CN-AutoMIC: Distilling Chinese Commonsense Knowledge from Pretrained Language Models

Chenhao Wang[1,2], **Jiachun Li**[1,2], **Yubo Chen**[1,2], **Kang Liu**[1,2,3] and **Jun Zhao**[1,2]
[1]National Laboratory of Pattern Recognition, Institute of Automation, CAS, Beijing, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3]Beijing Academy of Artificial Intelligence, Beijing, China
{chenhao.wang,jiachun.li,yubo.chen,kliu,jzhao}@nlpr.ia.ac.cn

Abstract

Commonsense knowledge graphs (CKGs) are increasingly applied in various natural language processing tasks. However, most existing CKGs are limited to English, which hinders related research in non-English languages. Meanwhile, directly generating commonsense knowledge from pretrained language models has recently received attention, yet it has not been explored in non-English languages. In this paper, we propose a large-scale Chinese CKG generated from multilingual PLMs, named as **CN-AutoMIC**, aiming to fill the research gap of non-English CKGs. To improve the efficiency, we propose generate-by-category strategy to reduce invalid generation. To ensure the filtering quality, we develop cascaded filters to discard low-quality results. To further increase the diversity and density, we introduce a bootstrapping iteration process to reuse generated results. Finally, we conduct detailed analyses on CN-AutoMIC from different aspects. Empirical results show the proposed CKG has high quality and diversity, surpassing the direct translation version of similar English CKGs. We also find some interesting deficiency patterns and differences between relations, which reveal pending problems in commonsense knowledge generation. We share the resources and related models for further study[1].

1 Introduction

Compiling large-scale commonsense knowledge resources is a long-term goal of the AI community. Recent efforts focus on constructing commonsense knowledge graphs (CKGs) via manually compiling (Speer et al., 2017; Sap et al., 2019; Mostafazadeh et al., 2020) or automatic extraction (Tandon et al., 2014; Romero et al., 2019; Zhang et al., 2020; Nguyen et al., 2021). These projects have shown benefits for a wide range of downstream tasks (Lin et al., 2019; Tian et al., 2020; Ammanabrolu et al., 2021).

However, most CKG projects are limited to English, which hinders further research in other languages. To go beyond such an English-centric trend in commonsense knowledge research, there are some challenging issues. First, directly translating English CKGs is not enough. For example, `(PersonX is a little girl, xWant, to ask Christmas presents)` is a triple from a CKG crowdsourced by English users (Sap et al., 2019), but it is not common in non-Christian cultures. In fact, such resources reflect only the commonsense perspectives of English-speaking communities. The translation will omit the cultural differences in other languages, and even implicitly exacerbate the English-centric bias (Mehrabi et al., 2021). Therefore, when creating CKGs for new languages, it is better to rely on native speakers and corpora. Second, current common practices in English CKG construction, i.e. manually compiling and automatic extraction, are difficult to generalize. For manually compiling, creating human-authored CKGs for each new language is very expensive. For automatic extraction, current pipelines rely on English-specific hand-craft patterns or language processing tools, which are not available to other languages. Therefore, when creating CKGs for new languages, the cost and availability of construction scheme should also be concerned.

Recent work reveals pretrained language models (PLMs) can be a new source to generate commonsense knowledge (Bosselut et al., 2019; Nguyen and Razniewski, 2022). The Latest research indicates the CKG built from huge PLMs (e.g. GPT-3) can surpass the crowdsourced ones in quantity and quality (West et al., 2021), and only a small amount of human-authored data are required for prompting and filtering. Interestingly, we find this way could be the ideal start point to construct CKGs for new languages, since PLMs can be trained on the corpora of target languages, and the generation process is low-cost and independent of language-

[1]http://github.com/wchrepo/cnautomic/

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 9253 - 9265

specific tools. However, up to now, work in this thread has not extended to non-English languages. The main challenge of this paradigm is that the generation quality and diversity are often conflicting and difficult to control. To sample diverse results, the generation should be run extensive times, and a large number of generated results are invalid and need to be filtered out. For new languages, as there is often no comparable PLM to GPT-3 (Brown et al., 2020) in English, the generated results will be even noisier. Therefore, we need to reduce unnecessary generation to increase the efficiency and take measures to ensure the quality.

In this paper, we propose a framework to distill commonsense knowledge from multilingual pretrained language models. To increase the generation efficiency, we propose a generate-by-category strategy to reduce invalid generation. To ensure the filtering quality, we propose cascaded filters to discard low-quality results. To further increase the diversity and density, we introduce a bootstrapping process. Based on the framework, we propose a large-scale Chinese commonsense knowledge graph, **CN-AutoMIC** (**Auto**matically Obtained **M**ach**I**ne **C**ommonsense). To the best of our knowledge, it is the first non-English CKG built from pretrained language models. The empirical results show the proposed CKG has high quality and diversity. We also discuss some interesting deficiencies that need further solutions. We summarize the contribution as follows.

- We propose a framework to distill commonsense knowledge from multilingual PLMs, incorporating several novel strategies to improve the generation efficiency and quality.
- We propose the first large-scale Chinese commonsense knowledge graph built with large-sized PLMs, **CN-AutoMIC**. Its high-quality subset contains 1.1M triples with an accuracy of **87.2%**. Besides the CKG, we also release small-sized commonsense models trained on it, named as **CN-COMET**.
- We conduct comprehensive evaluation and analysis for CN-AutoMIC and CN-COMET. Besides verifying the quality and diversity, we also get some valuable observations about the deficiencies of generation. Considering generating commonsense knowledge with PLMs is still not fully explored, our findings can provide more insights into this topic.

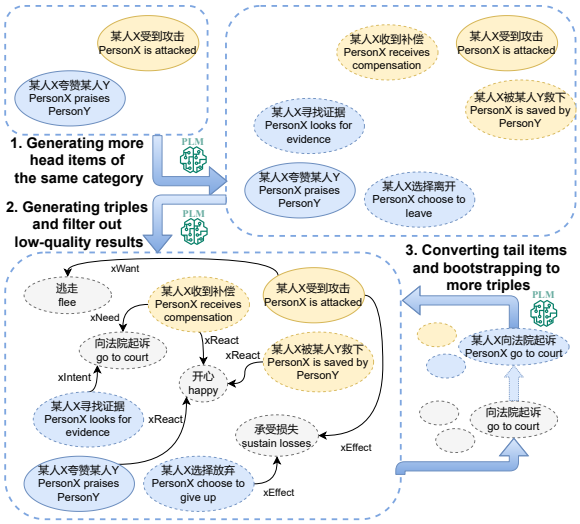

Figure 1: The construction demonstration of CN-AutoMIC. The dashed nodes the relations are generated and filtered with pretrained language models.

2 Related Work

2.1 Commonsense Knowledge Graphs

After some pioneers of strict logic formalization (Lenat et al., 1990), most recent commonsense knowledge resources, also known as commonsense knowledge graphs, represent commonsense knowledge in loosely-structured (*head*, *relation*, *tail*) triples, where the *head* and *tail* can be concepts or situations described in free-form phrases, and the *relation* can be various commonsense dimensions. Some of such resources are constructed by manually compiling or crowdsourcing (Speer et al., 2017; Sap et al., 2019; Hwang et al., 2021; Mostafazadeh et al., 2020). The others are mainly mined from corpora with human-crafted patterns or language processing tools (Tandon et al., 2014; Romero et al., 2019; Zhang et al., 2020; Fang et al., 2021; Nguyen et al., 2021).

Unfortunately, most of these projects are limited in English. Among the mainstream CKGs, ConceptNet (Speer et al., 2017) is the only multilingual one. It supports 10 core languages and more common languages. However, most of its non-English parts are taxonomic or lexical knowledge. The rest parts are limited in quantity and coverage. A remarkable recent work of Chinese commonsense knowledge resources is C^3KG (Li et al., 2022). It is based on the translation of ATOMIC (Sap et al., 2019; Hwang et al., 2021), which may fail to capture the cultural differences. Therefore, our work can be a strong complement to them.

2.2 Extracting Knowledge from PLMs

Since PLMs have seen a great number of documents and latently learned associations among concepts, there are extensive efforts to probe or extract relational knowledge from PLMs (Petroni et al., 2019; Sung et al., 2021; AlKhamissi et al., 2022). As for commonsense knowledge, some earlier work has tried to automatically complete CKGs by fine-tuning PLMs (Bosselut et al., 2019; Guo et al., 2020; Hwang et al., 2021), which still needs a large number of existing triples as training data. Recent research demonstrates that through natural language prompting, PLMs can adapt to generate commonsense knowledge under the few-shot setting (Da et al., 2021), or directly act as triple scorers without training (Davison et al., 2019). A significant progress is made by West et al. (2021). They use GPT-3 to generate a CKG from scratch. During the construction, only a small amount of human-authored data are required for in-context prompting generation and filtering results. They show the resulting CKG surpasses human-authored ATOMIC in quantity, quality, and diversity. Compared with West et al. (2021), our work proposes more improvement in generation and filtering and brings more comprehensive analysis for the paradigm from a non-English perspective.

3 Construction of CN-AutoMIC

For clarity, in this section, we first show the overview of the construction task, then describe the construction process of CN-AutoMIC in detail.

3.1 Overview

We expect to obtain commonsense knowledge represented in (*head*, *relation*, *tail*) triples via prompting generation. The demonstration of construction is shown in Figure 1. Similar to the construction workflow of crowdsourced CKGs, we hope to first collect *head* items (Section 3.2) and then collect *tail* items according to several pre-defined relations (Section 3.3). We limit the *head* items to the description of eventualities (events, activities and states) (Bach, 1986), such as "某人 X 离开家 (PersonX leaves home)". Following West et al. (2021), we consider seven relation types about eventualities from ATOMIC, which are listed in Table 2. Since the raw generated results are mixed with noise and degeneration, we introduce human supervision to train filter models to distinguish high-quality results (Section 3.4). To reuse the generated *tails* and increase the density, we propose a bootstrapping iteration process (Section 3.5).

3.2 Generating Head Items

We start from a minor size of head item seeds, using them as examples to prompt PLMs to generate more head items in the same format.

Notably, although previous work (West et al., 2021) treats all head items without distinction and collects knowledge about them for all predefined relations, we argue that it is necessary to further subdivide head items into different categories, because some heads and relations are in conflict and they cannot produce valid results. For example, we cannot infer the intent of X (xIntent) in "某人 X 受到攻击 (PersonX is attacked)", because he is passively involved in it rather than intentionally causes it. Therefore, we divide the head items in three categories (*voluntary occurrences*, *involuntary occurrences* and *states*) and match them with different relations, as illustrated in Table 1. Note that these categories are not strict, but they can hopefully reduce invalid generation.

Then, we collect head item seeds for the three categories. We mainly sample the high-quality head items from ATOMIC, manually categorizing and translating them. We intentionally discard the instances that are English-specific or rare in Chinese context, such as "PersonX has a baby shower". In total, we collect 100, 50, and 45 seeds for *voluntary occurrences*, *involuntary occurrences*, and *states*, respectively.

Through pilot studies, we empirically choose mT5-XXL (13B parameters) (Xue et al., 2021) for generation. It is one of the biggest publicly available multilingual PLMs[2], covering 101 languages, but still 13x smaller than GPT-3 used in West et al. (2021). During the generation, we use the prompt shown in Figure 2. For each generation cycle, we sample 10 examples from the seeds to construct the prompt, and use nucleus sampling with $p = 0.9$ to decode 100 results. The generation cycles for different head categories are *independently* conducted. After generating 600K raw results (2,000 cycles for each category), we discard 30% results of high negative log-likelihood and merge the duplicates in the remaining part, resulting in 100K head items.

[2]We note that mT5 has an additional multilingual advantage. It can conduct text-infilling generation for any position the special token <extra_id_0> appears. This is helpful for some languages (e.g. SOV languages) which cannot guarantee that the content to be generate is at the end of the prompt.

Category	Examples	Valid Relations
Voluntary Occurrences	某人X租房子；某人X学开车；某人X夸赞某人Y PersonX rents a house; PersonX learns to drive; PersonX praises PersonY	xWant, xReact, xEffect, xAttr, xNeed, xIntent, HinderedBy
Involuntary Occurrences	某人X受到攻击；某人X睡过头；某人X收到某人Y的来信 PersonX is attacked; PersonX oversleeps; PersonX receives a letter from PersonY	xWant, xReact, xEffect xAttr, xNeed, HinderedBy
States	某人X很疲惫；某人X头晕；某人X认识某人Y PersonX is tired; PersonX feels dizzy; PersonX knows PersonY	xWant, xAttr, xNeed, xEffect, HinderedBy

Table 1: Three categories of the head items: *voluntary occurrences* (PersonX intentionally cause it), *involuntary occurrences* (PersonX is involuntarily involved in it), and *states* (PersonX is in it for some time).

Relation	Description / Verbalizing Templates
xWant	{head}，在此之后，{X}想要{tail} {head}. As a result, {X} wants {tail}
xReact	{head}，对此，{X}感觉{tail} {head}. For that, {X} feels {tail}
xEffect	{head}。结果，{X}{tail} {head}. The outcome is that {X} {tail}
xAttr	{head}，据此，可以看出{X}是{tail} When {head}, people think {X} is {tail}
xNeed	{head}，在此之前，{X}需要{tail} {head}. Before that, {X} needed {tail}
xIntent	{head}，{X}的意图是{tail} {head}, {X}'s intent in it is {tail}
HinderedBy	{head}，这受到阻碍，因为{tail} {head} can be hindered by {tail}

Table 2: The relations and their verbalizing templates. The words in brackets are placeholders and will be replaced according to the head and tail.

1. 某人X买书;
(1. PersonX buys a book;)
.....
10. 某人X和某人Y一起打篮球;
(10. PersonX plays basketball with PersonY;)
11. <extra_id_0>
(11. <extra_id_0>)

Figure 2: The prompt for generating head items. The translation in parentheses is not actually in the prompt. <extra_id_0> is a special token to make the mT5 model do text-infilling generation.

请填写人物的意图，例如:
(Please write down the intent of the person, Example:)
1. 晓燕取回报纸，晓燕的意图是阅读报纸;
(1. Xiaoyan takes back the newspaper, Xiaoyan's intent is to read the newspaper;)
.....
9. 张三联系警察，张三的意图是 <extra_id_0>;
(9. Zhang calls the police, Zhang's intent is <extra_id_0>;)

Figure 3: The prompt for generating tail items according to a pre-defined relation.

3.3 Generating Triples

To obtain complete triples, we further generate tail items according to different relations. Similar to Section 3.2, we use example triples for prompting generation. To make effective use of the capabilities of PLMs, we need to convert triples into natural language sentences. For this sake, we use the templates in Table 2 to verbalize triples, and further replace "某人X (PersonX)" placeholders with random Chinese names.

We continue to use mT5-XXL model for generation. With the verbalized example triples, we construct the prompt as shown in Figure 3. For each relation, we use 8 example triples, which are sampled from ATOMIC and manually translated into Chinese. The prompt is used to generate 10 tail items for each (*head*,*relation*) pair with nucleus sampling ($p = 0.7$). As said before, each head item is only paired with the valid relations according to its category[3], so that the invalid generation results are reduced. After converting names back to placeholders and removing duplicated triples, this step produces 5.2M raw triples in total.

3.4 Filtering Results

Annotation To train filters that can distinguish high-quality triples, we randomly sample 4000 instances from the raw generated triples and ask three native Chinese speakers to annotate them. We intentionally give three questions for each triple. The annotators should first rate the head and tail alone to indicate whether they are acceptable. If not, the options of rejecting include syntax errors, abnormal or impossible situations (e.g. "某人X在天上游泳 (PersonX swims in the sky)") and other faults (e.g. containing real names rather than name placeholders). Then, if the annota-

[3]We tag head items with the category of examples used to generate them. It could be wrong sometimes, but we empirically find the overall Precision is moderate (Section 4.2).

	Heads	Tails	Triples
Acceptance Rate	85.2%	94.4%	47.6%

Table 3: Annotator's acceptance for sampled raw triples.

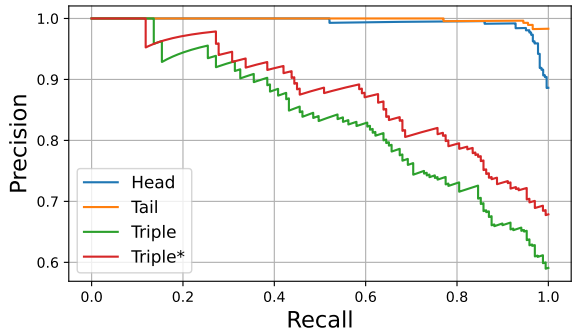

Figure 4: The precision-recall curves of filter models. "**Head**", "**Tail**", "**Triple**" are the performances of head classifier, tail classifier and triple classifier respectively. "**Triple***" is the performance of triple classifier on a subset, where the head and tail are already acceptable.

tors have accepted the head and tail, they should rate the triple as a whole, with options for acceptability: "always/often", "sometimes/likely", "far-fetched/never", or "invalid". The former two are considered "accepted". The latter two are "rejected". Table 3 shows the acceptance rate. For the overall results, we find the fleiss's κ (Fleiss, 1971) is 0.439, which indicates moderate agreement.

Training We use 80% of the annotated data for training, and the remaining parts for validating and testing. We train binary classifier models to predict whether the input is acceptable. Considering the acceptance differences in the annotation results(Table 3), we train single-use classifiers for heads, tails, and triples, respectively. We empirically choose a Chinese version[4] of ELECTRA (Clark et al., 2020) as the underlying model and fine-tune it for the tasks. We set learning rate to 5e-5 and batch size to 128 by grid search, and use early stopping to maximize the average precision (AP) on the validation data. We also tried other multilingual or Chinese PLMs of similar size, but the ELECTRA-based models obtain the best AP.

Cascaded Filtering We report the precision-recall curves of the trained models in Figure 4, including all three classifiers. We also report the performance of triple classifier on "clean data",

[4]The hfl/chinese-electra-180g-large checkpoint.

Relation	Tail-to-Head Example	Converted Category
xNeed, xWant, xIntent	去看医生→某人X去看医生 go to the doctor→PersonX goes to the doctor	Voluntary Occurrences
xEffect, HinderedBy	失去工作→ 某人X失去工作 lose the job→PersonX loses the job	Involuntary Occurrences
xReact	满意→某人X感觉满意 satisfied→PersonX feels satisfied	States

Table 4: Converting tails to heads for bootstrapping iteration.

where the head and tail in the triple are acceptable (**Triple***). From the curves, we can find head and tail classifiers can reach very high precision at almost all recall value. And triple classifier perform better on "clean data". It indicates that we can achieve better performance by **cascaded filtering**, i.e., applying the head and tail classifier first, and then using the triple classifier. We also find it is useful to set different thresholds for different relation types. Finally, we set the thresholds for head and tail classifier to ensure $precision > 0.98$ on the test data. We empirically search three groups of thresholds for the triple classifier, based on $precision = 0.9, 0.8, 0.75$ on each relation. We use these thresholds to get three subsets with different sizes and denote them as *high/mid/low* subsets.

3.5 Bootstrapping Iteration

We note that although the generated tail items have different formats from the head items, many of them can be converted into head items with simple templates, as shown in Table 4. After the above steps, many of the tail items have never appeared as head items. To further increase the diversity and density, we use the high-frequency tail items from the *mid* subset to generate more triples. We repeat the generating and filtering process described in Section 3.3 and Section 3.4, using the trained filters. Such bootstrapping iteration can be done several times, though we only conduct it once in this work. Finally, the resulting triples from different iterations are merged. We denote the merged sets as CN-AutoMIC$_{high/mid/low}$ respectively.

4 Evaluation and Analysis

In this section, we evaluate and analyze the resources in three parts. First, we comprehensively evaluate CN-AutoMIC in size, quality and diversity. In this step, we also evaluate the commonsense model (CN-COMET) trained on it. Second, we conduct specific analyses for different construc-

	Knowledge Graph	Construction	Unique Head Items	Unique Tail Items	Triples	Human Acceptance
English CKGs	ATOMIC$^{20}_{20}$ (Hwang et al., 2021)	Human	25,807	354,777	760,034	86.8*
	ATOMIC^{10X} (Unfiltered) (West et al., 2021)	Generation	165,783	874,417	6,456,300	78.5*
	ATOMIC^{10X} (High-Quality)	Generation	164,553	357,761	2,512,720	96.4*
Chinese CKGs	ATOMIC-zh (Li et al., 2022)	Translation	20,949	276,446	712,970	38.7
	(**ours**) CN-AutoMIC (Unfiltered)	Generation	114,364	1,101,556	6,868,766	47.6
	(**ours**) CN-AutoMIC$_{low}$	Generation	99,817	385,333	2,764,465	75.2
	(**ours**) CN-AutoMIC$_{mid}$	Generation	97,329	269,655	1,812,175	80.5
	(**ours**) CN-AutoMIC$_{high}$	Generation	89,738	182,893	1,140,840	**87.2**

Table 5: The statistics of CN-AutoMIC and related resources. The * results are from West et al. (2021)

tion steps. Third, we inspect the cases of culture-specific commonsense knowledge and generation deficiencies.

4.1 Evaluating the Graph

Setup We count the size of triples, the unique heads, and the unique tails in the graph to investigate the quantity and diversity. For comparison, we refer to two English CKGs, including human-authored ATOMIC$^{20}_{20}$ (Hwang et al., 2021) and automatically generated ATOMIC^{10x} (West et al., 2021). We also refer to a Chinese CKG (ATOMIC-zh) (Li et al., 2022) which is automatically translated from ATOMIC$^{20}_{20}$. Then, to test the quality, we sample 1000 triples from ATOMIC-zh and CN-AutoMIC and conduct a human evaluation. The annotation setting is similar to Section 3.4, but only the triples need to be rated. We keep the annotation results by majority vote and report the average acceptance for the triples[5].

Overall Statistics The overall statistics of CN-AutoMIC are shown in Table 5. From the results, we find: **(1)** Compared with the existing translated CKG, CN-AutoMIC contains a larger size of triples with better quality, as well as more unique head items. Interestingly, even the raw generated triples have better average acceptance than the translated CKG. We speculate that is because the translated CKG has a lot of syntax and translation errors. **(2)** After filtering, the human acceptance reaches up to 87.2 from 47.6, indicating the effect of the filtering process. As a trade-off, the diversity of tail items is decreased. **(3)** The English ATOMIC^{10x} is generated by GPT-3 and has better basic quality. After filtering, it can reach very high acceptance, surpassing human-authored ATOMIC$^{20}_{20}$ by 10 percent. By contrast, CN-AutoMIC struggles on reaching high acceptance, which shows the difficulty of generating non-English commonsense knowledge.

[5]The fleiss's κ is 0.535, indicating moderate agreement.

Relation	Unique Tail Items	Triples	Acceptance
xWant	23,673	179,861	84.8
xReact	1,985	145,431	95.4
xEffect	91,808	463,298	86.5
xAttr	1,660	28,973	88.9
xNeed	45,221	209,525	81.2
xIntent	20,575	79,012	88.1
HinderedBy	8,868	34,740	85.0

Table 6: Relation-level results of CN-AutoMIC$_{high}$.

Relation-level Results We show the relation-level results of CN-AutoMIC$_{high}$ in Table 6. We can find the acceptance on most of the relations is between 80 to 90 percent. However, the number of triples varies significantly. That is because we use different filter thresholds for different relations to achieve the same quality level. To some extent, the change of triple amounts reflects how much commonsense knowledge of a specific relation type exists in the PLM. According to the results, mT5 seems to be better at *xEffect* than *HinderedBy*.

Commonsense Model To examine the data quality from another perspective, we also train commonsense knowledge models (COMET) (Bosselut et al., 2019; Hwang et al., 2021) on CN-AutoMIC or ATOMIC-zh triples. We denote these models as CN-COMET. The models are based on mT5-base (580M), which is 20x smaller than T5-XXL. During training, we set the learning rate to 1e-4 and batch size to 128 by a small grid search. We linearly decay the learning rate for all training steps, and finally take the checkpoints with the lowest negative log-likelihood on the validation set. For fair com-

Model/Training Data	Train Data Acc.	Generation Acc.
CN-COMET (ATOMIC-zh)	38.7	29.7
CN-COMET (CN-AutoMIC$_{low}$)	75.2	54.3
CN-COMET (CN-AutoMIC$_{mid}$)	80.5	60.2
CN-COMET (CN-AutoMIC$_{high}$)	87.2	**66.9**

Table 7: The performance of commonsense knowledge model trained on different CKGs.

Category	Voluntary Occurences	Involuntary Occurences	States
Precision	84%	71%	76%

Table 8: The category precision of generated head items.

parison, we evaluate all these models on a held-out set of ATOMIC-zh, and manually check the results. During the evaluation, we remove the instances that has unreadable head items. The results are shown in Table 7. The model trained on CN-AutoMIC$_{high}$ achieves the best performance, indicating the high-quality CKG can make the small-sized model infer better commonsense knowledge. Nevertheless, the best performance is still not satisfactory. The quality of training data might be still not good enough. And the translation noise in test data could also exacerbate the difficulty.

4.2 Analyzing the Construction Steps

In this section, we analyze the effect of some intermediate steps.

Category Precision As described in Section 3.2, head items are generated with three categories of seeds, so we can reduce invalid generation according to the categories. However, the categories of generated head items are not guaranteed to be what they are generated from. Therefore, we manually check 100 generated head items for each category, and report the precision in Table 8. The results indicate that most of the head items belong to the category they are generated from. Based on this, we avoid nearly 10% invalid generation according to the category.

Effect of Cascaded Filtering We validate the effect of cascaded filtering by temporarily removing the head and tail classifiers during constructing CN-AutoMIC$_{high}$. That adds 47K new triples in total. We sample 500 triples from them and conduct manually checking. About 43.2% of them are bad triples. According to the estimation, removing head and tail classifiers can make the overall acceptance drops by 1.9%.

	Before Iteration	After Iteration
Unique Head Items	67,263	89,738
Unique Tail Items	143,195	182,893
Triples	768,124	1,140,840
Retaining Rate	0.148	0.166

Table 9: The changes of *high* subset after one iteration. Retaining rate means the proportion of *high* subset in all generated triples.

Effect of Bootstrapping Iteration In Table 9, we report the changes of *high* subset after a bootstrapping iteration. From the results, we find the quantities of unique heads, tails, and triples have substantially increased. Also, we find the retaining rate (i.e. the proportion of triples that are retained by the filters) also increases. We conjecture that it is because the filtered triples before iteration have high-quality tail items. Converting them to head items and conducting generation can get better performance.

4.3 Case Study

Culture-Specific Knowledge Since mT5 has been trained on Chinese corpora, it may generate commonsense knowledge specific to Chinese context. We find some explicitly culture-specific knowledge triples from the *high* subset and list them in Table 10. These examples involve festivals, traditional practices, games, and apps that are familiar to Chinese people but not popular in English communities. Therefore they cannot be found in current English CKGs. In contrast, CN-AutoMIC can capture such commonsense knowledge in Chinese perspectives to some extent.

Deficiency Patterns We show some regular generation mistakes in Table 11. We note two interesting error types: **(1)** Some head items cannot pair with some relations. Taking them as input will always result in errors. For example, "PersonX kills himself" cannot pair with `xWant` (after that, X wants), because he will lose consciousness and cannot want to do anything. Though we have set three head categories for such problems, there are still some intractable cases. The fundamental reason is that PLMs are unable to "reject" inappropriate input. Further research is needed to avoid such

Festival	某人X和某人Y共度中秋 → xNeed → 做月饼 PersonX spends the Mid-Autumn Festival with PersonY → xNeed → make moon cakes
Traditional Practice	某人X坐月子 → xIntent → 把身体照顾好 PersonX is in postpartum confinement → xIntent → take good care of (her) health
Game	某人X歇一整天 → xWant → 打麻将 PersonX takes a day off → xWant → play mahjong
App	某人X看视频 → xNeed → 打开优酷 PersonX watches videos → xNeed → open the Youku app.

Table 10: The cases of generated commonsense knowledge triples that are specific to Chinese context. *Mid-Autumn Festival* is a Chinese traditional festival. *Postpartum confinement* (or *lying-in*) is a traditional custom for new mothers. *Mahjong* is a game popular among Chinese. *Youku* is a website similar to Youtube/Netflix.

Conflict with the Relation	某人X杀死自己 → xWant → 参加葬礼 PersonX kills himself → xWant (after that x wants) → Attend the funeral
	某人X失去意识 → xReact→ 激动 PersonX is knocked unconscious → xReact (after that x feels) → excited
Negative Expression or Unfavorable Situation	某人X没有房屋→ xNeed → 赚钱 PersonX has no house → xNeed (before that x needs) → earn money
	某人X手很疼痛 → HinderedBy → 某人X没有止疼药 PersonX's hand is aching → HinderedBy → PersonX doesn't have painkillers

Table 11: The error cases of generated commonsense knowledge triples.

results. **(2)** For negative expressions or unfavorable situations, the model often performs badly generation for some relations. For example, in "PersonX has no house; before that, X needs", the model is trying to generate the methods to avoid the trouble (such as "earning money"), rather than the reason that makes X get in the trouble. This might be due to the ambiguity in the natural language prompts.

5 Discussion

What is the upper-bound size of the generation? The PLMs can conduct ever-lasting generation, but it seems there is a soft upper bound. The results generated later are easy to repeat previous results. Therefore, the cost of novel results would gradually increase and eventually become unaffordable. In this work, we have generated tens of millions of triples. It has still not reached the limit, but similar or repeated content has appeared in large numbers. For example, during generating head phrases, we observe that the proportion of non-repetitive results keeps descending (Figure 5).

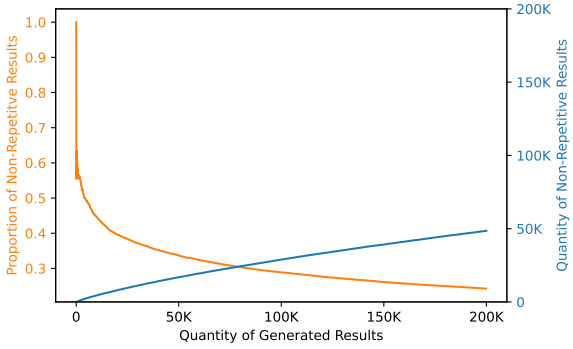

Figure 5: The change of quantity and proportion of non-repetitive results during generating head phrases.

If PLMs have already learned commonsense knowledge, why is it necessary to extract the knowledge from it First, according to the results in this paper, the quality of direct generated results is still poor. It indicates that even for a model with 13B+ parameters, the learned knowledge is still rough and full of mistakes. A distilling process can distinguish the clean knowledge and benefit smaller applicable models. Besides, recent work shows that even though explicitly training PLMs with knowledge, there is no guarantee that they can actually use such knowledge in target tasks(Wang et al., 2021). Therefore, we can exploit explicit symbolic knowledge as auxiliary information.

6 Conclusion

Considering the dilemma of lacking non-English commonsense knowledge resources, in this paper, we propose CN-AutoMIC, the first Chinese commonsense knowledge graph that is totally generated by pretrained language models. During the construction, we use prompting generation to obtain head and tail items, as well as introduce categorized generation, cascaded filtering and bootstrapping iteration to improve the quantity, quality and diversity. Through human evaluation, the resource is shown to have better quality than directly translated resources from English language. We discuss the culture-related phenomena and common deficiency patterns in the generated knowledge graph. Although our work is limited to Chinese, the basic framework and methods can be used to populate CKGs in more languages.

Limitations

The main limitations of this work include: **(1)** We require large-scale pretrained models. The generation performance is strongly dependent on the size of models. We use 4 RTX-A6000 GPUs for running the T5-XXL model. **(2)** Due to the lack of large-sized PLMs, the quantity and quality of this generated Chinese CKG still fall behind similar English resources. **(3)** We still cannot interpret the behavior of large PLMs. The specific source of the generated commonsense knowledge is hard to locate, and there are potential ethical risks since the results are not completely checked. **(4)** We still require extra human labor when applying the method to each new language. Although basically our methods and underlying models (mT5) can generalize for other languages, we still need human-crafted prompts and a minor size of annotations for training filter models.

Acknowledgements

This work is supported by the National Key Research and Development Program of China (No. 2020AAA0106400), and the National Natural Science Foundation of China (No. 61922085, 61976211, 62176257). This work is also supported by the Strategic Priority Research Program of Chinese Academy of Sciences (Grant No. XDA27020200), the Youth Innovation Promotion Association CAS, and Yunnan Provincial Major Science and Technology Special Plan Projects (No.202202AD080004).

References

Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona Diab, and Marjan Ghazvininejad. 2022. A review on language models as knowledge bases. *arXiv preprint arXiv:2204.06031*.

Prithviraj Ammanabrolu, Wesley Cheung, William Broniec, and Mark O Riedl. 2021. Automated storytelling via causal, commonsense plot ordering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5859–5867.

Emmon Bach. 1986. The algebra of events. *Linguistics and philosophy*, pages 5–16.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. COMET: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779, Florence, Italy. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. *arXiv:2005.14165 [cs]*. ArXiv: 2005.14165.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Jeff Da, Ronan Le Bras, Ximing Lu, Yejin Choi, and Antoine Bosselut. 2021. Analyzing commonsense emergence in few-shot knowledge models. In *3rd Conference on Automated Knowledge Base Construction*.

Joe Davison, Joshua Feldman, and Alexander Rush. 2019. Commonsense knowledge mining from pretrained models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1173–1178, Hong Kong, China. Association for Computational Linguistics.

Tianqing Fang, Hongming Zhang, Weiqi Wang, Yangqiu Song, and Bin He. 2021. DISCOS: Bridging the gap between discourse knowledge and commonsense knowledge. In *Proceedings of the Web Conference 2021*. ACM.

Joseph L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

Daya Guo, Duyu Tang, Nan Duan, Jian Yin, Daxin Jiang, and Ming Zhou. 2020. Evidence-aware inferential text generation with vector quantised variational AutoEncoder. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6118–6129, Online. Association for Computational Linguistics.

Jena D Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. 2021. (comet-) atomic 2020: On symbolic and neural commonsense knowledge graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6384–6392.

Douglas B. Lenat, R. V. Guha, Karen Pittman, Dexter Pratt, and Mary Shepherd. 1990. Cyc: toward programs with common sense. *Communications of the ACM*, 33(8):30–49.

Dawei Li, Yanran Li, Jiayi Zhang, Ke Li, Chen Wei, Jianwei Cui, and Bin Wang. 2022. C^3KG: A Chinese commonsense conversation knowledge graph. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1369–1383, Dublin, Ireland. Association for Computational Linguistics.

Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. KagNet: Knowledge-aware graph networks for commonsense reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2829–2839, Hong Kong, China. Association for Computational Linguistics.

Ninareh Mehrabi, Pei Zhou, Fred Morstatter, Jay Pujara, Xiang Ren, and Aram Galstyan. 2021. Lawyers are dishonest? quantifying representational harms in commonsense knowledge resources. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5033, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nasrin Mostafazadeh, Aditya Kalyanpur, Lori Moon, David Buchanan, Lauren Berkowitz, Or Biran, and Jennifer Chu-Carroll. 2020. GLUCOSE: GeneraLized and COntextualized story explanations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4569–4586, Online. Association for Computational Linguistics.

Tuan-Phong Nguyen and Simon Razniewski. 2022. Materialized knowledge bases from commonsense transformers. In *Proceedings of the First Workshop on Commonsense Representation and Reasoning (CSRR 2022)*, pages 36–42, Dublin, Ireland. Association for Computational Linguistics.

Tuan-Phong Nguyen, Simon Razniewski, and Gerhard Weikum. 2021. Advanced semantics for commonsense knowledge extraction. In *Proceedings of the Web Conference 2021*. ACM.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Julien Romero, Simon Razniewski, Koninika Pal, Jeff Z. Pan, Archit Sakhadeo, and Gerhard Weikum. 2019. Commonsense Properties from Query Logs and Question Answering Forums. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, pages 1411–1420, New York, NY, USA. Association for Computing Machinery.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019. ATOMIC: An Atlas of Machine Commonsense for If-Then Reasoning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:3027–3035.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4444–4451.

Mujeen Sung, Jinhyuk Lee, Sean Yi, Minji Jeon, Sungdong Kim, and Jaewoo Kang. 2021. Can language models be biomedical knowledge bases? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4723–4734, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Niket Tandon, Gerard de Melo, Fabian Suchanek, and Gerhard Weikum. 2014. WebChild: harvesting and organizing commonsense knowledge from the web. In *Proceedings of the 7th ACM international conference on Web search and data mining*, WSDM '14, pages 523–532, New York, NY, USA. Association for Computing Machinery.

Zhixing Tian, Yuanzhe Zhang, Kang Liu, Jun Zhao, Yantao Jia, and Zhicheng Sheng. 2020. Scene restoring for narrative machine reading comprehension. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3063–3073, Online. Association for Computational Linguistics.

Cunxiang Wang, Pai Liu, and Yue Zhang. 2021. Can generative pre-trained language models serve as knowledge bases for closed-book QA? In *Proceedings of the 59th Annual Meeting of the Association for*

Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 3241–3251, Online. Association for Computational Linguistics.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena D Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2021. Symbolic knowledge distillation: from general language models to commonsense models. *arXiv preprint arXiv:2110.07178*.

Genta Indra Winata, Andrea Madotto, Zhaojiang Lin, Rosanne Liu, Jason Yosinski, and Pascale Fung. 2021. Language models are few-shot multilingual learners. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Wei Zeng, Xiaozhe Ren, Teng Su, Hui Wang, Yi Liao, Zhiwei Wang, Xin Jiang, ZhenZhang Yang, Kaisheng Wang, Xiaoda Zhang, Chen Li, Ziyan Gong, Yifan Yao, Xinjing Huang, Jun Wang, Jianfeng Yu, Qi Guo, Yue Yu, Yan Zhang, Jin Wang, Hengtao Tao, Dasen Yan, Zexuan Yi, Fang Peng, Fangqing Jiang, Han Zhang, Lingfeng Deng, Yehong Zhang, Zhe Lin, Chao Zhang, Shaojie Zhang, Mingyue Guo, Shanzhi Gu, Gaojun Fan, Yaowei Wang, Xuefeng Jin, Qun Liu, and Yonghong Tian. 2021. Pangu-α: Large-scale autoregressive pretrained chinese language models with auto-parallel computation.

Hongming Zhang, Daniel Khashabi, Yangqiu Song, and Dan Roth. 2020. TransOMCS: From linguistic graphs to commonsense knowledge. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pages 4004–4010. International Joint Conferences on Artificial Intelligence Organization.

Zhengyan Zhang, Yuxian Gu, Xu Han, Shengqi Chen, Chaojun Xiao, Zhenbo Sun, Yuan Yao, Fanchao Qi, Jian Guan, Pei Ke, Yanzheng Cai, Guoyang Zeng, Zhixing Tan, Zhiyuan Liu, Minlie Huang, Wentao Han, Yang Liu, Xiaoyan Zhu, and Maosong Sun. 2021. CPM-2: Large-scale cost-effective pre-trained language models. *AI Open*, 2:216–224.

A Annotation Details

In this study, we mainly have two kinds of annotation tasks, annotating training data for filters and human evaluation for results. Both of them require the workers to review a bunch of triples and answer questions. We show the annotation page in Figure 6. There are three questions for the workers: (1) Whether the head is acceptable. (2) Whether the tail is acceptable. (3) Whether the triple is acceptable as a whole.

For question (1) and (2), we provide one option for acceptance and four options for rejection: abnormal expressions (syntax errors), violation of commonsense (impossible situations), unusable format (incomplete generation or containing names of real identities), and mismatch to the relation.

For question (3), there are four options about whether the triple is acceptable (plausible): "always/often", "sometimes/likely", "far-fetched/never", and "invalid".

Before annotation, each worker are shown with a instruction, which contains background knowledge of the annotation task and examples for each answer options.

B Alternative Models for Generation

During the pilot studies, we also compared several alternative text generation backbones besides mT5, including a multilingual autoregressive model XGLM (Winata et al., 2021) and two Chinese-specific models, Pangu-Alpha (Zeng et al., 2021) and CPM2 (Zhang et al., 2021). We construct a small set of test prompts to make the models complete some commonsense knowledge triples. For each model, We use its biggest publicly available checkpoint to generate 100 results for each prompt (directly sampling without hyper-parameter search).

We show the results and samples in Table 12 and Table 13. All these backbones can generate some plausible results. However, the XGLM (7.5B) model often generate <unk> tokens and CPM2 (11B) model sometimes give degenerate or irrelevant long results. In general, Pangu-Alpha (13B) and mT5-XXL (13B) have better generation quality. Based on comprehensive consideration of efficiency and feasibility, we conduct full experiments and analyses with mT5-XXL in this paper.

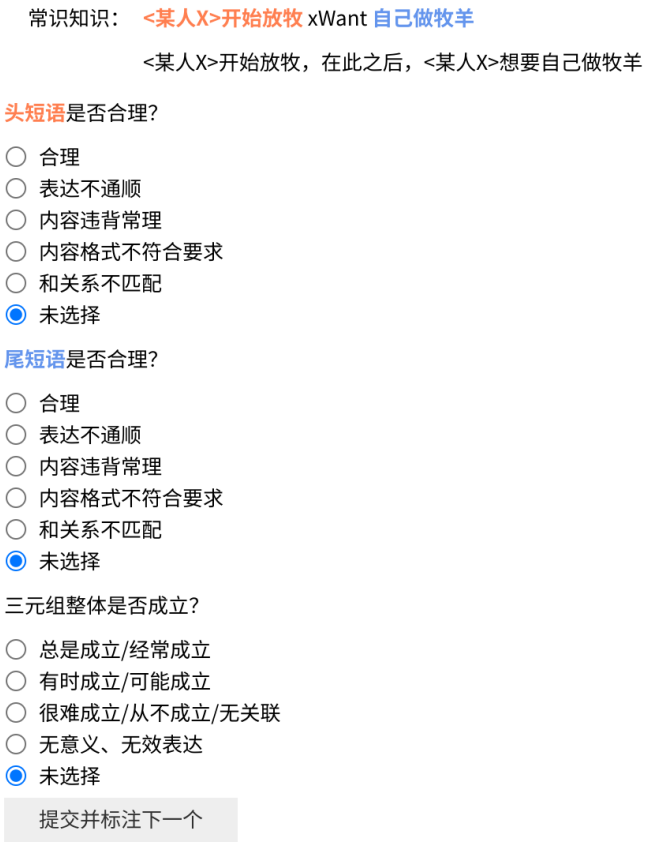

Figure 6: The annotation page.

	mT5-XXL (13B)	XGLM (7.5B)	Pangu-Alpha (13B)	CPM2 (11B)
Pilot Tested Triple Acceptance	0.58	0.42	0.61	0.46

Table 12: The acceptance of generated triples for different backbone generation models.

Prompt:	……X爱上Y，在此之后，X想要 .....X falls in love with Y. As a result, X wants to
mT5-XXL (13B)	["当Y的妻子", "娶Y", "看电视", "追求", "结婚"] ["become Y's wife", "marry Y (as the husband)", "watch TV", "chase", "get married"]
XGLM (7.5B)	["与李四保持联系", "追求李四", "去做手术", "和Y谈恋爱", "向Y表白"] ["keep in touch with Y", "chase Y", "go to surgery", "have a love affair with Y", "confess love to Y"]
Pangu-Alpha (13B)	["拆散二人", "拥有Y", "结婚", "追Y", "结识新朋友"] ["break up two", "own Y", "get married", "chase Y", "making new friends"]
CPM2 (11B)	["追回Y", "Y嫁给Z", "求她", "将Y追回来", "嫁给Y"] ["chase back Y", "Y marry Z, "beg her", "chase Y back", "marry Y (as the wife)"]
Prompt:	……在X买书之前，X需要 .....X buys book. Before that, X needs to
mT5-XXL (13B)	["借书", "钱", "书", "在书店", "看书", "有钱"] ["borrow books", "money", "book", "in the bookstore", "read books", "have money"]
XGLM (7.5B)	["买书", "花钱","拥有一本书","看一本书","将书带回家"] ["buy a book", "spend money", "own a book", "read a book", "take the book home"]
Pangu-Alpha (13B)	["拥有图书馆的任何一本书","准备考试","拥有书签","去拿书"] ["own any book in the library", "prepare for the exam", "own a bookmark", "get a book"]
CPM2 (11B)	["存钱10","能说服某人拿一堆书看","在11点完成手上的工作","赚到钱"] ["save money 10", "able to persuade someone to read a pile of books", "finish the work at 11 o'clock", "make money"]
Prompt:	……X发动战争，据此，可以看出X是 .....When X start a war, people think X is
mT5-XXL (13B)	["凶悍的", "自私的", "暴戾的", "野蛮的", "有胆量的"] ["ferocious", "selfish", "violent", "barbaric", "courageous"]
XGLM (7.5B)	["强<unk>的", "聪明的", "勇敢的", "胆小怕事的"] ["Str<unk>g", "smart", "brave", "timid"]
Pangu-Alpha (13B)	["粗暴的", "自私的", "有勇有谋的", "奋起抗争的", "莽撞的"] ["rude", "selfish", "Brave and resourceful", "arising to struggle", "reckless"]
CPM2 (11B)	[" 失去","凶残的","残暴的","作大仗的","偏激的"] [" loss", "ferocious", "brutal", (meaningless expression), "extreme"]
Prompt:	……X复习，对此，X感觉 ……X reviews (lessons). For that, X feels
mT5-XXL (13B)	["疲惫", "压力", "紧张", "无聊"] ["tired", "stress", "nervous", "boring"]
XGLM (7.5B)	["疲<unk>","轻松","精力充<unk>","紧张"] ["tir<unk>", "relaxed", "energe<unk>", "nervous"]
Pangu-Alpha (13B)	["考得不错","没意义","头疼","学习压力太大","轻松"] ["good grades", "meaningless", "headache", "too much study pressure", "relaxed"]
CPM2 (11B)	["寂寞", "紧张", "与以往不同", "压力大"] ["lonely", "nervous", "different from the past", "too much pressure"]

Table 13: The sampled generation results of different backbone models. We omit the seed examples in prompts and only show the part to be complete. For legibility, we also replace the person names with X, Y or Z.