Exploring the Effectiveness of Multi-Lingual Commonsense Knowledge-Aware Open-Domain Dialogue Response Generation

Sixing Wu1,2, Jiong Yu1,2, Tianshi Che3, Yang Zhou3, and Wei Zhou1,2*

1 National Pilot School of Software, Yunnan University, Kunming, China
2 Engineering Research Center of Cyberspace, Yunnan University, Kunming, China
3 Auburn University, Auburn, Alabama, USA
{wusixing, zwei}@ynu.edu.cn

Abstract

Prior works have shown the promising results of commonsense knowledge-aware models in improving informativeness while reducing the hallucination issue. Nonetheless, prior works often can only use monolingual knowledge whose language is consistent with the dialogue context. Except for a few high-resource languages, such as English and Chinese, most languages suffer from insufficient knowledge issues, especially minority languages. To this end, this work proposes a new task, Multi-Lingual Commonsense Knowledge-Aware Response Generation (MCKRG), which tries to use commonsense knowledge in other languages to enhance the current dialogue generation. Then, we construct a MCKRG dataset MCK-Dialog of seven languages with multiple alignment methods. Finally, we verify the effectiveness of using multi-lingual commonsense knowledge with a proposed MCK-T5 model. Extensive experimental results demonstrate the great potential of using multi-lingual commonsense knowledge in high-resource and low-resource languages. To the best of our knowledge, this work is the first to explore Multi-Lingual Commonsense Knowledge-Aware Response Generation.

1 Introduction

Open-domain dialogue response generation is always a tricky problem (Kann et al., 2022) because it allows users to converse in whatever they like, and then various knowledge will be involved (Zhou et al., 2022). Even with the expensive large-scale language models, models still suffer from the non-informative and hallucination problem if they can not access up-to-date and correct knowledge. The non-informative problem (Li et al., 2016) means generated responses are always boring and generic, such as ‘I don’t know.’; the hallucination problem (Maynez et al., 2020) means factually incorrect knowledge that violates the real world may be generated; for example, ‘iPhone 9 is my favorite.’.

Grounding dialogue response on the commonsense knowledge (Wu et al., 2020; Zhou et al., 2022; Cai et al., 2023) has shown great effectiveness in alleviating the aforementioned two problems. Large-scale commonsense knowledge such as ConceptNet (Speer et al., 2016) can provide various correct and informative real-world facts (Yu et al., 2020). Nonetheless, such works often can only leverage the knowledge of the native language. For example, an English system can only access the English knowledge base, and an Indonesian system can only access the Indonesian knowledge base. This restriction is not problematic for high-resource languages such as Chinese and English (Kann et al., 2022) because we can easily collect sufficient commonsense knowledge from the Internet. However, for many low-resource languages, collecting enough knowledge in their own language is exceptionally challenging. Without sufficient knowledge, knowledge-aware models face the problem of making bricks without straw, which would dramatically reduce their real-world practicalities (Wu et al., 2021, 2022b).

Given such challenges, this work proposes Multi-Lingual Commonsense Knowledge-Aware Response Generation (MCKRG), a new task to break the bottleneck that models can only use the knowledge of the native language. The primary challenge of MCKRG is the dataset. As of the beginning of this work, only KoWoW is a large-scale English-Korean cross-lingual knowledge-aware dataset (Kim et al., 2021). However, KoWoW is not grounded on commonsense knowledge and only involves two languages. Thus, this work constructs a multi-lingual knowledge-aligned conversational dataset MCK-Dialog with several automatic processes. MCK-Dialog uses the XPersona of 7 languages as the conversational corpus; hence, MCK-
Dialogue Context: I want to eat roast duck. I don’t know where it’s delicious?

Response: Quanjude’s Peking Duck is very delicious; I often eat it.

Figure 1: An illustration of MCKRG. In this example, English is the native language, and Chinese is used as an example to show how to use both the native-code and mixed-code multi-lingual knowledge.

**2 Related Work**

**Knowledge-Aware Response Generation** Traditional response generation methods (Sutskever et al., 2014) often generates non-informative dialogues (Li et al., 2016) and factually incorrect hallucination information. To this end, knowledge-aware works ground dialogue response on external knowledge (Yu et al., 2020). The knowledge can be gathered from text-based encyclopedia (Lin et al., 2020a; Zhao et al., 2020), commonsense knowledge (Zhang et al., 2020; Wu et al., 2020; Zhou et al., 2022), search engines (Shuster et al., 2022), and many others (Moghe et al., 2020; Li et al., 2022). Although such works have achieved promising results, they often only focus on high-resource English or Chinese (Kann et al., 2022). Once in the low-resource language scenarios that knowledge is insufficient (Wu et al., 2022b), such works will degenerate into traditional non-knowledge-aware works. Compared to them, this work explores using multi-lingual commonsense knowledge besides the native language. In addition, although the prior work KoWoW (Kim et al., 2021) learns to use Korean-English cross-lingual knowledge, this work is totally different because: 1) we use graph-based commonsense knowledge instead of text-based knowledge; 2) KoWoW only studies a much simpler two language scenario.

1. To our best knowledge, we are the first to use multi-lingual commonsense knowledge to enhance dialogue generation (i.e., MCKRG);
2. We have constructed the first MCKRG benchmark dataset MCK-Dialog of 7 languages and propose a MCK-T5 as the baseline;
3. Extensive experiments verified the effect of MCKRG and how to use multi-lingual commonsense knowledge in dialogue generation.
**Multi-Source Knowledge** Prior works try to seek more knowledge from multiple sources to enrich the dialogue generation (Liang et al., 2021; Wu et al., 2021, 2022b). This work has a similar idea but the difference is also significant because we seek knowledge of other languages but the same knowledge type, while prior works seek knowledge of other knowledge types but the same language.

**Multi-Lingual** It allows a model to handle data of multiple languages in a uniform model. Using multi-lingual pre-trained language models such as mBART (Liu et al., 2020) and mT5 (Xue et al., 2021) are a convenient way. Although multilingual systems have been paid much attention in NLP, only a few works in dialogue response generation. MulZDG (Liu et al., 2022) and XDailyDialog (Liu et al., 2023) studies the non-knowledge-aware multilingual response generation, X-Persona (Lin et al., 2020b) studies the multi-lingual personalized response generation, and ToD (Majewska et al., 2023) studies the multi-lingual task-oriented response generation. Compared to such works, this work is the first work to study the multi-lingual commonsense knowledge-aware open-domain dialogue response generation.

**Cross-Lingual** For its primary issue, translation, there are three common paradigms (Huang et al., 2023; Zheng et al., 2023): 1) **translation-first** conducts translation before the generation while 2) **translation-later** conducts generation before the translation. The last 3) **end2end** does not explicitly conduct translation. This work has explored all paradigms: 1) Sec 3.2/Sec 3.3 uses the translation-first/later methods to transform knowledge of other languages to the current native language; 2) Our **MCK-T5** uses mixed-code knowledge in an end2end way.

### 3 MCK-Dialog Dataset

#### 3.1 Conversations

**XPersona** (Lin et al., 2020b) is a multi-lingual extension of the monolingual English **Persona-Chat** (Zhang et al., 2018), which employed humans to generate dialogues grounded on an assigned persona. To obtain conversations in the other six languages, **XPersona** first used Neural Machine Translation (NMT) API with several human-in-the-loop mistake correction processes to construct the parallel training sets in other languages efficiently. Besides the NMT, the valid set and test set were well-revised by humans in the post-processing, which may change the dialogue or discard a dialogue; thus, as shown in Table 1, the valid/test sets are not in parallel across languages. **MCK-Dialog** keeps all dialogues of XPersona.

#### 3.2 Commonsense Knowledge

We adopt the multilingual commonsense knowledge base **ConceptNet** (Speer et al., 2016). As reported in Table 2, we can see that the abundances of Native knowledge across languages are different. High-resource languages have 1M+ facts while low-resource languages can only have 4K+ facts, or even no facts. To enrich the abundance of knowledge, we leverage the knowledge of other languages. We create two expanded sub-graphs **Transferred** and **Translated** for each specific language $l_1$ besides the original Native sub-graph:

- **Native**: For a fact $f = (e_{head}, e_{rel}, e_{tail})$ in the ConceptNet, if the language labels of $e_{head}$ and $e_{tail}$ are the same $l_1$, then it belongs to the Native$_{l_1}$.

- **Transferred**: We transfer the knowledge of other languages to Transferred$_{l_1}$ by using the cross-lingual synonyms facts. For example, given two English-Chinese synonym facts, (Apple Inc., Synonyms, 苹果公司) and (Tim Cook, Synonyms, 蒂姆·库克), then the English fact (Apple Inc., Related To, Tim Cook) can be transferred to a Chinese fact (苹果公司, Related To, 蒂姆·库克).

- **Translated**: Mainly for the low-resource and zero-resource languages. We employ the NMT API\(^2\) to translate a fact from the source language to

---

\(^2\)We use the Microsoft Azure Translation API.

---

### Table 1: The number of dialogue sessions/utterances.

<table>
<thead>
<tr>
<th></th>
<th>En</th>
<th>Cn</th>
<th>Fr</th>
<th>Jp</th>
<th>Ko</th>
<th>It</th>
<th>Id</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>16.9K/124K (The following units are also K)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Valid</td>
<td>17.3/0.22/1.7/1</td>
<td>0.25/1.9</td>
<td>0.28/2.1/1</td>
<td>0.32/3.0/1</td>
<td>0.14/1.1</td>
<td>0.48/3.8</td>
<td></td>
</tr>
<tr>
<td>Test</td>
<td>17.8/0.22/1.7/1</td>
<td>0.25/1.9</td>
<td>0.28/2.2/1</td>
<td>0.32/3.3</td>
<td>0.14/1.1</td>
<td>0.48/3.8</td>
<td></td>
</tr>
</tbody>
</table>

### Table 2: Knowledge abundance statistics of different sub-graphs. #E/#R/#F represents the number of distinct entities, relations and facts of each sub-graph.

<table>
<thead>
<tr>
<th></th>
<th>Native</th>
<th>Transferred</th>
<th>Translated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#E</td>
<td>#R</td>
<td>#F</td>
</tr>
<tr>
<td>En</td>
<td>1.17M</td>
<td>47</td>
<td>3.28M</td>
</tr>
<tr>
<td>Cn</td>
<td>154K</td>
<td>25</td>
<td>367K</td>
</tr>
<tr>
<td>Fr</td>
<td>367K</td>
<td>17</td>
<td>2.94M</td>
</tr>
<tr>
<td>Jp</td>
<td>84.4K</td>
<td>30</td>
<td>249K</td>
</tr>
<tr>
<td>Ko</td>
<td>3.51K</td>
<td>12</td>
<td>4.45K</td>
</tr>
<tr>
<td>It</td>
<td>511K</td>
<td>16</td>
<td>580K</td>
</tr>
<tr>
<td>Id</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>All</td>
<td>3.95M</td>
<td>47</td>
<td>9.51M</td>
</tr>
</tbody>
</table>
the target language. Similar to the Transferred, we only translate the head entity and the tail entity. All newly translated facts are placed into the extension Translated\textsubscript{i} for the specific language \textit{l}_{i}.

Note that although both Transferred\textsubscript{i} and Translated\textsubscript{i} are built with multi-lingual knowledge, they still represent knowledge in the native language \textit{l}_{i}. Hence, to avoid ambiguity, cross/multi-lingual is only used to tell the language relation between the dialogue context and the original knowledge source. Then, we use native-code to indicate the dialogue and knowledge are in the same language, and mixed-code to indicate the dialogue and knowledge may not be in the same language.

3.3 Knowledge Alignment

We then align the conversations with commonsense knowledge, which is a two-stage pipeline: the first Entity Recognition stage identifies entities, and the next Fact Selection stage selects fact candidates.

3.3.1 Entity Recognition

We propose a native-code \textit{ER}_{\textit{src}} to identify the entities of the native language and two mixed-code functions \textit{TTR}_{\textit{src-\textit{tgt}}} and \textit{RTR}_{\textit{src-\textit{tgt}}} to identify the entities of other languages.

Native-Code Given an utterance in language \textit{l}_{\textit{src}}, \textit{ER}_{\textit{src}} first tokenizes and lemmatizes (optional) it to a n-gram list. Then, all stop-words are removed and we only retain the nouns, verbs, adjectives, and adverbs for all 1-gram candidate entities. Lastly, we filter the candidate entities using the entity set of the sub-graphs of the corresponding language \textit{l}_{\textit{src}} to obtain the native-code entities. In the implementation, we employ several tools to handle each language. In detail, to tokenize a text, we use Natural Language Toolkit (NLTK) package (Bird et al., 2009) for English, French, Italian, and Indonesian; mecab-python3 package for Japanese; pymecabko package for Korean; splitting to characters for Chinese. Then, we use the same tool as the part-of-speech (POS) tagger for each non-Chinese language, and we directly remove all 1-gram Chinese entities. Furthermore, the Simplemma package is used as the lemmatizer for English, French, Italian, and Indonesian.

Mixed-Code The first \textit{TTR}_{\textit{src-\textit{tgt}}} is translation-then-recognition style. It first translates the given utterance into the target language \textit{l}_{\textit{tgt}} using the NMT API and then employs the corresponding \textit{ER}_{\textit{tgt}} to identify the entities in the target language.

Differently, another recognition-then-translation style \textit{RTR}_{\textit{src-\textit{tgt}}} first uses the native \textit{ER}_{\textit{src}} to identify the entities, then uses the cross-lingual synonym facts (None-Indonesian) or NMT API (Indonesian) to obtain the entities in \textit{l}_{\textit{tgt}}.

3.3.2 Fact Selection

Given a dialogue session \( H = (X_1, \ldots, X_n) \) of \( n \) turns and the identified entity set \((E_1, \ldots, E_n)\), we propose a Hard Match and a Soft Match to select the facts \( K \) that ground the current session \( H \).

Hard Match For a knowledge fact \( f = (e_{\text{head}}, e_{\text{rel}}, e_{\text{tail}}) \), if \( e_{\text{head}} \in E_{1:t-1} \) while \( e_{\text{tail}} \in E_{t} \), or \( e_{\text{tail}} \in E_{1:t-1} \) while \( e_{\text{head}} \in E_{t} \), then we assume \( H \) is partially grounded on \( f \).

Soft Match Hard Match method only considers the literal overlap but ignores the semantics, which may involve irrelevant knowledge. To this end, Soft Match first uses SentenceBERT (Reimers and Gurevych, 2020) to compute the fact embedding and the dialogue session embedding; then, Soft Match uses the Cosine function to estimate the relevance in the embedding space. Then, facts are selected based on similarity scores.

3.4 Statistics

Knowledge Abundance As mentioned in Section 3.2, there is an obvious abundance gap between high-resource and low-resource languages. To bridge this gap, we transfer/translate facts from other languages to the current language. Table 2 reports the final knowledge abundance of three sub-graphs. It can be seen that Transferred can significantly expand the knowledge abundance for non-zero-resource languages. By using NMT APT, Translated can significantly enrich the knowledge abundance for not only the non-zero-resource languages but also zero-resource Indonesian. Nevertheless, we can not say Translated is better because it would introduce more noise in the translation.

Knowledge Alignment Figure 2 reports the average matched commonsense knowledge facts per dialogue session after the alignment. Besides, we also report the number of possible related facts, which only require that at least one entity appears in the dialogue session. We first focus on the native-code knowledge (i.e., \textit{ER} bars). We can see that \( En, Cn, Fr, \) and \( Jp \) are high-resource languages because they have sufficient matched/related facts even if only use the Native ConceptNet (i.e., \textit{Native-ER}).
Unlike them, other languages are low-resource or zero-resource in the Naive ConceptNet. As for the next native-code knowledge Transferred, we can find its contribution (i.e., Transferred-ER) is limited for most languages after alignment. This is because its expansion relies on the cross-lingual synonym facts, which are also limited in ConceptNet. Unlike Transferred, the last native-code knowledge Translated can introduce more new knowledge facts, especially for the Indonesian (i.e., Translated-ER) with a higher tolerance for noise. By using all three sub-graphs, each language can have a similar knowledge abundance of the matched/related fact candidates. Then, we move to the mixed-code knowledge (TtR and RtT bars). English have the same counts among the three bars because we only retrieve knowledge of English besides the native language. It can be seen that the translation-later style RtT can retrieve more knowledge facts than the translation-first style TtR.

Overall, it must be noted that all counts illustrated in Figure 2 are non-repetitive in each language and the total number of available facts should sum all sub-factors together. Consequently, the average number of matched/related facts per session is about 800 and 89K. Thus, similar to the construction of other knowledge-aligned datasets, although the retrieved facts are relevant to the context to some extent, irrelevant knowledge still commonly exists, and this alignment is session-level. Therefore, selecting the most relevant knowledge at each turn of the dialogue generation is still needed.

4 MCK-T5 Model

To explore the effectiveness of MCKRG, we propose a MCK-T5 Model, which is built upon the multi-lingual mt5-base (582M) (Xue et al., 2021). Formally, each instance is denoted as \((H_i, K_i, R_i)\), where \(H_i = \{X_{i,1}, \cdots, X_{i,n}\}\) consists of \(n\)-turns dialogue histories, \(K_i = \{f_{i,j}\}_m\) consists of \(m\) knowledge facts, and \(R_i\) is the target response. Since the backbone mT5 is a plain-text-based Encoder-Decoder network, we have to convert the structural knowledge facts \(K_i\) into plain text and then concatenate the result with the dialogue history \(H_i\). Thus, the input is formatted as follows:

\[
\text{Input} = [\text{Prompt}_i; \Omega_K(K_i); \Omega_H(H_i)]
\]  

(1)

where \(\text{Prompt}_i\) is an instruction code to the task, which is ‘Generate a response based on the following terms:’. The next \(\Omega_K(K_i)\) linearizes all facts with identifiers; for example, ‘Contextual Facts: [0] (Apple Inc., RelatedTo, Steven Jobs); ....; [10] (iPhone, IsA, Device)’. The last \(\Omega_H(H_i)\) concatenates all dialogue histories with role labels.
just like 'Dialogue History: User: X1; Bot: X2; User: X3...'. mT5 does not need explicit labels to indicate the current language. Hence, we directly concatenate the $\Omega_K(K_i)$ and $\Omega_H(H_i)$ even if they are in different languages.

Training Thanks to the multi-lingual nature of mT5, this work considers two training strategies: 1) Singleton: We separately train a MCK-T5 model for each language; 2) Joint: All languages share one same MCK-T5 model in a specific scenario. For both strategies, we uniformly adopt the Maximum Likelihood Estimation in training, and the corresponding objective function is defined as:

$$\mathcal{L} = \sum_{r_i, t \in R_i} -\log(P(R_{i,1:t-1}|H_i, K_i)) \quad (2)$$

5 Experiments

5.1 Settings

Model Baselines consist of: 1) T5 and 2) mT5: the monolingual English T5 (t5-large, 738M) (Raffel et al., 2020) and the multi-lingual mT5 (mt5-base, 582M) (Xue et al., 2021); 3) BART and 4) mBART: the monolingual English BART (bart-large, 406M) (Lewis et al., 2020) and Chinese (fnlpc-bart-large, 407M) (Shao et al., 2021) BART, as well as the multi-lingual mBART (mbart-large, 611M) (Liu et al., 2020); 5) Bert2Bert, 6) M-Bert2Bert, 7) CausalBert and 8) M-CausalBert: models proposed by XPersona (Lin et al., 2020b), which use encoder-decoder models (5 and 6, 220-356M) or autoregressive models (7 and 8, 356M) for response generation. The prefix M- denotes it is a multi-lingual model, whereas a monolingual model. Implementation: We use the official codes for 5-8 models. Other models are implemented by PyTorch and Transformers. Similar to XPersona, we fine-tune each model with 5 epochs. We use the learning rate 1e-4 for T5/mT5-based models while 1e-5 for other models. The mini-batch size is 8, and the gradient accumulation step is 4. Experiments were run on Nvidia-A100 or RTX3090.

Metrics We adopt three representative metrics: 1) ES: We first use SentenceBERT\textsuperscript{3} to get the embedding of the ground-truth response and the generated response, and then use cosine function to compute the embedding similarity to estimate the semantic relevance; 2) F1: we use the CharF1 to compute the overlapping-based relevance and the precision of the generated response at the same time; 3) B4: We use the 4-gram BLEU to estimate the quality of the generated response. Besides, we use Mean-A to estimate the overall performance among multiple languages. It first computes the local geometric mean score of three metrics in each language and then computes the weighted (the size of the test set is regarded as a weight) mean score over all local scores. Mean-E excludes the Indonesian. We do not use perplexity because of the notable vocabulary differences among models.

5.2 Main Results

As shown in Table 3, the multi-lingual knowledge-aware MCK-T5s can significantly outperform the monolingual knowledge-aware MCK-T5s, especially in the low-resource languages (Ko, Id, see Native-ER in Figure 2), demonstrating the effectiveness of using the multi-lingual commonsense knowledge. Meanwhile, due to insufficient native knowledge, MCK-T5+MoCsk is beaten by the persona-aware mT5+Pers, indicating that monolingual commonsense knowledge can not provide information comparable to persona knowledge. Differently, MCK-T5+Csk uses multi-lingual commonsense knowledge and the overall performance is comparable to mT5+Pers. This result is incredible because the construction of such dialogues is totally grounded on persona knowledge rather than commonsense knowledge, and MCK-T5+Csk can be applied to more scenarios but mT5+Pers can only be used in the persona-aware scenario. Besides, the best XPersona baseline, M-CausalBERT, is worse than mT5+Pers and our MCK-T5+Csk. We think the reason is the backbone PLM of M-CausalBERT is worse than mT5.

For a more fair comparison in the persona scenario, MCK-T5+Csk+Pers is allowed to access the persona knowledge. In this case, our MCK-T5 undoubtedly achieves the best performance in every dimension. This result demonstrates commonsense knowledge is essential in dialogue generation and would not conflict with other knowledge.

5.3 Native-Code Comparisons

Here, both the dialogue and the knowledge are in the same language. We test 3 configurations: 1) Base uses the Native sub-graph; 2) Ext additionally uses the Transferred sub-graph; and 3) Tri uses Native+Transferred+Translated sub-graphs. We also test the proposed two fact selection methods

\textsuperscript{3}paraphrase-multilingual-MiniLM-L12-v2
Table 3: Model Comparisons. In Know column, None uses no knowledge, Pers uses the persona provided by XPersona, Csk uses 30 commonsense knowledge facts retrieved by the best configuration ‘Soft Match, Tri, ER+Rt+Tri, Joint Training’ found in Table 5, MoCsk uses only the monolingual native commonsense knowledge.

Table 4: Native-code knowledge comparisons. Models use the Singleton training. Except for Vanilla, HM randomly selects 20 facts for each dialogue session while SF select the top 20 most relevant.

Table 5: Mixed-code knowledge comparisons. Except for Vanilla, models use the English and/or the native Tri sub-graphs, and use SF to select the 20 facts. Models use the Singleton training except for the last uses Join.

Hard Match (HM) and Soft Match (SF). Table 4 reports the results, and findings are discussed in the following paragraphs.

The necessity of knowledge. Compared to the non-knowledge Vanilla, all knowledge-aware models have achieved notable gains. It clearly indicates that 1) commonsense knowledge is necessary for open-domain dialogue response generation, and 2) our commonsense knowledge construction and alignment methods are at least not ineffective.

The necessity of native-code multi-lingual knowledge. Besides the Native, Ext and Tri use the knowledge transferred/translated from other languages. We can find Ext is better than Base and Tri is the best no matter if considering the Indonesian or not. It shows that using knowledge of other languages is an effective way to alleviate the knowledge shortage in the native language in both the high-resource and low-resource scenarios.

The necessity of semantic-driven alignment. Soft-Match can further consider the semantics during the alignment. The results have indicated this necessity because SF is always better than HM. Meanwhile, the performance gain of SF becomes weaker when using all three sub-graphs. This is because we only adopt the non-fine-tuned Sentence-BERT. We believe the more advanced fine-tuned rankers, such as cross/poly-encoders (Urbanek et al., 2019; Humeau et al., 2019; Wu et al., 2022a), can take a step further and leave this as future work.
5.4 Mixed-Code Comparisons

We also consider using the knowledge in other languages without transferring or translating.

**English Only** We assume only English knowledge is available. For other languages, to use the cross-lingual English knowledge in a mixed-code way, TiTR first translates dialogues to English and then recognizes the corresponding English entities; on the contrary, RiT first recognizes the native language entities and then translates them to English. As reported in Table 5, we can find that both TiTR and RiT can significantly outperform the non-knowledge Vanilla, demonstrating 1) knowledge can be directly transferred from one knowledge to another knowledge in the open-domain dialogue response generation; 2) comparing such two models, TiTR is a little better. We think the reason is the translation-first TiTR can reduce the error distortion between the entity recognition process and the knowledge alignment process.

**English Plus** Then, we assume commonsense sub-graphs of both the native language knowledge and English are available. We can find that the corresponding ER+TiR, ER+RiT, and ER+RiT+TiR are better than the English-only TiR and RiT, showing more knowledge is better. Compared to ER+RiT+TiR, the last ER+RiT+TiR+Joint further uses the Joint Training strategy to train all instances of different languages in a uniform model. It can be seen that performance has increased significantly, indicating the knowledge of a specific language can be implicitly transferred to another language via joint training and a multi-lingual backbone language model.

5.5 Case study

Table 6 has shown two real cases sampled from our MCK-Dialog dataset. In the first Chinese case, four responses are generated with different knowledge configurations. The first w/o. csk does not use any knowledge; thus, the response is just a simple repetition. The second w/. Chinese csk only uses Chinese knowledge in the original ConceptNet. We find there is no enough relevant knowledge facts that can be used. Thus, the response is unchanged. The next w/ TiTR English csk only uses the English knowledge in the ConceptNet via the TiR and knows what is vegan. However, the information is still not enough. The last w/ Multi-lingual csk uses the mixed-code multi-lingual knowledge; thus, the response is most relevant and informative. In the next case, we select an English case that is parallel to the previous Chinese case. Unlike the first case, no vegan-related fact has appeared because 1) there are too many general English facts, and our Soft Match is difficult to put vegan-related facts at the top; 2) we only use facts of the native language and English in this experiment, this makes English the least profitable of all languages. Nevertheless, the generated dialogues are still acceptable.

### Table 6: Case Study

<table>
<thead>
<tr>
<th>Case</th>
<th>Response</th>
<th>English Only</th>
<th>English Only</th>
<th>English Only</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>Yes, I like vegetarian lunch. What about you?</td>
<td>Yes, I don’t. I love animals. Do you?</td>
<td>Yes, I like Vegetarianism. What about you?</td>
<td>Yes, I’m not a Vegetarianism, but I like cooking.</td>
</tr>
<tr>
<td>Case 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

6 Future Work and Conclusion

6.1 Future Work

This work focuses on proposing the task and constructing the dataset/baselines. Based on the foundation, there can be some new research topics:

1) **Knowledge Selection:** As mentioned, there are 800/89K+ matched/related facts in each dialogue session. Obviously, a model is difficult to handle such a large number of facts, and a response would not involve so many facts. Thus, how to select
proper facts are important.

2) Knowledge Conflicts: Using knowledge from multiple sources and languages simultaneously would inevitably lead to knowledge conflicts, such as translation errors, repetitions across languages, and many others. Eliminating these conflicts is necessary for further improving efficiency.

6.2 Conclusion

Prior commonsense knowledge-aware dialogue response generation works have achieved promising results. However, they only study high-resource languages such as English and Chinese. Differently, this paper focuses on both the high-resource and low-resource languages. Hence, we first propose a new Multi-Lingual Commonsense Knowledge-Aware Response Generation (MCKRG) task, which tries to use commonsense knowledge in other languages to enhance dialogue generation in the current language. Then, we construct a dataset MCK-Dialog of seven languages and a baseline model MCK-T5. Extensive experimental results demonstrate the great potential of using multi-lingual commonsense knowledge in high-resource and low-resource languages. To our best knowledge, this work is the first to explore MCKRG.

Acknowledgement

This work is supported in part by the National Natural Science Foundation of China under Grant 62162067 and 62101480, Research and Application of Object detection based on Artificial Intelligence, in part by the Yunnan Province expert workstations under Grant 202205AF150145.

Limitation

This work studies the multi-lingual commonsense knowledge-aware dialogue response generation. Besides the future work mentioned in Section 6.1, there are still some limitations in the work.

Benchmark Dataset

MCK-Dialog only targets to be a benchmark dataset because the construction mainly relied on automatic methods, which may impact the quality besides the English dialogues. This can be attributed to the backbone conversational corpus XPersona was built in a similarly automatic way. Nevertheless, we still believe that building benchmark datasets in an automated manner is still necessary for multi-lingual studies, as the cost of manual construction is too high, especially when the number of languages studied increases.

Human Evaluation

This work does not conduct a human evaluation. The reason is two-fold: 1) This work involves seven different languages, posing great challenges in employing volunteers, designing the human evaluation process and quality control; 2) We have conducted an early human evaluation for English and Chinese dialogues, we find this evaluation may not have enough persuasiveness. Non-English conversations were translated from English, and their texts are quite different from human-written texts, even though they have been revised by humans (Lin et al., 2020b).

Ethical Consideration

This paper studies a widely studied topic, namely, open-domain dialogue response generation, and uses open-released resources. We do not introduce any new ethical statement/consideration in this work. Consequently, we think this work has no additional ethical considerations compared to previous work from the technical perspective.

References

Steven Bird, Ewan Klein, and Edward Loper. 2009. Natural language processing with Python: analyzing text with the natural language toolkit. " O’Reilly Media, Inc.".


