CERD: A Comprehensive Chinese Rhetoric Dataset for Rhetorical Understanding and Generation in Essays

Anonymous EMNLP submission

Abstract

Existing rhetorical understanding and generation datasets or corpora primarily focus on single coarse-grained categories or fine-grained categories, neglecting the common interrelations between different rhetorical devices by treating them as independent sub-tasks. In this paper, we propose the **C**hinese **E**ssay **R**hetoric **D**ataset (CERD), consisting of 4 commonly used coarse-grained categories including metaphor, personification, hyperbole and parallelism and 23 fine-grained categories across both form and content levels. CERD is a manually annotated and comprehensive Chinese rhetoric dataset with five interrelated sub-tasks. Unlike previous work, our dataset aids in understanding various rhetorical devices, recognizing corresponding rhetorical components, and generating rhetorical sentences under given conditions, thereby improving the author's writing proficiency and language usage skills. Extensive experiments are conducted to demonstrate the interrelations between multiple tasks in CERD, as well as to establish a benchmark for future research on rhetoric. The experimental results indicate that Large Language Models achieve the best performance across most tasks, and jointly fine-tuning with multiple tasks further enhances performance. The dataset and code will be released in a future version.

1 Introduction

Rhetoric, a form of linguistic expression frequently used in Chinese, is often employed in literary works to enhance the effectiveness and persuasiveness of writing. In the learning process of primary and middle school students, rhetorical devices are a key component of writing skills, with metaphor, personification, hyperbole and parallelism being the most commonly used (Chen, 2019). Examples of four mentioned coarse-grained categories are shown in Figure 1. With the advancement of educational technology, several studies explored automatic essay evaluation (Wang et al., 2016; Yuan et al., 2020; Zhong and Zhang, 2020) where rhetoric is a key component because the use of rhetorical devices in writing reflects the literary quality and language expression ability of an essay (Burstein et al., 2001; Ishioka and Kameda, 2006).

Figure 1: An excerpt from an essay illustrating four commonly used rhetorical devices. It is worth noting that a sentence can employ one or more rhetorical devices, or it can be a literal sentence.

Popular rhetoric benchmarks often excessively focus on a single category of rhetoric and neglect the intrinsic connections between different rhetorical devices, leading to a limited and one-sided understanding of rhetorical phenomena. For example, Shutova (2010) and Li et al. (2022b) mainly considered metaphors, while Liu et al. (2018) and Chakrabarty et al. (2020) only considered similes. Specifically, Liu et al. (2018) focused only on similes and the rhetorical components are fixed as tenors and vehicles with a specific comparator in the sentences. Besides, Li et al. (2022b) introduced a corpus containing metaphorical sentences, treating personification as a type of metaphor. This results in a lack of full utilization of the interrelations between different rhetorical devices.

To address the challenges, as illustrated in Figure 2, we propose the **C**hinese **E**ssay **R**hetoric **D**ataset (CERD), a comprehensive Chinese rhetoric dataset with five sub-tasks, constructed from essays written by primary and middle school students in real-world scenarios. CERD addresses the afore-

Figure 2: An example of five sub-tasks in CERD. An overview of the five tasks is discussed in Section 4.1.

mentioned limitations in prior work: **Firstly**, our dataset includes 4 coarse-grained categories and 23 fine-grained categories across both form and content levels, providing a broader and deeper perspective for rhetorical understanding. **Secondly**, we abstract the types of rhetorical components across different fine-grained categories, enabling their extraction within a unified framework. This approach highlights the intrinsic connections between different rhetorical devices, facilitating a more comprehensive understanding. **Thirdly**, unlike previous benchmarks that only required generating parts of the rhetorical components, our dataset provides more context for generating complete rhetorical sentences under certain conditions because the annotation was conducted at the essay level.

The contributions of CERD are listed as follows:

- We propose the manually annotated **C**hinese **E**ssay **R**hetoric **D**ataset (CERD) which consists of five interrelated sub-tasks for rhetorical understanding and generation in essays.
- Extensive experiments are conducted on CERD as a benchmark for future research on rhetoric.
- We demonstrate the interrelations between the sub-tasks, highlighting that the annotations from one task can provide additional information to other tasks.

2 Related Work

Rhetoric studies primarily focus on two categories: understanding and generation.

Rhetoric Datasets For rhetorical understanding related datasets, Shutova (2010) sampled metaphorical texts from various genres including literature and newspaper articles. Liu et al. (2018) introduced an annotated Chinese essay corpus focusing on simile. Chinese Literary Grace Corpus (CLGC) presented by Li et al. (2022a) includes coarse-grained categories of metaphor, personification and parallelism while not further including fine-grained categories or annotations on rhetorical components. For rhetorical generation related datasets, Chakrabarty et al. (2020) presented a parallel corpus consisting of a large number of similes from collected from Reddit. Li et al. (2022b) introduced a labeled Chinese Metaphor Corpus (CMC) and a large-scale unlabeled Chinese Literature Corpus (CLC). MAPS-KB (He et al., 2023) is a million-scale probabilistic simile knowledge base including tenor and vehicle triplets for generating parts of rhetorical components. Distinct from previous work, CERD incorporates 4 commonly used coarse-grained categories in a unified framework with 5 interrelated sub-tasks.

Rhetoric Tasks and Approaches For rhetorical understanding tasks, Liu et al. (2018) presented the neural network-based approaches that outperform all rule-based (Niculae, 2013; Niculae and Yaneva, 2013; Qadir et al., 2015, 2016) and feature-based baselines (Li et al., 2008) on simile related tasks. Zeng et al. (2020) used the Chinese essay corpus introduced by Liu et al. (2018) as a benchmark and propose a cyclic multi-task learning model with a pre-trained BERT (Devlin et al., 2018) encoder that stacks sub-tasks and forms a loop by connecting the last to the first. Wang et al. (2022) used the same benchmark and present a model that merges the input-side features as a heterogeneous graph and leverages decoding features via distillation. For rhetorical generation tasks, Chakrabarty et al. (2020) proposed a fine-tuned BART model (Lewis et al., 2019) to generate sentences using similes based on literal sentences. Stowe et al. (2021) presented a fine-tuned T5 model (Raffel et al., 2020) to generate simile sentences in both free-text generation and controllable text generation scenarios. He et al. (2023) proposed a framework for large-scale simile knowledge base construction.

3 Dataset Construction

In this section, we discuss the construction process of CERD.The definitions and descriptions of tasks in CERD are introduced in Section 4.

3.1 Dataset Overview

We collected 503 essays from primary and middle school students' examinations and daily practice, averaging approximately 20.57 sentences and 706.47 tokens per essay. Essays written by students, whose first language is Chinese, are chosen because rhetoric is commonly used in their writing, especially since most of their essays are narrative than argumentative. Furthermore, the essays are written in real-world scenarios, genuinely reflecting the students' ability to use rhetoric.

CERD consists of five tasks, including **(1) Rhetoric Classification (Task RC)**, **(2) Form Classification (Task FC)**, **(3) Content Classification (Task CC)**, **(4) Component Extraction (Task CE)** and **(5) Rhetoric Generation (Task RG)**, covering both rhetoric understanding and generation. The annotation was conducted at the essay level, while the results are at the sentence level, except for Task RG.

3.2 Dataset Annotation

3.2.1 Dataset Annotation Guidelines

We developed the annotation guidelines based on the linguistic definitions of rhetoric (Li, 2020), categorizing the coarse-grained categories into four types: metaphor, personification, hyperbole and parallelism. We further categorize them into fine-grained categories at both form and content levels. More details are introduced in Appendix A.1.

Fine-grained Form-level Categories The coarse-grained categories are subdivided into 12 fine-grained form-level categories based on the parts of speech or structure of rhetorical components. Fine-grained form-level categories improve the understanding of the structures of rhetorical sentences, facilitating both the analysis of sentence grammar and the extraction of rhetorical components from the sentence.

Fine-grained Content-level Categories The coarse-grained categories are subdivided into 11 fine-grained content-level categories based on the property of rhetorical components. Fine-grained content-level categories enhance the recognition of the contents and topics of rhetorical sentences, thereby improving the understanding of rhetorical descriptions.

Rhetorical Components In general, rhetorical components are categorized into three types: connectors, objects and contents. Connectors are used to link the objects and contents or to represent significant markers in a sentence. Objects represent people or things described rhetorically in a sentence. Contents refer to the rhetorical descriptions in a sentence. For different form-level categories, the specific rhetorical components may have various meanings.

3.2.2 Dataset Annotation Process

During the entire annotation process, as illustrated in Figure 9 (Appendix A.2), four annotators with backgrounds in Education or Chinese Language and Literature participated. We first developed draft annotation guidelines and conducted a pre-annotation on 50 essays. After assesing the Inter-Annotator Agreements (IAA) (Cohen, 1960) between the annotators, we refined the draft annotation guidelines. Finally, 503 essays were divided into four batches, with the last 20 essays annotated by Annotator A being the same as the first 20 essays annotated by Annotator B, and so on. These overlapped annotations are used to check the IAA. More details are introduced in Appendix A.2.

3.3 Dataset Statistics

3.3.1 Inter-Annotator Agreements

We use Cohen's Kappa κ (Cohen, 1960) to evaluate the IAA, defined as Equation 1,

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{1}$$

where p_o is the empirical probability of agreement on the label assigned to any sample and p_e is the expected agreement when both annotators assign labels randomly. To calculate the IAA for Tasks RC, FC and CC, we use the weighted means of Cohen's Kappa across different categories. For Tasks CE and RG, we remove the tokens that are not part of any rhetorical component and calculate Cohen's Kappa at the token level. The IAA scores across five tasks of CERD are shown in Table 1.

Annotators	Cohen's Kappa κ (%)			
	RC	FC	CC	CE/RG
A & B	77.67	76.01	76.87	55.89
B & C	59.00	58.55	58.17	45.06
C & D	62.69	62.00	62.22	50.55
Average	66.45	65.54	65.76	50.50

Table 1: Inter-Annotator Agreements across five tasks of CERD. A, B, C and D denote the four annotators.

3.3.2 Dataset Distributions

The distribution of coarse-grained categories across five tasks is shown in Table 2. Sentences using metaphor and personification are more frequent than those employing hyperbole and parallelism, indicating that these are the most commonly rhetorical devices used in students' essays.

The distribution of fine-grained form-level categories is illustrated in Figure 3 (a), showing that the form categories of simile and verb are the most frequently used. We also assess the distribution of fine-grained content-level categories, displayed in Figure 3 (b), demonstrating that the content categories of concrete and personification are the most frequently used.

Task	#Met	#Per	#Hyp	#Par	#Lit
RC	509	220	130	150	150
FC	524	229	132	151	150
CC	522	221	130	151	150
CE	572	271	136	152	150
RG	449	260	135	0	0

Table 2: Distribution of coarse-grained categories across five tasks. "Met", "Per", "Hyp", "Par", "Lit" refer to metaphor, personification, hyperbole, parallelism and literal, respectively. A sentence can employ several rhetorical devices, which are not counted redundantly in the Task RC. Furthermore, Task RG excludes all sentences that use parallelism and literal sentences.

4 Experiments

4.1 Tasks Overview

CERD includes five tasks, covering multiple task types such as multi-label classification, named entity recognition and controllable text generation, providing comprehensive support for rhetorical understanding and generation.

Rhetoric/Form/Content Classification Tasks RC/FC/CC are multi-label classification problems. Given a sentence x as input, a model is asked to predict which rhetorical devices $y \subset Y$ the sentence employs, where the set Y denotes all the possible categories in a task. In particular, a sentence may employ multiple rhetorical devices. Therefore, $|y|$ should satisfy $1 \leq |y| \leq |Y|$. For Task RC, there are 5 possible coarse-grained categories, including the case of literal sentences. For Task FC, there are 13 possible fine-grained form-level categories, including the case of literal sentences. For Task CC, there are 12 possible fine-grained content-level categories, including the case of literal sentences.

Component Extraction Task CE is a named entity recognition problem. Given a sentence x with N tokens as input, a model is expected to extract all the possible rhetorical components y in the sentence, where $y = \{S_{\text{literals}}, S_{\text{connectors}}, S_{\text{objects}}, S_{\text{contents}}\}$ is a tuple. The set S consists of multiple ordered pairs (i, j), where $1 \leq i \leq j \leq N$ denotes the indices of the literal or rhetorical components in the sentence.

Rhetoric Generation Task RG is a controllable text generation problem. For an essay with N sentences, given the preceding context with at most k consecutive sentences $s = \{s_{i-k}, \ldots, s_{i-2}, s_{i-1}\}$,

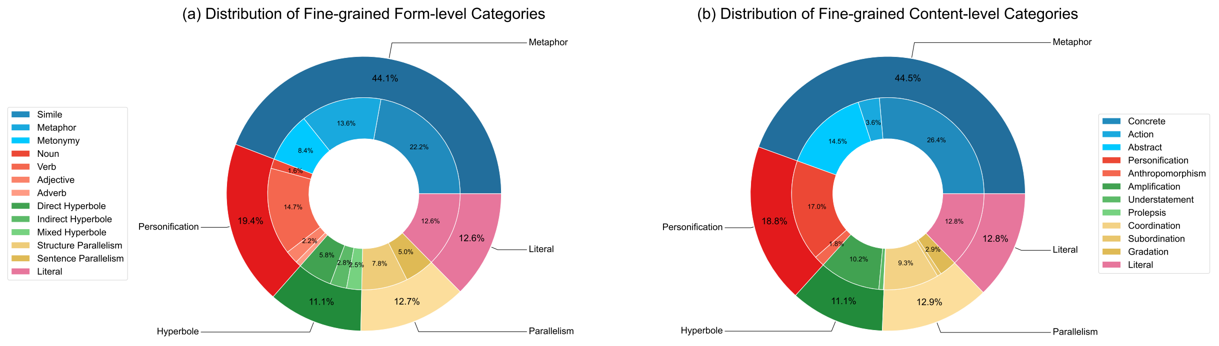

Figure 3: Distribution of fine-grained categories is illustrated in Figure (a) for form-level categories and in Figure (b) for content-level categories.

the objects of the i-th sentence, and the coarse-grained categories the i-th sentence employs as inputs, a model is asked to generate the sentence s_i satisfying the conditions, where $1 \leq i \leq N, k = \min\{k, i-1\}$.

Interrelations between the Tasks There are interrelations between multiple tasks in CERD, where the annotations from one task can provide additional information to other tasks. Tasks FC and CC rely on the coarse-grained categories provided by Task RG. Furthermore, Task CE relies on the fine-grained form-level categories from Task FC. Additionally, Task RG relies on the coarse-grained categories from Task RC and the rhetorical components extracted by Task CE.

4.2 Baselines and Evaluation Metrics

Baselines We evaluate RoBERTa (Liu et al., 2019), a BERT-based (Devlin et al., 2018) pretrained model on Task RC, FC, CC and CE. Furthermore, we test LLMs such as GPT-3.5 (OpenAI, 2022), GPT-4 (Achiam et al., 2023) and Qwen1.5 (Bai et al., 2023) on all the tasks. In particular, for RoBERTa, we choose RoBERTa$_{\text{BASE}}$[1] pretrained on Chinese corpus CLUECorpusSamll (Xu et al., 2020). For GPT-3.5 and GPT-4, we use gpt-3.5-turbo-0125 and gpt-4-turbo-2024-04-09 respectively. For Qwen1.5, we adopt both zero-shot learning and LoRA (Hu et al., 2021) fine-tuning for all the tasks. Details of the experimental setups are provided in Appendix C.

Evaluation Metrics To evaluate Tasks RC, FC, CC and CE, we utilize the metrics such as Exact Match, Precision, Recall and F1 score. In particular, seqeval (Ramshaw and Marcus, 1999; Nakayama, 2018), a framework for sequence labeling evaluation, is used to assess Task CE. To evaluate Task RG, we adopt automatic evaluation metrics such as BLEU (Papineni et al., 2002), ROUGE (Lin, 2004) and PPL (Jelinek et al., 1977), and we also use LLMs like GPT-4o (OpenAI, 2024) to evaluate the quality of the models' generations. Specifically, we design two LLM-based evaluation metrics: Single-answer Rating and Pairwise Ranking. The Single-answer Rating metric asks the LLM to rate the generations on a scale from 1 to 5. The Pairwise Ranking metric asks the LLM to compare the generated sentences with the original ones written in the essays.

4.3 Results and Analysis

4.3.1 Rhetoric Classification

As shown in Table 3, Qwen1.5-7B with multi-task fine-tuning outperforms all other models in classifying coarse-grained categories. Besides, RoBERTa fine-tuned on the task surpasses all the LLMs in zero-shot performance but scores slightly lower than Qwen1.5-7B with single-task fine-tuning.

The experimental results indicate that BERT-based model outperform LLMs when there are relatively few categories and the differences between coarse-grained categories are significant.

4.3.2 Form Classification

As shown in Table 4, for a more complicated multi-label classification problem, RoBERTa performs competitively with LLMs. In particular, RoBERTa outperforms Qwen1.5-7B with both single-task fine-tuning and multi-task fine-tuning on the micro-F1 score. However, Qwen1.5-7B with fine-tuning performs significantly better than RoBERTa on the macro-F1 score, while Qwen1.5-7B with zero-shot

[1] https://huggingface.co/uer/chinese_roberta_L-12_H-768

Models	EM	micro-P	micro-R	micro-F1	macro-P	macro-R	macro-F1
RoBERTa	63.31	72.40	<u>76.81</u>	74.54	68.75	<u>69.00</u>	68.36
GPT-3.5	20.16	37.39	64.26	47.27	30.61	51.95	36.10
GPT-4	54.44	61.46	70.34	65.50	54.21	63.36	57.11
Qwen1.5-7B	27.82	40.54	68.44	50.92	31.43	54.35	38.69
w/ single-task FT	<u>71.77</u>	<u>77.25</u>	74.90	<u>76.06</u>	<u>73.05</u>	68.29	<u>70.27</u>
w/ multi-task FT	**75.40**	**80.56**	**77.19**	**78.83**	**76.71**	**70.02**	**72.68**

Table 3: Results (in %) of Rhetoric Classification Task.

approaches the performance of RoBERTa and GPT-4 in zero-shot settings.

4.3.3 Content Classification

As shown in Table 5, RoBERTa outperforms all the LLMs on all metrics except for macro-Recall and macro-F1, while Qwen1.5-7B with multi-task fine-tuning approaches the performance of RoBERTa. Notably, GPT-4 surpasses all other baselines on the macro-F1 score by approximately 15% compared to the second best model.

The experimental results of Tasks FC and CC on the macro-F1 scores highlight that LLMs are more capable of understanding imbalanced fine-grained categories than BERT-based model. This is possibly because LLMs learn the concepts and differences of various categories through prompts, which will be further discussed in Appendix D.

Furthermore, compared to Task RC, Qwen1.5-7B with multi-task fine-tuning surpasses the model fine-tuned on the single task, demonstrating that it learns the interrelations between different tasks. A possible explanation is that the model learns the mappings of coarse-grained and fine-grained categories through multi-task fine-tuning. As illustrated in Figure 4, the given sentence employs both metaphor and personification, while Qwen1.5-7B with single-task fine-tuning classifies it as personification. Additionally, for Task FC, the model predicts the sentence as indirect hyperbole, which is a fine-grained category of hyperbole rather than personification. The mismatched mapping between coarse-grained and fine-grained categories also occurs in Task CC, indicating that the model fails to establish the correct mappings through single-task fine-tuning. Further analysis of the mappings between categories is discussed in Section 5.1.

4.3.4 Component Extraction

As shown in Table 6, Qwen1.5-7B with multi-task fine-tuning is competitive with RoBERTa on both the micro-F1 and macro-F1 scores. Additionally, GPT-4 with zero-shot achieves the best performance on Recall metrics.

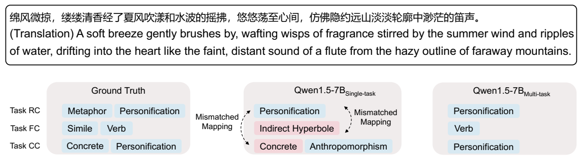

Figure 4: Case study on Rhetoric Classification Task, Form Classification Task and Content Classification Task. A mismatched mapping refers to a fine-grained category that does not belong to its predicted corresponding coarse-grained category.

As illustrated in Figure 5, the fine-grained form-level category of the given sentence is simile, which requires comparator, tenor and vehicle as its rhetorical components. Qwen1.5-7B with single-task fine-tuning fails to extract the comparator from the sentence, even though the model classifies it as a simile sentence. Further analysis of mappings between rhetorical components and fine-grained form-level categories is discussed in Section 5.2.

Figure 5: Case study on Component Extraction Task. A mismatched mapping refers to the extracted rhetorical components that do not fully satisfy the requirements of the predicted corresponding fine-grained form-level category.

4.3.5 Rhetoric Generation

As shown in Table 7, Qwen1.5-7B and GPT-4 with zero-shot exhibit competitive performances across multiple metrics. Specifically, for automatic evaluation metrics, Qwen1.5-7B achieves

Models	EM	micro-P	micro-R	micro-F1	macro-P	macro-R	macro-F1
RoBERTa	<u>50.81</u>	**76.63**	<u>52.03</u>	**61.98**	**86.13**	29.93	33.92
GPT-3.5	2.42	12.86	29.89	17.98	33.85	25.97	20.02
GPT-4	24.60	33.06	43.91	37.72	37.48	<u>30.78</u>	30.39
Qwen1.5-7B	5.24	14.39	35.42	20.47	20.13	25.22	28.99
w/ single-task FT	41.94	47.98	43.91	45.86	<u>52.09</u>	24.92	<u>40.20</u>
w/ multi-task FT	**54.03**	<u>59.60</u>	**54.98**	<u>57.20</u>	51.46	**31.81**	**55.04**

Table 4: Results (in %) of Form Classification Task.

Models	EM	micro-P	micro-R	micro-F1	macro-P	macro-R	macro-F1
RoBERTa	**54.44**	**67.95**	**59.77**	**63.60**	**75.55**	40.44	43.49
GPT-3.5	2.82	16.35	32.71	21.80	21.34	31.76	31.80
GPT-4	12.50	23.84	28.95	26.15	25.79	29.31	**58.26**
Qwen1.5-7B	2.42	16.90	35.71	22.95	18.69	35.95	33.89
w/ single-task FT	46.77	51.21	47.74	49.42	<u>66.49</u>	35.92	36.96
w/ multi-task FT	<u>53.63</u>	<u>59.68</u>	<u>56.77</u>	<u>58.19</u>	55.19	**42.27**	<u>43.85</u>

Table 5: Results (in %) of Content Classification Task.

Models	Acc	micro-P	micro-R	micro-F1	macro-P	macro-R	macro-F1
RoBERTa	**89.23**	38.84	<u>40.61</u>	**39.70**	42.26	<u>43.49</u>	<u>42.83</u>
GPT-3.5	52.09	10.01	29.98	15.01	12.66	29.88	17.07
GPT-4	71.20	29.10	**44.40**	35.16	30.01	**46.73**	36.51
Qwen1.5-7B	56.17	11.34	33.40	16.93	11.41	36.39	17.20
w/ single-task FT	<u>83.82</u>	<u>40.82</u>	32.07	35.92	**51.72**	31.63	37.14
w/ multi-task FT	82.64	**41.81**	37.76	<u>39.68</u>	<u>46.21</u>	40.32	**43.00**

Table 6: Results (in %) of Component Extraction Task.

the best performance on BLEU-2 and PPL, while GPT-4 surpasses other baselines on BLEU-4 and ROUGE-L. For LLM-based evaluation metrics, GPT-4 achieves the highest Single-answer rating score, indicating its capability to generate fluent and expressive rhetorical sentences. Furthermore, Qwen1.5-7B performs the best on the Pairwise Ranking metric, demonstrating that 69.23% of its generated rhetorical sentences are better than the references in essays. However, it is worth noting that compared to Qwen1.5-7B with zero-shot, the model fine-tuned on Task RG or multi-task performs worse. A potential reason is that the model overfits on the training set and therefore loses its generalization capability.

An example of rhetorical sentences generated by various models is illustrated in Figure 6, indicating that GPT-3.5, GPT-4 and Qwen1.5-7B generate the rhetorical sentences satisfying the given conditions. Besides, the generation closely relates to the preceding context. For example, GPT-3.5 and Qwen1.5-7B mention the fragrance of flowers that appeared earlier in the text, while GPT-4 references the previously mentioned breeze.

Figure 6: Case study on Rhetoric Generation Task.

5 Discussion

5.1 Effect of Rhetoric Classification Task

As mentioned in Section 4.1 and Section 4.3.3, Task RC provides information on coarse-grained categories, while Tasks FC and CC require the model to classify sentences at fine-grained levels. Intuitively, it is much more complicated for a model to directly solve Tasks FC and CC because the num-

Models	BLEU-2 (%) ↑	BLEU-4 (%) ↑	ROUGE-L (%) ↑	PPL ↓	Rating ↑	Ranking (%) ↑
GPT-3.5	6.55	3.23	19.13	81.10	4.01	59.17
GPT-4	6.82	**3.43**	**20.33**	45.79	**4.61**	66.27
Qwen1.5-7B	**8.27**	3.24	17.43	**45.17**	4.14	**69.23**
w/ single-task FT	6.96	2.77	14.74	154.39	1.67	19.53
w/ multi-task FT	5.83	1.69	14.61	125.96	1.97	29.59

Table 7: Results of Rhetoric Generation Task. "Rating" refers to Pairwise-answer Rating, a score from 1 to 5. "Ranking" refers to Pairwise Ranking, indicating the percentage of generated sentences better than the references.

ber of fine-grained categories is larger than that of coarse-grained ones. Therefore, learning the mappings between coarse-grained categories and their corresponding fine-grained categories may help the model solve Tasks FC and CC.

We define the correct mapping rate as the percentage of instances where a model correctly maps all coarse-grained categories in Task RC to their corresponding fine-grained form-level or content-level categories in Tasks FC or CC. As displayed in Figure 7, RoBERTa and Qwen1.5-7B fine-tuned on the single task show similar but relatively low performance on correct mapping rates. When Task RC is removed from the multi-task fine-tuning stage, there are no significant differences on correct mapping rates compared to Qwen1.5-7B with single-task fine-tuning. However, reintroducing Task RC data during multi-task fine-tuning significantly improves the performance of Qwen1.5-7B on correct mapping rate. Therefore, the experiment demonstrates the effect of Task RC on the mappings between coarse-grained and fine-grained categories.

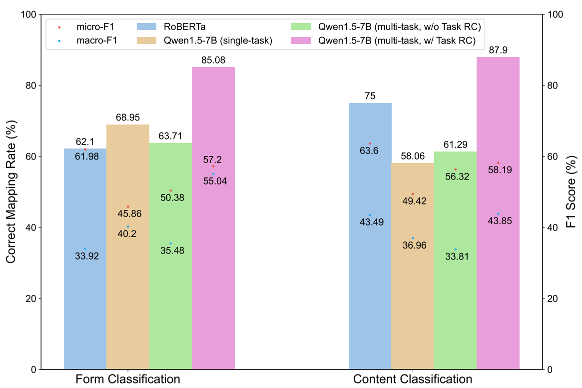

Figure 7: Effect of Task RC during multi-task fine-tuning. The bars represent the correct mapping rates, while the points represent the F1 scores.

5.2 Effect of Form Classification Task

Similar to the correct mapping rate in Section 5.1, the correct mapping rate of Task CE is defined as the percentage of instances where a model extracts all the necessary rhetorical components in a given sentence according to its form-level categories. As shown in Figure 8, compared to RoBERTa and Qwen1.5-7B fine-tuned without Task FC, Qwen1.5-7B with multi-task fine-tuning improves the correct mapping rate. The results demonstrate the importance of Task FC in extracting correct rhetorical components from the sentences.

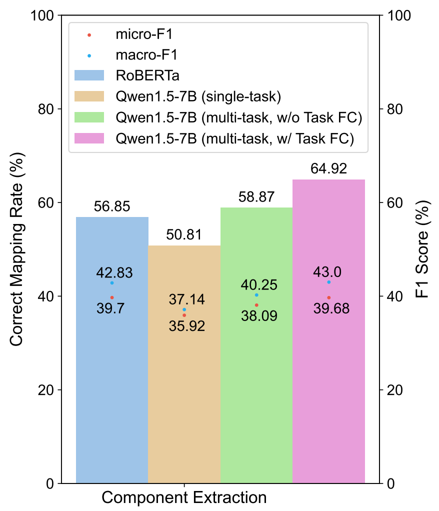

Figure 8: Effect of Task FC during multi-task fine-tuning. The bars represent the correct mapping rates, while the points represent the F1 scores.

6 Conclusion

In this paper, we propose the **C**hinese **E**ssay **R**hetoric **D**ataset (CERD), a comprehensive Chinese rhetoric dataset consisting of five sub-tasks. We conduct extensive experiments as a benchmark for future research on rhetoric. The experimental results indicate that both GPT-4 and Qwen1.5-7B with fine-tuning are superior baseline models, achieving competitive performances across multiple sub-tasks. Furthermore, we demonstrate the interrelations between different sub-tasks in CERD and the significance of task settings.

Limitations

The data collected to construct CERD comes from real-world scenarios. Although it does not affect the recognition and understanding of rhetoric, there may inevitably be some typographical errors due to the limited language proficiency of primary and middle school students.

Ethics Statement

All the participating annotators were compensated for their contributions, with each annotator's hourly wage being approximately 45% higher than the local minimum wage. Additionally, all the essays in CERD have been authorized for use. Moreover, to protect the privacy of the authors, we adopted data anonymization in CERD, removing all personal information related to them.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Jill C Burstein, Lisa Braden-Harder, Martin S Chodorow, Bruce A Kaplan, Karen Kukich, Chi Lu, Donald A Rock, and Susanne Wolff. 2001. System and method for computer-based automatic essay scoring. US Patent 6,181,909.

Tuhin Chakrabarty, Smaranda Muresan, and Nanyun Peng. 2020. Generating similes effortlessly like a pro: A style transfer approach for simile generation. *arXiv preprint arXiv:2009.08942*.

Lifei Chen. 2019. A study on the present situation of rhetoric use in primary school students compositions. Master's thesis, Shanghai Normal University.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Qianyu He, Xintao Wang, Jiaqing Liang, and Yanghua Xiao. 2023. Maps-kb: A million-scale probabilistic simile knowledge base. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 6398–6406.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Tsunenori Ishioka and Masayuki Kameda. 2006. Automated japanese essay scoring system based on articles written by experts. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 233–240.

Fred Jelinek, Robert L Mercer, Lalit R Bahl, and James K Baker. 1977. Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1):S63–S63.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Bin Li, Li-li Yu, Min Shi, and Wei-guang Qu. 2008. Computation of chinese simile with "xiang". *J. Chin. Inf. Process.*, 22(6):27–32.

Qingrong Li. 2020. *Modern Practical Chinese Rhetoric*. BEIJING BOOK CO. INC. In Chinese.

Yi Li, Dong Yu, and Pengyuan Liu. 2022a. Clgc: A corpus for chinese literary grace evaluation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5548–5556.

Yucheng Li, Chenghua Lin, and Frank Geurin. 2022b. Nominal metaphor generation with multitask learning. *arXiv preprint arXiv:2206.05195*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Lizhen Liu, Xiao Hu, Wei Song, Ruiji Fu, Ting Liu, and Guoping Hu. 2018. Neural multitask learning for simile recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1543–1553.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Hiroki Nakayama. 2018. seqeval: A python framework for sequence labeling evaluation. Software available from https://github.com/chakki-works/seqeval.

Vlad Niculae. 2013. Comparison pattern matching and creative simile recognition. In *Proceedings of the Joint Symposium on Semantic Processing. Textual Inference and Structures in Corpora*, pages 110–114.

Vlad Niculae and Victoria Yaneva. 2013. Computational considerations of comparisons and similes. In *51st Annual Meeting of the Association for Computational Linguistics Proceedings of the Student Research Workshop*, pages 89–95.

OpenAI. 2022. Introducing chatgpt. https://openai.com/blog/chatgpt/. Accessed: 2024-03-10.

OpenAI. 2024. Hello gpt-4o. https://openai.com/index/hello-gpt-4o/. Accessed: 2024-05-13.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Ashequl Qadir, Ellen Riloff, and Marilyn Walker. 2015. Learning to recognize affective polarity in similes. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 190–200.

Ashequl Qadir, Ellen Riloff, and Marilyn Walker. 2016. Automatically inferring implicit properties in similes. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1223–1232.

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

Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.

Ekaterina Shutova. 2010. Automatic metaphor interpretation as a paraphrasing task. In *Human language technologies: the 2010 annual conference of the North American chapter of the association for computational linguistics*, pages 1029–1037.

Kevin Stowe, Nils Beck, and Iryna Gurevych. 2021. Exploring metaphoric paraphrase generation. In *Proceedings of the 25th conference on computational natural language learning*, pages 323–336.

Xiaoyue Wang, Linfeng Song, Xin Liu, Chulun Zhou, and Jinsong Su. 2022. Getting the most out of simile recognition. *arXiv preprint arXiv:2211.05984*.

YH Wang, ZJ Li, YY He, W Chao, and J Zhou. 2016. Research on key technology of automatic essay scoring based on text semantic dispersion. *Journal of Chinese Information Processing*, 30(6):173–181.

Liang Xu, Xuanwei Zhang, and Qianqian Dong. 2020. Cluecorpus2020: A large-scale chinese corpus for pre-training language model. *arXiv preprint arXiv:2003.01355*.

Shuai Yuan, Tingting He, Huan Huang, Rui Hou, and Meng Wang. 2020. Automated chinese essay scoring based on deep learning. *Computers, Materials & Continua*, 65(1):817–833.

Jiali Zeng, Linfeng Song, Jinsong Su, Jun Xie, Wei Song, and Jiebo Luo. 2020. Neural simile recognition with cyclic multitask learning and local attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9515–9522.

Q Zhong and J Zhang. 2020. Chinese composition scoring algorithm embedded with language deep perception. *Comput. Eng. Appl*, 56:124–129.

A Dataset Annotation Details

A.1 Details of Dataset Annotation Guidelines

The annotation guidelines for form-level and content-level categories in CERD is shown in Table 8 and 9 respectively. We subdivide the coarse-grained categories into fine-grained form-level and content-level categories based on specific criteria. Specifically, the fine-grained form-level categories include:

- For metaphor, it is subdivided into simile, metaphor and metonymy.
- For personification, it is subdivided into noun, verb, adjective and adverb.
- For hyperbole, it is subdivided into direct hyperbole, indirect hyperbole and mixed hyperbole.
- For parallelism, it is subdivided into structure parallelism and sentence parallelism.

Besides, the fine-grained content-level categories include:

- For metaphor, it is subdivided into concrete, action and abstract.
- For personification, it is subdivided into personification and anthropomorphism.
- For hyperbole, it is subdivided into amplification, understatement and prolepsis.
- For parallelism, it is subdivided into coordination, subordination and gradation.

Coarse-grained Category	Criteria	Form-level Category	Explanation
Metaphor	The explicitness of rhetorical components	Simile	Tenor, vehicle and comparator are used explicitly in the sentence.
		Metaphor	Tenor and vehicle are used explicitly in the sentence.
		Metonymy	Only vehicle is used explicitly in the sentence.
Personification	The parts of speech of rhetorical components	Noun	Use nouns for people/objects to describe objects/people.
		Verb	Use verbs for people/objects to describe objects/people.
		Adjective	Use adjectives for people/objects to describe objects/people.
		Adverb	Use adverbs for people/objects to describe objects/people.
Hyperbole	The form of hyperbole	Direct Hyperbole	Directly exaggerate something.
		Indirect Hyperbole	Exaggerate something else to exaggerate a thing.
		Mixed Hyperbole	Exaggerate using other rhetorical devices.
Parallelism	The component of parallelism item	Structure Parallelism	The item servers as a specific grammatical component in the sentence.
		Sentence Parallelism	The item servers as a complete sentence on its own.

Table 8: Annotation guidelines for fine-grained form-level categories in CERD.

Coarse-grained Category	Criteria	Content-level Category	Explanation
Metaphor	The property of tenor	Concrete	The tenor can be seen, touched or imagined.
		Action	The tenor is an action, behavior or event.
		Abstract	The tenor is an abstract concept.
Personification	The property of content	Personification	Write about a non-human as if it were human.
		Anthropomorphism	Write about something that is not A as if it were A, where A is non-human.
Hyperbole	The direction of hyperbole	Amplification	Exaggeration towards large, many, long or high.
		Understatement	Exaggeration towards small, few, short or low.
		Prolepsis	Mentioning a later event before an earlier event.
Parallelism	The relationship between items	Coordination	Changing the order of the items does not affect the coherence.
		Subordination	A logical order of precedence between items exists.
		Gradation	The meanings and emotions expressed by each item progressively intensify.

Table 9: Annotation guidelines for fine-grained content-level categories in CERD.

Additionally, the annotation guidelines for rhetorical components are shown in Table 10. As mentioned in Section 3.1, we abstract the rhetorical components into three types: connectors, objects and contents. Specifically, for different coarse-grained categories or fine-grained form-level categories, the rhetorical components have various meanings:

- For metaphor, if the form-level category is simile, the rhetorical components include the comparator (as the connector), the tenor (as the object) and the vehicle (as the content). If the form-level category is metaphor, the rhetorical components include the tenor (as the object) and the vehicle (as the content). If the form-level category is metonymy, the rhetorical components only include the vehicle (as the content).

Coarse-grained Category	Criteria	Form-level Category	Rhetorical Components		
			Connector	Object	Content
Metaphor	Tenor: the object or concept being compared Vehicle: the object or concept used for comparison Compartor: the word connects the tenor and vehicle	Simile Metaphor Metonymy	Comparator - -	Tenor -	Vehicle
Personification	Personification Object: the person/thing being described Personification Content: the similarities to the object	-	- -	Personification Object	Personification Content
Hyperbole	Hyperbole Object: the thing being described Hyperbole Content: the exaggerated description	-	- -	Hyperbole Object	Hyperbole Content
Parallelism	Parallelism Item: the markers	-	Parallelism Marker	-	-

Table 10: Annotation guidelines for rhetorical components in CERD.

- For personification, regardless of the form-level category, the rhetorical components include the personification object (as the object) and the personification content (as the content).
- For hyperbole, regardless of the form-level category, the rhetorical components include the hyperbole object (as the object) and the hyperbole content (as the content).
- For parallelism, regardless of the form-level category, the rhetorical components only include the parallelism marker (as the connector).

A.2 Details of Dataset Annotation Process

The annotation process is illustrated in Figure 9 and introduced briefly in Section 3.2.2. In this section, we further discuss more details of the annotation process.

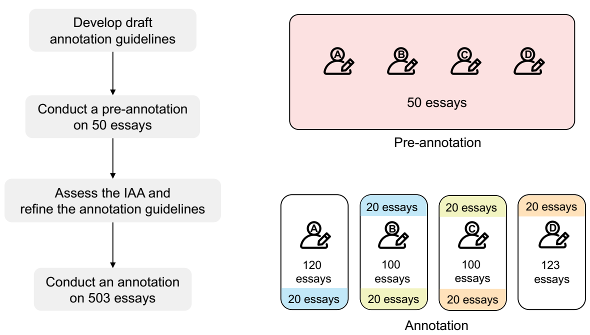

Figure 9: Annotation process of CERD.

The entire annotation process, from developing the draft annotation guidelines to conducting an annotation on 503 essays, took three months. To ensure the efficiency and quality of annotation, we held weekly online discussions to address common issues encountered during both the pre-annotation on 50 essays and the annotation on 503 essays. Furthermore, the 50 essays annotated during the pre-annotation process were not re-annotated or used subsequently.

B Dataset Statistics Details

The statistics of essays used to construct CERD are shown in Table 11. The total number of sentences in 503 essays is 10,349, with 355,352 tokens.

#Total Sentences	10,349
#Total Tokens	355,352
Avg. #Sentences per Essay	20.57
Avg. #Tokens per Essay	706.47
Avg. #Tokens per Sentence	34.34

Table 11: Statistics of essays used to construct CERD.

C Experimental Setups

We split CERD into training/validation/test sets, displayed in Table 12. To prevent data leakage, the dataset is split at the essay level, ensuring that the essays containing sentences in the training or validation sets are not included in the test set for any task.

Tasks	Type	#Sentences	#Tokens
RC/FC/CC/CE	Train	634	29,517
	Val	225	11,748
	Test	248	12,186
	Sum	**1,107**	**53,451**
RG	Train	404	52,969
	Val	158	22,246
	Test	169	24,239
	Sum	**731**	**99,454**

Table 12: Dataset splits of CERD.

We perform full parameter fine-tuning of RoBERTa on 24GB RTX 3090 GPUs and LoRA (Hu et al., 2021) fine-tuning of Qwen1.5-7B on 80GB A100 GPUs. The hyperparameters used in our experiments are listed in Table 13. Our models are fine-tuned using AdamW (Loshchilov and Hutter, 2017) optimizer and cosine learning rate scheduler.

Models	lr	bs	steps	r	α
RoBERTa	6×10^{-5}	32	30 epochs	-	-
Qwen1.5-7B$_{\text{Single}}$	2×10^{-4}	32	50 steps	32	32
Qwen1.5-7B$_{\text{Multi}}$	2×10^{-4}	32	250 steps	32	32

Table 13: Hyperparameters for fine-tuning RoBERTa and Qwen1.5-7B. "lr" refers to the learning rate. "bs" refers to the batch size. "r" and "α" refer to the hyperparameters used in LoRA.

D Prompt Templates

For all tasks and models, the prompt templates are used for both inference and fine-tuning. The prompt templates and inputs are originally written in Chinese. The English translations of the prompt templates are displayed in Figure 10, Figure 11 and Figure 12 respectively.

Prompt Template for Rhetoric Classification Task

Classify rhetorical devices into "metaphor", "personification", "hyperbole" and "parallelism". Each sentence may be literal or employ one, or multiple rhetorical devices.
Select one or more coarse-grained categories from "metaphor", "personification", "hyperbole" and "parallelism" and "literal" without repetition. Output directly in JSON format, with the field name "rhetoric" as an array, without explanation.
Output format:
{
"rhetoric": ["selected coarse-grained categories"]
}
Based on the requirements, directly output the answer in JSON format.
Sentence: `{{ sentence }}`
Rhetoric:

Prompt Template for Form Classification Task

Classify rhetorical devices into "metaphor", "personification", "hyperbole" and "parallelism". Each sentence may be literal or employ one, or multiple rhetorical devices.
Classify the form-level categories of metaphor into "simile", "metaphor" and "metonymy" based on the explicitness of rhetorical components. Simile includes comparator, tenor and vehicle, metaphor includes tenor and vehicle, and metonymy includes only the vehicle.
Classify the form-level categories of personification into "noun", "verb", "adjective" and "adverb" based on the parts of speech of rhetorical components. Noun refers to using nouns for people/objects to describe objects/people. Verbs refers to using verbs for people/objects to describe objects/people. Adjective refers to using adjectives for people/objects to describe objects/people. Adverb refers to using adverbs for people/objects to describe objects/people.
Classify the form-level categories of hyperbole into "direct hyperbole", "indirect hyperbole", and "mixed hyperbole" based on the form of hyperbole. Direct hyperbole directly exaggerates something, indirect hyperbole exaggerates something else to exaggerate a thing, and mixed hyperbole exaggerates using other rhetorical devices.
Classify the form-level categories of parallelism into "structure parallelism" and "sentence parallelism" based on the component of the parallelism item. Structure parallelism refers to the item servers as a specific grammatical component in the sentence, while sentence parallelism refers to the item serves as a complete sentence on its own.
Select one or more fine-grained form-level categories from "simile", "metaphor", "metonymy", "noun", "verb", "adjective", "adverb", "direct hyperbole", "indirect hyperbole", "mixed hyperbole", "structure parallelism", "sentence parallelism" and "literal" without repetition. Output directly in JSON format, with the field name "form" as an array, without explanation.
Output format:
{
"form": ["selected fine-grained form-level categories"]
}
Based on the requirements, directly output the answer in JSON format.
Sentence: `{{ sentence }}`
Form:

Prompt Template for Content Classification Task

Classify rhetorical devices into "metaphor", "personification", "hyperbole" and "parallelism". Each sentence may be literal or employ one, or multiple rhetorical devices.
Classify the content-level categories of metaphor into "concrete", "action" and "abstract" based on property of tenor. Concrete refers to the tenor can be seen, touched or imagined. Action refers to the tenor is an action, behavior or event. Abstract refers to the tenor is an abstract concept.
Classify the content-level categories of personification into "personification" and "anthropomorphism" based on the property of content. Personification refers to write about a non-human as if it were human. Anthropomorphism refers to write about something that is not A as if it were A, where A is non-human.
Classify the content-level categories of hyperbole into "amplification", "understatement" and "prolepsis". Amplification refers to exaggeration towards large, many, long or high. Understatement refers to exaggeration towards small, few, short or low. Prolepsis refers to mention a latter event before an earlier event.
Classify the content-level categories of parallelism into "coordination", "subordination" and "gradation". Coordination refers to changing the order of the items does not affect the coherence. Understatement refers to a logical order of precedence between items exists. Prolepsis refers to the meanings and emotions expressed by each item progressively intensify.
Select one or more fine-grained rhetorical content types from "concrete", "action", "abstract", "personification", "anthropomorphism", "amplification", "understatement", "prolepsis", "coordination", "subordination", "gradation," and "literal" without repetition. Output directly in JSON format, with the field name "content" as an array, without explanation.
Output format:
{
"content": ["selected fine-grained content-level categories"]
}
Based on the requirements, directly output the answer in JSON format.
Sentence: `{{ sentence }}`
Content:

Figure 10: Prompt templates for Tasks RC, FC and CC. `{{sentence}}` represents the input sentence.

Prompt Template for Component Extraction Task

Classify rhetorical devices into "metaphor", "personification", "hyperbole" and "parallelism". Each sentence may be literal or employ one, or multiple rhetorical devices.
Rhetorical components are categorized into three types: "connector", "object" and "content". The specific definitions for different rhetorical devices are as follows:
For metaphor, the connector is "comparator" and the object is "tenor" and the content is "vehicle". The comparator is the word connecting the tenor and the vehicle. The tenor is the object or concept being compared. The vehicle is the object or concept used for comparison.
For personification, the object is "personification object" and the content is "personification content". The personification object is the person or thing being described. The personification content is the similarities to the object.
For hyperbole, the object is "hyperbole object" and the content is "hyperbole content". The hyperbole object is the thing being described. The hyperbole content is the exaggerated description.
For parallelism, the connector is "parallelism item". The parallelism item is the parallelism marker.
Extract all rhetorical components from the sentence completely. Use JSON format for output, with "connector" as an array for connectors, "object" as an array for objects, and "content" as an array for contents. Do not explain. If there are no corresponding rhetorical components, the field value should be null.
Output format:
{
"connector": ["connectors in the sentence"],
"object": ["objects in the sentence"],
"content": ["contents in the sentence"]
}
Based on the requirements, directly output the answer in JSON format.
Sentence: `{{sentence}}`
Rhetorical Components:

Figure 11: Prompt template for Tasks CE. `{{sentence}}` represents the input sentence.

Prompt Template for Rhetoric Generation Task

Classify rhetorical devices into "metaphor", "personification", "hyperbole" and "parallelism". Each sentence may be literal or employ one, or multiple rhetorical devices.
Generate a sentence using the `{{ rhetoric }}` rhetorical device, with the requirement that the sentence includes `{% if rhetoric == 'metaphor' %}`the tenor is `{{ object }}{% elif rhetoric == 'personification' %}`the personification object is `{{ object }}{% else %}`the hyperbole object is `{{ object }}{% endif %}`. Use JSON format for output, with the field name "generation." Do not explain.
`{% if previous_sentences is not none %}`
The preceding sentences are as follows:
`{% for previous_sentence in previous_sentences %}`
`{{ previous_sentence }}`
`{% endfor %}`
`{% endif %}`
Output format:
{
"generation": "Generated sentence"
}
Based on the requirements, directly output the answer in JSON format.
Output:

Figure 12: Prompt template for Tasks RG. `{{rhetoric}}` represents the target coarse-grained category. `{{object}}` represents the target object. `{{previous_sentence}}` represents the preceding context.