A Simple but Effective Approach to Improve Structured Language Model Output for Information Extraction

Anonymous ACL submission

Abstract

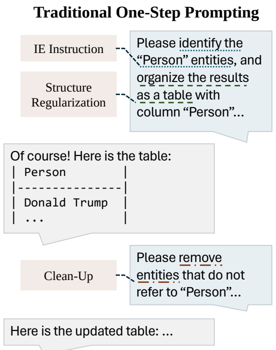
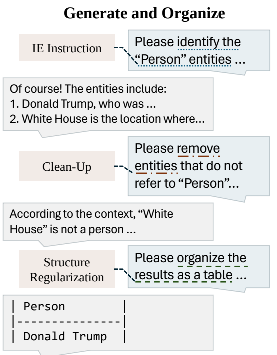
Large language models (LLMs) have demonstrated impressive abilities in generating unstructured natural language according to instructions. However, their performance can be inconsistent when tasked with producing text that adheres to specific structured formats, which is crucial in applications like named entity recognition (NER) or relation extraction (RE). To address this issue, this paper introduces an efficient method, G&O, to enhance their structured text generation capabilities. It breaks the generation into a two-step pipeline: initially, LLMs generate answers in natural language as intermediate responses. Subsequently, LLMs are asked to organize the output into the desired structure, using the intermediate responses as context. G&O effectively separates the generation of content from the structuring process, reducing the pressure of completing two orthogonal tasks simultaneously. Tested on zero-shot NER and RE, the results indicate a significant improvement in LLM performance with minimal additional efforts. This straightforward and adaptable prompting technique can also be combined with other strategies, like self-consistency, to further elevate LLM capabilities in various structured text generation tasks.

1 Introduction

Information extraction (IE) is a critical task that involves retrieving specific information, such as named entities and relationships, from unstructured or semi-structured texts, and converting this information into a structured format (Cowie and Lehnert, 1996; Li et al., 2023a). Traditionally, IE models have relied heavily on fully supervised learning, necessitating extensive labeled datasets for training. This approach not only demands significant human effort but also restricts the scope of extractable information to a limited set of predefined types, such as “`person`” for Named Entity Recognition (NER) and “`born in`” for Relation Extraction (RE). This limitation is particularly prominent in specialized fields like materials science, where resources are scarce. Earlier attempts to mitigate these challenges, such as weak supervision (Ren et al., 2020; Liang et al., 2020; Zhang et al., 2021; Li et al., 2022), have introduced methods that utilize noisy heuristic labeling functions (LFs) to reduce the reliance on manually labeled data. However, the effectiveness of these methods often hinges on the quality of the LFs, which is not always consistent.

The advent of Large Language Models (LLMs) like GPTs (OpenAI, 2022, 2023) has promoted an attention shift towards universal IE approaches. These methods aim to extract a wide range of information without the need for task-specific labels. Strategies include directly prompting LLMs with instructions for specific tasks (Wang et al., 2023a; Han et al., 2023; Xie et al., 2023; Zhang et al., 2023) or fine-tuning them on either true labels or pseudo labels generated by GPTs (Wang et al., 2023b; Zhou et al., 2023; Sainz et al., 2023; Zaratiana et al., 2023; Jiao et al., 2023). Nonetheless, the inherent mismatch between the unstructured data language models are typically trained on and the structured output requirement presents a challenge. Previous studies have employed specialized prompts to guide the model in generating structured outputs, such as lists (Zhou et al., 2023) or tables (Jiao et al., 2023). However, integrating task instructions with these organizational prompts has sometimes resulted in formatting issues or compromised IE performance.

In response to these challenges, this paper introduces a simple but effective methodology, Generate and Organize (G&O), designed to enhance the capability of LLMs in performing structured zero-shot IE tasks, with a focus on NER and RE. Our approach divides the generation into two distinct components: 1) generating IE responses in a free-form natural language format; followed by 2) structuring these responses into a predefined for-

Please identify the "Person" entities, and organize the results as a table with column "Person"...

Of course! Here is the table:

```
| Person         |
|----------------|
| Donald Trump   |
| ...            |
```

Please remove entities that do not refer to "Person"...

Here is the updated table: ...

Please identify the "Person" entities ...

Of course! The entities include:
1. Donald Trump, who was ...
2. White House is the location where...

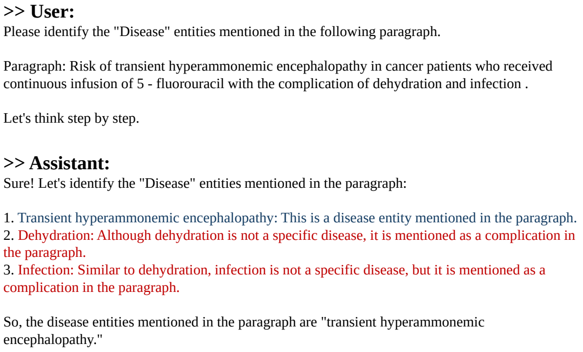
Please remove entities that do not refer to "Person"...

According to the context, "White House" is not a person ...

Please organize the results as a table ...

```
| Person         |
|----------------|
| Donald Trump   |
```

Figure 1: The pipeline of G&O for NER, compared with Traditional One-Step prompting methods.



>> User:
Please identify the "Disease" entities mentioned in the following paragraph.

Paragraph: Risk of transient hyperammonemic encephalopathy in cancer patients who received continuous infusion of 5 - fluorouracil with the complication of dehydration and infection .

Let's think step by step.

>> Assistant:
Sure! Let's identify the "Disease" entities mentioned in the paragraph:

1. Transient hyperammonemic encephalopathy: This is a disease entity mentioned in the paragraph.
2. Dehydration: Although dehydration is not a specific disease, it is mentioned as a complication in the paragraph.
3. Infection: Similar to dehydration, infection is not a specific disease, but it is mentioned as a complication in the paragraph.

So, the disease entities mentioned in the paragraph are "transient hyperammonemic encephalopathy."

Figure 2: GPT-3.5's natural language responses tend to include irrelevant entities (marked by red). Although clearly explained, irrelevant terms still pose a difficulty for GPT-3.5 during format organization.

mat. Furthermore, we incorporate a clean-up component to eliminate any potential noise from the free-form responses before structuring. Through extensive experimentation, we demonstrate that our method boosts zero-shot IE performance across various LLMs. Additionally, we show that each component of our approach contributes to its overall effectiveness. Beyond NER and RE, G&O is versatile enough to be integrated with other techniques, such as self-consistency (Wang et al., 2023c), and can be applied to a broad spectrum of tasks requiring formatted outputs. To support further research, we have made our methodology, including the code and experimental results, publicly available at nonymous.4open.science/r/GnO-IE-F44C/.

2 Generate and Organize

To enhance the capability of LLMs on zero-shot IE tasks that necessitate structured outputs, our prompting pipeline integrates three key components, as depicted in Figure 1: 1) *free-form response generation*, which prompts LLMs to identify the required information from the provided context without imposing any syntactic or structural constraints on the result; 2) *answer clean-up*, tailored to the specific task at hand, filters out extraneous information to maintain the integrity of the final structured output; and 3) *structure organization*, which is responsible for transforming the unstructured responses into organized formats, such as Markdown tables or lists, based on the LLMs' prior responses within the conversation history. In addition, we add zero-shot CoT (Kojima et al., 2022) to further improve the IE performance.

Although our modification appears minor compared to traditional IE prompts that combine components 1 and 3, it enhances alignment with the inherent semantic progression of natural language, and yields responses that are both more coherent and informative, according to the theory of Xie et al. (2022). In addition, clean-up also plays a crucial role. As illustrated in Figure 2, our empirical analysis reveals that while models efficiently identify relevant entities or relationships, they often include unrelated information that does not pertain to the requested types. This phenomenon largely stems from the models' training to be "helpful" through RLHF (Ouyang et al., 2022). Despite the identification of irrelevant entities, their presence complicates the task of formatting the useful information during the structure organization phase. Hence, the clean-up phase is crucial to ensuring that the output is concise and focused solely on the entities of interest. In the final step, we opt for Markdown tables as the structured format due to their prevalence in LLM training datasets and to maintain consistency with the RE pipeline.

3 Experiment and Discussion

3.1 Named Entity Recognition

Datasets Our research utilizes diverse NER datasets , including CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003) from the general domain, NCBI Disease (Dogan et al., 2014) and BC5CDR (Li et al., 2016) from the biomedical sector, and PolyIE (Cheung et al., 2023) from the field of materials science. Please refer to appendix B.2 for statistics and details on data processing.

Baselines A fundamental baseline of G&O is **One-Step** prompting, which consolidates identification and organization into a single prompt. While there are variations in implementation, this

	CoNLL 2003		BC5CDR		NCBI Disease		PolyIE		Macro Average	
	Partial	Full	Partial	Full	Partial	Full	Partial	Full	Partial	Full
AEiO	0.5370	0.4965	0.6199	0.5058	-	-	0.1300	0.0935	0.4290	0.3653
One-Step	0.4741	0.4477	0.7030	0.6041	0.6500	0.5131	0.4669	0.3207	0.5735	0.4714
G&O-NER	0.6569	0.6192	0.7610	0.6079	0.6935	0.5047	0.5449	0.3823	0.6641	0.5285
− CoT	0.6572	0.6079	0.6634	0.5544	0.5653	0.4059	0.4551	0.3068	0.5853	0.4688
− clean-up	0.7003	0.6436	0.7421	0.5861	0.6475	0.4541	0.5103	0.3421	0.6501	0.5065
+ CR	0.6775	0.6394	0.7724	0.6186	-	-	0.6011	0.4236	0.6837	0.5605
+ FT	0.7175	0.6800	0.7949	0.6838	0.7703	0.5507	0.7608	0.5533	0.7609	0.6170

Table 1: The F_1 scores of GPT-3.5 on the NER datasets with different prompting strategies. "Partial" and "Full" refer to the partial and full matching criteria; "+" and "−" indicate the addition and removal of the corresponding components. "CR" stands for Conflict Resolution, and "FT" for BERT fine-tuning.

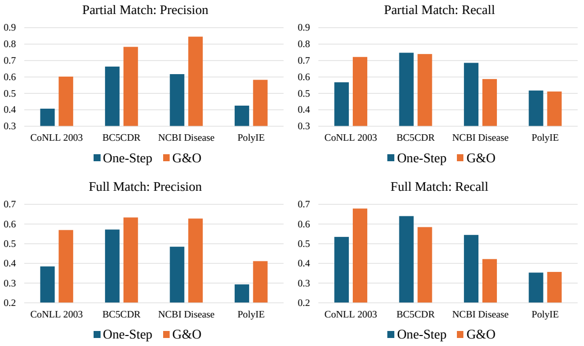


Figure 3: Comparing the precision and recall of G&O-NER with One-Step on NER datasets.

method is currently the dominant approach among LLMs for tasks demanding structured outputs. We also consider another straightforward benchmark termed All-Entity-in-One (**AEiO**), which instructs the model to concurrently identify entities of various types, *e.g.*, "Identify `person`, `location`, and `organization` entities within the given paragraph". Note that One-Step also incorporates a clean-up phase, and AEiO differs from G&O primarily in the number of entity types it addresses. Please refer to appendix B.3 and B.4 for details.

Metrics and Evaluation We employ micro-averaged precision, recall, and F_1 score as metrics. However, the strict span-level matching criterion disproportionately penalizes predictions that overlap with the ground truth without matching exactly, as observed by Zhou et al. (2023). Therefore, we use both **partial** and **full** matching scores. The former acknowledges overlapping spans as true positives, whereas the latter demands identical entities. Please refer to appendix B.6 for more details.

Main Results As our main NER results, Table 1 presents the F_1 scores achieved by GPT-3.5 5 using various prompting strategies. The effectiveness of G&O is evident when compared against the One-Step approach, where G&O-NER is superior in nearly all datasets under both partial and full matching criteria. On average, the separation of task instruction and organization prompts yields a 15.8% increase in partial-match F_1 and a 12.1% improvement in full. Furthermore, the comparison with the AEiO baseline highlights the benefits of entity-specific instructions on information extraction performance. Analysis of Figure 3 reveals that G&O-NER consistently boosts precision relative to the One-Step method without significantly affecting recall on average. It shows that the opportunity to explain its results, facilitated by the CoT process, encourages GPT-3.5 to produce more precise final outputs through self-verification and clean-up.

Ablation Studies To assess the contribution of key elements within our approach, we conduct two straightforward ablation studies: excluding the CoT prompting and omitting the cleanup process. Results in Table 1 show that the former leads to a 11.87% partial F_1 drop on average and the latter 2.11%. On a dataset-specific basis, these features exhibit minimal or even adverse effects on the CoNLL 2003 dataset. However, they play a pivotal role in enhancing performance on scientific datasets. As discussed in § 2, LLMs are prone to integrating discussions about irrelevant scientific terms in their responses, a tendency less prevalent with general entities such as person names in the CoNLL dataset. Moreover, the encouragement for models to articulate responses in natural language proves more advantageous for scientific datasets, where entities tend to be more complex and varied.

Resolving Entity Type Conflict Given G&O-NER processes each entity type separately, a no-

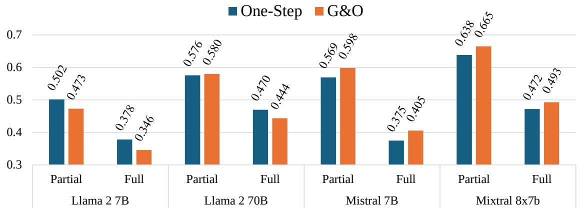


Figure 4: F_1 scores of differnt LMs with G&O and One-Step promptings, macro-averaged on the all datasets.

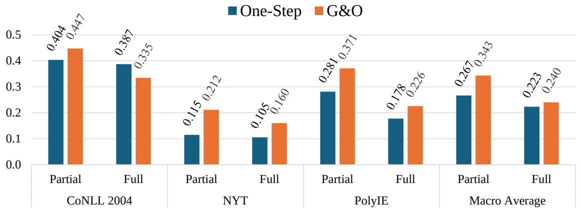


Figure 5: The F_1 scores of GPT-3.5 with different prompting approaches on RE datasets.

table challenge is the potential for a single entity span to be categorized under multiple types. To mitigate this issue, we implement two strategies: 1) Conflict Resolution (**CR**), prompts LLMs to resolve any conflict of entity types as it arise; and 2) BERT Fine-Tuning (**FT**), which entails the fine-tuning of a pre-trained Transformer encoder (Devlin et al., 2019) using pseudo labels generated by GPT. Detailed setup is provided in appendix B.3. As indicated in Table 1, both approaches enhance the overall effectiveness of G&O-NER, with FT being superior. FT not only addresses the type conflict issue but also acts as a filter that discerns high-level entity patterns from the pseudo labels. This process effectively refines the GPT-generated outputs by eliminating random inaccuracies.

Other LLMs In exploring the adaptability of our approach with various LLMs, we extended G&O to 4 open-source LLMs, including Llama 2 7B/70B (Touvron et al., 2023), Mistral 7B (Jiang et al., 2023), and Mixtral 8x7B (Jiang et al., 2024), specifically their chat/instruct variants. As depicted in Figure 4, the impact of G&O is less pronounced with these LLMs compared to GPT-3.5, indicating a dependency on the models' capacity for reasoning and following instructions. Notably, Llama 2 models rarely produce explanations for their outputs, which renders G&O virtually equivalent to One-Step prompting, albeit less robust due to an increased likelihood of error propagation. Conversely, G&O effectively encourages Mi[s/x]tral to provide detailed explanations in natural language, achieving a more consistent enhancement over the One-Step approach. It can be concluded that G&O is more suited for LLMs that are designed with a focus on reasoning abilities and the capacity to engage in multi-round conversations.

3.2 Relation Extraction

In our study on RE, we evaluate 3 datasets: CoNLL 2004 (Roth and Yih, 2004), NYT (Zeng et al., 2018), and PolyIE. Our RE approach is end-to-end, predicting entities and their relations within the same iteration, which mirrors real-world scenarios and presents a greater challenge. We primarily assess G&O-RE against the One-Step method using GPT-3.5, focusing on partial and full match precision, recall, and F_1 scores. Elaboration on G&O-RE is provided in appendix B.4.

Figure 5 illustrates that G&O enhances GPT-3.5's performance on RE tasks, registering an average F_1 score improvement of 28.5% for partial matches and 7.6% for full matches. Notably, across both NER and RE tasks, enhancements in partial matches consistently surpass those in full matches. Analysis of LLM responses reveals that G&O tends to produce longer entity descriptions that incorporate attributes and modifiers not always present in the original annotations. When applied to other LLMs, G&O-RE maintains consistent performance boosts, underscoring its versatility and applicability across diverse IE tasks.

4 Conclusion

In this paper, we propose a simple yet effective approach—G&O—to improve structured prediction from LLMs for IE tasks. Different from conventional prompting approaches, G&O separates the identification and formatting steps into two stages, which allows the model to focus on each step independently and facilitates the generation of organized results. Tested under the zero-shot IE settings with GPT 3.5, this simple adjustment brings significant performance gains, demonstrating the effectiveness of G&O. The improvement is further validated by ablation studies and the generalizability of G&O to other LLMs, which can be further improved by resolving prediction conflicts using both prompting and fine-tuning methods. We hope our work can bring insights and inspirations for further research on structured prediction from LLMs and contribute to the development of more effective and interpretable IE systems.

Limitations

Given the limitations of our computational resources, our evaluation was conducted on a select number of datasets and tasks. While these experiments effectively illustrate the efficacy of G&O, we recognize that incorporating additional datasets and tasks could enhance the robustness of our conclusions. Moreover, we exclusively utilized Markdown tables for structuring the final output, aiming for consistency across NER and RE tasks. However, we did not investigate alternative formats such as lists or JSON, which are potentially more compatible with GPT models. Such investigations are earmarked for future research endeavors.

Another intriguing aspect, not covered in this study, is the potential for fine-tuning open-source LLMs to align with our prompting format through methods like supervised fine-tuning and reinforcement learning. We hypothesize that such an approach could significantly augment the zero-shot capabilities of universal IE tasks. Addressing these areas of interest remains a key objective for our subsequent research efforts.

References

Zhijun Chen, Hailong Sun, Wanhao Zhang, Chunyi Xu, Qianren Mao, and Pengpeng Chen. 2023. Neural-hidden-crf: A robust weakly-supervised sequence labeler. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023*, pages 274–285. ACM.

Jerry Junyang Cheung, Yuchen Zhuang, Yinghao Li, Pranav Shetty, Wantian Zhao, Sanjeev Grampurohit, Rampi Ramprasad, and Chao Zhang. 2023. Polyie: A dataset of information extraction from polymer material scientific literature. *CoRR*, abs/2311.07715.

Jim Cowie and Wendy Lehnert. 1996. Information extraction. *Commun. ACM*, 39(1):80–91.

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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Rezarta Islamaj Dogan, Robert Leaman, and Zhiyong Lu. 2014. NCBI disease corpus: A resource for disease name recognition and concept normalization. *J. Biomed. Informatics*, 47:1–10.

Yucan Guo, Zixuan Li, Xiaolong Jin, Yantao Liu, Yutao Zeng, Wenxuan Liu, Xiang Li, Pan Yang, Long Bai, Jiafeng Guo, and Xueqi Cheng. 2023. Retrieval-augmented code generation for universal information extraction. *CoRR*, abs/2311.02962.

Ridong Han, Tao Peng, Chaohao Yang, Benyou Wang, Lu Liu, and Xiang Wan. 2023. Is information extraction solved by chatgpt? an analysis of performance, evaluation criteria, robustness and errors. *CoRR*, abs/2305.14450.

Xu Han, Tianyu Gao, Yuan Yao, Deming Ye, Zhiyuan Liu, and Maosong Sun. 2019. Opennre: An open and extensible toolkit for neural relation extraction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019 - System Demonstrations*, pages 169–174. Association for Computational Linguistics.

Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. Fewrel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4803–4809. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts. *CoRR*, abs/2401.04088.

Yizhu Jiao, Ming Zhong, Sha Li, Ruining Zhao, Siru Ouyang, Heng Ji, and Jiawei Han. 2023. Instruct and extract: Instruction tuning for on-demand information extraction. In *Proceedings of the 2023*

Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023, pages 10030–10051. Association for Computational Linguistics.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Ouyu Lan, Xiao Huang, Bill Yuchen Lin, He Jiang, Liyuan Liu, and Xiang Ren. 2020. Learning to contextually aggregate multi-source supervision for sequence labeling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2134–2146. Association for Computational Linguistics.

Hunter Lang, Aravindan Vijayaraghavan, and David A. Sontag. 2022. Training subset selection for weak supervision. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

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

Yinghao Li, Colin Lockard, Prashant Shiralkar, and Chao Zhang. 2023a. Extracting shopping interest-related product types from the web. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 7509–7525. Association for Computational Linguistics.

Yinghao Li, Pranav Shetty, Lucas Liu, Chao Zhang, and Le Song. 2021. Bertifying the hidden markov model for multi-source weakly supervised named entity recognition. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 6178–6190. Association for Computational Linguistics.

Yinghao Li, Le Song, and Chao Zhang. 2022. Sparse conditional hidden markov model for weakly supervised named entity recognition. In *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, pages 978–988. ACM.

Yongqi Li, Yu Yu, and Tieyun Qian. 2023b. Type-aware decomposed framework for few-shot named entity recognition. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 8911–8927. Association for Computational Linguistics.

Chen Liang, Yue Yu, Haoming Jiang, Siawpeng Er, Ruijia Wang, Tuo Zhao, and Chao Zhang. 2020. BOND: bert-assisted open-domain named entity recognition with distant supervision. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1054–1064. ACM.

Pierre Lison, Jeremy Barnes, and Aliaksandr Hubin. 2021. skweak: Weak supervision made easy for NLP. In *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL 2021 - System Demonstrations, Online, August 1-6, 2021*, pages 337–346. Association for Computational Linguistics.

Pierre Lison, Jeremy Barnes, Aliaksandr Hubin, and Samia Touileb. 2020. Named entity recognition without labelled data: A weak supervision approach. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 1518–1533. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Tingting Ma, Huiqiang Jiang, Qianhui Wu, Tiejun Zhao, and Chin-Yew Lin. 2022. Decomposed meta-learning for few-shot named entity recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1584–1596, Dublin, Ireland. Association for Computational Linguistics.

OpenAI. 2022. Introducing ChatGPT. (Accessed on Jun 18, 2023).

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022,* *NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. 2018. Learning to reweight examples for robust deep learning. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 4331–4340. PMLR.

Wendi Ren, Yinghao Li, Hanting Su, David Kartchner, Cassie Mitchell, and Chao Zhang. 2020. Denoising multi-source weak supervision for neural text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3739–3754, Online. Association for Computational Linguistics.

Dan Roth and Wen-tau Yih. 2004. A linear programming formulation for global inference in natural language tasks. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, pages 1–8, Boston, Massachusetts, USA. Association for Computational Linguistics.

Esteban Safranchik, Shiying Luo, and Stephen H. Bach. 2020. Weakly supervised sequence tagging from noisy rules. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 5570–5578. AAAI Press.

Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, and Eneko Agirre. 2023. Gollie: Annotation guidelines improve zero-shot information-extraction. *CoRR*, abs/2310.03668.

Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2895–2905. 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.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023a. GPT-NER: named entity recognition via large language models. *CoRR*, abs/2304.10428.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, Jihua Kang, Jingsheng Yang, Siyuan Li, and Chunsai Du. 2023b. Instructuie: Multi-task instruction tuning for unified information extraction. *CoRR*, abs/2304.08085.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023c. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An explanation of in-context learning as implicit bayesian inference. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Tingyu Xie, Qi Li, Jian Zhang, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2023. Empirical study of zero-shot NER with chatgpt. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 7935–7956. 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.

Yue Yu, Simiao Zuo, Haoming Jiang, Wendi Ren, Tuo Zhao, and Chao Zhang. 2021. Fine-tuning pre-trained language model with weak supervision: A contrastive-regularized self-training approach. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1063–1077, Online. Association for Computational Linguistics.

Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. 2023. Gliner: Generalist model for named entity recognition using bidirectional transformer. *CoRR*, abs/2311.08526.

Xiangrong Zeng, Daojian Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018. Extracting relational facts by an end-to-end neural model with copy mechanism. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 506–514, Melbourne, Australia. Association for Computational Linguistics.

Jieyu Zhang, Yue Yu, Yinghao Li, Yujing Wang, Yaming Yang, Mao Yang, and Alexander Ratner. 2021. WRENCH: A comprehensive benchmark for weak supervision. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Kai Zhang, Bernal Jimenez Gutierrez, and Yu Su. 2023. Aligning instruction tasks unlocks large language models as zero-shot relation extractors. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 794–812. Association for Computational Linguistics.

Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, and Hoifung Poon. 2023. Universalner: Targeted distillation from large language models for open named entity recognition. *CoRR*, abs/2308.03279.

Dawei Zhu, Xiaoyu Shen, Marius Mosbach, Andreas Stephan, and Dietrich Klakow. 2023. Weaker than you think: A critical look at weakly supervised learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 14229–14253. Association for Computational Linguistics.

A Related Works

In the landscape of supervised neural networks, researchers are actively seeking methods to reduce reliance on labeled data for Information Extraction (IE) tasks, acknowledging the effort and limitations associated with manually labeled data. A prominent approach includes the development of weak and distant supervision techniques (Liang et al., 2020; Lan et al., 2020; Lison et al., 2020, 2021; Zhang et al., 2021; Yu et al., 2021; Li et al., 2021, 2022; Chen et al., 2023). These methods aim to lessen the annotation workload through the use of heuristic labeling functions (LFs). These functions, whether singular (Ren et al., 2018; Liang et al., 2020; Yu et al., 2021) or multiple (Lison et al., 2020; Li et al., 2021, 2022; Chen et al., 2023; Lang et al., 2022), are designed to generate noisy labels for unlabeled data. Subsequently, models are trained to refine and amalgamate these labels for improved prediction accuracy. However, some critics argue that the efficacy of single-LF methods is highly dependent on the quality of the clean validation set (Zhu et al., 2023), and the creation of multiple LFs can be a labor-intensive process (Safranchik et al., 2020; Lison et al., 2021).

Another research trajectory involves few-shot and zero-shot learning techniques (Han et al., 2018, 2019; Soares et al., 2019; Yang and Katiyar, 2020; Ma et al., 2022; Li et al., 2023b). These methods are directed towards adapting IE models to new domains using minimal labeled examples. In line with the rapid advancements in Large Language Models (LLMs), some studies have explored directly prompting these models for open-type IE tasks (Wang et al., 2023a; Han et al., 2023; Xie et al., 2023; Guo et al., 2023). Additionally, there is an emerging focus on fine-tuning generative LLMs to better align with specific prompts or task formats (Zhou et al., 2023; Zhang et al., 2023; Sainz et al., 2023; Jiao et al., 2023). Nonetheless, these studies have overlooked the issue of LLMs' suboptimal performance in structured prediction tasks when using mixed prompts, which is the central topic of our research.

B Detailed Experiment Setup

B.1 Models

Our research primarily centers on GPT-3.5, particularly the `gpt-3.5-turbo-0613` version, as documented by (OpenAI, 2022).[1]. Despite not being the most current iteration, we have opted to continue using this version to ensure the continuity of our experimental work. In terms of open-source LLMs, our selection includes Llama 2 7B (`Llama-2-7b-chat-hf`,[2] Touvron et al., 2023), Llama 2 70B (`Llama-2-70b-chat-hf`,[3] Touvron et al., 2023), Mistral 7B (`Mistral-7B-Instruct-v0.2`,[4] Jiang et al., 2023), and Mixtral 8x7B (`Mixtral-8x7B-Instruct-v0.1`,[5] Jiang et al., 2024). Our experimental procedure involves only forward inference without any model fine-tuning. The inference process for GPT 3.5 utilizes the OpenAI API via Azure, whereas the open-source LLMs are run using HuggingFace Transformers library (Wolf et al., 2019) and vllm (Kwon et al., 2023). The deployment of Llama 2 7B and Mistral 7B is each on an individual NVIDIA A100 80G GPU, while Mixtral 8x7B and Llama 2 70B are deployed on two GPUs each.

	CoNLL 03	BC5CDR	NCBI	PolyIE
n-instance	3,453	1,000	940	96 / 1,170
avg. l-text	70	148	147	2,761 / 188
n-entity-type	3	2	1	3
n-entity-mention	4,945	2,074	957	4,803

Table 2: NER dataset statistics. "avg. l-text" denotes the average number of characters in each text instance. The statistics of PolyIE is shown as "Paragraph-level / Sentence-level".

B.2 Datasets

In the NER task, we incorporate datasets from several sources: CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003), NCBI Disease (Dogan et al., 2014), BC5CDR (Li et al., 2016), and PolyIE (Cheung et al., 2023). The CoNLL 2003, NCBI Disease, and BC5CDR datasets are obtained as prepared by Wang et al. (2023b), while the PolyIE dataset is sourced directly from Cheung et al. (2023). We apply minor modifications to these datasets to tailor them to our study's needs. Specifically, for CoNLL 2003, we remove the "`MISC`" entities due to their lack of informativeness and rare usage in practical scenarios. In the case of PolyIE, "`Condition`" entities are excluded owing to the ambiguity surrounding their definition. For BC5CDR, we limit

---

[1] platform.openai.com/docs/models/gpt-3-5-turbo

[2] huggingface.co/meta-llama/Llama-2-7b-chat-hf

[3] huggingface.co/meta-llama/Llama-2-70b-chat-hf

[4] huggingface.co/mistralai/Mistral-7B-Instruct-v0.2

[5] huggingface.co/mistralai/Mixtral-8x7B-Instruct-v0.1

	CoNLL 04	**NYT**	**PolyIE**
n-instance	288	369	96
avg. l-text	159	199	2,761
n-relation-type	5	7	1
n-ary-relations	2	2	3
n-relation-mention	422	265	527

Table 3: RE dataset statistics. “n-ary-relations” indicates the number of entities in a relation tuple (group).

our testing to 1,000 randomly selected samples to minimize computational demands. Preliminary experiments indicate that model performance on these subsets aligns with results obtained using the full datasets. Furthermore, we adapt the input context for PolyIE to account for differences in the capabilities of GPT-3.5 and other language models. For GPT-3.5, we input entire paragraphs, whereas for other LLMs, we broke the paragraphs into sentences and processed them individually, acknowledging their constrained history comprehension and memory abilities. We report the performance on the test set of each dataset, and the detailed statistics are shown in Table 2.

In terms of RE, we utilize CoNLL 2004 (Roth and Yih, 2004), NYT (Zeng et al., 2018), and PolyIE (Cheung et al., 2023). Similar to NER, we obtain CoNLL 2004 and NYT from Wang et al. (2023b) and PolyIE from Cheung et al. (2023). The infrequent relation types are also removed from the datasets to ensure a more focused and affordable evaluation. For PolyIE, we only keep the “`Material-Property-Value`” relations and test all models using paragraph-based instances as the relations usually span multiple sentences. The statistics of the RE datasets are shown in Table 3.

B.3 NER Implementation Details

G&O and Baselines Our experiments focus on revealing the difference between the zero-shot performance of LLMs when prompted with G&O and One-Step. Therefore, we use the same prompt for both methods to ensure a fair comparison, except for the position of structure organization instruction. Specifically, for the NER task, G&O-NER prompts are designed as Listing 1:

Listing 1: G&O-NER prompts.

```
>> SYSTEM PROMPT
You are a knowledgeable assistant specialized in
    recognizing and understanding named entities
    and their interrelations. If requested to
    organize information in tabular format, you are
     adept at filtering and presenting only the
    relevant and valid results. You will exclude
    any results that are not pertinent or are
    inaccurate from the table according to the
    discussion history.

>> USER PROMPT  # Step 1. Free-form response
    generation
Please identify the "<ENTITY TYPE>" entities in the
    following paragraph.

Paragraph: <PARAGRAPH>

# optional zero-shot CoT prompt
Let's think step by step.

>> ASSISTANT ANSWER

# varies from case to case, omitted

>> USER PROMPT  # Step 2. Clean-up (optional)

Please remove irrelevant entities and only keep the
    entities that clearly refer to <ENTITY TYPE>.

>> ASSISTANT ANSWER

# varies from case to case, omitted

>> USER PROMPT  # Step 3. Structure organization

Please present the valid entities as a Markdown
    table with one column "<ENTITY TYPE>".

Make sure to present the entities precisely in the
    same words as in the original paragraph.

>> ASSISTANT ANSWER

# varies from case to case, omitted
```

In the prompts, the “entity types” are rephrased so that they are more comprehensible to the models. For example, “`PER`” is rephrased as “`person`”; “`CN`” is rephrased as “`Material Name`”, *etc.*. The prompting format is kept consistent across all models with only one difference: the system prompt is not applied to open-source LLMs. Of course, the specific prompt string is adjusted to match each model’s prompting style. Similarly, the One-Step NER prompts are shown below:

Listing 2: One-Step NER prompts.

```
>> SYSTEM PROMPT
You are a knowledgeable assistant specialized in
    recognizing and understanding named entities
    and their interrelations. If requested to
    organize information in tabular format, you are
     adept at filtering and presenting only the
    relevant and valid results. You will exclude
    any results that are not pertinent or are
    inaccurate from the table according to the
    discussion history.

>> USER PROMPT  # Step 1. Free-form response
    generation
Please identify the "<ENTITY TYPE>" entities in the
    following paragraph and present the valid
    entities as a Markdown table with one column "<
    ENTITY TYPE>". Make sure to present the
    entities precisely in the same words as in the
    original paragraph.

Paragraph: <PARAGRAPH>

# optional zero-shot CoT prompt
Let's think step by step.

>> ASSISTANT ANSWER

# varies from case to case, omitted

```

```
>> USER PROMPT  # Step 2. Clean-up (optional)

Please remove irrelevant entities and only keep the
     entities that clearly refer to <ENTITY TYPE>
     and present the valid entities as a Markdown
     table with one column "<ENTITY TYPE>". Make
     sure to present the entities precisely in the
     same words as in the original paragraph.

>> ASSISTANT ANSWER

# varies from case to case, omitted
```

Notice that the term "One-Step" refers to the IE generation and structure organization being performed in a single step. The pipeline could also contain a standalone clean-up step, which is set up as default in our experiments (as revealed in Listing 2).

For both G&O and One-Step, we ask LLMs to generate the entities of one type at a time, which is most frequently adopted in previous works (Wang et al., 2023b; Xie et al., 2023). To validate the effectiveness of such an approach, we introduce AEiO, a method that generates all entities at once. It is derived from G&O, with the only difference being the instruction to generate and organize all entities simultaneously. The specific prompt for AEiO is shown in Listing 3.

Listing 3: AEiO NER prompts.

```
>> SYSTEM PROMPT
You are a knowledgeable assistant specialized in
     recognizing and understanding named entities
     and their interrelations. If requested to
     organize information in tabular format, you are
      adept at filtering and presenting only the
     relevant and valid results. You will exclude
     any results that are not pertinent or are
     inaccurate from the table according to the
     discussion history.

>> USER PROMPT  # Step 1. Free-form response
     generation
Please identify the "<ENTITY TYPE 1, ENTITY TYPE 2,
     ..., ENTITY TYPE n>" entities in the following
     paragraph.

Paragraph: <PARAGRAPH>

# optional zero-shot CoT prompt
Let's think step by step.

>> ASSISTANT ANSWER

# varies from case to case, omitted

>> USER PROMPT  # Step 2. Clean-up (optional)

Please remove entities that do not clearly refer to
     any of the following entity types: "<ENTITY
     TYPE 1, ENTITY TYPE 2, ..., ENTITY TYPE n>".

>> ASSISTANT ANSWER

# varies from case to case, omitted

>> USER PROMPT  # Step 3. Structure organization

Please present the valid entities as a Markdown
     table with columns ["Entity", "Entity Type"].

Make sure to present the entities precisely in the
     same words as in the original paragraph and the
      Entity Types are in "<ENTITY TYPE 1, ENTITY
     TYPE 2, ..., ENTITY TYPE n>".

>> ASSISTANT ANSWER

# varies from case to case, omitted
```

As NCBI Disease contains only one entity type, AEiO is identical to G&O and is not applied.

Conflict Resolution As mentioned in § 3.1, we adopt two strategies to resolve the entity type conflict in NER results: Conflict Resolution (CR) and BERT Fine-Tuning (FT). CR first identifies the conflicting entity spans and then prompts the model to resolve the conflict within the previous conflicting LLM outputs, also following the generation-organization prompting format. An example of CR is shown in Listing 4.

Listing 4: An example of conflict resolution on the CoNLL 2003 dataset.

```
>> SYSTEM PROMPTn
You are a helpful assistant who is good at
     identifying named entities and their relations.

>> USER PROMPT
According to the following paragraph, please
     identify and resolve the conflicts in the Named
      Entity Recognition (NER) results:

Paragraph: But China saw their luck desert them in
     the second match of the group , crashing to a
     surprise 2 - 0 defeat to newcomers Uzbekistan .

NER Response 1:
| Location    |
|-------------|
| China       |
| Uzbekistan  |

NER Response 2:
| Organization |
|--------------|
| Uzbekistan   |


Let's think step by step.

>> ASSISTANT ANSWER

Step 1: Identify the conflicts in the NER results.

In NER Response 1, both "China" and "Uzbekistan" are
      identified as locations, which is consistent
     with the context of the paragraph.

In NER Response 2, "Uzbekistan" is identified as an
     organization, which is not consistent with the
     context of the paragraph. "Uzbekistan" should
     be identified as a location.

Step 2: Resolve the conflicts in the NER results.

Based on the context of the paragraph, the resolved
     NER results should be:

Resolved NER Response 1:
| Location    |
|-------------|
| China       |
| Uzbekistan  |

Resolved NER Response 2:
| Location |
|--------------|
| Uzbekistan   |

Therefore, the conflicts in the NER results have
     been resolved.
```

```
>> USER PROMPT

Please present the entities with corrected entity
     types as a Markdown table with columns ["Entity
     ", "Entity Type"]. Make sure the Entities are
     expressed in the same words as the original
     answers and the Entity Types in ["location", "
     organization", "person"].

>> ASSISTANT ANSWER

Here are the corrected entity types presented as a
     Markdown table:

|   Entity   | Entity Type |
|------------|-------------|
|   China    |  location   |
| Uzbekistan |  location   |
```

In the example, we directly take the conflicting LLM outputs (“NER Response n”) as the input context and ask the model to resolve the conflict. We are aware that this may not be the most effective way to resolve conflicts, but it is the most straightforward to demonstrate the necessity of CR.

Supervised BERT Fine-Tuning Another strategy to resolve the entity type conflict and further boost the performance is to fine-tune a Transformer encoder model on LLM-generated content. To achieve this goal, we first align the LLM-generated entities to the tokenized paragraph (which is the same as our evaluation process discussed in appendix B.6), generating a set of token-level labels. Specifically, we use BIO2 tagging scheme, where “B” denotes the beginning of an entity, “I” denotes the inside of an entity, and “O” denotes the outside of an entity. If a conflict occurs, we randomly select one of the conflicting entities to be the pseudo label. With the training dataset established, we fully fine-tune DeBERTa V3 (He et al., 2023),[6] a state-of-the-art pre-trained Transformer encoder model, following the supervised learning paradigm with Cross Entropy loss. One tricky part is that our setting is more similar to transductive learning, as the model is fine-tuned and evaluated on the same dataset, although the gold labels are different. To prevent overfitting to pseudo labels, we apply a dropout rate of 0.3 to both self-attention and feed-forward layers. In addition, we use a relatively large learning rate of 1×10^{-4} with AdamW optimizer (Loshchilov and Hutter, 2019) and a linear learning rate scheduler with warm-up ratio of 0.1. On all datasets, the model is updated for around 300 *steps*—roughly 6 epochs for CoNLL 2003, 10 epochs for NCBI Disease and BC5CDR, and 15 epochs for PolyIE, with batch sizes adjusted from 16 to 64 accordingly. We do not apply any early stopping strategy due to the lack of a reliable validation signal. All experiments are conducted on a single NVIDIA A100 80G GPU with full-precision floating point numbers (float32), implemented with PyTorch (Paszke et al., 2019) and HuggingFace Transformers library (Wolf et al., 2019). No factorization or efficient tuning approach is adopted.

B.4 RE Implementation Details

RE poses a greater challenge than NER because it demands that the model not only discern entities within the text but also understand their contextual relationships in an end-to-end manner. Many relation labels, such as “`place lived`” or “texttt-location contains”, present ambiguity that can be difficult for LLMs to comprehend. To mitigate this, we tailor prompts for each type of relation to enhance the model’s comprehension. Specifically, we leverage GPT-4 (OpenAI, 2023) with Web UI to craft prompts based on a simple slot-filling template designed for GPT-3.5, enabling it to recognize specific relations from the textual context. An example of this process is provided in Listing 5, showcasing we guide GPT-4 in generating relation extraction prompts for GPT-3.5.

Listing 5: An example of GPT-4’s instruction to generate RE prompts for GPT 3.5.

```
>> USER PROMPT
Please rephrase the following prompt so that GPT 3.5
      can detect the "location contains"
     relationships between Florida and Boca Raton:

Original Prompt: Please identify the "location
     contains" relationships between the "Location"
     and "Location" entities in the following
     paragraph.

Paragraph: Graveside service Monday January 31 ,
     2:00 P.M. at Riverside Cemetery , Rochelle Park
      , N.J. Donations may be made to Hospice By The
      Sea , Boca Raton , Florida .

Let's thinks step by step.
```

Notice that the “original prompt” is simply modified from the template “Please identify the `<relation type>` relationships between the `<head entity type>` and `<tail entity type>` entities in the following paragraph.” These example paragraphs and labels used for prompt construction are drawn randomly from the training partition of datasets, ensuring no test data is exposed.

Another distinct aspect of RE, compared to NER, is the integration of the clean-up process into the structural organization phase, rather than treating it as a separate step. For RE, we introduce an additional column during the structuring phase, so that the result table not only lists entities linked

---

[6] huggingface.co/microsoft/deberta-v3-base

by the desired relation but also indicates the presence of that relation. In the post-processing stage, any entity pairs without a confirmed relation in this additional column are excluded. This integrated approach has proven more effective in preliminary tests than the isolated clean-up process traditionally used in NER. Examples of G&O-RE in Listing 6 conversation pipeline on the CoNLL 2004 dataset for the relation type "`organization-based-in`" illustrate this methodology, and one example output is demonstrated in Listing 7.

Listing 6: An example of G&O-RE.

```
>> SYSTEM PROMPT
You are a knowledgeable assistant specialized in
     recognizing and understanding named entities
     and their interrelations. When requested to
     organize information in tabular format, you are
      adept at filtering and presenting only the
     relevant and valid results. You will exclude
     any results that are not pertinent or are
     inaccurate from the table according to the
     discussion history.

>> USER PROMPT  # Step 1. Free-form response
     generation
Please analyze the given text to identify
     relationships where an organization is
     headquartered or primarily operates in a
     specific location. Look for patterns that
     indicate this type of relationship.

Paragraph: An art exhibit at the Hakawati Theatre in
      Arab east Jerusalem was a series of portraits
     of Palestinians killed in the rebellion .

Let's think step by step.

>> ASSISTANT ANSWER

# varies from case to case, omitted

>> USER PROMPT  # Step 2. Structure organization and
      clean-up

If exists, please present the valid relationships as
      a Markdown table with columns ["Organization",
      "Location", "Whether the Organization is based
      in the Location"]. Make sure the table items
     are from the original paragraph.

>> ASSISTANT ANSWER

# varies from case to case, omitted
```

Listing 7: An example of RE output. The column headers are slightly modified for better visualization.

```
|Organization|Location|Whether ORG based in LOC|
|-----------------------|----------|-----|
| Bolshoi Ballet        | Moscow   | Yes |
| Kirov Ballet          | Leningrad| Yes |
| Armenian opera singers| Yerevan  | Yes |
```

Furthermore, the One-Step RE prompt merges the two steps of the aforementioned pipeline, simplifying the process into a single prompt, as shown in Listing 8.

Listing 8: An example of One-Step prompting for RE.

```
>> SYSTEM PROMPT
You are a knowledgeable assistant specialized in
     recognizing and understanding named entities
     and their interrelations. When requested to
     organize information in tabular format, you are
      adept at filtering and presenting only the
     relevant and valid results. You will exclude
     any results that are not pertinent or are
     inaccurate from the table according to the
     discussion history.

>> USER PROMPT
Please analyze the given text to identify
     relationships where an organization is
     headquartered or primarily operates in a
     specific location. Look for patterns that
     indicate this type of relationship. If exists,
     please present the valid relationships as a
     Markdown table with columns ["Organization", "
     Location", "Whether the Organization is based
     in the Location"]. Make sure the table items
     are from the original paragraph.

Paragraph: An art exhibit at the Hakawati Theatre in
      Arab east Jerusalem was a series of portraits
     of Palestinians killed in the rebellion .

Let's think step by step.

>> ASSISTANT ANSWER

# varies from case to case, omitted
```

For a comprehensive review of the prompts designed for each relation type, we refer readers to the meta files accompanying each dataset within our code repository.

B.5 Justification for Using Markdown

As noted in § 2, we utilize Markdown tables to format the output from LLMs for both NER and RE tasks to ensure uniformity in the structured output. Additionally, we find that instructing LLMs to format outputs into Markdown tables, with column names such as `Organization`, `Location`, and `Whether the Organization is based in the Location`, simplifies the process compared to using JSON. JSON formatting requires more detailed prompts and can lead to inconsistencies in the output (*e.g.*, variations in key naming or decisions regarding the use of dictionaries versus lists as the primary structure). Consequently, we prefer Markdown tables for their simplicity in both RE and NER tasks, ensuring consistency across outputs. Examples and justifications for this formatting choice will be included in our revised manuscript.

B.6 Post-Processing and Evaluation

While extracting entities or relationships, the outputs from LLMs may not always align perfectly with the terminology or phrasing in the source text. Issues such as extraneous or missing spaces, variations in tense, and unnecessary clarification of acronyms, are common, particularly for smaller models. To address this, we employ a fuzzy matching algorithm using Python's difflib library[7] to bet-

---

[7] docs.python.org/3/library/difflib.html

	user+assistant	**assistant-only**
One-Step	157.78	60.41
G&O	246.35	149.59
− CoT	168.72	70.26
− clean-up	148.51	57.96
+ CR	+51.58	+29.15

Table 4: Average token consumption for annotating a single input instance using One-Step baseline prompting and G&O. "user+assistant" represents the combined input tokens from the user and the output tokens from the language model; "assistant-only" focuses solely on the model's output tokens. The "+ CR" row indicates the **additional** tokens required to resolve conflicts per instance.

ter correlate the LLM outputs with the original text.

Subsequently, we evaluate the model's precision, recall, and F_1 score based on how well the predicted entity spans match the actual ground truth spans. As discussed in § 3.1, our evaluation encompasses both full and partial match scores to provide a thorough assessment of model accuracy. A full match necessitates complete agreement between the predicted and ground truth spans, aligning with traditional evaluation methods. Partial matching, however, accounts for overlaps between predicted and actual spans, thus accommodating minor discrepancies. For instance, in the sentence "He's working for the White House", a ground truth entity labeled "White House → `Organization`" and a predicted span "the White House → `Organization`" (with an added "the" in the span) would be acknowledged as a true positive prediction in a partial match scenario, but considered both a false positive and a false negative in a full match evaluation. Conversely, labeling "White House" as a "`Location`" would be incorrect under both matching criteria.

For RE tasks, achieving a partial or full match on all entity spans in the relation group is required for a prediction to be considered correct in CoNLL 2004 and NYT. In the PolyIE dataset, we adopt a more lenient approach, accepting a relation prediction as correct if at least one set of mapped entity spans within a paragraph corresponds to the ground truth. This flexibility is due to the frequent mention of each entity within a relation group across the paragraph, owing to its length (Table 3). Imposing a strict criterion for matching all entity spans in PolyIE could lead to counterintuitive evaluation outcomes.

C Additional Results

C.1 Token Consumption

Table 4 summarizes the token consumption for annotating a single input instance using various prompting methods and their variants. It indicates that G&O uses more tokens than One-Step, primarily because of the extra tokens derived from the CoT reasoning processes in natural language, as demonstrated in the comparisons within the first three rows. One approach to address this issue is to use LLMs to annotate a subset of the labels, followed by training a discriminative model like DeBERTa to manage the remaining annotations. However, determining the optimal number of annotations to balance efficiency and effectiveness remains a challenge and warrants further investigation.

C.2 Complete Results

Tables 5 and 6 present the complete results of our experiments on the NER and RE tasks, respectively. Due to the limiation of computational resources, we do not conduct the full set of ablation studies on open-source LLMs or the RE task, and only validate our key point, G&O, by comparing it to the One-Step approach. The key findings are presented using tables and figures in the main paper and will not be repeated here.

		CoNLL 2003		**BC5CDR**	
		Partial	Full	Partial	Full
GPT-3.5	AEiO	0.5370 (0.6819 / 0.4429)	0.4965 (0.6323 / 0.4088)	0.6199 (0.8254 / 0.4963)	0.5058 (0.6794 / 0.4028)
	One-Step	0.4741 (0.4070 / 0.5678)	0.4477 (0.3850 / 0.5349)	0.7030 (0.6632 / 0.7479)	0.6041 (0.5720 / 0.6401)
	G&O-NER	0.6569 (0.6027 / 0.7219)	0.6192 (0.5695 / 0.6784)	0.7610 (0.7835 / 0.7398)	0.6079 (0.6334 / 0.5845)
	− CoT	0.6572 (0.6648 / 0.6498)	0.6079 (0.6185 / 0.5977)	0.6634 (0.8001 / 0.5666)	0.5544 (0.6776 / 0.4691)
	− clean-up	0.7003 (0.6482 / 0.7616)	0.6436 (0.5992 / 0.6950)	0.7421 (0.7153 / 0.7712)	0.5861 (0.5699 / 0.6032)
	+ CR	0.6775 (0.6386 / 0.7213)	0.6394 (0.6043 / 0.6788)	0.7724 (0.7954 / 0.7506)	0.6186 (0.6447 / 0.5946)
	+ FT	0.7175 (0.6496 / 0.8012)	0.6800 (0.6161 / 0.7585)	0.7949 (0.8166 / 0.7743)	0.6838 (0.7068 / 0.6622)
Llama 2 7B	One-Step	0.4237 (0.3234 / 0.6139)	0.3929 (0.3005 / 0.5672)	0.6426 (0.5766 / 0.7256)	0.5087 (0.4608 / 0.5678)
	G&O-NER	0.4281 (0.3193 / 0.6495)	0.3787 (0.2830 / 0.5725)	0.6408 (0.6300 / 0.6520)	0.5169 (0.5131 / 0.5207)
Llama 2 70B	One-Step	0.4685 (0.3711 / 0.6353)	0.4428 (0.3514 / 0.5983)	0.7532 (0.7118 / 0.7997)	0.6389 (0.6081 / 0.6730)
	G&O-NER	0.5476 (0.4490 / 0.7016)	0.5128 (0.4213 / 0.6550)	0.7260 (0.7251 / 0.7270)	0.5792 (0.5849 / 0.5737)
Mistral 7B	One-Step	0.4884 (0.3970 / 0.6344)	0.4075 (0.3334 / 0.5240)	0.7246 (0.7012 / 0.7496)	0.5244 (0.5149 / 0.5342)
	G&O-NER	0.5693 (0.4963 / 0.6676)	0.4963 (0.4349 / 0.5780)	0.7318 (0.7967 / 0.6766)	0.5300 (0.5870 / 0.4831)
Mixtral 8x7B	One-Step	0.5827 (0.5023 / 0.6936)	0.5213 (0.4517 / 0.6163)	0.7815 (0.7574 / 0.8072)	0.6038 (0.5923 / 0.6157)
	G&O-NER	0.6575 (0.6029 / 0.7230)	0.5937 (0.5471 / 0.6489)	0.7697 (0.7840 / 0.7560)	0.6098 (0.6309 / 0.5902)
		NCBI Disease		**PolyIE**	
		Partial	Full	Partial	Full
GPT-3.5	AEiO	-	-	0.1300 (0.7440 / 0.0712)	0.0935 (0.5383 / 0.0512)
	One-Step	0.6500 (0.6175 / 0.6860)	0.5131 (0.4851 / 0.5445)	0.4669 (0.4253 / 0.5177)	0.3207 (0.2936 / 0.3533)
	G&O-NER	0.6935 (0.8458 / 0.5877)	0.5047 (0.6278 / 0.4220)	0.5449 (0.5830 / 0.5115)	0.3823 (0.4117 / 0.3569)
	− CoT	0.5653 (0.8260 / 0.4297)	0.4059 (0.6101 / 0.3041)	0.4551 (0.4901 / 0.4249)	0.3068 (0.3342 / 0.2836)
	− clean-up	0.6475 (0.6775 / 0.6200)	0.4541 (0.4886 / 0.4242)	0.5103 (0.5036 / 0.5173)	0.3421 (0.3396 / 0.3448)
	+ CR	-	-	0.6011 (0.6723 / 0.5572)	0.4236 (0.4685 / 0.3866)
	+ FT	0.7703 (0.8822 / 0.6837)	0.5507 (0.6356 / 0.4859)	0.7608 (0.7044 / 0.8270)	0.5533 (0.5034 / 0.6141)
Llama 2 7B	One-Step	0.5405 (0.4670 / 0.6416)	0.3474 (0.3076 / 0.3992)	0.3994 (0.3701 / 0.4338)	0.2629 (0.2440 / 0.2849)
	G&O-NER	0.5342 (0.5481 / 0.5209)	0.3000 (0.3163 / 0.2853)	0.2881 (0.2898 / 0.2864)	0.1874 (0.1886 / 0.1863)
Llama 2 70B	One-Step	0.6390 (0.5910 / 0.6957)	0.4608 (0.4340 / 0.4911)	0.4421 (0.4307 / 0.4541)	0.3355 (0.3255 / 0.3461)
	G&O-NER	0.5992 (0.6098 / 0.5890)	0.3736 (0.3902 / 0.3584)	0.4466 (0.4313 / 0.4630)	0.3093 (0.2980 / 0.3215)
Mistral 7B	One-Step	0.6715 (0.7211 / 0.6282)	0.3842 (0.4306 / 0.3469)	0.3923 (0.3357 / 0.4720)	0.1821 (0.1586 / 0.2138)
	G&O-NER	0.6588 (0.7987 / 0.5606)	0.3892 (0.4897 / 0.3229)	0.4335 (0.4313 / 0.4357)	0.2060 (0.2072 / 0.2048)
Mixtral 8x7B	One-Step	0.6674 (0.6734 / 0.6615)	0.4484 (0.4667 / 0.4316)	0.5210 (0.4717 / 0.5817)	0.3144 (0.2873 / 0.3472)
	G&O-NER	0.7049 (0.8629 / 0.5958)	0.4413 (0.5613 / 0.3636)	0.5275 (0.4609 / 0.6165)	0.3268 (0.2880 / 0.3776)

Table 5: The complete results of different LLMs on NER datasets, presented as "F_1 (precision / recall)".

	CoNLL 2004		**NYT**		**PolyIE**	
	Partial	Full	Partial	Full	Partial	Full
GPT-3.5						
One-Step	0.404 (0.363 / 0.455)	0.387 (0.261 / 0.324)	0.116 (0.088 / 0.170)	0.106 (0.080 / 0.155)	0.281 (0.427 / 0.210)	0.178 (0.256 / 0.136)
G&O-RE	0.447 (0.436 / 0.459)	0.335 (0.333 / 0.337)	0.212 (0.145 / 0.393)	0.160 (0.110 / 0.298)	0.371 (0.390 / 0.353)	0.226 (0.221 / 0.231)
Llama 2 70B						
One-Step	0.334 (0.234 / 0.584)	0.224 (0.159 / 0.378)	0.182 (0.103 / 0.765)	0.152 (0.086 / 0.645)	0.358 (0.413 / 0.316)	0.254 (0.277 / 0.234)
G&O-RE	0.361 (0.258 / 0.605)	0.252 (0.184 / 0.404)	0.191 (0.111 / 0.683)	0.156 (0.091 / 0.559)	0.371 (0.449 / 0.317)	0.272 (0.313 / 0.241)
Mixtral 8x7B						
One-Step	0.423 (0.418 / 0.428)	0.260 (0.264 / 0.257)	0.261 (0.179 / 0.477)	0.187 (0.128 / 0.343)	0.242 (0.367 / 0.107)	0.134 (0.193 / 0.102)
G&O-RE	0.441 (0.397 / 0.496)	0.294 (0.270 / 0.323)	0.237 (0.152 / 0.541)	0.170 (0.109 / 0.389)	0.364 (0.416 / 0.324)	0.226 (0.240 / 0.213)

Table 6: Comprehensive performance metrics of various LLMs on RE Datasets, expressed as F_1 (precision / recall). Results from smaller-scale models, Llama 2 7B and Mistral 7B, are omitted due to their inability to produce valid responses in initial testing on NYT and PolyIE.