classifier.

- **NNShot** and **StructShot** (Yang and Katiyar, 2020): Two metric-based few-shot learning methods for NER.

- **TemplateNER** (Cui et al., 2021): A template-based prompt method through a generative pre-trained LM.

- **LST-NER** (Zheng et al., 2022): Adopts graph matching to address cross-domain data scarcity and label mismatches.

C Additional Experiments

C.1 Low-Resource Study

C.1.1 Few-shot Study

To evaluate the performance of our method in few-shot settings, we conduct 20-shot and 50-shot experiments using the CoNLL2003 as the source domain, with results presented in Table 4. Our method significantly outperforms the baselines. When the target domain data size decreases from 50-shot to 20-shot, the average F1 score of the Flan-T5-base across the five domains drops by only 2.26%, while the Flan-T5-large drops by 2.20%. Compared to other baselines, our method demonstrates greater robustness in few-shot settings. In particular, in the AI domain, Flan-T5-base achieves an F1 score of 72.36% under the 50-shot setting. From Table 1, we observe that the prior SOTA method, **TOPT**, achieves an F1 score of 72.34% in the standard setting. In other domains, our method under few-shot settings surpasses the performance of many baselines under their standard settings. These results indicate that our method can deliver strong performance even with limited data. By leveraging the external knowledge and reasoning capabilities of the LLM, our method effectively enhances the model's generalization ability in few-shot settings.

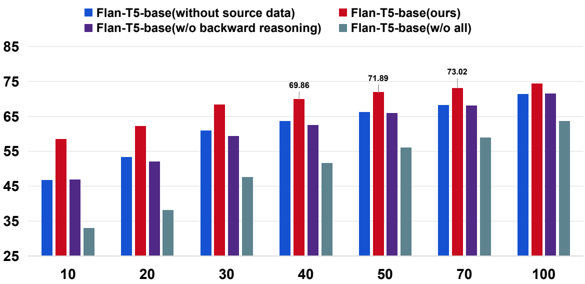


Figure 4: The effect of AI domain sample size on F1 score in four different settings. CoNLL2003 as the source domain.

C.1.2 Impact of Sample Size on Performance

To further analyze the performance of our method in low-resource settings, we conduct more detailed experiments in the AI domain using the Flan-T5-base (250M). As shown in Figure 4, We compare our method (Flan-T5-base (ours)) with several baselines: one that removes both MCQM and PBRC (w/o all), one that removes the MCQM (w/o backward reasoning), and one that evaluates our method without utilizing source domain training data (without source data). The results demonstrate that our method significantly improves F1 scores, and the improvement becomes more evident as the sample size decreases. Specifically, under the setting of 50 samples, our method achieves an F1 score of 71.89%, which is nearly equivalent to the performance of **TOPT** (Table 1) that uses Flan-T5-base

Domain	Flan-T5-base (w/o all)			Flan-T5-base (ours)		
	Precision	Recall	F1 score	Precision	Recall	F1 score
science	71.31 (83.11)	68.66 (81.10)	69.96 (82.09)	83.05 (88.12)	79.88 (84.49)	81.43 (86.27)
politics	71.30 (87.45)	69.29 (86.99)	70.28 (87.22)	84.62 (91.15)	82.47 (88.01)	83.53 (89.55)
music	77.07 (88.46)	73.65 (82.99)	75.32 (85.64)	85.34 (89.45)	82.10 (87.85)	83.69 (88.64)
literature	68.28 (83.67)	67.47 (83.98)	67.87 (83.82)	80.48 (87.67)	77.64 (84.12)	79.03 (85.86)
AI	63.42 (81.15)	63.70 (81.64)	63.56 (81.39)	75.74 (86.07)	72.90 (83.01)	74.29 (84.51)

Table 5: Type classification performance comparison. CoNLL2003 as the source domain.

as a backbone under the setting of 100 samples. Besides, when we remove the backward reasoning, the model shows more evident reductions in F1-score as the sample size decreases. We believe that limited samples make it challenging for the model to learn the feature differences among similar fine-grained entity types, exacerbating entity type confusion. The backward reasoning helps the model differentiate between easily confused entity types during type classification, highlighting the significant advantage of the confusion analysis of MCQM in low-resource scenarios. Additionally, we observe that the gap in F1 scores between training with and without the source domain increases as the sample size decreases. This indicates that in low-resource settings, our method effectively enables the model to leverage coarse-grained features learned from the source domain, thereby enhancing the model's transferability of knowledge from the source domain.

C.2 Entity Type Classification Performance

To verify that the model's performance improvement is attributed to enhanced entity type classification capabilities, we evaluate its performance in both entity span detection and overall performance using precision, recall, and F1 score. The results are shown in Table 5, where the values in parentheses indicate the performance on entity span detection. We observe that our method improves the F1 score for entity span detection by 2%–4% across the five domains, with slight improvements also observed in precision and recall. More notably, the overall performance increases by over 10% on average. These results demonstrate that our method significantly enhances the model's ability in entity type classification and effectively mitigates the model's entity type confusion.

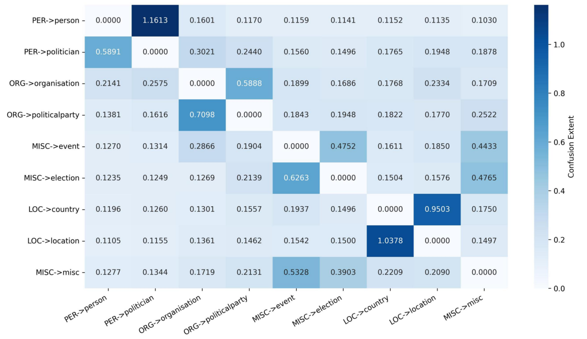


Figure 5: Visualization of confusion analysis. For A->B, source entity type A is a prefix of target entity type B.

C.3 Visualization Analysis

We utilize a heatmap to visualize the confusion analysis results of MCQM for Flan-T5-large (780M) under the CoNLL2003 → politics setting, with each target domain entity type labeled with its corresponding source domain prefix. As shown in Figure 5, we observe that target domain entity types sharing the same prefix tend to exhibit higher confusion extent, which demonstrates the effectiveness of employing the two-stage progressive reasoning strategy in PBRC. For certain entity types (e.g., person, country), their most easily confused entity type is distinctly identifiable. For instance, the confusion extent between person and politician is substantially higher than that between person and other types. Our experiments show that the bidirectional reasoning in PBRC brings more significant improvements for such types. In contrast, for other entity types (e.g., event, misc), these types are associated with multiple entities that show similar confusion extent. We find that these entity types often involve more diverse entity forms and contextual scenarios in the real world, making it challenging to pinpoint their most easily confused entity types. Overall, our MCQM effectively quantifies the model's confusion extent between fine-grained

Domain	Entity Type	Method	precision	recall	F1 score
science	scientist	w/o all	83.51	95.73	89.20
		Flan-T5-large (ours)	94.93 ↑11.42	96.95 ↑1.22	95.93 ↑6.73
	person	w/o all	85.85	59.48	70.27
		Flan-T5-large (ours)	83.22 ↓2.63	81.05 ↑21.57	82.12 ↑11.85
	chemicalcompound	w/o all	75.80	61.66	68.00
		Flan-T5-large (ours)	81.52 ↑5.72	77.72 ↑16.06	79.50 ↑11.50
	chemicalelement	w/o all	28.57	26.09	27.27
		Flan-T5-large (ours)	88.24 ↑59.67	65.22 ↑39.13	75.00 ↑47.73
politics	politician	w/o w/o all	56.99	92.21	70.45
		Flan-T5-large (ours)	91.19 ↑34.20	93.19 ↑0.98	92.18 ↑21.73
	person	w/o all	95.52	22.38	36.26
		Flan-T5-large (ours)	93.61 ↓1.91	87.06 ↑64.68	90.22 ↑53.96
	organisation	w/o all	74.01	64.73	69.06
		Flan-T5-large (ours)	82.31 ↑8.30	84.22 ↑19.49	83.26 ↑14.20
	politicalparty	w/o all	85.96	86.61	86.28
		Flan-T5-large (ours)	91.63 ↑5.67	87.37 ↑0.76	89.45 ↑3.17
music	song	w/o all	89.76	83.71	86.63
		Flan-T5-large (ours)	93.18 ↑3.42	92.13 ↑8.42	92.66 ↑6.03
	album	w/o all	80.68	84.40	82.50
		Flan-T5-large (ours)	93.09 ↑12.41	90.78 ↑6.38	91.92 ↑9.42
	musicalinstrument	w/o all	22.22	4.76	7.84
		Flan-T5-large (ours)	84.38 ↑62.16	64.29 ↑59.53	72.97 ↑65.13
	misc	w/o all	27.59	25.81	26.67
		Flan-T5-large (ours)	43.48 ↑15.89	32.26 ↑6.45	37.04 ↑10.37

Table 6: Some easily confused entity types (CoNLL2003 as the source domain). Use precision, recall, and F1 for analysis. Blue marks the absolute increase.

entity types. In Figure 7, we present the visualization results for Flan-T5-large from CoNLL2003 to five domains.

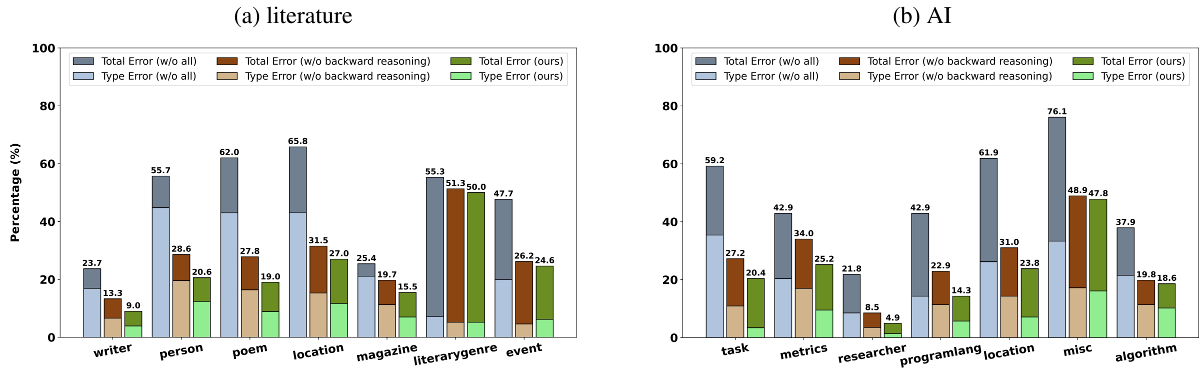
C.4 Case Study

C.4.1 Confusion Type

To further analyze the effectiveness of our method, we select several entity types from different target domains based on MCQM, as shown in Table 6, including unidirectional confusion relationships, where type A's most easily confused type is B, but B's most easily confused type is not A ($A \rightarrow B$), and bidirectional confusion relationships, where A and B are the most easily confused types with each other ($A \leftrightarrow B$). Based on the analysis of MCQM (Figure 7), the relationships are as follows: chemicalelement → chemicalpound, musicalinstrument → misc, song ↔ album, politician ↔ person, organization ↔ politicalparty, scientist ↔ person. Comparisons are made using three metrics: precision (P), recall (R), and F1 score.

Bidirectional Confusion. In Table 6, we observe that our method significantly improves the precision, recall, and F1 score for entity types. In the case of politician and person, under the w/o setting, the precision of politician (56.99%) is much lower than the recall (92.21%), and the recall of person (22.38%) is much lower than its precision (95.52%). According to the formulas for precision and recall, this can be attributed to many entities of person being misclassified as politician, which is consistent with the confusion analysis result of MCQM (Figure 5). After incorporating PBRC, the precision of politician (91.19%) increases by 34.20%, and the recall of person (87.06%) improves by 64.68%, with F1 scores for both types increasing by 21.73% and 64.68%, respectively. Similarly, for scientist and person, as well as for album and song, similar results are observed. PBRC, informed by the confusion analysis of MCQM, incorporates bidirectional reasoning to guide the model to capture fine-grained entity features, effectively mitigating entity type confusion.

Unidirectional Confusion. After employing our method, chemicalelement shows improvements of over 40% in precision (88.24%), recall (65.22%), and F1 (75.00%), with improvements observed for chemicalpound. Musical instrument shows a 62.16% increase in precision (84.38%), a 59.53% increase in recall (64.29%), and a 65.13% increase in F1 (72.97%). Specifically, although the precision, recall, and F1 scores for misc also show improvement, the overall performance remains suboptimal, a trend also observed in other domains. Upon analysis, we attribute this to the inherent complexity of the misc type, which can be summarized by the following two factors: (1) misc consists of a more diverse set of entities, meaning its features


(a) literature

(b) AI

Figure 6: Error analysis on entity types under three different settings.

are more varied and complex, making it difficult for the model to learn effectively with limited samples; (2) the diversity of misc causes challenges in confusion analysis, making it difficult to pinpoint its most easily confused entity type.

C.4.2 Error Analysis on Entity Types

To evaluate the impact of our method on specific entity types, we report the total error proportion (including both missed entities and misclassified entities) and the type classification error proportion for each entity type under three settings: **w/o all, w/o backward reasoning, and ours**, as shown in Figure 6. We use Flan-T5-large as the backbone model and CoNLL2003 as the source domain. We observe that under the w/o backward reasoning setting, both domains show a significant decrease in the total error proportion and entity type classification error proportion for most entity types. Moreover, for some entity types, the ratio of entity type classification errors to total errors also drops considerably. This indicates that the reasoning information provided by the LLM effectively mitigates entity type confusion and improves the model's generalization ability in the entity type classification. When backward reasoning is further incorporated, the entity type classification error proportion continues to decrease for most entity types, confirming the effectiveness of the confusion analysis of MCQM. However, the event in the literature domain and the misc and algorithm in the AI domain do not show such improvements. Based on the confusion analysis of MCQM (Figure 7), we believe the absence of clear confusion types for these entity types causes this. In addition, the literarygenre in the literature domain does not exhibit notable reductions under either setting. We observe that the LLM performs poorly in entity span detection for this entity type, resulting in a large number of missed entities. Similarly, the backbone model also exhibits weak performance on this entity type, and no significant complementary advantage is observed between the two.

Method	sci.	pol.	mus.	lit.	AI
base(w/o all)	70.0	70.3	75.3	67.9	63.6
large(w/o all)	71.8	73.2	78.6	69.7	65.4
GPT-4o-mini	74.2	77.9	81.8	73.2	68.5
base(ours)	**81.4**	**83.5**	**83.7**	**79.0**	**74.3**
large(ours)	**82.3**	**85.0**	**85.6**	**80.0**	**76.0**

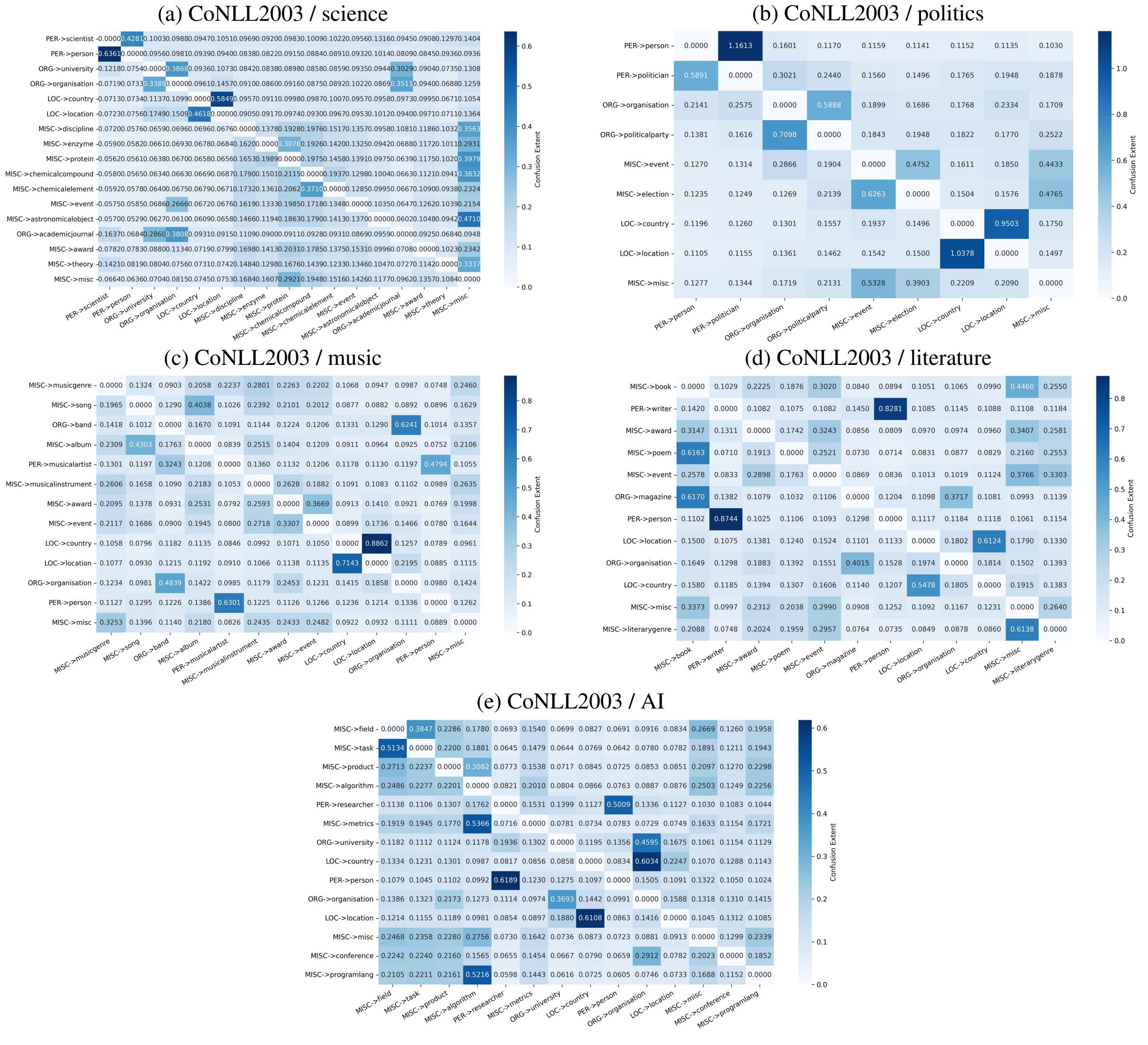
Table 7: Model performance (F1 score) comparison.

C.5 Individual vs. Combined Model

To evaluate the advantages of our approach, we present the results in Table 7, which compares the performance of fine-tuning the Flan-T5 model alone (w/o all), using the LLM for NER, and our method. We use CoNLL2003 as the source domain. For the LLM, we apply the same template (Figure 2) and provide 10 in-context learning examples, enabling it to generate NER results in the specified format. Experimental results show that, regardless of whether the backbone model is Flan-T5-base or Flan-T5-large, our method outperforms both the fine-tuned Flan-T5 and the method only based on the LLM in terms of F1 score. This demonstrates that our method effectively combines the advantages of fine-tuning the backbone model and the general-purpose LLM. General-purpose LLMs have advantages in parameter scale and knowledge breadth, as they are pretrained on a wide range of domain-specific corpora and possess some reasoning ability, which makes them particularly effective for low-resource entities. However, due to the hallucination phenomenon (Huang et al., 2025; Zhang et al., 2023), the LLM may introduce errors during the reasoning process. Although the reasoning path is often logically sound, the final output may still be incorrect, leading to degraded performance. On the other hand, Flan-T5, fine-tuned on the NER task, optimizes entity recognition by learning domain-specific context patterns through task-specific training data, and thus exhibits stronger performance on entity types with abundant training samples. However, when the training samples for certain entity types are scarce, the performance of Flan-T5 drops significantly.

Our method combines the broad knowledge of the LLM with the precise recognition capabilities of the fine-tuned T5 model, using the reasoning information generated by the LLM as supplementary features to significantly improve performance. The semantic cues and external knowledge provided by the LLM help the model better understand the context, especially for complex and low-resource entities. Meanwhile, through supervised fine-tuning, Flan-T5 learns how to effectively leverage the reasoning information generated by the LLM, mitigating the negative impact of hallucinations and incorrect reasoning information.

We provide several concrete examples: In Figure 8, the LLM supplies external knowledge about "Liberal International," enabling the Flan-T5 model to correctly classify this entity. In Figure 9, the LLM mistakenly identifies "Results of Astronomical Observations" as an entity, but the Flan-T5 model is not misled. In Figure 10, the LLM misses the entity "kernel methods," while Flan-T5 successfully recognizes it. In Figure 11, the LLM incorrectly classifies the entities as "location." However, since Flan-T5 is fine-tuned on the training dataset, it correctly classifies them as "country" based on the keyword "Empire". These cases demonstrate that our method effectively leverages the complementary strengths of LLMs and fine-tuned smaller models.



(a) CoNLL2003 / science

(b) CoNLL2003 / politics

(c) CoNLL2003 / music

(d) CoNLL2003 / literature

(e) CoNLL2003 / AI

Figure 7: Visualization of the confusion analysis results of MCQM for Flan-T5-large (CoNLL2003 as the source domain). For A→B, source type A is a prefix of target type B.

Text	The Freedom Party was subsequently expelled from the Liberal International , and the remaining liberals seceded to found the Liberal Forum (Liberales Forum , member Liberal International , Alliance of Liberals and Democrats for Europe Party) in 1993 .
Gold Label	"organisation": ["Liberal International", "Liberal International"], "politicalparty": ["Freedom Party", "Liberal Forum", "Liberales Forum", "Alliance of Liberals and Democrats for Europe Party"]
Two-stage Reasoning	**Initial Reasoning:** The phrase "was subsequently expelled from" suggests that "Liberal International" has a formal membership system, reinforcing its classification as an ORG. Similarly, "Freedom Party," "Liberales Forum," and "Alliance of Liberals and Democrats for Europe Party" are described as entities that were either expelled, seceded, or founded, implying they are organized groups. The phrase "the remaining liberals seceded to found" suggests an internal division within an existing organization, further indicating that these entities operate in a structured manner. Externally, these names are recognized as political groups, reinforcing their classification as ORG. **Bidirectional Reasoning:** Liberal International is classified as an organisation rather than a political party because it functions as an overarching entity that encompasses multiple political parties. The sentence describes Liberal International as being "expelled from," indicating it is an organized structure that other political parties can belong to, rather than a single political party itself. The absence of the term "Party" in its name and its role in representing multiple political parties also points to its classification as an organisation. Externally, Liberal International is recognized as a global federation of political parties, further confirming its classification as an organisation. The Freedom Party, Liberales Forum, and Alliance of Liberals and Democrats for Europe Party are classified as political parties rather than organisations because the sentence explicitly describes them as entities that were "founded" after seceding. This suggests they are newly established, independent political groups created to pursue political goals. The use of the term "Party" in their names further indicates their classification as political parties. The focus on political action and division from another group suggests they are political entities, rather than broader, non-political organisations.

Prediction	**1.Flan-t5-base(w/o):**
	"politicalparty": ["Freedom Party", "Liberal International", "Liberal Forum", "Liberales Forum", "Liberal International", "Alliance of Liberals and Democrats for Europe Party"] ✗
	2.Flan-t5-base(ours):
	"organisation": ["Liberal International", "Liberal International"], "politicalparty": ["Freedom Party", "Liberal Forum", "Liberales Forum", "Alliance of Liberals and Democrats for Europe Party"] √
	3.Flan-t5-large(w/o):
	"politicalparty": ["Freedom Party", "Liberal International", "Liberal Forum", "Liberales Forum", "Liberal International", "Alliance of Liberals and Democrats for Europe Party"] ✗
	4.Flan-t5-large(ours):
	"organisation": ["Liberal International", "Liberal International"], "politicalparty": ["Freedom Party", "Liberal Forum", "Liberales Forum", "Alliance of Liberals and Democrats for Europe Party"] √

Figure 8: A specific example in the politics domain.

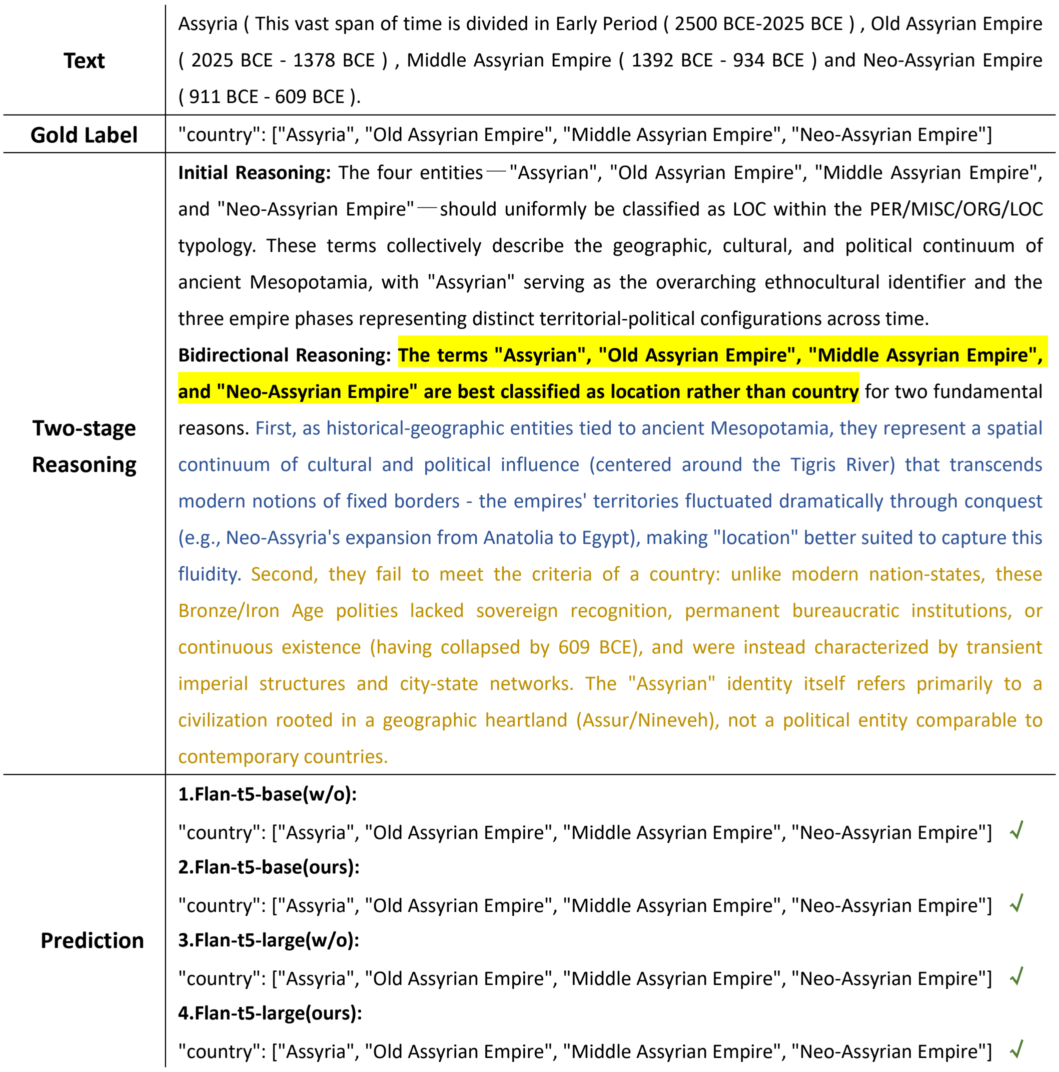
Text	The name was suggested by John Herschel (son of William Herschel , discoverer of Mimas and Enceladus) in his 1847 publication **Results of Astronomical Observations** made at the Cape of Good Hope , in which he advocated naming the moons of Saturn after the Titans , brothers and sisters of the Titan Cronus (whom the Romans equated with their god Saturn) .
Gold Label	"scientist": ["John Herschel", "William Herschel"], "person": ["Titan Cronus"], "location": ["Cape of Good Hope"], "astronomicalobject": ["Mimas", "Enceladus", "Saturn", "Titans", "Saturn"], "misc": ["Romans"]
Two-stage Reasoning	**Initial Reasoning:** John and William Herschel are astronomers, as mentioned in the context, and Titan Cronus is a mythological figure associated with Saturn, so all are classified as PER. The LOC includes the Cape of Good Hope, where John Herschel conducted observations, as stated in the text, making it a geographical location. **Results of Astronomical Observations** is a publication by John Herschel, classified as MISC. Romans refers to a cultural group. Mimas, Enceladus, Saturn and Titans are celestial objects mentioned in the context, so they are MISC. **Bidirectional Reasoning:** John Herschel and William Herschel are classified as scientists rather than persons because they are explicitly mentioned as astronomers in the context, with a connection to scientific research. John Herschel proposed naming Saturn's moons after the Titans, and William Herschel is mentioned as the discoverer of Mimas and Enceladus. Titan Cronus is classified as a person rather than a scientist because he is a figure from Greek mythology. In the text, Titan Cronus is referred to as a mythological character with no direct connection to astronomy or scientific activities. Cape of Good Hope is classified as a location rather than a country because it is a specific geographical site where John Herschel conducted astronomical observations. A country typically refers to a political entity, whereas Cape of Good Hope refers to a unique geographic location, so it is categorized as location. Mimas, Enceladus, Saturn, and Titans are classified as astronomical objects rather than miscellaneous because these entities are clearly defined astronomical objects. **Results of Astronomical Observations because it refer to is a publication by John Herschel.** Romans are classified as miscellaneous rather than protein because it refer to a cultural and ethnic group, while protein usually refers to biological molecules, which are unrelated to the subject of this text.
Prediction	**1.Flan-t5-base(w/o):** "scientist": ["John Herschel", "William Herschel"], "location": ["Cape of Good Hope"], "astronomicalobject": ["Mimas", "Enceladus"], "misc": ["Saturn", "Titans", "Saturn"] Unrecognized Entities: ("Romans", "Titan Cronus") ✗ **2.Flan-t5-base(ours):** "scientist": ["John Herschel", "William Herschel"], "person": ["Titan Cronus"], "location": ["Cape of Good Hope"], "astronomicalobject": ["Mimas", "Enceladus", "Saturn", "Titans", "Saturn"], "misc": ["Romans"] ✓ **3.Flan-t5-large(w/o):** "scientist": ["John Herschel", "William Herschel", "Titan Cronus"], "location": ["Cape of Good Hope"], "astronomicalobject": ["Mimas", "Enceladus"], "misc": ["Saturn", "Titans", "Saturn", "Romans"] ✗ **4.Flan-t5-large(ours):** "scientist": ["John Herschel", "William Herschel"], "person": ["Titan Cronus"], "location": ["Cape of Good Hope"], "astronomicalobject": ["Mimas", "Enceladus", "Saturn", "Titans", "Saturn"], "misc": ["Romans"] ✓

Figure 9: A specific example in the science domain.

Text	Application areas of **kernel methods** are diverse and include geostatistics , kriging , inverse distance weighting , 3D reconstruction , bioinformatics , chemoinformatics , information extraction and handwriting recognition .
Gold Label	"field": ["geostatistics", "bioinformatics", "chemoinformatics"], "task": ["3D reconstruction", "information extraction", "handwriting recognition"], "algorithm": [**"kernel methods"**, "kriging", "inverse distance weighting"]
Two-stage Reasoning	**Initial Reasoning:** The terms "geostatistics," "kriging," "inverse distance weighting," "3D reconstruction," "bioinformatics," "chemoinformatics," "information extraction," and "handwriting recognition" are classified as MISC because, within the context of the sentence, they represent specific methods, tasks, or specialized areas of study. The phrase "application areas of kernel methods are diverse" signals that the listed terms are various areas and techniques that are applied in different domains, reinforcing their classification as MISC. **Bidirectional Reasoning:** Geostatistics, 3D reconstruction, bioinformatics, and chemoinformatics are classified as fields rather than tasks because they represent broad academic disciplines or research areas that involve a range of methods, techniques, and applications. These are not specific tasks, but rather comprehensive fields encompassing various activities and approaches. information extraction and handwriting recognition are classified as tasks because they focus on specific, well-defined operations within broader domains. Information extraction refers to the task of identifying and extracting structured data from unstructured text, and handwriting recognition is the task of converting handwritten text into a machine-readable format. These are not fields because they do not represent broad areas of research or study. Instead, they are narrow, specific processes within the larger fields of natural language processing and pattern recognition, respectively. They are targeted actions with clearly defined objectives, which distinguishes them from fields that are more expansive in scope. Kriging and inverse distance weighting are classified as algorithms rather than misc because they are defined computational methods used to solve specific problems, particularly in the field of geostatistics. These techniques are structured, precise methods, distinguishing them from miscellaneous items, which typically lack a clear, specialized purpose.

Prediction	**1.Flan-t5-base(w/o):**
	"field": ["geostatistics", "kriging", "inverse distance weighting", "3D reconstruction", "bioinformatics", "chemoinformatics", "information extraction", "handwriting recognition"], "algorithm": [**"kernel methods"**] ✗
	2.Flan-t5-base(ours):
	"field": ["geostatistics", "3D reconstruction", "handwriting recognition", "bioinformatics", "chemoinformatics"], "task":["information extraction"], "algorithm": [**"kernel methods"**, "kriging", "inverse distance weighting"] ✗
	3.Flan-t5-large(w/o):
	"field": ["geostatistics", "kriging", "inverse distance weighting", "3D reconstruction", "bioinformatics","chemoinformatics"], "task":["information extraction", "handwriting recognition"], "algorithm": [**"kernel methods"**] ✗
	4.Flan-t5-large(ours):
	"field": ["geostatistics", "bioinformatics", "chemoinformatics"], "task": ["3D reconstruction", "information extraction", "handwriting recognition"], "algorithm": [**"kernel methods"**, "kriging", "inverse distance weighting"] √

Figure 10: A specific example in the AI domain.



Text	Assyria (This vast span of time is divided in Early Period (2500 BCE-2025 BCE) , Old Assyrian Empire (2025 BCE - 1378 BCE) , Middle Assyrian Empire (1392 BCE - 934 BCE) and Neo-Assyrian Empire (911 BCE - 609 BCE).
Gold Label	"country": ["Assyria", "Old Assyrian Empire", "Middle Assyrian Empire", "Neo-Assyrian Empire"]
Two-stage Reasoning	**Initial Reasoning:** The four entities—"Assyrian", "Old Assyrian Empire", "Middle Assyrian Empire", and "Neo-Assyrian Empire"—should uniformly be classified as LOC within the PER/MISC/ORG/LOC typology. These terms collectively describe the geographic, cultural, and political continuum of ancient Mesopotamia, with "Assyrian" serving as the overarching ethnocultural identifier and the three empire phases representing distinct territorial-political configurations across time. **Bidirectional Reasoning:** **The terms "Assyrian", "Old Assyrian Empire", "Middle Assyrian Empire", and "Neo-Assyrian Empire" are best classified as location rather than country** for two fundamental reasons. First, as historical-geographic entities tied to ancient Mesopotamia, they represent a spatial continuum of cultural and political influence (centered around the Tigris River) that transcends modern notions of fixed borders - the empires' territories fluctuated dramatically through conquest (e.g., Neo-Assyria's expansion from Anatolia to Egypt), making "location" better suited to capture this fluidity. Second, they fail to meet the criteria of a country: unlike modern nation-states, these Bronze/Iron Age polities lacked sovereign recognition, permanent bureaucratic institutions, or continuous existence (having collapsed by 609 BCE), and were instead characterized by transient imperial structures and city-state networks. The "Assyrian" identity itself refers primarily to a civilization rooted in a geographic heartland (Assur/Nineveh), not a political entity comparable to contemporary countries.

Prediction	**1.Flan-t5-base(w/o):**
	"country": ["Assyria", "Old Assyrian Empire", "Middle Assyrian Empire", "Neo-Assyrian Empire"] √
	2.Flan-t5-base(ours):
	"country": ["Assyria", "Old Assyrian Empire", "Middle Assyrian Empire", "Neo-Assyrian Empire"] √
	3.Flan-t5-large(w/o):
	"country": ["Assyria", "Old Assyrian Empire", "Middle Assyrian Empire", "Neo-Assyrian Empire"] √
	4.Flan-t5-large(ours):
	"country": ["Assyria", "Old Assyrian Empire", "Middle Assyrian Empire", "Neo-Assyrian Empire"] √

Figure 11: A specific example in the politics domain.