Optimizing Chinese Lexical Simplification Across Word Types: A Hybrid Approach

Anonymous ACL submission

Abstract

This paper addresses the task of Chinese lexical simplification (CLS), which aims to replace complex words in a given sentence with simpler alternatives that retain the original meaning. One of the challenges in CLS is the scarcity of data resources. Previous unsupervised methods exhibit limited performance, while supervised methods struggle because of the lack of annotated data. We begin by evaluating the few-shot performance of several dialogue models at various scales on CLS, discovering that their effectiveness is sensitive to different word types. For large but expensive Large Language Models (LLMs), such as GPT-4, excel at simplifying in-dictionary common words and Chinese idioms compared to smaller models. Therefore, we propose an automatic knowledge distillation approach that generates training data for common words and Chinese idioms using GPT-4, and then use the training data to fine-tune smaller models in a unified but word-type aware manner. Besides, even GPT-4 encounters difficulties with out-of-dictionary (OOD) words. To address this, we employ a retrieval-based interpretation augmentation strategy, injecting relevant information from external sources into context. The experimental results show that the fine-tuned small models can obtain superior performance than GPT-4 for simplifying common words and idioms, which optimizes the balance between CLS performance and computational cost. The interpretation augmentation strategy can improve the performance of most models for simplifying OOD words.

1 Introduction

Lexical Simplification (LS) is the task of replacing complex words in a sentence with simpler alternatives while preserving their structure and original meaning. LS enhances text readability, benefiting a wide range of people, such as students (De Belder and Moens, 2010), non-native speakers (Paetzold and Specia, 2016), and individuals with cognitive impairments (Saggion, 2017). However, LS is a challenging task that requires both linguistic knowledge and contextual awareness.

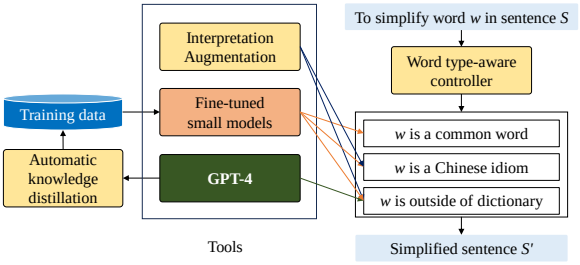

Figure 1: The general framework of the proposed word type-aware Chinese lexical simplification method.

This paper focuses on Chinese lexical simplification. One big barrier for CLS is the scarcity of training data. Consequently, recent researches concentrate on unsupervised methods that based on pre-trained language models (PLMs), e.g., the state-of-the-art CLS system, BERT-LS (Qiang et al., 2021), generates candidate words based on the pre-trained masked language model (MLM) BERT (Devlin et al., 2018). Despite its simplicity, BERT cannot fully understand the task, resulting in conservative substitutions and performance bottleneck.

We observe that recent large generative pre-trained language models, such as GPT-4 (Achiam et al., 2023), demonstrate superior task comprehension through task instructions and a few demonstrations. In contrast, smaller model (ranging from 700M to 7B parameters in this paper) fails to achieve satisfactory performance, highlighting the critical influence of model scale. However, the costs of training, maintaining, and invoking large language models are enormous. We face a trade-off between performance and cost when choosing between small and large models.

In this paper, we aim to improve small models by learning from and collaborating with GPT-4, expecting small models to achieve competitive per-

formance compared to GPT-4 while significantly reducing inference costs. To accomplish this goal, we present the following contributions:

- First, we conduct a comprehensive analysis of unsupervised CLS methods based on small and large language models to gain a deeper understanding of their strengths and weaknesses. We discover that GPT-4 has advantages in task understanding that minimizes the semantic loss of substitute compared to small models. Besides, linguistic resources can benefit small models when simplifying common words in the dictionary. However, all models exhibit a need for improvement in handling OOD words.

- Second, we propose a knowledge distillation framework called **PivotKD**, which expends in-dictionary common words oriented training data generated by GPT-4 for CLS task. PivotKD samples pivot words from a Chinese word dictionary and utilizes GPT-4 to generate sentences containing pivot words, which are then automatically replaced with alternatives belonging to varying levels of word complexity. The evaluation results demonstrate that small models fine-tuned on the training data yield superior performance compared to GPT-4 in simplifying common words and Chinese idioms.

- Third, we propose a retrieval-based interpretation augmentation strategy to enhance simplification of OOD words. This strategy involves querying search engine to acquire an interpretation of target complex word, which is then injected into the input sentence of model through a certain prompt template. Experimental results show that GPT-4 and most of the assessed fine-tuned small models exhibit improvements in simplifying OOD words.

Our research suggests that we can select the most appropriate model for each type of complex word to balance CLS performance and computational cost between models.

2 Related Work

Lexical simplification primarily follows a pipeline consisting of three main stages: the identification of complex words, the generation of substitution candidates, and the selection and ranking of candidates according to their simplicity and perplexity.

The identification of complex words aims to determine which word is considered complex in a sentence by a specific target population (Shardlow, 2013; Yimam et al., 2018; Dehghan et al., 2022). Aligning with current baseline method, we do not focus on this stage since complex words are given in test dataset. For further information, the reader is encouraged to consult a recent survey (North et al., 2023).

Knowledge-based methods Early research on lexical simplification relied on lexical knowledge databases to generate substitutes (Carroll et al., 1998; Drndarevic and Saggion, 2012). However, databases are not only expensive to develop and maintain, but also limited in word coverage.

Word embedding-based methods With the advent of deep learning, semantic similarity computation based on word embeddings has become a popular method for substitute generation and ranking (Paetzold and Specia, 2017). The cost of training word embedding models is significantly lower than that of manually constructing knowledge databases, and these methods also largely alleviate the problem of insufficient word coverage.

PLM-based methods Subsequently, pre-trained language models (PLMs) show strong ability in capturing contextual semantic information, and have been proposed for lexical simplification. For example, BERT-LS (Qiang et al., 2020) introduced an unsupervised method that employs BERT to generate substitutions for complex words based on the encoding of the surrounding context. PromptLS (Vásquez-Rodríguez et al., 2022) found that fine-tuning PLMs can achieve better performance compared to unsupervised approaches. ConLS (Sheang et al., 2022) fine-tuned an encoder-decoder model T5 for substitute generation, which naturally predicts simple words with multiple tokens. One challenge in fine-tuning is the scarcity of supervised training data for certain languages, such as Chinese.

LLM-based methods Recently, large language models have been applied for lexical simplification through prompt learning-based methods. It shows that GPT-3 is capable of comprehending the task and learning task instructions with a few demonstrations, achieving good performance in the English language (Aumiller and Gertz, 2022). This indicates that LLMs have extensive linguistic knowledge and exhibit strong in-context learning

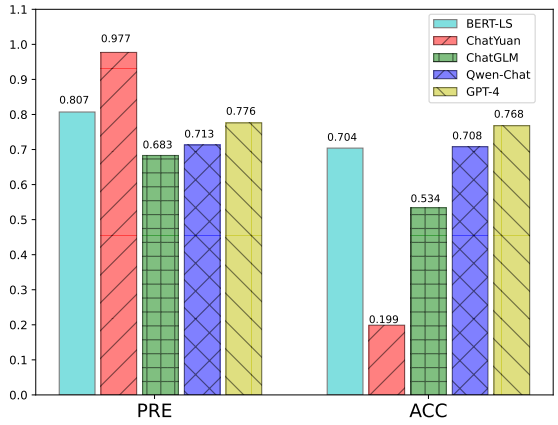

Figure 2: Overall results of BERT-LS and three LLMs in the few-shot setting.

Models	Common		Idioms		OOD	
	PRE	ACC	PRE	ACC	PRE	ACC
BERT-LS	86.8	76.9	70.8	**41.7**	34.0	28.3
ChatYuan	**98.4**	22.4	**91.7**	8.30	**94.3**	3.80
ChatGLM	72.0	58.3	62.5	29.2	39.6	22.6
Qwen-Chat	75.6	74.9	25.0	25.0	55.6	55.6
GPT-4	81.6	**80.7**	29.2	29.2	66.0	**66.0**

Table 1: Detailed results of simplifying three types of complex words.

ability. Thus, the critical challenge lies in identifying effective strategies to guide LLMs to generate the desired predictions. Nonetheless, the processes of training, deploying, and utilizing LLMs remain highly expensive.

Knowledge distillation Knowledge distillation (KD) aims to enhance the performance of a smaller student model by leveraging the knowledge from a larger teacher model (Kim and Rush, 2016). In this study, we adopt a black-box KD method since only predictions and hyper-parameters are available when utilizing LLMs like GPT-4 to obtain high-quality training data for CLS.

3 Task, Data and In-Depth Analysis

In this section, we provide a brief introduction to data resource and evaluation metrics, followed by an analysis of representative baselines that use BERT and LLM.

3.1 Lexical Simplification Settings

An LS system first identifies complex words in a sentence and then generates candidate substitutes, which is known as substitute generation (SG). Considering complex word identification depends on a target population, we assume that a sentence and a target complex word are given following previous work (Qiang et al., 2021).

Formally, given a sentence $\mathbf{s}$ and a complex word w in $\mathbf{s}$, the task is to generate a simpler alternative v, a word or a group of words, to form a simpler sentence $\mathbf{s}'$, which is expected to be smooth, clear, and maintain the same meaning as $\mathbf{s}$.

3.2 Dataset and Metrics

3.2.1 Dataset

We use the publicly available Chinese lexical simplification dataset HanLS (Qiang et al., 2021). HanLS includes 524 sentences, each containing a complex word from the advanced level of the Chinese Proficiency Test (Hanyu Shuiping Kaoshi, HSK), and each complex word has 8.51 annotated simple substitutes on average as reference answers.

Evaluation Metrics Following previous work (Paetzold and Specia, 2016), we use precision and accuracy as metrics.

Precision (PRE): The proportion of predicted substitutes that are the original complex word itself or appear in the reference answers.

Accuracy (ACC): The proportion of predicted substitutes that are different from the original complex word and appear in the reference answers.

PRE and ACC are complementary metrics. A higher PRE indicates a lower probability of predicting misleading or incorrect words, reflecting the system's robustness. Considering a conservative system may retain a large number of original words to achieve high PRE, thus ACC is involved to measure its simplification ability.

3.3 Baseline Systems

We adopt BERT-LS (Qiang et al., 2021) and several dialogue models of different scale using few-shot learning as baselines, analyzing their behaviors, and gaining a deeper understanding of their strengths and weaknesses.

3.3.1 BERT-LS

The input of BERT-LS is formed by concatenating the original sentence with its copy, in which the target complex word is replaced with [MASK]. BERT then predicts substitutes in a masked position.

Since a Chinese word often consists of multiple Chinese characters and BERT's tokenizer operates at character level, BERT-LS accommodates predictions with varying numbers of [MASK] tags (e.g., one to four). If complex word is listed in the Chinese synonymy thesaurus (Mei, 1983), its

Instruction	任务是将句子中给定的难词(由#标记)替换为一个简单的词，同时保持句子的结构和意思不变并尽量流畅。 (The task is to replace the complex word (marked by #) in the sentence with a simple word, while keeping the structure and meaning of the sentence unchanged and as smooth as possible.)
Input 1	练习与理解不是#截然#对立的，而是相辅相成的。 (Practice and understanding are not #thoroughly# opposed but complement each other.)
Response	练习与理解不是#完全#对立的，而是相辅相成的。 (Practice and understanding are not #completely# opposed but complement each other.)
Input 2	他#呕心沥血#写了这本书。
Response	[Let LLM generate the response]

Figure 3: An example of instruction and demonstration design for prompting LLMs for CLS.

Original sentence	小猪#似懂非懂#，心想幸福怎么会是我的尾巴呢？ The little pig #seemed to understand but didn't really understand#, thinking how could happiness be my tail?
BERT-LS	小猪#不解#，心想幸福怎么会是我的尾巴呢？ The little pig #puzzled#, thinking how could happiness be my tail?
ChatGPT	小猪#有些糊涂#，心想幸福怎么会是我的尾巴呢？ The little pig #was a bit confused#, thinking how could happiness be my tail?

Figure 4: The outputs of BERT-LS and ChatGPT on simplifying a Chinese idiom.

synonyms are used as substitutes. Finally, BERT-LS ranks these substitutes with multiple sources of evidence, including word embeddings, BERT scores, and word frequencies.

3.3.2 Dialogue models

We use GPT-4-1106-preview (GPT-4 for short) and explore its performance through few-shot learning, incorporating task instructions and three demonstrations within the context. Figure 3 shows an illustrative example. We also investigate three open-source small Chinese dialogue models Qwen1.5-7B-Chat (Qwen-Chat for short), ChatGLM2-6B (ChatGLM for short) (Du et al., 2022) and ChatYuan-large-v2 (700m parameters, ChatYuan for short) (Xuanwei Zhang and Zhao, 2022) under the same setting for comparison. Specifically, for GPT-4, we extract predictions from its responses; For Qwen, ChatGLM and ChatYuan, we extract the predicted top 10 substitutes and rank them the same as BERT-LS.

3.4 Analysis and Discussion

3.4.1 Overall Results

Figure 2 shows the overall results. ChatYuan does not perform much simplification, as shown by its high PRE scores and low ACC scores. For dialogue models, the performance is directly correlated with model scale. BERT-LS also achieves impressive results with the aid of external linguistic resources. Overall, GPT-4 demonstrates the most robust capabilities of task understanding and instruction following, suggesting its potential for knowledge distillation.

Original sentence	我最近网上冲浪的时候总能刷到好多#镁铝#哦！ I always see a lot of #magnesium aluminum# when I surf the internet recently!
BERT-LS	我最近网上冲浪的时候总能刷到好多#金属#哦！ I always see a lot of #metal# when I surf the internet recently!
ChatGPT	我最近网上冲浪的时候总能刷到好多#美食#哦！ I always see a lot of #gourmet food# when I surf the internet recently!

Figure 5: The outputs of BERT-LS and ChatGPT on simplifying an OOD word.

3.4.2 Analysis

We analyze the relation between the models' performance and the types of complex words. Specifically, we categorize complex words into three types:

- **Common words**: Refer to non-idiomatic words included in the Chinese word dictionary named Xinhua Zidian, which covers more than 320k words.
- **Chinese idioms**: Idioms or Chengyu, an crucial component of the Chinese language, typically composed of four Chinese characters that convey a moral or lesson in a concise and elegant manner.
- **Out-of-dictionary (OOD) words**: Refer to words excluded in Xinhua Zidian, majorly consists of new words like internet slang.

Table 1 presents the performance of different models on these types of complex words. GPT-4 surpasses other models in simplifying common words, but lags behind BERT-LS on Chinese idioms. In addition, GPT-4 demonstrates a remarkable advantage in simplifying OOD words, although none of the models achieves satisfactory results.

We compare the predictions of GPT-4 and BERT-LS in simplifying Chinese idioms and find that GPT-4 is undervalued. Figure 4 shows an example. GPT-4 often generates phrases that are more

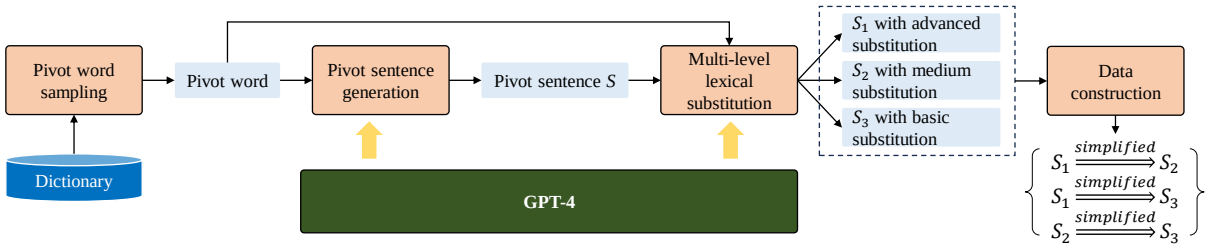

Figure 6: The main workflow of the PivotKD framework for generating CLS data based on ChatGPT.

任务是将句子中给定的词(由#标记)替换为同义词，同时保持句子的结构和意思不变并尽量流畅。词语难度可以分为高级、中级和基本三个等级，请为每个等级分别生成n个替换词。
(The task is to replace the given words in the sentence (marked by #) with synonyms, while maintaining the structure and meaning of the sentence as much as possible. The synonyms should be categorized into three levels of difficulty: advanced, intermediate, and basic. Please generate n replacement words for each level.)

Figure 7: An instruction for 3-level lexical substitution.

coherent and smooth than single words. In contrast, BERT-LS primarily predicts single words, aligning with the format of annotated gold answers. Therefore, annotations for idioms in HanLS are expected to be extended to phrase level.

Simplifying OOD words is a challenge for all models. Figure 5 shows an example. The term "magnesium-aluminum" is a Chinese internet slang that sounds like "beauty" and refers to beautiful women. Neither BERT-LS nor ChatGPT can produce suitable answers in most cases, probably because they have limited knowledge of these OOD words.

In summary, our in-depth analysis reveals the following key observations: 1) Small models struggle to grasp the task effectively when not provided with adequate supervision. 2) The performance of the task is highly dependent on the type of complex words.

4 The Proposed Method

We propose a framework, comprising three key modules: automatic knowledge distillation, retrieval-based interpretation augmentation, and a word type-aware controller.

4.1 Automatic Knowledge Distillation

Our objective is to develop a high-quality CLS training dataset through the knowledge distillation of GPT-4. We anticipate that the generated sentences will be grammatically correct, free of spelling errors, cover a diverse range of topics, and include precise substitutes. Specifically, we propose an automatic knowledge distillation strategy named **PivotKD**, which relies on GPT-4 and does not require human intervention. Figure 6 illustrates its main workflow.

4.1.1 Pivot Word Sampling

We sample words from the Xinhua Zidian dictionary and refer to these sampled words as *pivot words*. Our preliminary analysis indicates that GPT-4 performs well on simplifying common words and idioms but struggles with OOD words. To ensure accurate generation of substitutes, we avoid collect OOD words as pivot words. Besides, We limit the word to be a noun, verb, adjective, adverb or idiom. To enhance diversity, each word can be sampled only once.

4.1.2 Pivot Sentence Generation

Given a pivot word, we instruct GPT-4 to generate a sentence containing this word, which leverages the strengths of GPT-4 in the following aspects: 1) GPT-4 is capable of generating correct and coherent sentences, thereby avoiding the spelling and grammar errors commonly found in data collected from the web or existing corpus; 2) GPT-4 can generate sentences covering diverse topics since we do not constrain the topics during the sampling of pivot words and sentence generation. we can assume that the generated dataset is topic-independent.

4.1.3 Multi-level Lexical Substitution

Following the generation of a pivot sentence with a pivot word, we then direct GPT-4 to generate substitutes at three distinct levels of word complexity (*advanced*, *intermediate*, and *basic*) to replace the pivot word in the sentence. This step eliminates the need to predefine the complexity of pivot words, allowing GPT-4 to act as the judge. The advantage is that the absolute complexity of word is indeterminate and varies among different users, while

the relative complexity between words is objective and certain. We convey the requirements to GPT-4 through instructions as shown in Figure 7. We allow GPT-4 to generate a word as a substitute for common complex words while a word or a phrase as a substitute for Chinese idioms.

4.1.4 Data Construction

For one pivot word, we employ GPT-4 to generate n simpler substitutes across each level of word complexity, from which $3n$ sentences are collected. Then we construct a set of sentence pairs based on the complex-to-simple criteria. Specifically, a sentence pair $(\mathbf{s}, \mathbf{s}')$ is chosen if the word complexity level of the substitute in $\mathbf{s}$ is higher than that in $\mathbf{s}'$. Notice that the sentence pair may not necessarily contain the pivot word itself, which serves solely to provide semantic guidance.

4.1.5 Instruction Fine-tuning

We conduct instruction fine-tuning with Qwen-Chat, ChatGLM and ChatYuan. For ChatYuan, all parameters are fine-tuned, while LoRA (Hu et al., 2021) is used to fine-tune Qwen-Chat and ChatGLM. The training data, derived from the constructed sentence pairs $\{(\mathbf{s}, \mathbf{s}')\}$, is transformed into a question-answer format. The input question incorporates task-specific instructions, i.e., to simplify common complex words or Chinese idioms, and demarcates the complex word with the tag #. The output answer is the corresponding simplified sentence $\mathbf{s}'$, with a substitute that is marked with the tag # as well.

4.2 Retrieval-based Interpretation Augmentation

OOD words present a huge challenge for simplification because new internet slang emerge continuously, while pre-trained models remain static. Motivated by recent work on retrieval-augmented LLMs (Lewis et al., 2020; Nakano et al., 2021), we propose a retrieval-based interpretation augmentation approach that dynamically collect word interpretations from the web to alleviate the knowledge gap.

Retrieving Word Interpretation The majority of OOD words are internet slang, for which pre-trained models may lack relevant knowledge. However, interpretation for these words are often available online. We crawled the search results from Baidu ZhiDao search engine through a query "*What does the word* [***complex word***] *mean?*", and extract the content of the top k snippets containing the word as the interpretation.

Injecting Interpretation for Inference The obtained interpretation is injected into the input question through a manually designed prompt template, which is applicable for both GPT-4 and the fine-tuned small models, offering flexibility and ease of integration as a plug-in.

4.3 Word-type aware Controlled Inference

Currently, we have several models for handling CLS, including GPT-4, fine-tuned small models, and retrieval-based interpretation augmentation. We aim to find an effective and efficient way to integrate these components.

Our solution is based on the observation that the performance of CLS is highly sensitive to the type of complex words. We suggest to use small models as the basic model, supplemented by GPT-4 or retrieval-based interpretation augmentation for handling Chinese idioms or OOD words. The following evaluation section will discuss optimal strategies for different types of complex words.

5 Evaluation

5.1 Experimental Settings

For PivotKD, we sample 5,000 pivot words from Xinhua Zidian. We avoid using the complex words in test dataset HanLS as pivot words to prevent data leakage. Since the sentences are entirely generated by GPT-4, there is no overlap with HanLS.

The multi-level lexical substitution module generates $n = 1$ substitution for each level of word complexity. Finally, we establish a training dataset comprising 8,962 sentence pairs, covering 4,269 distinct substitutions.

We also conduct a human evaluation on 500 samples from the dataset. For each sentence pair, we employ two persons to judge whether the relative word complexity between them are *clearly reasonable*, *hard to distinguish*, or *contradiction or irrelevant*. The proportions of these options are 70%, 25%, and 5% respectively, indicating that the quality of the dataset produced by PivotKD is acceptable.

We fine-tune ChatYuan for 1 epoch, and fine-tune Qwen-Chat and ChatGLM with LoRA for 3 epochs. For retrieval-based interpretation augmentation, we use the top $k = 1$ snippet since it already provides satisfactory performance.

Models	Common		Idiom		OOD		All	
	PRE	ACC	PRE	ACC	PRE	ACC	PRE	ACC
BERT-LS (frozen)	86.8	76.9	70.8	41.7	34.0	28.3	80.7	70.4
ChatGPT (frozen)	81.6	80.7	29.2	29.2	66.0	66.0	77.6	76.8
+RIA	-	-	-	-	**79.3**	**79.3**	-	-
ChatYuan (frozen)	*98.4*	22.4	*91.7*	8.3	*94.3*	3.8	*97.7*	19.9
ChatYuan (full-tuning)	85.2	80.5	**75.0**	**70.8**	56.6	45.3	81.8	76.5
+RIA	**86.3**	**82.5**	**75.0**	**70.8**	**68.0**	**62.3**	**84.0**	**80.0**
ChatGLM (frozen)	72.0	58.3	62.5	29.2	39.6	22.6	68.3	53.4
ChatGLM (LoRA)	83.0	82.1	**66.7**	**66.7**	60.4	58.5	80.0	79.0
+RIA	**83.6**	**82.5**	58.3	58.3	**64.2**	**62.3**	**80.5**	**79.4**
Qwen-Chat (frozen)	75.6	74.9	25.0	25.0	56.6	56.6	70.8	71.3
Qwen-Chat (LoRA)	84.1	**83.6**	**66.7**	**66.7**	49.1	49.1	**79.7**	**79.4**
+RIA	**84.3**	83.2	**66.7**	**66.7**	45.3	45.3	79.5	78.6
Hyb-CLS	84.1	**83.6**	**75.0**	**70.8**	**79.3**	**79.3**	83.2	**82.6**

Table 2: System comparisons on HanLS. RIA indicates utilizing retrieval-based interpretation augmentation during inference. The results with the highest accuracy are bolded, and the best results obtained across all models are marked with underlines.

5.2 Experimental Results

5.2.1 Auto-Evaluation

Table 2 presents the overall results of BERT-LS, GPT-4, ChatYuan, ChatGLM, Qwen-Chat and some variants, and specified with the separated results on three types of complex words. We observe several trends:

(1) **The effects of PivotKD** The results indicate that small models significantly benefit from supervised instruction fine-tuning. These models exhibit a better understanding of the task and achieve substantial improvements compared to their unsupervised frozen counterparts, surpassing GPT-4 and markedly outperforming the baseline method, BERT-LS, in overall performance. This demonstrates the effectiveness of PivotKD and the quality of the constructed training dataset.

The effect of increasing the number of training samples on the performance of the fine-tuned ChatYuan and ChatGLM models is illustrated in Figure 8. Generally, the performance of the models can be enhanced by increasing the number of training samples. ChatGLM can reach a steady performance using relatively fewer samples, while the performance of ChatYuan has a consistent improvement with the increase of the training samples, indicating that smaller models may need more training data.

(2) **The effects of retrieval-based interpretation augmentation (RIA)** The application of RIA significantly enhances the performance of most models in simplifying OOD words, confirming that the retrieved word interpretations provide effective information that can be effectively utilized by the models. However, this improvement is not observed in Qwen-Chat, as well as simplification on common words and idioms. One possible reason is that injected interpretations may introduce noise when the model has already embedded adequate knowledge through pre-training on extensive and broad-domain corpus.

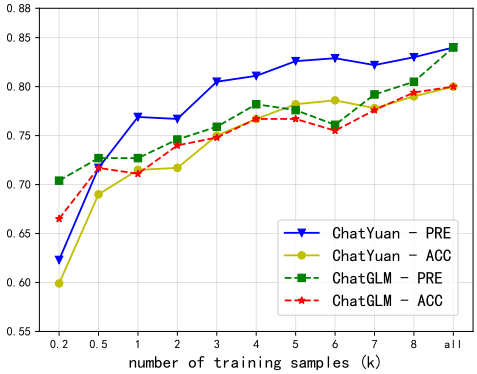

Figure 8: The effects of the number of training samples for fine-tuning ChatYuan and ChatGLM.

(3) **Performance on different word types** The performance of the fine-tuned models on different types of complex words remains highly variable, performing better on common words than on Chinese idioms and OOD words. Notably, ChatYaun, ChatGLM and Qwen-Chat both surpass GPT-4 on common words and idioms but still lag behind on OOD words, demonstrating that fine-tuned smaller models can compete with LLMs on words that are easier to simplify.

(4) **A hybrid approach** Since the performance

of CLS is word-type sensitive, we suggest that we can optimize performance by selecting the best-performing method for each word type across all models and variants in practice. Thus we refer to this strategy as **Hyb-CLS** (a hybrid approach for CLS). For example, we can employ the fine-tuned Qwen-Chat with LoRa for common words, the fine-tuned ChatYuan with RIA for Chinese idioms, and GPT-4 with RIA for handling OOD words.

Please note that the constructed training dataset and the search results for OOD words in HanLS will be open-sourced following further review and verification. The resources are expected to be available at GitHub.

5.2.2 Human Evaluation

The system outputs may be reasonable but outside the reference answers. So we conduct a human evaluation. We sample 20 common words, 20 Chinese idioms, and 20 OOD words from HanLS. Three raters rate the mixed outputs of different systems according to the following criteria:

- 4 points: The substitute is simpler and has the same meaning as the complex word without any information loss, and the resulting sentence is smooth.
- 2 points: The substitute is simpler and has the same meaning as the complex word, but there is a loss of information in terms of details and degree, or the output is not so smooth.
- 0 points: The substitute is not simpler or its meaning differs from the complex word.

Table 3 shows the averaged human evaluation results. We can see that GPT-4 still has an advantage in simplifying idioms and OOD words, indicating the strong ability of very large language models. The fine-tuned small models achieve similar performance and the performance is also close to GPT-4. RIA is verified to be effective as well. The human evaluation confirms that with proper manipulation of the fine-tuned small models and large models, it is possible to keep a balance between performance and cost.

Models	Common	Idioms	OOD	All
BERT-LS	3.17	1.23	0.6	1.67
ChatGPT + RIA	**3.70**	**3.17**	**2.60**	**3.16**
ChatGLM	3.37	2.60	1.93	2.63
+RIA	3.50	2.83	2.53	2.95
ChatYuan	3.37	2.60	1.50	2.49
+RIA	3.43	2.83	2.2	2.82

Table 3: Human evaluation of three models in different settings. The rating ranges from 0 (worst) to 4 (best).

6 Conclusion

This paper presents a word-type aware approach for Chinese lexical simplification. The core idea is to consider the types of complex words to integrate small and large language models effectively and efficiently. For common complex words and Chinese idioms, we propose an automatic knowledge distillation framework, PivotKD, to generate training data using GPT-4 for fine-tuning small models, which can outperform GPT-4 in simplifying common words. For addressing the issue of OOD words, we propose a retrieval-based interpretation augmentation strategy, which effectively improves the performance on OOD words. Consequently, we are able to control the inference strategy according to the type of complex words, thus efficiently combining small and large models to achieve optimal performance.

7 Limitations

There are three possible limitations of this work. First, our evaluation is based on the HanLS dataset, which is limited in size and coverage. We plan to extend the dataset. Second, we assume that ChatGPT understand the lexical difficulty levels, but we verify this assumption by analyzing the relative lexical difficulty between a pair of words in the generated data. More detailed and specially designed probing analysis can be conducted. Third, this paper focuses on Chinese lexical simplification, but the proposed method can be potentially applied to other languages. We plan to address these limitations in the future work.

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*.

Dennis Aumiller and Michael Gertz. 2022. Unihd at tsar-2022 shared task: Is compute all we need for lexical simplification? In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 251–258.

John Carroll, Guido Minnen, Yvonne Canning, Siobhan Devlin, and John Tait. 1998. Practical simplification

of english newspaper text to assist aphasic readers. In *Proceedings of the AAAI-98 Workshop on Integrating Artificial Intelligence and Assistive Technology*, pages 7–10. Citeseer.

Jan De Belder and Marie-Francine Moens. 2010. Text simplification for children. In *Prroceedings of the SIGIR workshop on accessible search systems*, pages 19–26. ACM; New York.

Mohammad Dehghan, Dhruv Kumar, and Lukasz Golab. 2022. GRS: Combining generation and revision in unsupervised sentence simplification. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 949–960, Dublin, Ireland. Association for Computational Linguistics.

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*.

Biljana Drndarevic and Horacio Saggion. 2012. Towards automatic lexical simplification in spanish: An empirical study. In *PITR@ NAACL-HLT*, pages 8–16.

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. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Jiaju Mei. 1983. *Synonymy Thesaurus of Chinese Words*. Shanghai Lexicographical Publishing House, Shanghai.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. 2021. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*.

Kai North, Marcos Zampieri, and Matthew Shardlow. 2023. Lexical complexity prediction: An overview. *ACM Computing Surveys*, 55(9):1–42.

Gustavo Paetzold and Lucia Specia. 2016. Unsupervised lexical simplification for non-native speakers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.

Gustavo Paetzold and Lucia Specia. 2017. Lexical simplification with neural ranking. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 34–40.

Jipeng Qiang, Yun Li, Yi Zhu, Yunhao Yuan, and Xindong Wu. 2020. Lexical simplification with pretrained encoders. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8649–8656.

Jipeng Qiang, Xinyu Lu, Yun Li, Yunhao Yuan, and Xindong Wu. 2021. Chinese lexical simplification. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:1819–1828.

Horacio Saggion. 2017. Automatic text simplification. *Synthesis Lectures on Human Language Technologies*, 10(1):1–137.

Matthew Shardlow. 2013. A comparison of techniques to automatically identify complex words. In *51st Annual Meeting of the Association for Computational Linguistics Proceedings of the Student Research Workshop*, pages 103–109, Sofia, Bulgaria. Association for Computational Linguistics.

Kim Cheng Sheang, Daniel Ferrés, and Horacio Saggion. 2022. Controllable lexical simplification for english. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 199–206.

Laura Vásquez-Rodríguez, Nhung Nguyen, Matthew Shardlow, and Sophia Ananiadou. 2022. Uom&mmu at tsar-2022 shared task: Prompt learning for lexical simplification. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 218–224.

Liang Xu Xuanwei Zhang and Kangkang Zhao. 2022. Chatyuan: A large language model for dialogue in chinese and english. https://github.com/clue-ai/ChatYuan.

Seid Muhie Yimam, Chris Biemann, Shervin Malmasi, Gustavo Paetzold, Lucia Specia, Sanja Štajner, Anaïs Tack, and Marcos Zampieri. 2018. A report on the complex word identification shared task 2018. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 66–78, New Orleans, Louisiana. Association for Computational Linguistics.

A Details in Dataset Construction

We sampled 5,000 pivot words for data generation. After constructing sentence pairs according to the

A sentence pair	
Complex	他#悄无声息#地走进房间，以免吵醒熟睡中的孩子。 (He slipped into the room #stealthily#, so as not to wake the sleeping child.)
Simple	他#偷偷#地走进房间，以免吵醒熟睡中的孩子。 (#quietly#.)
A training sample	
Prompt	他#悄无声息#地走进房间，以免吵醒熟睡中的孩子。 (Same as the sentence above) 你的任务是将句子中给定的难词#悄无声息#替换为一个简单的词或短语，同时保持句子结构和意思不变并尽量流畅。 (The task is to replace the complex word #stealthily# in the sentence with a simple word or phrase, while keeping the structure and meaning of the sentence unchanged and as smooth as possible.)
Response	他#偷偷#地走进房间，以免吵醒熟睡中的孩子。 (#quietly#.)

Table 4: An example of a constructed sentence pair and the corresponding training sample.

difficulty levels of the substitutions, we use some rules to further reduce noise.

Firstly, we excluded substitutions that exist in the target complex word list of HanLS, thus there is no overlap between the augmented data and the test data. Secondly, we constrain that for the complex word in each constructed sentence pair should be in the Xinhua Zidian dictionary.

Some basic statistics of the final dataset for fine-tuning the small models are shown in Table 5. Table 4 shows a constructed sentence pairs and the corresponding training sample.

B Parameter Settings

Table 6 shows the infrastructure for conducting our experiments. The hyper-parameters used for fine-tuning ChatYuan-large-v2, ChatGLM2-6B and Qwen1.5-7B-Chat are listed in Table 7, Table 8 and Table 9respectively. We set the value of the temperature parameter of ChatGPT API as 0 because we emphasize the generation quality and control the diversity through pivot words.

Attribute	Value
Sentence pairs	8,962
Avg. length of sentences	22.38
Distinct substitutions	4,269
Avg. length of substitutes	2.04

Table 5: Basic statistics of the augmented dataset via PivotKD.

Settings	Value
GPU	Nvidia A6000
GPU memory	48 GB
CPU	AMD EPYC 7542
OS	Ubuntu 20.04.5 LTS
Pytorch version	1.31.1
CUDA version	11.6

Table 6: Infrastructure for conducting our experiments.

C Human Rating

We conducted human evaluation for ChatGLM, ChatYuan, ChatGPT and BERT-LS. We sampled 20 words for each type of complex word, merging the predictions of the models for human evaluation.

We set the ratings to be 0,2,or 4 according to the criteria introduced in the main content. We employed three raters who are students in a normal university. They are volunteers and unaware of the model information of these predictions. We reported the average rating for each prediction. The mean variance of the ratings between different raters is 0.55. Table 10 demonstrate one example of predictions and human ratings.

Hyper-parameters	Value
max_seq_length	512
num_epoch	1
learning_rate	5e-5
scheduler	cosine
batch_size	16

Table 7: Hyper-parameter settings used for fine-tuning ChatYuan-large-v2.

Hyper-Parameters	Value
max_seq_length	512
num_epoch	3
learning_rate	5e-5
scheduler	cosine
batch_size	16
lora_rank	8
lora_alpha	32
lora_dropout	0.1

Table 8: Hyper-parameter settings used for fine-tuning ChatGLM2-6B.

Hyper-Parameters	Value
max_seq_length	512
num_epoch	3
learning_rate	5e-5
scheduler	cosine
batch_size	4
lora_rank	8
lora_alpha	16
lora_dropout	0

Table 9: Hyper-parameter settings used for fine-tuning Qwen1.5-7B-Chat.

Sentence	那个年代，汤姆有一点叛逆，有一个梦想就是去当#绿林好汉#. (In that era, Tom was a bit rebellious, and he had a dream of becoming an #outlaw hero#.)
Prediction	那个年代，汤姆有一点叛逆，有一个梦想就是去当#土匪(bandit)#.
Rater A	2
Rater B	0
Rater C	2

Table 10: One example of predictions and human ratings.