ChatUIE: Exploring Chat-based Unified Information Extraction using Large Language Models

Jun Xu, Mengshu Sun*, Zhiqiang Zhang and Jun Zhou
Ant Group, Hangzhou, China
{xujun.xj, mengshu.sms, lingyao.zzq, jun.zhoujun}@antgroup.com

Abstract

Recent advancements in large language models have shown impressive performance in general chat. However, their domain-specific capabilities, particularly in information extraction, have certain limitations. Extracting structured information from natural language that deviates from known schemas or instructions has proven challenging for previous prompt-based methods. This motivated us to explore domain-specific modeling in chat-based language models as a solution for extracting structured information from natural language. In this paper, we present ChatUIE, an innovative unified information extraction framework built upon ChatGLM. Simultaneously, reinforcement learning is employed to improve and align various tasks that involve confusing and limited samples. Furthermore, we integrate generation constraints to address the issue of generating elements that are not present in the input. Our experimental results demonstrate that ChatUIE can significantly improve the performance of information extraction with a slight decrease in chatting ability.

Keywords: information extraction, large language models, reinforcement learning

1. Introduction

Information extraction (IE) is a structured prediction task that aims to identify and structure user-specified information from unstructured texts (Andersen et al., 1992; Grishman, 2019; Lu et al., 2022; Cao et al., 2022; Jiang et al., 2021; Xu and Sun, 2022). IE tasks are highly diversified due to their varying targets (entities, relations, events, etc.), heterogeneous structures (spans, triplets, records, etc.), and domain-specific schemas (Lou et al., 2023; Zeng et al., 2022; Du et al., 2022). The primary studies (Jiang et al., 2021; Li et al., 2022; Xu et al., 2018; Ye et al., 2022; Cao et al., 2022; Sheng et al., 2021; Zhang et al., 2023; Tang et al., 2022; Xu et al., 2022) of information extraction are task-specialized, which results in dedicated architectures, isolated models, and specialized knowledge sources for different IE tasks. Several improved methods (Lu et al., 2022; Lou et al., 2023; Wei et al., 2023; Wang et al., 2023) have been proposed for the unified modeling of information extraction tasks, including prompt-based extractive and generative models. However, these methods are highly tailored to pre-defined schemas or fixed instructions, which makes it extremely challenging to facilitate natural language extraction. As shown in Figure 1, UIE relies on a pre-defined schema and prompt template. Deviating from this consistency can significantly degrade model performance, especially for zero-shot tasks where the schema was not seen during training. In contrast, InstructUIE uses a set of instructions for information extraction. However, since these instructions are not tailored to the specific task (what to extract?), the model is restricted to the known dataset. When faced with a new schema, using task-irrelevant instructions makes it difficult to produce satisfactory results.

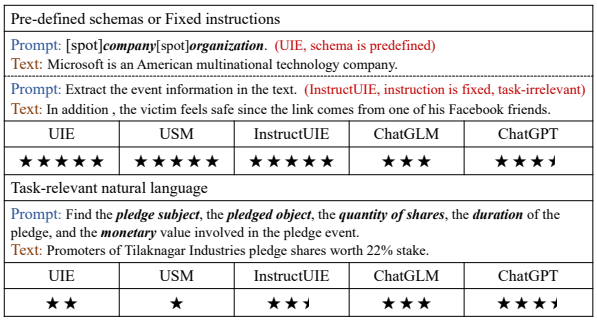

Figure 1: The approximate performance of unified information extraction framework in various application scenarios.

Generally, previous instruction-based methods focused more on memorizing instructions rather than comprehending them. While ChatGLM outperforms InstructUIE in task-relevant natural language scenarios, there is room for improvement in closed domain. However, enhancing the information extraction capabilities while preserving the general chat capabilities of ChatGLM presents challenges. Firstly, conflicts between domain-specific knowledge and the knowledge embedded in LLMs may result in knowledge forgetting. Secondly, the scarcity of annotated domain knowledge and uneven sample distribution make it arduous for LLM to effectively adapt and accommodate. These issues cannot be adequately ad-

*Corresponding author

LREC-COLING 2024, pages 3146–3152
20-25 May, 2024.

dressed through supervised fine-tuning alone. In order to address this issue, we have introduced reinforcement learning to align various tasks. In contrast to other tasks, the result for information extraction is derived from the input. To ensure that the generated elements remain within the input, we utilize generation constraint decoding.

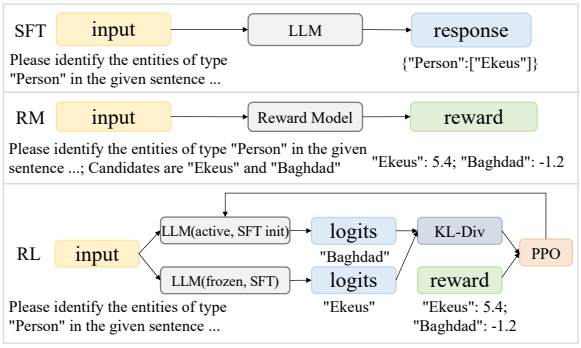

Figure 2: The overall framework of chat-based unified information extraction using LLMs.

2. Methodology

The architecture of our framework is illustrated in Figure 2, which mainly consists of three stages. Initially, supervised fine-tuning is used to incorporate domain knowledge into LLMs. Next, the reward learning model is utilized to enhance the learning of confusing samples and data with limited samples. Lastly, we combine the trained supervised fine-tuning model and reward model using reinforcement learning to align various tasks.

2.1. Domain Knowledge Integration

We utilize supervised learning to fine-tune ChatGLM using a variety of information extraction datasets. The input of the SFT model is divided into two parts: instruction and context. The instruction, with M tokens, and the context, with N tokens, are encoded by GLM to derive vector representations $x = [x_{i1}, ..., x_{iM}; x_{c1}, ..., x_{cN}] \in \mathbb{R}^{(M+N)\times D}$, where D represents the dimension of the embedding. The probability of generating each token is as follows:

$$p(y_i|x) = \frac{p(v(y_i)|c(x; y_{i-1}))}{\sum_{y' \in \mathcal{V}} p(v(y')|c(x; y_{i-1}))} \quad (1)$$

where $\mathcal{V}$ represents a vocabulary of 130, 528 tokens. v is a MLP , and c is the decoder of GLM. The dimension of $p(y_i|x)$ is $\mathbb{R}^{|\mathcal{V}|}$. The objective function of SFT is to maximize the likelihood:

$$\mathcal{L}_{SFT} = \frac{1}{L} \sum_{i=0}^{L} CE(y_i, p(y_i|x)) \quad (2)$$

where L represents the target sequence length. CE is the cross-entropy loss. Unified information extraction stands out from other text generation tasks as it necessitates the generated content to be a span within the input. Moreover, our framework adopts a JSON format to effectively represent the structured relationships. Therefore, we introduce generative constraint decoding. As shown in Figure 3, for instance, after generating token 'A', the next token must be either 'B' or '"'.

	Special token for JSON							Prompt			Input	
	{	}	[	]	,	:	"	A	B	C	D	E
{		1					1					
}				1								
[	1			1			1					
]		1			1		1					
,	1						1					
:			1				1					
"		1	1	1	1	1		1	1	1	1	1
A							1		1			
B							1			1		
C							1				1	
D							1					1
E							1					

Figure 3: Strategies for generating constraints in information extraction tasks.

2.2. Reinforcement Learning

In unified information extraction, the presence of diverse data sources and types often leads to challenges, such as type confusion and uneven distribution of samples, in the supervised fine-tuning model. To address these issues, reinforcement learning is introduced as a solution. The reward model of reinforcement learning also uses ChatGLM as the backbone. It takes an instruction, a context, and a positive or negative response as input, and outputs a scalar reward. We use the logits of the EOS token to represent the scalar reward:

$$r(x, y) = v(y_{eos}|c(x; y)) \quad (3)$$

Our goal is to maximize the difference between the rewards of positive and negative samples. Therefore, the objective of reward modeling can be expressed as follows:

$$\mathcal{L}_{RM} = -log(\sigma(r(x_p, y_p) - r(x_n, y_n))) \quad (4)$$

where σ is a sigmoid function. Contrary to the supervised fine-tuning model, the reward model does not exclusively dependent on training samples from the training set. The construction of training samples encompasses diverse methods, including: (1) substituting sample results with different types of confusion to generate negative samples, and (2) using the SFT and ChatGPT models to forecast extraction results for external

analogous datasets. Moreover, ChatGPT is used to score the extraction results, thereby increasing the amount of limited sample data. Then, the optimization strategy for reinforcement learning adopts PPO. Finally, the objective function in RL training can be expressed as follows:

$$\mathcal{L}_{RL} = r(x, y) - \beta log(\frac{p^{RL}(y|x)}{p^{SFT}(y|x)}) \quad (5)$$

where $r(x, y)$ represents a scalar reward, and β is a scalar coefficient of KL-div. $p^{RL}(y|x)$ denotes the logits of the active model, while $p^{SFT}(y|x)$ denotes the logits of the reference model.

3. Experimental Settings and Results

3.1. Experimental Settings

Dataset We conducted our experiments on multiple widely-used datasets, including Resume for NER; CoNLL-2004 for RE; FewFC for EE (Zhang and Yang, 2018; Roth and Yih, 2004; Zhou et al., 2021); WebQA, CEval, CMMLU, and MMLU (Li et al., 2016; Huang et al., 2023; Li et al., 2023; Hendrycks et al., 2021) for general chat. For detailed information on data division, please refer to Table 1. The training of ChatUIE requires two sets of data: one for the reward model (<instruction, context, positive response, negative response>) and the other for supervised fine-tuning and reinforcement learning (<instruction, context, response>). Negative responses consist of confusing data or data that the SFT model cannot fit. For example, an instruction like "Please identify the entities of type Person in the given sentence" and a context like "U.N. official Ekeus heads for Baghdad". The training data for SFT and RL is constructed as follows:

		SFT	RM	RL
Train	Resume	10,472	5,489	10,472
	CoNLL	2,647	1,153	2,647
	FewFC	25,665	8,299	25,665
	Sum	**38,784**	**14,941**	**38,784**
Dev	Resume	1,258	657	1,258
	CoNLL	659	288	659
	FewFC	3,263	1,050	3,263
	Sum	**5,180**	**1,996**	**5,180**
Test	Resume	1,253	664	1,253
	CoNLL	641	288	641
	FewFC	3,244	1,049	3,244
	Sum	**5,138**	**2,001**	**5,138**

Table 1: Details of the datasets: The division of these datasets is consistent with previous work, and ChatUIE performed training/dev data augmentation based on the divided results.

```
{
  "instruction": "Please identify the
      entities of type Person in the given
      sentence",
  "context": "U.N. official Ekeus heads for
      Baghdad",
  "ouput": '[{"Person": ["EKeus"]}]'
}
```

The training data for RM is constructed as follows:

```
{
  "instruction": "Please identify the
      entities of type `Person' in the given
      sentence",
  "context": "U.N. official Ekeus heads for
      Baghdad",
  "ouput": [
       '[{"Person": ["EKeus"]}]',
       '[{"Person": ["Baghdad"]}]'
    ]
}
```

Evaluation Metrics. For the NER task, we follow a span-level evaluation setting, where the entity span and entity type must be correctly predicted. For the RE task, a relation triple is correct if the model correctly predicts the span of subject and object and the relation between subject and object. For the EE task, we report two evaluation metrics: (1) Event Trigger: an event trigger is correct if the event type and the trigger span are correctly predicted. (2) Event Argument: an event argument is correct if its role type and event type match a reference argument mention. For general chat task, we use ROUGE-1 as the metric, it refers to the overlap of unigrams between the generation and reference response.

Implementation Details. We compare the proposed ChatUIE with the following strong baseline models: **ChatGLM** [1] takes information extraction as a generation problem. The input and output of the test set are consistent with ChatUIE, but ChatGLM has not been trained, and the output is inferred directly. **ChatGPT** gets responses through the official interface. **UIE** implementation and parameters are consistent with the author's official code [2], the English dataset uses the UIE English base model [3], and the Chinese dataset uses the mT5 base model. Our model **ChatUIE** is trained on 8×V100-32G. For other hyper-parameters and details, please refer to Table 2. The implementation of supervised fine-tuning and reward modeling refers to ChatGLM (Du et al., 2022; Robinson and Wingate, 2023), and the implementation of reinforcement learning refers to TRL (von Werra

[1] https://github.com/THUDM/ChatGLM-6B
[2] https://github.com/universal-ie/UIE
[3] https://huggingface.co/luyaojie/uie-base-en

	SFT	RM	RL
batch size	64	32	8
fine-tuning type	LoRA	LoRA	LoRA
train epochs	0-15	3	2
lora rank	8	8	8
lora dropout	0.1	0.1	0.1
lora target	QKV	QKV	QKV
learning rate	1e-4	2e-5	1e-6
max input length	450	450	450
max output length	600	600	600
KL-div β	-	-	0.1

Table 2: Hyperparameters of different models.

et al., 2020). The results reported in the experiment are the average of 5 different random seeds (0,1,2,3,4).

3.2. Results

3.2.1. Overall Results

The results reported in the experiment are the average of five different random seeds. Table 3 and Table 4 present the comparisons between our model and other baselines. As demonstrated, our model surpassed the generative model UIE (with supervised fine-tuning) by 3.89%, 1.27%, and 5.75% in F1 score on the Resume, CoNLL, and FewFC datasets, respectively. Additionally, it can be observed that the performance of ChatGLM and ChatGPT has significantly decreased without domain-specific data training. After applying reinforcement learning (RL), ChatUIE exhibited performance improvements of 1.56% in Resume, 2.43% in CoNLL, and 3.59% in FewFC. Furthermore, the generation constraints (GC) also showed noticeable enhancements across different datasets. As shown in Figure 4, using FewFC as an example, the categories of Acquisition, Transfer, and Investment are susceptible to confusion. However, after applying reinforcement learning techniques, the performance of all three types improves significantly. Notably, the sample sizes for Judgment and Charge are relatively small, constituting less than one-third of the Investment samples. By incorporating external homologous data through reinforcement learning, the overall effect improves by approximately 1 to 3 percentage points.

3.2.2. Results For Chatting

To assess the overall chat capability of ChatUIE, we have chosen WebQA as the dataset for evaluating common sense question and answer performance. As shown in Figure 5, compared to ChatGLM, ChatUIE demonstrated a significant improvement of 30.77% on Resume, 24.28% on CoNLL, and 41.22% when the epoch is 3. However, there is only a slight decrease of 0.89% in the general question answering task, which is still within an acceptable range. As the number of training epochs increases, the performance of the domain-specific datasets improves. However, there is a risk of losing the general chatting ability. Simultaneously, we also evaluate the overall capabilities of ChatUIE using a dataset for large-scale model assessment. As depicted in Table 5 and Table 6, a minor decline is observed in the overall chat capabilities of ChatUIE. At the same time, it is evident that the absence of reinforcement learning in ChatUIE results in a partial decrease in its overall chat capabilities. Reinforcement learning effectively mitigates knowledge decay by integrating diverse tasks.

Model	Resume (%)			CoNLL (%)		
	P	R	F1	P	R	F1
ChatGLM	41.05	86.98	55.78	11.60	70.82	19.93
ChatGPT	80.47	63.49	70.98	48.03	24.93	32.83
UIE	94.03	93.67	93.85	75.56	73.72	74.63
ChatUIE	95.58	95.82	**95.70**	75.82	75.35	**75.58**
w/o RL	93.51	94.97	94.23	74.25	73.34	73.79
w/o GC	94.37	94.55	94.46	74.03	73.82	73.92

Table 3: Overall results of Resume and CoNLL.

Model	TC (%)			AC (%)		
	P	R	F1	P	R	F1
ChatGLM	20.33	75.87	32.08	10.33	61.21	17.68
ChatGPT	83.86	42.56	56.47	76.80	29.54	42.67
UIE	89.64	76.90	82.78	78.82	61.42	69.04
ChatUIE	87.04	79.69	**83.20**	73.63	72.39	**73.01**
w/o RL	86.58	76.81	81.40	78.53	63.93	70.48
w/o GC	86.84	77.92	82.14	76.78	66.56	71.31

Table 4: Overall results of event extraction on FewFC. TC is trigger classification, while AC stands for argument classification.

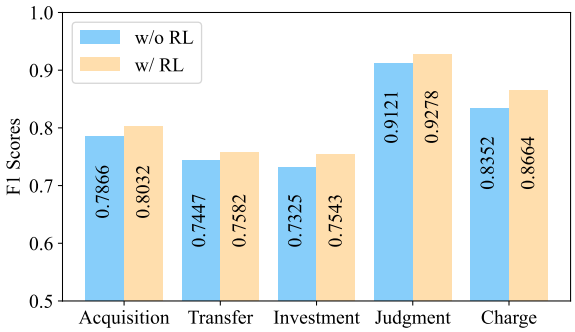

Figure 4: Performance comparison of confusing and limited samples after reinforcement learning in FewFC.

3.2.3. Zero-Shot Information Extraction

We evaluate the zero-shot performance of ChatUIE by testing it on unseen information extraction datasets. These include MSRA (Levow, 2006) for

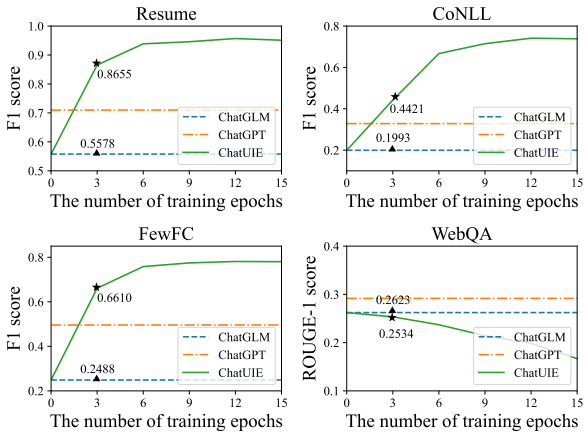

Figure 5: Performance of different tasks as the number of training epochs increase.

		STEM	Hum	SSci	Other	Avg
CEval	ChatGLM	35.84	44.24	45.14	40.07	40.30
	ChatUIE	35.12	43.93	45.02	39.46	39.72
	- RL	34.89	43.64	44.37	39.05	39.33
MMLU	ChatGLM	34.93	43.03	45.40	42.47	40.83
	ChatUIE	35.02	43.22	44.17	41.23	40.25
	- RL	34.85	43.11	43.83	40.96	40.02

Table 5: Overall results of CEval and MMLU.

NER; SemEval (Hendrickx et al., 2010) for RE; and iFLYTEK for EE. As can been seen from Table 7 and Table 8, our model outperformed the baseline model (ChatGLM) by 18.85%, 9.58%, and 8.89% in F1 score on the MSRA, SemEval, and iFLYTEK datasets, respectively. Since the training dataset of InstructUIE includes SemEval, zero-shot testing is not conducted. ChatUIE surpasses UIE and InstructUIE as it doesn't rely on pre-defined schema or fixed instructions, enabling it to comprehend natural language more effectively.

	STEM	Hum	SSci	Other	CSp	Avg
ChatGLM	31.53	40.52	41.30	39.88	38.59	38.35
ChatUIE	30.97	40.23	40.88	38.24	39.12	37.86
- RL	30.25	40.34	40.37	38.02	38.78	37.49

Table 6: Overall results of CMMLU.

Model	MSRA (%)			SemEval (%)		
	P	R	F1	P	R	F1
ChatGLM	14.74	88.31	25.26	6.49	32.84	10.84
ChatGPT	59.63	28.73	38.78	58.53	9.21	15.92
InstructUIE	29.35	33.56	31.31	-	-	-
UIE	24.33	40.97	30.52	9.87	47.26	16.33
ChatUIE	30.73	78.14	**44.11**	15.21	31.03	**20.42**

Table 7: Overall results of zero-shot NER on MSRA and RE on SemEval.

Model	TC (%)			AC (%)		
	P	R	F1	P	R	F1
ChatGLM	9.79	37.97	15.57	8.21	27.41	12.63
ChatGPT	38.50	11.66	17.90	33.25	10.04	15.42
InstructUIE	13.39	16.78	14.89	10.55	12.83	11.58
UIE	11.39	31.78	16.77	10.37	22.72	14.24
ChatUIE	22.82	26.41	**24.49**	20.07	23.25	**21.54**

Table 8: Overall results of zero-shot event extraction on iFLYTEK.

3.2.4. Case Study

As shown in Figure 6, it is clear that ChatGPT does not strictly follow the specified format, and some of the generated content in ChatGLM does not match the input. However, ChatUIE addresses these issues by utilizing reinforcement learning and generation constraints.

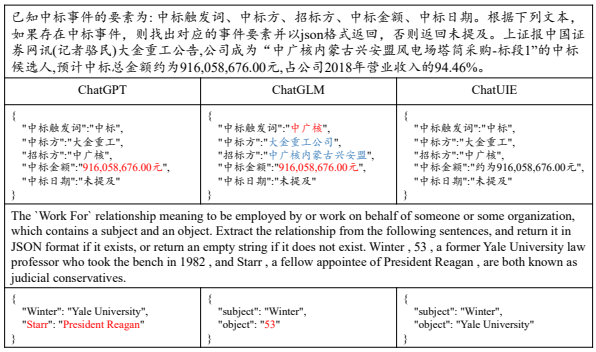

Figure 6: The impacts of various chat models on event extraction and relation extraction.

4. Conclusion

We have presented ChatUIE, a chat-like unified information extraction framework based on ChatGLM. Our framework effectively improves the performance of ChatGLM on domain-specific datasets while preserving its ability to chat. Empirical comparisons and analytical experiments have verified its effectiveness. Moreover, our work may have implications for other complex structured generation tasks.

5. Limitations

Nonetheless, these results must be interpreted with caution and several limitations should be borne in mind. Firstly, due to GPU limitations, we only trained our model based on ChatGLM-6B. Secondly, the processing speed of generative information extraction is significantly slower than that of extractive information extraction, making it more suitable for interactive applications. Thirdly, while the performance on domain-specific datasets is improved, there may be a slight loss of the ability to chat.

6. Acknowledgements

We would like to thank the anonymous reviewers for their constructive comments. This work was supported by Ant Group.

7. Bibliographical References

Peggy M. Andersen, Philip J. Hayes, Steven P. Weinstein, Alison K. Huettner, Linda M. Schmandt, and Irene B. Nirenburg. 1992. Automatic extraction of facts from press releases to generate news stories. In *3rd Applied Natural Language Processing Conference, ANLP 1992, Trento, Italy, March 31 - April 3, 1992*, pages 170–177. ACL.

Hu Cao, Jingye Li, Fangfang Su, Fei Li, Hao Fei, Shengqiong Wu, Bobo Li, Liang Zhao, and Donghong Ji. 2022. OneEE: A one-stage framework for fast overlapping and nested event extraction. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1953–1964, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. GLM: general language model pretraining with autoregressive blank infilling. pages 320–335.

Ralph Grishman. 2019. Twenty-five years of information extraction. *Nat. Lang. Eng.*, 25(6):677–692.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, Yao Fu, Maosong Sun, and Junxian He. 2023. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. *arXiv preprint arXiv:2305.08322*.

Haoming Jiang, Danqing Zhang, Tianyu Cao, Bing Yin, and Tuo Zhao. 2021. Named entity recognition with small strongly labeled and large weakly labeled data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1775–1789, Online. Association for Computational Linguistics.

Gina-Anne Levow. 2006. The third international Chinese language processing bakeoff: Word segmentation and named entity recognition. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 108–117, Sydney, Australia. Association for Computational Linguistics.

Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. 2023. Cmmlu: Measuring massive multitask language understanding in chinese.

Jingye Li, Hao Fei, Jiang Liu, Shengqiong Wu, Meishan Zhang, Chong Teng, Donghong Ji, and Fei Li. 2022. Unified named entity recognition as word-word relation classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10965–10973.

Peng Li, Wei Li, Zhengyan He, Xuguang Wang, Ying Cao, Jie Zhou, and Wei Xu. 2016. Dataset and neural recurrent sequence labeling model for open-domain factoid question answering.

Jie Lou, Yaojie Lu, Dai Dai, Wei Jia, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2023. Universal information extraction as unified semantic matching. *AAAI*, abs/2301.03282.

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5755–5772, Dublin, Ireland. Association for Computational Linguistics.

Joshua Robinson and David Wingate. 2023. Leveraging large language models for multiple choice question answering. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

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.

Jiawei Sheng, Shu Guo, Bowen Yu, Qian Li, Yiming Hei, Lihong Wang, Tingwen Liu, and Hongbo Xu. 2021. CasEE: A joint learning framework with cascade decoding for overlapping event extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 164–174, Online. Association for Computational Linguistics.

Wei Tang, Benfeng Xu, Yuyue Zhao, Zhendong Mao, Yifeng Liu, Yong Liao, and Haiyong Xie. 2022. UniRel: Unified representation and interaction for joint relational triple extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7087–7099, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, and Nathan Lambert. 2020. Trl: Transformer reinforcement learning. `https://github.com/lvwerra/trl`.

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. 2023. Instructuie: Multi-task instruction tuning for unified information extraction.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. 2023. Zero-shot information extraction via chatting with chatgpt.

Jun Xu, Siqi Shen, Dongsheng Li, and Yongquan Fu. 2018. A network-embedding based method for author disambiguation. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, pages 1735–1738. ACM.

Jun Xu and Mengshu Sun. 2022. DPNPED: dynamic perception network for polysemous event trigger detection. *IEEE Access*, 10:104801–104810.

Jun Xu, Weidi Xu, Mengshu Sun, Taifeng Wang, and Wei Chu. 2022. Extracting trigger-sharing events via an event matrix. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1189–1201, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Deming Ye, Yankai Lin, Peng Li, and Maosong Sun. 2022. Packed levitated marker for entity and relation extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4904–4917, Dublin, Ireland. Association for Computational Linguistics.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, Peng Zhang, Yuxiao Dong, and Jie Tang. 2022. GLM-130B: an open bilingual pre-trained model. *CoRR*, abs/2210.02414.

Sheng Zhang, Hao Cheng, Jianfeng Gao, and Hoifung Poon. 2023. Optimizing bi-encoder for named entity recognition via contrastive learning. *ICLR*.

Yue Zhang and Jie Yang. 2018. Chinese NER using lattice LSTM. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1554–1564, Melbourne, Australia. Association for Computational Linguistics.

Yang Zhou, Yubo Chen, Jun Zhao, Yin Wu, Jiexin Xu, and Jinlong Li. 2021. What the role is vs. what plays the role: Semi-supervised event argument extraction via dual question answering. In *Proceedings of AAAI-21*. AAAI Press.