X-RiSAWOZ: High-Quality End-to-End Multilingual Dialogue Datasets and Few-shot Agents

♣Mehrad Moradshahi[1] ♣Tianhao Shen[2] Kalika Bali[3] Monojit Choudhury[3]
Gaël de Chalendar[4] Anmol Goel[5] Sungkyun Kim[6] Prashant Kodali[5]
Ponnurangam Kumaraguru[5] Nasredine Semmar[4] Sina J. Semnani[1] Jiwon Seo[6]
Vivek Seshadri[3,7] Manish Shrivastava[5] Michael Sun[1] Aditya Yadavalli[7]
Chaobin You[2] ♦Deyi Xiong[2] ♦Monica S. Lam[1]

[1] Computer Science Department, Stanford University, Stanford, USA [2] College of Intelligence and Computing, Tianjin University, Tianjin, China
[3] Microsoft Research India, Bangalore, India [4] Université Paris-Saclay, CEA, List, Palaiseau, France
[5] International Institute of Information Technology, Hyderabad, India
[6] Deep Learning & Big Data Systems Lab, Hanyang University, Seoul, South Korea [7] Karya Inc., India

Abstract

Task-oriented dialogue research has mainly focused on a few popular languages like English and Chinese, due to the high dataset creation cost for a new language. To reduce the cost, we apply manual editing to automatically translated data. We create a new multilingual benchmark, X-RiSAWOZ, by translating the Chinese RiSAWOZ to 4 languages: English, French, Hindi, Korean; and a code-mixed English-Hindi language. X-RiSAWOZ has more than 18,000 human-verified dialogue utterances for each language, and unlike most multilingual prior work, is an end-to-end dataset for building fully-functioning agents.

The many difficulties we encountered in creating X-RiSAWOZ led us to develop a toolset to accelerate the post-editing of a new language dataset after translation. This toolset improves machine translation with a hybrid entity alignment technique that combines neural with dictionary-based methods, along with many automated and semi-automated validation checks.

We establish strong baselines for X-RiSAWOZ by training dialogue agents in the zero- and few-shot settings where limited gold data is available in the target language. Our results suggest that our translation and post-editing methodology and toolset can be used to create new high-quality multilingual dialogue agents cost-effectively. Our dataset, code, and toolkit are released open-source.[1]

1 Introduction

In recent years, tremendous effort has been put into the research and development of task-oriented dialogue agents; yet, it has been mainly focused on only a handful of popular languages, hindering the adoption of dialogue technology around the globe. Collecting dialogue data from scratch for a new language is ideal but prohibitively expensive and time-consuming, leading to the current lack of reliable multilingual dialogue benchmarks.

In recent years, several non-English task-oriented dialogue (ToD) datasets have been created. These datasets are either collected from scratch (Quan et al., 2020; Zhu et al., 2020a), synthesized using a state machine with manually written templates, and paraphrased for fluency by crowdworkers (Lin et al., 2021), or manually translated from another language (Li et al., 2021b). All of these approaches are labor-intensive, costly, and time-consuming; such investment is unlikely to be made for less widely spoken languages.

This motivates the development of zero and few-shot techniques that can produce a usable agent in a new language with no or only a few gold training dialogues in the target language. Concurrent with this work, Ding et al. (2022); Zuo et al. (2021); Hung et al. (2022b) adopt a *translation and manual post-editing* process where data is translated with neural machine translation models first, and then post-edited by crowdworkers. This approach has shown promise on MultiWOZ; however, reported zero- and few-shot accuracies show a big degradation in performance compared to full-shot accuracy in the source language. Besides, the performance of the agent in the original language was not good to begin with, in part due to misannotations in the dataset (Eric et al., 2019a). Lastly, these datasets either focus only on the subtask of Dialogue State Tracking (DST) (Ding et al., 2022) or auxiliary tasks such as Response Retrieval (Hung et al., 2022b), or are too small (Zuo et al., 2021) to train end-to-end dialogue agents that require policy, interactions with databases, and response generation components.

Our overall goal is to make task-oriented dialogue research in major languages available to low-resource languages. The key is to produce high-quality few-shot training, validation, and test set

[1] https://github.com/stanford-oval/dialogues

♣ Co-first authors ♦ Co-corresponding authors

	Dataset		
	Few-shot	Validation	Test
# Domains	12	12	12
# Dialogues	100	600	600
# Utterances	1,318	8,116	9,286
# Slots	140	148	148
# Values	658	2,358	3,571

Table 1: Statistics for the few-shot, validation, and test.

with as little manual effort as possible to enable zero-shot or few-shot training. We describe below our contributions towards this goal.

1.1 Data Translation Techniques and Toolset

Machine translation followed by human post-editing has been used as a method for extending monolingual NLP datasets to new languages (Yang et al., 2019; Ziemski et al., 2016; Giannakopoulos et al., 2011; Conneau et al., 2018). However, we discovered human post-editing to be the main pain point in creating new dialogue datasets. The process is costly, requiring a lot of back-and-forth among developers, translators, and annotators. Even after several rounds, the results are still not adequate. To alleviate this, we devised a scalable methodology and an associated toolkit that automates parts of this process, and aids translators and annotators to iteratively check their work themselves without developer supervision. This allows fast and accurate creation of a new dialogue dataset annotated with slot values for a new language.

We show that the entity-aware translation technique proposed by Moradshahi et al. (2023) is also applicable to other end-to-end dialogue datasets. We combine this technique with a dictionary-based alignment where multiple translations are generated for each entity individually (i.e. without context), using the same translation model used to translate the sentence. Then, the translated sentence is scanned to match any of the translation candidates, resulting in an improvement in the agent's performance.

Furthermore, we automatically check each step of data translation to ensure annotation consistency between dialogue utterances and API calls to the database. We are releasing this toolkit open-source for reproducibility as well as a resource for others.

1.2 X-RiSAWOZ Dataset

We created X-RiSAWOZ, a multi-domain, large-scale, and high-quality task-oriented dialogue benchmark, produced by translating the Chinese RiSAWOZ data to four diverse languages: English, French, Hindi, and Korean; and one code-mixed English-Hindi language. X-RiSAWOZ is an improvement over previous works in several aspects:

- **End-to-End**: Contains translations for all parts of dialogue including user and agent utterances, dialogue state, agent dialogue acts, and database results.
- **Larger**: RiSAWOZ is larger than MultiWOZ and covers a total of 11,200 dialogues with 151,982 turns. It also covers 12 domains compared to 7. In addition to translating validation and test data, we also sample 100 dialogue examples from the training set and translate them using the same process to use as few-shot training data. This way, X-RiSAWOZ can be used to experiment with few-shot techniques as well as zero-shot.
- **Higher Quality**: We choose RiSAWOZ as it exhibits the lowest misannotation rate among popular dialogue benchmarks as shown by Moradshahi et al. (2021). The data translation methodology described above reduces the mismatch between entities in the sentence and annotations, meaning that our translation process does not introduce new misannotations.
- **Cheaper**: First, the methodology and toolset reduce the amount of post-editing effort needed. Second, instead of using commercial translation systems such as Google Translate, we rely on open-source multilingual translation models such as MBART (Liu et al., 2020) for the translation of training data. This reduces the translation cost by at least 100x which could otherwise be a prohibiting factor when building datasets for new languages.

1.3 Experimental Results

We establish strong baseline results for our new X-RiSAWOZ dataset. In the full-shot setting, our model produces a new SOTA on the original Chinese dataset. With few-shot training, across languages, our model achieves between 60.7-84.6% accuracy for Dialogue State Tracking (DST), 38.0-70.5% accuracy for Dialogue Act (DA), and 28.5-46.4% for BLEU score when evaluated using gold data as the conversational context. Cumulatively over a conversation, our model achieves 17.2%, 11.9%, 11.3%, 10.6%, and 2.3% on Dialogue Success Rate (DSR), respectively. The remaining gap between zero or few-shot results on new languages and the full-shot results on Chinese creates opportunities for research and finding new techniques to further improve the dialogue agent performance.

2 Related Work

2.1 Multilingual Dialogue Datasets

MultiWOZ (Budzianowski et al., 2018; Ramadan et al., 2018; Eric et al., 2019b), CrossWOZ (Zhu et al., 2020a), and RiSAWOZ (Quan et al., 2020) are three monolingual Wizard-Of-Oz multi-domain dialogue datasets for travel dialogue agents. For the 9th Dialog System Technology Challenge (DSTC-9) (Gunasekara et al., 2020), MultiWOZ was translated to Chinese and CrossWOZ was translated to English using Google Translate. A portion of their evaluation and test sets were post-edited by humans, while the training set remained entirely machine translated. Moradshahi et al. (2021) translated RiSAWOZ to English and German using open-source machine translation models with alignment. However, the validation and test data were not verified by humans, resulting in potentially over-estimating the accuracy of agents. Several works (Ding et al., 2022; Zuo et al., 2021; Hung et al., 2022a) continued translation of MultiWOZ to other languages. For example, GlobalWOZ translates to several languages, with human translators post-editing machine-translated dialogue templates, and filling them with newly collected local entities. However, these works address only one or two subtasks of a full dialogue, and therefore training an end-to-end agent is not possible with them.

Different from these translation-based approaches, Lin et al. (2021) introduced BiToD, the first bilingual dataset for *end-to-end* ToD modeling. BiToD uses a dialogue simulator to generate dialogues in English and Chinese, and asks crowdworkers to paraphrase them for naturalness. This simulation-based approach eliminates the need for translation but requires hand-engineered templates and savvy developers with knowledge of the target language and dialogue systems. Besides, paraphrasing the entire dataset is costly.

2.2 Cross-Lingual Approaches for ToD

With the advent of pre-trained language models, contextual embeddings obtained from pre-trained multilingual language models (Devlin et al., 2018; Xue et al., 2021; Liu et al., 2020) have been used to enable cross-lingual transfer in many natural language tasks, including task-oriented dialogue agents. Unfortunately, most of this work has only focused on the DST subtask, which is a limitation we aim to rectify with this paper.

To further improve the cross-linguality of these embeddings, Tang et al. (2020) and Moghe et al. (2021) proposed fine-tuning multilingual BERT on a synthetic code-switching dataset. Glavaš et al. (2020) performed language adaptation by using intermediate masked language modeling in the target languages and improving zero-shot cross-lingual transfer for hate speech detection task.

Using machine translation for multilingual dialogue tasks has also been studied. Uhrig et al. (2021) used machine translation during inference to translate to English for semantic parsing. Instead, Sherborne et al. (2020) use machine translation to generate semantic parsing data to train a semantic parser in the target language which leads to better results. Moradshahi et al. (2023); Nicosia et al. (2021) proposed using alignment to improve the quality of translated data by ensuring entities are translated faithfully.

3 The End-to-End ToD Task

In end-to-end task-oriented dialogues, a user speaks freely with an agent over several turns to accomplish their goal according to their intents (e.g., "book a hotel with at least 5 stars"). In each turn, the agent must access its database if necessary to find the requested information (e.g., find a hotel that meets user constraints), decide on an action (e.g., present the information to the user or ask for additional information), and finally respond to the user in natural language based on the action it chooses. Following (Moradshahi et al., 2023), we decompose a dialogue agent into four subtasks:

1. *Dialogue State Tracking (DST)*: Generate the new belief state, for the current turn based on the previous belief state, the last two agent dialogue acts, and the current user utterance.
2. *API Call Detection (ACD)*: Determine if an API call is necessary to query the database.
3. *Dialogue Act Generation (DAG)*: Generate the agent dialogue act based on the current belief state, the last two agent dialogue acts, the user utterance, and the result from the API call.
4. *Response Generation (RG)*: Convert the agent dialogue act to produce the new agent utterance.

4 The Common Dialogue Interface

Over the years, various ToD datasets have been introduced (Budzianowski et al., 2018; Byrne et al., 2019; Zhu et al., 2020b; Quan et al., 2020; Lin et al., 2021), each with its own representation, making it difficult for researchers to experiment with different datasets. To facilitate experimentation, we have developed Common Dialogue, a standard interface for ToD tasks. This interface defines a unified format for datasets, their annotations, ontologies, and API interfaces. We show that the most widely-used recent dialogue datasets (such as MultiWoZ, RiSAWOZ, and BiToD) can be converted to this representation with a simple script. The standardization lets all different datasets be processed with the same software and models, significantly reducing the implementation time and cost.

Previously, other libraries such as ParlAI (Miller et al., 2017), ConvLab (Zhu et al., 2020c, 2022), and Nemo (Kuchaiev et al., 2019) were introduced so researchers can work with different dialogue datasets and interact with the trained models. However, these libraries are limited. They either do not provide a standard abstraction, making it difficult to add new datasets, or a modular interface that can connect with other code bases, requiring new models to be implemented in their repository before they can be used. Additionally, the training code needs to be modified to support a new dataset or language for an existing dataset.

5 Dataset Creation

In this section, we describe the process used to extend RiSAWOZ to the new languages. The original RiSAWOZ dataset is in Chinese. We manually translate the validation data (600 dialogues), test data (600 dialogues), and 1% of the training dataset (100 dialogues), which we refer to as *few-shot*, from Chinese to English. For other languages, we use English as the source language, since bilingual speakers of English and the target language are more accessible than Chinese and the target language. Since the English data is manually translated, this approach avoids double translationese (Vanmassenhove et al., 2021) and ensures the best data quality. We machine-translate the English data and manually post-edit the translation for fluency and correctness. Besides the few-shot data, we also machine-translate all of the Chinese training data into each of the languages (including English) and train with them; we refer to training with just this data set as *zero-shot*, since no human labor is used during dataset creation.

In the following, we discuss the steps and methods for preparing data for translation, including building alignment between entities and performing iterative quality checks. We also describe how to create the target language ontology, which serves as a database for API calling and provides a mapping between source and target language entities.

5.1 Translation and Alignment for Few-Shot, Validation, and Test Data

5.1.1 From Chinese to English

Figure 1 shows the process used to translate the Chinese dataset to English. First, human professional translators manually translate the Chinese dialogue utterances and ontology in the validation, test, and few-shot training data sets to English. We provide the translators with an annotation tool (Figure 2) to navigate through data examples, perform translation, and highlight entity spans in the translated sentence. The tool helps verify the consistency of slot value translations between user/agent utterances and their annotations after translation.

For each utterance in a dialogue, our tool automatically identifies the values in dialogue states and user/agent actions. Slots are *canonicalized* before calling the database, meaning that their values must lexically match those in the ontology. Since slot values appearing in the utterances may differ from the canonicalized version, we ask translators to manually identify and mark the non-canonicalized form of slot values and their word spans in the utterances.

The tool automatically checks the number of highlighted spans to prevent missing entity translations. After checking, the annotation tool outputs the English dialogue texts and a correspondence (i.e. alignment) between source and target language slot values.

5.1.2 From English to Other Languages

Automatic Translation. For validation, test, and few-shot data, we use commercial translation models since (1) translation is done only once, (2) data size is smaller so it is affordable, and (3) higher data quality reduces post-editing effort.

Manual Post Editing. We hire bilingual speakers of English and the target language to post-edit the translations for fluency and correctness. We instruct them to update the alignment if they modify the translated entities. We provide several tools that automatically check their work and help them during the process. We describe the details in Section 5.4.2.

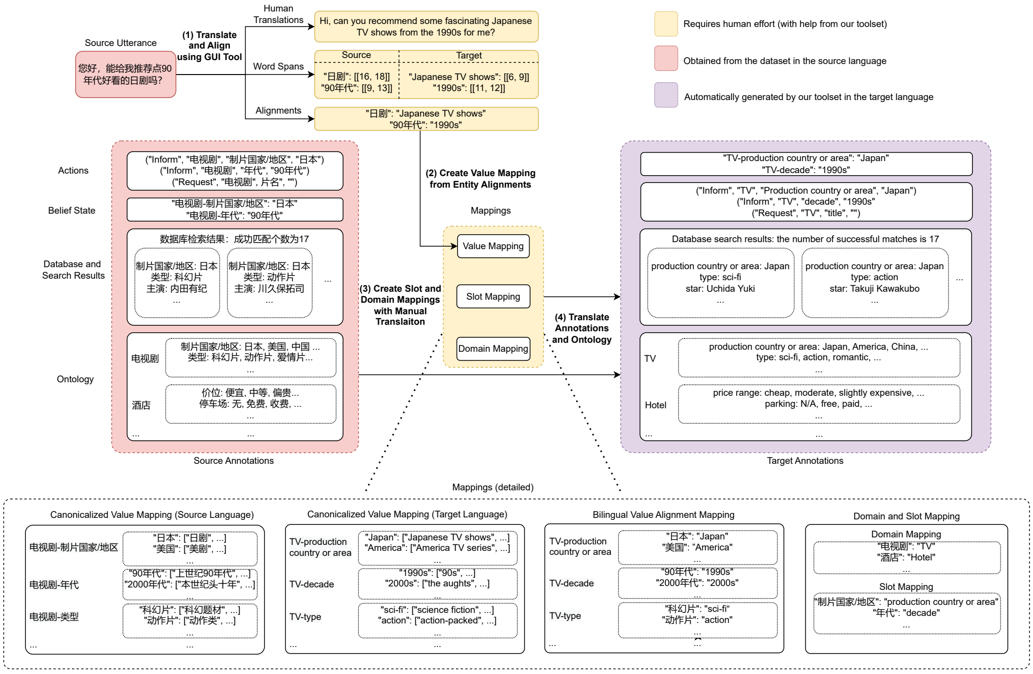

Source Utterance
您好，能给我推荐点90年代好看的日剧吗?
(1) Translate and Align using GUI Tool
Human Translations
Hi, can you recommend some fascinating Japanese TV shows from the 1990s for me?
Word Spans
Source
Target
"日剧": [[16, 18]]
"90年代": [[9, 13]]
"Japanese TV shows": [[6, 9]]
"1990s": [[11, 12]]
Alignments
"日剧": "Japanese TV shows"
"90年代": "1990s"
Requires human effort (with help from our toolset)
Obtained from the dataset in the source language
Automatically generated by our toolset in the target language
Actions
("Inform", "电视剧", "制片国家/地区", "日本")
("Inform", "电视剧", "年代", "90年代")
("Request", "电视剧", 片名", "")
Belief State
"电视剧-制片国家/地区": "日本"
"电视剧-年代": "90年代"
Database and Search Results
数据库检索结果：成功匹配个数为17
制片国家/地区: 日本
类型: 科幻片
主演: 内田有纪
...
制片国家/地区: 日本
类型: 动作片
主演: 川久保拓司
...
Ontology
电视剧
制片国家/地区: 日本, 美国, 中国 ...
类型: 科幻片, 动作片, 爱情片...
酒店
价位: 便宜, 中等, 偏贵...
停车场: 无, 免费, 收费, ...
Source Annotations
(2) Create Value Mapping from Entity Alignments
Mappings
Value Mapping
Slot Mapping
Domain Mapping
(3) Create Slot and Domain Mappings with Manual Translaiton
(4) Translate Annotations and Ontology
"TV-production country or area": "Japan"
"TV-decade": "1990s"
("Inform", "TV", "Production country or area", "Japan")
("Inform", "TV", "decade", "1990s"
("Request", "TV", "title", "")
Database search results: the number of successful matches is 17
production country or area: Japan
type: sci-fi
star: Uchida Yuki
production country or area: Japan
type: action
star: Takuji Kawakubo
TV
production country or area: Japan, America, China, ...
type: sci-fi, action, romantic, ...
Hotel
price range: cheap, moderate, slightly expensive, ...
parking: N/A, free, paid, ...
Target Annotations
Mappings (detailed)
Canonicalized Value Mapping (Source Language)
电视剧-制片国家/地区
"日本": ["日剧", ...]
"美国": ["美剧", ...]
电视剧-年代
"90年代": ["上世纪90年代", ...]
"2000年代": ["本世纪头十年", ...]
电视剧-类型
"科幻片": ["科幻题材", ...]
"动作片": ["动作类", ...]
Canonicalized Value Mapping (Target Language)
TV-production country or area
"Japan": ["Japanese TV shows", ...]
"America": ["America TV series", ...]
TV-decade
"1990s": ["90s", ...]
"2000s": ["the aughts", ...]
TV-type
"sci-fi": ["science fiction", ...]
"action": ["action-packed", ...]
Bilingual Value Alignment Mapping
TV-production country or area
"日本": "Japan"
"美国": "America"
TV-decade
"90年代": "1990s"
"2000年代": "2000s"
TV-type
"科幻片": "sci-fi"
"动作片": "action"
Domain and Slot Mapping
Domain Mapping
"电视剧": "TV"
"酒店": "Hotel"
Slot Mapping
"制片国家/地区": "production country or area"
"年代": "decade"

Figure 1: The translation and annotation process of X-RiSAWOZ from Chinese to English. There are 4 major steps: (1) Translate utterances and provide entity alignments between source and target sentences using the UI tool. (2) Create the value mapping using entity alignments. (3) Create slot and domain mappings by manually translating them from Chinese. (4) Translate slot values in the annotations and ontology using the value mapping.

5.2 Zero-Shot Training Data Translation & Alignment

To create the zero-shot training datasets for the target languages (including English), we use open-source machine translation models to translate the Chinese data to the target language. We pick open-source models since (1) their results are reproducible, (2) open-source models provide access to model weights necessary for hybrid alignment (described below), (2) they allow tuning text generation hyperparameters such as temperature (Ficler and Goldberg, 2017) or beam size (Freitag and Al-Onaizan, 2017) and (3) they cost less, thus allowing effective scaling to more languages.

Hybrid Alignment for NMT. Previous work (Moradshahi et al., 2021; Li et al., 2021a) proposed using alignment for tracking the position of entities during translation to ensure they can be replaced with the desired translation both in the utterance and the belief state. For this purpose, the encoder-decoder cross-attention weights of the neural machine translation model were used in a method called *neural alignment*. Although neural alignment often works well, it can produce incorrect spans as it is a probabilistic approach and has particularly low recall on long multi-token entities.

Ideally, if there exists a dictionary that provides a mapping between each source entity and all possible translations in the target language, we can directly scan the translated sentence to see if there is a match. We call such an approach *dictionary alignment*. Unfortunately, there is no such dictionary. We propose to build such a dictionary for each sentence on-the-fly. To do so, we first extract the entities from the sentence, then translate each individually and use nucleus sampling (Holtzman et al., 2019) with different temperature values to generate K translation candidates. This way, we build a mapping between each entity and possible translations which serves as the dictionary for dictionary alignment. Finally, we combine the two methods in a *hybrid* approach: We try to use dictionary alignment first, and if there is no matching translation in the output, we fall back to neural alignment.

5.3 Creating English-Hindi Code-Mixed Zero-Shot Training Data

For generating English-Hindi code-mixed train set, we implemented a pipeline combining GCM (Rizvi et al., 2021), and alignment based word substitution. An overview of the pipeline is shown in Fig. 3. GCM automatically generates code-mixed text given parallel data in two languages, based on two linguistic theories of code-mixing, the Equivalence Constraint theory (Poplack, 1980) and the

Matrix Language theory (Scotton, 1993).

We take the Chinese training set as source and translate user and agent utterances to English (en) and Hindi (hi). The translated sentences are fed as input to GCM, which produces code-mix utterances. For sentences where GCM fails to generate any candidate, we rely on word-alignment-based word substitution to generate a code-mixed utterance. Alignments are generated using cosine similarities between sub-word representations from mBERT in a parallel sentence pair (Dou and Neubig, 2021).

5.4 Translation of Annotations

The next step is to translate the slot values in the belief state, user and agent acts, and database search results in the source language to the target language. Since the translations of the same slot value may vary according to the context (e.g., "是" corresponds to is, does, has or other words indicating affirmative), we create a one-to-many mapping between source language slot values and corresponding translations based on the slot value alignments obtained above. We ask human translators to select the most appropriate expression from all candidate translations as the canonicalized translation. We follow two basic principles in this process:

Part-of-Speech (POS) Consistency. The translator should pick, for each slot, values with the same POS tags where possible. For example, for the "production country/region" slot in the TV series domain, we will use the unified noun form (i.e., "America"/"India") instead of mixing the noun and adjective form (i.e., "American"/"India").

Value Consistency. The translator should use the same translation across domains and slots. For example, the Chinese word "中等" when used as a "price-range" can be translated into "moderate" or "medium". We consistently map "中等" to "moderate" for all "price-range" slots across all domains.

5.4.1 Creating Ontology and Databases

We found that ontology construction should be done in tandem with dataset translation. In prior work, using a predefined ontology limited fluency and diversity of the translations (Zuo et al., 2021), and replacing entities in sentences after translation without careful attention to parts of speech or context resulted in grammatically incorrect sentences (Moradshahi et al., 2020; Ding et al., 2022). Each value in the source database is automatically mapped to its canonicalized translation. Note that since not all slot values are seen in the training dataset, translators are asked to provide canonicalized translations for those values.

The original RiSAWOZ dataset only provides final search results from databases instead of intermediate API calls. We hence also restore the API calls through the dialogue state, database, and search results for complete database interactions. This improves the extensibility of the dataset and helps to generalize RiSAWOZ to other languages and domains in the future.

5.4.2 Annotation Checker

Manual errors are inevitable, especially for translators who are unfamiliar with the process. We have developed an annotation checker to automatically flag and correct errors where possible:

Entity Checking. Our annotation checker ensures that changes made in the English translation of entities are propagated to the downstream translation for other target languages. It compares the revised annotations with current annotations and deleted incorrect or redundant slots. Additionally, it locates missing entities or entities that need re-annotation to help annotators quickly synchronize the latest changes.

API Checking. Some datasets such as RiSAWOZ, include the ground truth database search results. For these datasets, we can check the consistency of the API by comparing the results of the API call with the provided ground truth. Our checker resolves observed discrepancies by automatically deleting redundant slots and values in constraints and adding the differences to the slot value mappings. It also shows the precise locations of changes for annotators to review.

6 Experiment

The goal of our experiments is to create an agent in a *target* language, given full training data in the source (Chinese) language, and a varying amount of training data in the target language. We also assume we have access to a machine translation model from Chinese to the target language. We perform our experiments on different target languages in X-RiSAWOZ. Table 1 shows statistics of different data splits used in the experiments, which is the same across all *target* languages.

6.1 Setting

Full-Shot (mono-lingual). This setting is only possible for Chinese since we do not have full training data for target languages. In the full-shot experiments, all of the original Chinese training data is used for training. Note that this setting is not a cross-lingual experiment per se, but a point of comparison for other settings.

Zero-Shot (cross-lingual). In our zero-shot experiments, no manually created target language data is available for training. Instead, we automatically create training data by machine translation of the source language as described in Section 5.1.2. Additionally, we perform two ablations on our automatic training data translation approach: (1) Only using neural alignment (− Dictionary Align) (2) No alignment of any type (− Neural Align).

Few-Shot (cross-lingual). In the few-shot setting, we start from a zero-shot model (with its various ablations) and further fine-tune it on the few-shot dataset in the target language. So the model is trained on both machine translated data and few-shot manually created dataset. In this setting, we also perform an ablation where we only train on the few-shot training data and no machine translated data (*Few-shot Only*).

6.2 Models

In all our experiments, we use the m2m100 (Fan et al., 2020) model for Korean and mBART (Liu et al., 2020) for all other languages. We found mBART to be especially effective in zero-shot settings as the language of its outputs can be controlled by providing a language-specific token at the beginning of decoding. Additionally, its denoising pre-training objective improves its robustness to the remaining translation noise. In each setting, all four dialogue subtasks are done with a single model, where we specify the task by prepending a special token to the input.

Since the dataset for target languages is introduced in this paper, there is only prior work on the Chinese dataset. In Section 7.3, we compare our results to the best previously reported result on RiSAWOZ from Moradshahi et al. (2021) that achieved SOTA on the DST subtask using an mBART model, and from Quan et al. (2020) for other subtasks which use DAMD (Zhang et al., 2020), a Seq2Seq RNN end-to-end dialogue model. We use seven widely-used automatic metrics to compare different models. Please see Section A.2 for details of each metric.

7 Results and Discussion

We first evaluate the models for each turn, assuming that all previous subtasks and steps are correct. We then evaluate the end-to-end accuracy for the whole conversation.

7.1 Turn by Turn Evaluation

To understand how each component performs independently, our first experiment uses the gold data of all the previous turns and subtasks as input in our evaluation (Table 2). In this scenario, errors do not propagate from one subtask to the next in each turn. *Ours* refers to our main approach, which combines all techniques. Each ablation incrementally takes away one of the techniques.

In the zero-shot setting, results vary across added languages, where the agent achieves between 34.6-84.2% on DST, 42.8-67.3% on DA, and 10.2-29.9% on BLEU score. Fine-tuning on the few-shot data improves all metrics for all languages, with the agent achieving between 60.7-84.6% on DST, 38.0-70.5% on DA, and 28.5-46.4% on BLEU score. The improvement in DST is particularly prominent for Hindi, Korean, and English-Hindi, where the quality of machine translation may not be as good. Nonetheless, adding automatically translated data to training greatly improves the accuracy for these languages over the “few-shot only” result.

7.2 Error Analysis

To better understand the inference limitations of our trained agents, we manually inspected the model predictions by randomly selecting 100 validation turns for each domain where the prediction was incorrect. The following are the most common error patterns we observed across all languages:

Implicit Entities: In X-RiSAWOZ dialogues, some entities are not mentioned explicitly in the user’s utterance and need to be *inferred*. These entities include the corresponding price range for a luxury diner, a speaker’s desired attraction for a date with their partner, and hotel rating. These errors are partly due to the limited common-sense capability of the pre-trained language model used (Zhou et al., 2020) and partly due to the training data encouraging the model to copy entities verbatim from the input instead of making logical reasoning. This category accounts for 27% of errors observed.

Multiple Correct Dialogue Acts: In X-RiSAWOZ, the agent often provides an answer as soon as it receives the API call results. However, in some

Language	Setting	DST Acc. ↑	DA Acc. ↑	BLEU ↑
Full-Shot				
Chinese	Ours	**96.43**	**91.74**	**51.99**
Zero-Shot				
English	Ours	**84.23**	**67.27**	**27.14**
	− Dictionary Align	83.42	66.51	22.67
	− Neural Align	82.33	67.79	13.24
French	Ours	**70.75**	**59.27**	**29.88**
	− Dictionary Align	68.22	56.32	25.43
	− Neural Align	64.53	53.33	18.12
Hindi	Ours	**52.09**	**56.06**	**27.42**
	− Dictionary Align	50.12	54.34	23.43
	− Neural Align	48.11	53.21	18.32
Korean	Ours	**34.55**	49.56	**10.17**
	− Dictionary Align	31.47	**50.17**	9.87
	− Neural Align	29.87	49.51	4.59
English-Hindi	Ours	**49.95**	**42.78**	**11.31**
Few-Shot				
Chinese	Few-shot Only	**82.75**	**77.33**	**38.87**
English	Ours	**84.62**	69.44	**46.37**
	− Dictionary Align	83.37	69.74	46.16
	− Neural Align	82.01	**70.45**	45.43
	Few-shot Only	74.52	58.97	45.53
French	Ours	**73.12**	**61.11**	**42.21**
	− Dictionary Align	71.12	60.21	40.12
	− Neural Align	69.68	57.12	38.14
	Few-shot Only	67.55	50.96	44.77
Hindi	Ours	75.16	**59.02**	**38.38**
	− Dictionary Align	**75.32**	57.66	37.54
	− Neural Align	73.21	54.32	34.32
	Few-shot Only	55.77	49.88	38.18
Korean	Ours	**71.17**	**53.52**	**34.93**
	− Dictionary Align	69.57	52.37	34.75
	− Neural Align	69.91	52.00	33.80
	Few-shot Only	60.65	41.47	32.76
English-Hindi	Ours	**60.67**	**37.97**	26.77
	Few-shot Only	56.53	36.50	**28.54**

Table 2: Results on the validation set of X-RiSAWOZ, obtained by feeding the gold input for each subtask in each turn. The best result in each section is in bold. ↑ indicates higher number shows better performance.

cases, the agent asks follow-up questions (e.g., "how many seats do you want for the car?") to narrow down the search results. Since the dataset is constructed via human interactions and not simulation, there are no well-defined policies governing the agent's behavior. Thus, there are examples where multiple dialogue acts can be correct given the input and API constraints. Since during evaluation we can only check the model output against the one response provided as gold, another perfectly fine response can be deemed as incorrect. We discovered that 38% of errors are of this nature.

Incorrect Entities: In DST and DA subtasks, the accuracy is highly dependent on identifying the correct entities in the input. However, there are cases where the model (1) predicts a wrong entity, (2) predicts part of the entity, (3) predicts the entity along with prepositions, articles, etc. (4) omits the entity, or (5) fully hallucinates an entity. We found (1) and (2) to be the most common patterns. (3) can be addressed by a simple pre-processing or text lemmatization. (4) happens with complex sentences with many entities where the model often mispredicts the slot names as well as slot values. (5) is usually caused by data mis-annotations or errors in data processing, where a slot is missing from the input and the model generates the most probable value for it. The remaining 35% of errors fall under this category.

For each language, we also performed a similar analysis to understand if there are language-specific attributes that affect the accuracy and quality of the translated datasets. The result of these analyses is included in the appendix (A.4-A.7).

7.3 Full Conversation Evaluation

The main results of our experiments are reported in Table 3. Following Lin et al. (2021), the evaluation for these experiments is performed end-to-end meaning for each turn, the model output from the previous subtask is used as input for the next subtask. This reflects a real-world scenario where an agent is conversing with the user interactively.

Overall, in the full-shot setting, when training on the Chinese dataset, we improve the state of the art in Joint Goal Accuracy (JGA) by 1.33%, Task Success Rate (TSR) by 5.04%, Dialogue Success Rate (DSR) by 5.35%, and BLEU by 6.82%.

Language	Setting	JGA ↑	TSR ↑	DSR ↑	API ↑	DAA ↑	BLEU ↑	SER ↓
				Full-Shot				
Chinese	Ours	**78.23**	**53.67**	**45.67**	**72.70**	**73.68**	**34.72**	**26.41**
	SOTA	76.90	48.63	40.32	–	–	27.90	30.32
				Zero-Shot				
English	Ours	**43.64**	**22.46**	**16.00**	**44.95**	**40.81**	**14.12**	**47.08**
	− Dictionary Align	38.70	19.22	13.50	39.84	37.35	11.34	49.64
	− Neural Align	38.96	9.50	5.67	40.95	41.96	8.21	59.90
French	Ours	**24.04**	**12.58**	**7.17**	**34.20**	**38.32**	**10.88**	**58.45**
	− Dictionary Align	20.32	5.43	4.18	28.51	35.78	9.72	60.25
	− Neural Align	19.43	3.23	2.11	24.64	28.36	6.81	68.89
Hindi	Ours	**20.32**	**10.11**	**4.32**	**32.32**	**34.23**	**9.13**	**60.43**
	− Dictionary Align	18.31	5.15	3.98	30.12	32.31	8.11	65.43
	− Neural Align	17.32	3.12	3.10	28.51	28.13	7.00	67.25
Korean	Ours	**21.41**	**10.75**	**5.00**	**32.08**	**36.57**	7.27	**64.33**
	− Dictionary Align	19.53	9.46	4.83	27.75	36.33	**7.55**	35.84
	− Neural Align	17.77	8.77	3.67	27.19	25.45	7.12	38.98
English-Hindi	Ours	**9.22**	**4.81**	**2.03**	**10.43**	**26.47**	**5.41**	**63.26**
				Few-Shot				
Chinese	Few-shot Only	**37.69**	**28.04**	**21.00**	**40.73**	**42.30**	**13.89**	**45.44**
English	Ours	**48.91**	**23.13**	**17.17**	**50.06**	**42.45**	**26.33**	**44.93**
	− Dictionary Align	48.40	22.79	16.67	50.03	42.26	25.29	45.01
	− Neural Align	46.31	22.68	16.50	47.61	42.54	25.78	44.78
	Few-shot Only	29.87	16.09	10.50	32.30	30.45	20.00	52.79
French	Ours	**30.85**	**17.17**	**11.83**	**39.97**	**45.03**	**20.92**	**46.26**
	− Dictionary Align	28.51	16.11	9.54	38.11	43.41	19.91	48.35
	− Neural Align	26.45	15.54	9.13	35.74	42.15	16.99	49.26
	Few-shot Only	19.43	3.23	2.11	24.64	28.36	6.81	68.89
Hindi	Ours	**25.62**	**15.67**	**11.31**	**37.54**	**41.32**	**18.51**	**44.26**
	− Dictionary Align	23.12	15.11	10.32	35.14	39.51	16.34	46.76
	− Neural Align	21.12	13.22	8.61	34.11	34.12	15.33	48.97
	Few-shot Only	18.48	8.16	4.50	19.09	23.41	13.15	62.24
Korean	Ours	**26.24**	**14.32**	**10.60**	**35.42**	**38.42**	**20.32**	**43.21**
	− Dictionary Align	24.13	12.53	8.45	23.42	33.34	19.32	47.32
	− Neural Align	23.54	10.23	7.54	22.31	30.42	18.34	50.33
	Few-shot Only	20.66	9.16	5.17	19.39	23.56	17.81	54.57
English-Hindi	Ours	**21.80**	**4.13**	1.83	**22.64**	**21.69**	**5.29**	**66.31**
	Few-shot Only	16.07	3.69	**2.33**	15.65	16.97	3.93	69.61

Table 3: End-to-end results and ablations on the test set of X-RiSAWOZ. The best result in each section is in bold. ↓ indicates lower number shows better performance and vice versa.

The improvements are due to the improved and succinct dialogue representation we have created (Section 4), and contextual representations of transformer models.

In the zero-shot setting, results vary across languages, where the English, French, Hindi, Korean, and English-Hindi agents achieve 35%, 16%, 9%, 11%, and 4% of the DSR score of the full-shot Chinese agent, respectively. In the few-shot setting, the ratio improves to 38%, 26%, 25%, 23%, and 5%. The smallest and biggest improvements are on the English and Hindi dataset respectively. This suggests that the impact of few-shot data is greater when the quality of the pretraining data is lower, which is related to the quality of the translation model between Chinese and the target language.

The Response Generation subtask receives the largest improvement in performance when provided with human supervision in the few-shot data, with a BLEU score improvement of over 10%. This suggests that while translation with alignment is effective for understanding user input, it is not as effective for generating output text. This is partly due to the agent model used, mBART, which is trained with a denoising objective and is thus able to handle noisy input text better.

8 Conclusion

This paper presents a solution for balancing the trade-offs between standard machine translation and human post-editing. By standardizing and establishing best practices for "translation with manual post-editing", and releasing associated toolkits, post-editing can be made faster, more efficient, and cost-effective. We use our methodology to create X-RiSAWOZ, a new end-to-end, high-quality, and large multi-domain multilingual dialogue dataset, covering 5 diverse languages and 1 code-mixed language. We also provide strong baselines for zero/few-shot creation of dialogue agents via cross-lingual transfer. In the few-shot setting, our agents achieve between 60.7-84.6% on DST, 38.0-70.5% on DA, and 28.5-46.4% on RG subtasks across different languages. Overall, our work paves the way for more efficient and cost-effective development of multilingual task-oriented dialogue systems.

9 Limitations

We would have liked to evaluate the generalization of our cross-lingual approach on more languages. For instance, we partially rely on machine translation models for Chinese-to-English translation. Available translation models for other language pairs, especially from/to low-resource languages have much lower quality, and it would be desirable to measure the effect of that in our experiments.

The ontology used for new languages is derived by translating the Chinese ontology. As a result, the entities are not localized. Creating local ontology requires manual effort as one would need to identify websites or databases for scraping or collecting the entities. Once the local entities are collected, we can automatically replace translated entities with local ones to localize the dataset.

Another limitation is the lack of human evaluation for agent responses. BLEU score does not correlate well with human judgment (Sulem et al., 2018), and SER only accounts for the factuality of the response but not grammar or fluency. In future work, we wish to address this by conducting human evaluations in addition to automatic metrics.

10 Ethical Considerations

We do not foresee any harmful or malicious misuse of the technology developed in this work. The data used to train models is about seeking information about domains like restaurants, hotels and tourist attractions, does not contain any offensive content, and is not unfair or biased against any demographic. This work does focus on widely-spoken languages, but we think the cross-lingual approach we proposed can improve future dialogue language technologies for a wider range of languages.

We fine-tune multiple medium-sized (several hundred million parameters) neural networks for our experiments. We took several measures to avoid wasted computation, like performing one run instead of averaging multiple runs (since the numerical difference between different models is large enough to draw meaningful conclusions), and improving batching and representation that improved training speed, and reduced needed GPU time. Please refer to Appendix A.1 for more details about the amount of computation used in this paper.

Acknowledgements

We would like to thank Ruchi Jain for helping us validate the automatically translated Hindi dialogues. This work is supported in part by the National Science Foundation under Grant No. 1900638, the Alfred P. Sloan Foundation under Grant No. G-2020-13938, the Verdant Foundation, Microsoft, KDDI, JPMorgan Chase, and the Stanford Human-Centered Artificial Intelligence (HAI) Institute. This work is also co-funded by the Natural Science Foundation of Xinjiang Uygur Autonomous Region (No. 2022D01D43), Zhejiang Lab (No. 2022KH0AB01). This project has also received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 101021797 (Starlight), and the European Union's Horizon Europe research and innovation programme under grant agreement N° 101070192 (CORTEX^2). This work is also supported in part by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2020-0-01373 and IITP-2022-2021-0-01817).

References

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Ultes Stefan, Ramadan Osman, and Milica Gašić. 2018. Multiwoz - a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Bill Byrne, Karthik Krishnamoorthi, Chinnadhurai Sankar, Arvind Neelakantan, Ben Goodrich, Daniel Duckworth, Semih Yavuz, Amit Dubey, Kyu-Young Kim, and Andy Cedilnik. 2019. Taskmaster-1: Toward a realistic and diverse dialog dataset. 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 4506–4517.

Giovanni Campagna, Silei Xu, Mehrad Moradshahi, Richard Socher, and Monica S. Lam. 2019. Genie: A generator of natural language semantic parsers for virtual assistant commands. In *Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI 2019, pages 394–410, New York, NY, USA. ACM.

Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. *arXiv preprint arXiv:1809.05053*.

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*.

Bosheng Ding, Junjie Hu, Lidong Bing, Mahani Aljunied, Shafiq Joty, Luo Si, and Chunyan Miao. 2022. GlobalWoZ: Globalizing MultiWoZ to develop multilingual task-oriented dialogue systems. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1639–1657, Dublin, Ireland. Association for Computational Linguistics.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2112–2128, Online. Association for Computational Linguistics.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyag Gao, and Dilek Hakkani-Tur. 2019a. Multiwoz 2.1: Multi-domain dialogue state corrections and state tracking baselines. *arXiv preprint arXiv:1907.01669*.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyag Gao, and Dilek Hakkani-Tur. 2019b. Multiwoz 2.1: Multi-domain dialogue state corrections and state tracking baselines. *arXiv preprint arXiv:1907.01669*.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. 2020. Beyond english-centric multilingual machine translation. arxiv e-prints, page. *arXiv preprint arXiv:2010.11125*.

Jessica Ficler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. *arXiv preprint arXiv:1707.02633*.

Markus Freitag and Yaser Al-Onaizan. 2017. Beam search strategies for neural machine translation. *arXiv preprint arXiv:1702.01806*.

George Giannakopoulos, Mahmoud El-Haj, Benoit Favre, Marina Litvak, Josef Steinberger, and Vasudeva Varma. 2011. Tac 2011 multiling pilot overview. *TAC*.

Goran Glavaš, Mladen Karan, and Ivan Vulić. 2020. XHate-999: Analyzing and detecting abusive language across domains and languages. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6350–6365, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Chulaka Gunasekara, Seokhwan Kim, Luis Fernando D'Haro, Abhinav Rastogi, Yun-Nung Chen, Mihail Eric, Behnam Hedayatnia, Karthik Gopalakrishnan, Yang Liu, Chao-Wei Huang, Dilek Hakkani-Tür, Jinchao Li, Qi Zhu, Lingxiao Luo, Lars Liden, Kaili Huang, Shahin Shayandeh, Runze Liang, Baolin Peng, Zheng Zhang, Swadheen Shukla, Minlie Huang, Jianfeng Gao, Shikib Mehri, Yulan Feng, Carla Gordon, Seyed Hossein Alavi, David Traum, Maxine Eskenazi, Ahmad Beirami, Eunjoon, Cho, Paul A. Crook, Ankita De, Alborz Geramifard, Satwik Kottur, Seungwhan Moon, Shivani Poddar, and Rajen Subba. 2020. Overview of the ninth dialog system technology challenge: Dstc9.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Chia-Chien Hung, Anne Lauscher, Ivan Vulić, Simone Ponzetto, and Goran Glavaš. 2022a. Multi2WOZ: A robust multilingual dataset and conversational pretraining for task-oriented dialog. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3687–3703, Seattle, United States. Association for Computational Linguistics.

Chia-Chien Hung, Anne Lauscher, Ivan Vulić, Simone Paolo Ponzetto, and Goran Glavaš. 2022b. Multi2woz: A robust multilingual dataset and conversational pretraining for task-oriented dialog. *arXiv preprint arXiv:2205.10400*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Oleksii Kuchaiev, Jason Li, Huyen Nguyen, Oleksii Hrinchuk, Ryan Leary, Boris Ginsburg, Samuel Kriman, Stanislav Beliaev, Vitaly Lavrukhin, Jack Cook, et al. 2019. Nemo: a toolkit for building ai applications using neural modules. *arXiv preprint arXiv:1909.09577*.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Haoran Li, Abhinav Arora, Shuohui Chen, Anchit Gupta, Sonal Gupta, and Yashar Mehdad. 2021a. MTOP: A comprehensive multilingual task-oriented semantic parsing benchmark. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2950–2962, Online. Association for Computational Linguistics.

Jinchao Li, Qi Zhu, Lingxiao Luo, Lars Liden, Kaili Huang, Shahin Shayandeh, Runze Liang, Baolin Peng, Zheng Zhang, Swadheen Shukla, Ryuichi Takanobu, Minlie Huang, and Jianfeng Gao. 2021b. Multi-domain task-oriented dialog challenge ii at dstc9. In *AAAI-2021 Dialog System Technology Challenge 9 Workshop*.

Zhaojiang Lin, Andrea Madotto, Genta Indra Winata, Peng Xu, Feijun Jiang, Yuxiang Hu, Chen Shi, and

Pascale Fung. 2021. BiToD: A bilingual multi-domain dataset for task-oriented dialogue modeling. *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1 pre-proceedings (NeurIPS Datasets and Benchmarks 2021)*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

A. H. Miller, W. Feng, A. Fisch, J. Lu, D. Batra, A. Bordes, D. Parikh, and J. Weston. 2017. Parlai: A dialog research software platform. *arXiv preprint arXiv:1705.06476*.

Nikita Moghe, Mark Steedman, and Alexandra Birch. 2021. Cross-lingual intermediate fine-tuning improves dialogue state tracking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1137–1150, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mehrad Moradshahi, Giovanni Campagna, Sina Semnani, Silei Xu, and Monica Lam. 2020. Localizing open-ontology QA semantic parsers in a day using machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5970–5983, Online. Association for Computational Linguistics.

Mehrad Moradshahi, Sina Semnani, and Monica Lam. 2023. Zero and few-shot localization of task-oriented dialogue agents with a distilled representation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 886–901, Dubrovnik, Croatia. Association for Computational Linguistics.

Mehrad Moradshahi, Victoria Tsai, Giovanni Campagna, and Monica S Lam. 2021. Contextual semantic parsing for multilingual task-oriented dialogues. *arXiv preprint arXiv:2111.02574*.

Massimo Nicosia, Zhongdi Qu, and Yasemin Altun. 2021. Translate & fill: Improving zero-shot multilingual semantic parsing with synthetic data. *arXiv preprint arXiv:2109.04319*.

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

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037.

Shana Poplack. 1980. Sometimes i'll start a sentence in spanish y termino en espaÑol: toward a typology of code-switching 1. *Linguistics*, 18:581–618.

Jun Quan, Shian Zhang, Qian Cao, Zizhong Li, and Deyi Xiong. 2020. RiSAWOZ: A large-scale multi-domain Wizard-of-Oz dataset with rich semantic annotations for task-oriented dialogue modeling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 930–940, Online. Association for Computational Linguistics.

Osman Ramadan, Paweł Budzianowski, and Milica Gasic. 2018. Large-scale multi-domain belief tracking with knowledge sharing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 432–437.

Mohd Sanad Zaki Rizvi, Anirudh Srinivasan, Tanuja Ganu, Monojit Choudhury, and Sunayana Sitaram. 2021. GCM: A toolkit for generating synthetic code-mixed text. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 205–211, Online. Association for Computational Linguistics.

Carol Scotton. 1993. *Duelling languages : grammatical structure in codeswitching*. Clarendon Press Oxford University Press, Oxford, Eng. New York.

Tom Sherborne, Yumo Xu, and Mirella Lapata. 2020. Bootstrapping a crosslingual semantic parser.

Elior Sulem, Omri Abend, and Ari Rappoport. 2018. Bleu is not suitable for the evaluation of text simplification. *arXiv preprint arXiv:1810.05995*.

Chuanxin Tang, Chong Luo, Zhiyuan Zhao, Wenxuan Xie, and Wenjun Zeng. 2020. Joint time-frequency and time domain learning for speech enhancement. *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*.

Sarah Uhrig, Yoalli Garcia, Juri Opitz, and Anette Frank. 2021. Translate, then parse! a strong baseline for cross-lingual AMR parsing. In *Proceedings of the 17th International Conference on Parsing Technologies and the IWPT 2021 Shared Task on Parsing into Enhanced Universal Dependencies (IWPT 2021)*, pages 58–64, Online. Association for Computational Linguistics.

Eva Vanmassenhove, Dimitar Shterionov, and Matthew Gwilliam. 2021. Machine translationese: Effects of algorithmic bias on linguistic complexity in machine translation. *arXiv preprint arXiv:2102.00287*.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. *arXiv preprint arXiv:1508.01745*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

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. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. Paws-x: A cross-lingual adversarial dataset for paraphrase identification. *arXiv preprint arXiv:1908.11828*.

Yichi Zhang, Zhijian Ou, and Zhou Yu. 2020. Task-oriented dialog systems that consider multiple appropriate responses under the same context. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 05, pages 9604–9611.

Xuhui Zhou, Yue Zhang, Leyang Cui, and Dandan Huang. 2020. Evaluating commonsense in pre-trained language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9733–9740.

Qi Zhu, Christian Geishauser, Hsien chin Lin, Carel van Niekerk, Baolin Peng, Zheng Zhang, Michael Heck, Nurul Lubis, Dazhen Wan, Xiaochen Zhu, Jianfeng Gao, Milica Gašić, and Minlie Huang. 2022. Convlab-3: A flexible dialogue system toolkit based on a unified data format. *arXiv preprint arXiv:2211.17148*.

Qi Zhu, Kaili Huang, Zheng Zhang, Xiaoyan Zhu, and Minlie Huang. 2020a. CrossWOZ: A large-scale Chinese cross-domain task-oriented dialogue dataset. *Transactions of the Association for Computational Linguistics*, 8:281–295.

Qi Zhu, Kaili Huang, Zheng Zhang, Xiaoyan Zhu, and Minlie Huang. 2020b. Crosswoz: A large-scale chinese cross-domain task-oriented dialogue dataset. *Transactions of the Association for Computational Linguistics*, 8:281–295.

Qi Zhu, Zheng Zhang, Yan Fang, Xiang Li, Ryuichi Takanobu, Jinchao Li, Baolin Peng, Jianfeng Gao, Xiaoyan Zhu, and Minlie Huang. 2020c. Convlab-2: An open-source toolkit for building, evaluating, and diagnosing dialogue systems. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The united nations parallel corpus v1. 0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534.

Lei Zuo, Kun Qian, Bowen Yang, and Zhou Yu. 2021. Allwoz: Towards multilingual task-oriented dialog systems for all. *arXiv preprint arXiv:2112.08333*.

A Appendix

A.1 Implementation details

Our code is implemented in PyTorch (Paszke et al., 2019) using GenieNLP (Campagna et al., 2019) [2] library for training and evaluation. We use our newly written library (described in Section 4) for data preprocessing and evaluation which will be released upon publication. We use pre-trained models available through HuggingFace's Transformers library (Wolf et al., 2019). We use *m2m100-418M* model for Korean and *mbart-large-50* for other languages as the neural model for our agent. Both models use a standard Seq2Seq architecture with a bidirectional encoder and left-to-right autoregressive decoder. mBART uses sentence-piece (Kudo and Richardson, 2018) for tokenization, and is pre-trained on text reconstruction task in 50 languages.

In each setting, all four dialogue subtasks are done with a single model, where we specify the task by prepending a special token to the input. We found mBART to be especially effective in zero-shot settings as the language of its outputs can be controlled by providing a language-specific token at the beginning of decoding. Additionally, its denoising pre-training objective improves its robustness to the remaining translation noise.

For translation, we use the publicly available *mbart-large-50-many-to-many-mmt* (~611M parameters) and *m2m100-418M* (~1.2B parameters) models which can directly translate text from any of the 50 supported languages.

We use greedy decoding and train our models using teacher-forcing and token-level cross-entropy loss. We use Adam (Kingma and Ba, 2014) as our optimizer with a start learning rate of 2×10^{-5} and linear scheduling. These hyperparameters were chosen based on a limited hyperparameter search on the validation set. For the numbers reported in the paper, due to cost, we performed only a single run for each experiment.

Our models were trained on virtual machines with a single NVIDIA V100 (16GB memory) GPU on the Azure platform. For a fair comparison, all models were trained for the same number of iterations of 200K in the full-shot setting. In the few-shot setting, we fine-tuned the model for 10K

[2] https://github.com/stanford-oval/genienlp

steps on the few-shot data. Sentences are batched based on their input and approximate output token count for better GPU utilization. We set the total number of tokens per batch to 720. Training and evaluating each model takes about 15 GPU hours on average.

At inference time, we use the predicted belief state as input to subsequent turns instead of ground truth. However, to avoid the conversation from diverging from its original direction, similar to Lin et al. (2021), we use ground-truth agent acts as input for the next turn. We made sure the settings are equivalent for a fair comparison. Additionally, we noted that in many examples the prediction is similar to the gold truth except for small differences such as in case (e.g., "district" vs "District"), or extra punctuation in the predicted output. To address this, during evaluation, we apply entity normalization by using canonical mapping and string pattern matching to map entities to their canonicalized form.

A.2 Evaluation Metrics

Following Moradshahi et al. (2023), we use the following metrics to compare different models. Scores are averaged over all turns unless specified otherwise.

- **Joint Goal Accuracy (JGA)** (Budzianowski et al., 2018): The standard metric for evaluating DST. JGA for a dialogue turn is 1 if all slot-relation-value triplets in the generated belief state match the gold annotation, and is 0 otherwise.
- **Task Success Rate (TSR)** (Lin et al., 2021): A task, defined as a pair of domain and intent, is completed successfully if the agent correctly provides all the user-requested information and satisfies the user's initial goal for that task. TSR is reported as an average over all tasks.
- **Dialogue Success Rate (DSR)** (Lin et al., 2021): DSR is 1 for a dialogue if all user requests are completed successfully, and 0 otherwise. DSR is reported as an average over all dialogues. We use this as the main metric to compare models, since the agent needs to complete all dialogue subtasks correctly to obtain a full score on DSR.
- **API**: For a dialogue turn, is 1 if the model correctly predicts to make an API call, and all the constraints provided for the call match the gold. It is 0 otherwise.
- **Dialogue Act Accuracy (DAA)**: For a dialogue turn, is 1 if the model correctly predicts all the dialogue acts including entities, and is 0 otherwise.
- **BLEU** (Papineni et al., 2002): Measures the natural language response fluency based on n-gram matching with the human-written gold response. BLUE is calculated at the corpus level.
- **Slot Error Rate (SER)** (Wen et al., 2015): It complements BLEU as it measures the factual correctness of natural language responses. For each turn, it is 1 if the response contains all entities present in the gold response, and is 0 otherwise.

A.3 Human Post-Editing

Bilingual speakers of the source and output language were recruited as human translators and post-editors by each team. A user interface (see Fig. 2) was provided for them to perform translation and alignment tasks. The translators were instructed to ensure that the resulting translations were both accurate and fluent. Compensation for their work was provided at the standard rate in their respective countries.

A.4 Error Analysis: French

For the French language, we focused on the Response Generation subtask. We selected 300 of the prediction examples marked as false as not matching exactly the reference. In this set 42.8% of the predictions are completely wrong. Polite forms are particularly problematic. As an example, we can cite the case of the expression *"Tout le plaisir est pour moi, au revoir.* (*It's my pleasure, goodbye.*)" which has four different wrong predictions *"Pas de courtoisie, au revoir.* (*No courtesy, goodbye.*)", *"Pas de gentillesse, au revoir.* (*No kindness, goodbye.*)", *"Je suis heureux de vous servir, au revoir.* (*I am pleased to serve you, goodbye.*)" and *"Pas de bonheur, adieu!* (*No happiness, goodbye*)". The root cause most likely stems from the literal translation of Chinese idioms used in polite expressions (不要客气) to French in the zero-shot training data. However, most of the polite expressions should be easy enough to correct.

We noted that 7.4% of the predictions are just slightly off semantically. For example (*"Je recommande l'université de Xi'an Jiaotong-Liverpool.* (*I recommend Xi'an Jiaotong-Liverpool University.*)" vs. *"l'université de Xi'an Jiaotong-Liverpool de Liverpool est très bien.* (*Xi'an Jiaotong-Liverpool University from Liverpool is very good.*)" with the wrong insertion of *"de Liverpool* (*from Liverpool*)".

On the other hand, 15.4% of the predictions are semantically correct, but with minor errors (syntactic errors, repetitions, etc.). For instance, the meaning of the sentence *"Il y aura une brise sans direction continue samedi prochain.* (*There will be a continuous directionless breeze next Saturday.*)" is the same than the one of the sequence of words *"Le vent une brise sans direction continue vent doit être doux.* (*The wind a breeze without continuous wind direction must be gentle.*)" but this sequence of words is syntactically wrong. The date reference is also missing but it was not mandatory for the correctness of the dialogue. Finally, 34.4% of the supposedly wrong generated responses are in fact correct but just expressed differently, like in *"Elle a une note de 4,3.* (*It has a rating of 4.3.*") vs. *"La note de ce lieu est de 4,3.* (*The rating for this location is 4.3.*)". We think that this kind of difference could be handled by a computation of sentence embedding distance.

As we focused on the Response Generation component, we did not carry out a large-scale qualitative analysis of the Slot-Relation subtask but a quick look at the data seems to indicate that some given slots are often missing from the generated part, like *"the_most_suitable_people"*. For some other slots like "*metro_station*", some values seem to be missing from normalization data like "*peut*" which should be equivalent to "*pouvoir*" and "*true*". This latter error will be quite simple to correct.

A.5 Error Analysis: Hindi

We sample 10% of the errors from each domain from the Hindi validation dataset and analyze these examples manually. The following are the error patterns we observe:

Response Generation. As discussed in Section A.7, there are multiple ways to generate a sentence while matching the semantic content of the gold truth. While such RGs should ideally be marked as correct, their BLEU scores are low. Such instances amount to over 65% of all RG errors. In addition to such kind of errors, we observe that approximately 18% of all the RG error samples are largely accurate but they lack fluency. Here is one such instance where the model is trying to say bye to the user: "ajib hai, alvida!", which translates to "That's strange, bye!". In this example, the model conveys the right message but not in the most polite way. Such instances become more common when the model has to fill the "general" slot, used mainly in greetings. This is possibly because the model finds it more difficult to generate open-ended text than content-guided text.

Erroneous Slot-Relation Values. In some cases, the model predicts the right slot-relation values, but they are deemed incorrect because it predicts the synonym of the gold truth. This amounts to 28% of all the erroneous slot-relation value examples. In addition to such instances, we observe that some slot-relation values are marked as incorrect because of minor differences between the gold truth and the model prediction. These include extra spaces, punctuations, stop words, and the usage of synonyms. Such kind of errors amount to 17.8% of the sampled erroneous slot relation values. Lastly, our analysis reveals that there seems to be an increased amount of confusion between the following pairs of slots: "inform", "request" and "date", "time".

A.6 Error Analysis: Korean

The Korean language poses some unique challenges. In Korean, a word can be made up of multiple characters, and an *eo-jeol* is formed by one or more words to convey a coherent meaning. Spaces are used to delimit an eo-jeol. For instance, postpositions, or *jo-sa* in Korean, are connected to a noun to form an eo-jeol to indicate its grammatical relation to other words in a sentence.

Consider "저는 샤먼에 갈거에요", a sentence containing 3 eo-jeol and 9 characters that means "I will go to Xiamen". "샤먼에" is an eo-jeol meaning "to Xiamen", where "샤먼" is "Xiamen" and "에" (which is a jo-sa) means "to". Because the two words are connected into a single eo-jeol, the annotation is more prone to mistakes. Furthermore, extracting an entity in an eo-jeol is more difficult. This leads to more "incorrect entities" problems in the results for Korean.

Furthermore, Korean possesses distinctive auxiliary verbs/adjectives known as *bo-jo yong-eon*. These bo-jo yong-eon can be connected to the main verbs/adjectives either within a single combined eo-jeol or across multiple eo-jeols, leading to similar challenges in entity annotation. For example, both "친구들과 갈" and "친구들과 가는" means "to go with friends". Here, both "-ㄹ" (the character at the bottom of "갈") and "는" are bo-jo yong-eon meaning "to go". A single English auxiliary verb can map to a wide variety of bo-jo yong-eon depending on the context.

We modified the annotation tool so that it works at the character level instead of word level. To identify number entities, we use heuristics to extract the jo-sa from eo-jeol composed of a number and a jo-sa. Despite this, our analysis suggests that the eo-jeol and bo-jo yong-eon issues account for approximately 5% of the errors encountered.

Another issue that we ran into is how negative questions are answered differently in Korean. For example, when asked "isn't it hot?", "yes" means "it is hot" in English, but "it is not hot" in Korean. This discrepancy caused issues during the annotation process. At times, translators mistakenly mapped "yes" in English to mean "no" in Korean for negative questions, or they transformed them into positive inquiries, which we discovered later on. The former case of mapping "yes" to "no" resulted in inconsistency in entity mapping, especially when both positive and negative questions are present in the dataset for the domain and slots. To address this, we manually corrected the annotation results to ensure consistent entity mappings, which resolved the majority of the errors.

A.7 Error Analysis: English-Hindi Code-mixed

To understand the errors of English-Hindi (en-hi) code-mix set, we also sampled 10% of the erroneous examples for each domain from the en-hi validation set. In addition to the error categories noticed for English (Section 7.2), we observe the following patterns:

Response Generation (45%). Model prediction for Response Generation step is low on BLEU score because there can be multiple ways of code-mixing a sentence. The response could be monolingual, or can be code-mixed to various degrees, or different spans within a sentence could be switched, and such errors account to 19% of the total errors. For example, the gold truth is "*yah* 179 minutes *tak chalta hai*" and the model output is "movie *ki* duration 179 minutes *hai*"[3]. For around 20% samples, the generated responses are incoherent, malformed sentences or unnatural code-mixed sentences. We also observed that the generated sentences are low on fluency, while matching the semantic content of the gold truth, accounting for 6% of total errors. It is our conjecture that the erroneous code-mixed text generation can be ascribed to mBART's restricted ability to generate code-mixed sentences.

Erroneous slot-relation-value (35%). In some cases the model predicts additional slot-relation-values, in addition to the correct slot-relation-values (10% of the erroneous samples). For example, gold truth is "(weather) date equal_to next Tuesday" and the predicted output is "(weather) city equal_to Suzhou, date equal_to next Tuesday". It is likely that the model is copying additional slot-relation-value tuples that are available in the knowledge part of the input. In 23% of the analyzed erroneous samples, the model output has the wrong action, domain, slot, relation or slot values. About 1% of the erroneous samples hallucinated slot values.

Language and Script Difference (20%). Across the DST, DA, and RG steps, the gold truth differs from prediction in terms of the script or the language or both. For instance, the slot value could be in Hindi in the Devanagari script, whereas the model prediction is in English or/and in the Roman script. In some cases, although the values match, differences in script/languages can cause the automatic approach to identify them as an error. For example, the gold truth "(train) date equal_to 'next Sunday morning' , seat_type equal_to 'second class ticket' " differs only slightly from the model output "(train) date equal_to 'next Sunday morning', seat_type equal_to 'second class' ". The measured error rate may not reflect the correct model performance because some of these errors can be reduced by accounting for the semantic match between the generated output and the gold truth.

A.8 Dialogue Example

In Table 4, we show two turns of an example in the original dataset, and its translation to other languages.

A.9 Example of the Checking Process

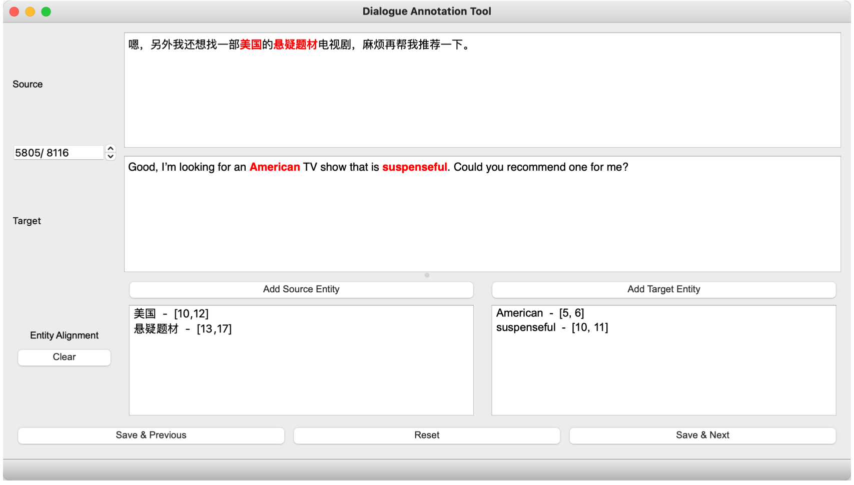
Figure 4 shows an example of our checking process described in Section 5 during the translation from English to French.

[3] In the examples, Hindi tokens (in italic) are written in romanized format for ease of reading. In the datasets, Hindi tokens are in Devanagri script.

Turn 1	DST	Input (EN)	DST: <state> null <endofstate> <history> USER: Hi, my friend is coming to Suzhou to visit me, I want to take him to a commercial center in the mid-price range. Do you have anything to recommend? <endofhistory>
		Output (EN)	(attraction) consumption " mid " , type " commercial center "
		Input (ZH)	DST: <state> null <endofstate> <history> USER: 你好,我朋友要来苏州找我玩,我想带他找一个消费中等的商业中心逛逛,求推荐。 <endofhistory>
		Output (ZH)	(attraction) consumption " 中等 " , type " 商业中心 "
	API	Input (EN)	API: <knowledge> null <endofknowledge> <state> (attraction) consumption " mid " , type " commercial center " <endofstate> <history> USER: Hi, my friend is coming to Suzhou to visit me, I want to take him to a commercial center in the mid-price range. Do you have anything to recommend? <endofhistory>
		Output (EN)	yes
		Input (ZH)	API: <knowledge> null <endofknowledge> <state> (attraction) consumption " 中等 " , type " 商业中心 " <endofstate> <history> USER: 你好,我朋友要来苏州找我玩,我想带他找一个消费中等的商业中心逛逛,求推荐。 <endofhistory>
		Output (ZH)	yes
	DA	Input (EN)	DA: <knowledge> (attraction) address " Guanqian Street, Gusu District, Suzhou City. " , area " Gusu District " , available_options " 4 " , consumption " moderate " , metro_station " true " , name " Guanqian Street " , opening_hours " all day " , phone_number " N/A " , score " 4.3 " , the_most_suitable_people " friends " , ticket_price " free " , type " commercial center " <endofknowledge> <state> (attraction) consumption " mid " , type " commercial center " <endofstate> <history> USER: Hi, my friend is coming to Suzhou to visit me, I want to take him to a commercial center in the mid-price range. Do you have anything to recommend? <endofhistory>
		Output (EN)	(attraction) recommend name " Guanqian Street "
		Input (ZH)	DA: <knowledge> （ attraction ） address " 苏州市姑苏区观前街 " , area " 姑苏区 " , available_options " 4 " , consumption " 中等 " , metro_station " 是 " , name " 观前街 " , opening_hours " 全天 " , phone_number " 无 " , score " 4.3 " , the_most_suitable_people " 朋友出游 " , ticket_price "免费 " , type " 商业中心 " <endofknowledge> <state> (attraction) consumption " 中等 " , type " 商业中心 " <endofstate> <history> USER: 你好,我朋友要来苏州找我玩,我想带他找一个消费中等的商业中心逛逛,求推荐。<endofhistory>
		Output (ZH)	(attraction) recommend name " 观前街 "
	RG	Input (EN)	RG: <actions> (attraction) recommend name " Guanqian Street " <endofactions> <history> USER: Hi, my friend is coming to Suzhou to visit me, I want to take him to a commercial center in the mid-price range. Do you have anything to recommend? <endofhistory>
		Output (EN)	You can go to Guanqian Street.
		Input (ZH)	API: <knowledge> null <endofknowledge> <state> (attraction) consumption " 中等 " , type " 商业中心 " <endofstate> <history> USER: 你好,我朋友要来苏州找我玩,我想带他找一个消费中等的商业中心逛逛,求推荐。 <endofhistory>
		Output (ZH)	去观前街吧。

Turn 2	DST	Input (EN)	DST: <state> (attraction) consumption " mid " , type " commercial center " <endofstate> <history> AGENT_ACTS: (attraction) recommend name " Guanqian Street " USER: Oh yeah, why didn't I think of that? When is it open? <endofhistory>
		Output (EN)	(attraction) consumption " mid " , name " Guanqian Street " , the_most_suitable_people " friend " , type " commercial center "
		Input (ZH)	DST: <state> (attraction) consumption " 中等 " , type " 商业中心 " <endofstate> <history> AGENT_ACTS: (attraction) recommend name " 观前街 " USER: 对呀,我怎么没想起来呢,什么时间开放呀? <endofhistory>
		Output (ZH)	(attraction) consumption " 中等 " , name " 观前街 " , type " 商业中心 "
	API	Input (EN)	API: <knowledge> (attraction) address " 苏州市姑苏区观前街 " , area " 姑苏区 " , available_options " 4 " , consumption " 中等 " , metro_station " 是 " , name " 观前街 " , opening_hours " 全天 " , phone_number " 无 " , score " 4.3 " , the_most_suitable_people " 朋友出游 " , ticket_price " 免费 " , type " 商业中心 " <endofknowledge> <state> (attraction) consumption " 中等 " , name " 观前街 " , type " 商业中心 " <endofstate> <history> AGENT_ACTS: (attraction) recommend name " 观前街 " USER: 对呀,我怎么没想起来呢,什么时间开放呀? <endofhistory>
		Output (EN)	yes
		Input (ZH)	API: <knowledge> null <endofknowledge> <state> (attraction) consumption " 中等 " , type " 商业中心 " <endofstate> <history> USER: 你好,我朋友要来苏州找我玩,我想带他找一个消费中等的商业中心逛逛,求推荐。 <endofhistory>
		Output (ZH)	yes
	DA	Input (EN)	DA: <knowledge> (attraction) address " Guanqian Street, Gusu District, Suzhou City. " , area " Gusu District " , available_options " 1 " , consumption " moderate " , features " You can try food from time-honored Suzhou brands, such as Songhelou Restaurant, Huang Tianyuan, and visit Xuanmiao Temple, the place that gave the street its name. " , metro_station " true " , name " Guanqian Street " , opening_hours " all day " , phone_number " N/A " , score " 4.3 " , the_most_suitable_people " friends " , ticket_price " free " , type " commercial center " <endofknowledge> <state> (attraction) consumption equal_to " mid " , name equal_to " Guanqian Street " , the_most_suitable_people equal_to " friend " , type equal_to " commercial center " <endofstate> <history> AGENT_ACTS: (attraction) recommend name equal_to " Guanqian Street " USER: Oh yeah, why didn't I think of that? When is it open? <endofhistory>
		Output (EN)	(attraction) inform opening_hours " all day "
		Input (ZH)	DA: <knowledge> (attraction) address " 苏州市姑苏区观前街 " , area " 姑苏区 " , available_options " 4 " , consumption " 中等 " , metro_station " 是 " , name " 观前街 " , opening_hours " 全天 " , phone_number " 无 " , score " 4.3 " , the_most_suitable_people " 朋友出游 " , ticket_price "免费 " , type " 商业中心 " <endofknowledge> <state> (attraction) consumption " 中等 " , type " 商业中心 " <endofstate> <history> USER: 你好,我朋友要来苏州找我玩,我想带他找一个消费中等的商业中心逛逛,求推荐。<endofhistory>
		Output (ZH)	(attraction) inform opening_hours " 全天 "
	RG	Input (EN)	RG: <actions> (attraction) inform opening_hours " all day " <endofactions> <history> USER: Oh yeah, why didn't I think of that? When is it open? <endofhistory>
		Output (EN)	It's open all day.
		Input (ZH)	RG: <actions> (attraction) inform opening_hours " 全天 " <endofactions> <history> USER: 对呀,我怎么没想起来呢,什么时间开放呀? <endofhistory>
		Output (ZH)	全天开放哟。

Table 4: An example from X-RiSAWOZ validation set in Chinese and English. For brevity, only the first 2 turns are shown.



5805/ 8116

Source: 嗯，另外我还想找一部**美国**的**悬疑题材**电视剧，麻烦再帮我推荐一下。

Target: Good, I'm looking for an **American** TV show that is **suspenseful**. Could you recommend one for me?

美国 - [10,12]
悬疑题材 - [13,17]

American - [5, 6]
suspenseful - [10, 11]

Figure 2: A screenshot of the annotation tool used by translators to translate a sentence from Chinese to English and mark the entity spans to create the slot value alignment. The entity spans show the position of words for English and characters for Chinese.

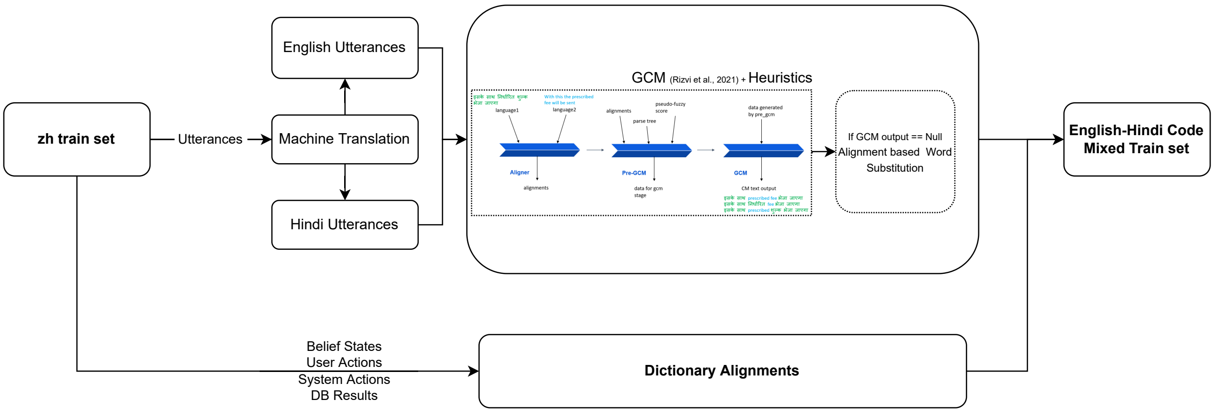


Figure 3: English-Hindi code-mixed train set is generated using a pipeline that combines GCM (Rizvi et al., 2021), and word alignment to generate code-mixed utterances. Entities in Belief states, user and system actions are substituted using dictionary alignment.

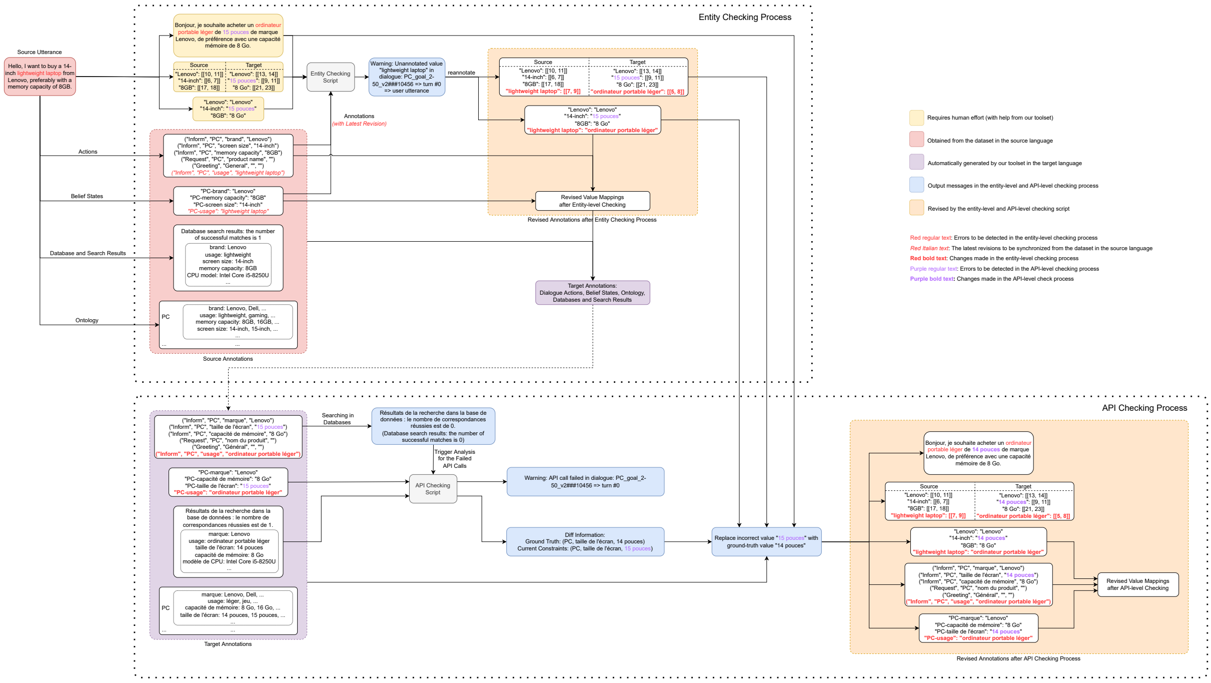

Entity Checking Process
Source Utterance
Hello, I want to buy a 14-inch lightweight laptop from Lenovo, preferably with a memory capacity of 8GB.
Bonjour, je souhaite acheter un ordinateur portable léger de 15 pouces de marque Lenovo, de préférence avec une capacité mémoire de 8 Go.
Source
Target
"Lenovo": [[10, 11]]
"14-inch": [[6, 7]]
"8GB": [[17, 18]]
"Lenovo": [[13, 14]]
"15 pouces": [[9, 11]]
"8 Go": [[21, 23]]
"Lenovo": "Lenovo"
"14-inch": "15 pouces"
"8GB": "8 Go"
Entity Checking Script
Warning: Unannotated value "lightweight laptop" in dialogue: PC_goal_2-50_v2###10456 => turn #0 => user utterance
reannotate
Annotations
(with Latest Revision)
Actions
Belief States
Database and Search Results
Ontology
("Inform", "PC", "brand", "Lenovo")
("Inform", "PC", "screen size", "14-inch")
("Inform", "PC", "memory capacity", "8GB")
("Request", "PC", "product name", "")
("Greeting", "General", "", "")
("Inform", "PC", "usage", "lightweight laptop")
"PC-brand": "Lenovo"
"PC-memory capacity": "8GB"
"PC-screen size": "14-inch"
"PC-usage": "lightweight laptop"
Database search results: the number of successful matches is 1
brand: Lenovo
usage: lightweight
screen size: 14-inch
memory capacity: 8GB
CPU model: Intel Core i5-8250U
...
PC
brand: Lenovo, Dell, ...
usage: lightweight, gaming, ...
memory capacity: 8GB, 16GB, ...
screen size: 14-inch, 15-inch, ...
...
...
...
Source Annotations
Source
Target
"Lenovo": [[10, 11]]
"14-inch": [[6, 7]]
"8GB": [[17, 18]]
"lightweight laptop": [[7, 9]]
"Lenovo": [[13, 14]]
"15 pouces": [[9, 11]]
"8 Go": [[21, 23]]
"ordinateur portable léger": [[5, 8]]
"Lenovo": "Lenovo"
"14-inch": "15 pouces"
"8GB": "8 Go"
"lightweight laptop": "ordinateur portable léger"
Revised Value Mappings after Entity-level Checking
Revised Annotations after Entity Checking Process
Target Annotations: Dialogue Actions, Belief States, Ontology, Databases and Search Results
Requires human effort (with help from our toolset)
Obtained from the dataset in the source language
Automatically generated by our toolset in the target language
Output messages in the entity-level and API-level checking process
Revised by the entity-level and API-level checking script
Red regular text: Errors to be detected in the entity-level checking process
Red Italian text: The latest revisions to be synchronized from the dataset in the source language
Red bold text: Changes made in the entity-level checking process
Purple regular text: Errors to be detected in the API-level checking process
Purple bold text: Changes made in the API-level check process
API Checking Process
("Inform", "PC", "marque", "Lenovo")
("Inform", "PC", "taille de l'écran", "15 pouces")
("Inform", "PC", "capacité de mémoire", "8 Go")
("Request", "PC", "nom du produit", "")
("Greeting", "Général", "", "")
("Inform", "PC", "usage", "ordinateur portable léger")
Searching in Databases
Résultats de la recherche dans la base de données : le nombre de correspondances réussies est de 0.
(Database search results: the number of successful matches is 0)
Trigger Analysis for the Failed API Calls
"PC-marque": "Lenovo"
"PC-capacité de mémoire": "8 Go"
"PC-taille de l'écran": "15 pouces"
"PC-usage": "ordinateur portable léger"
API Checking Script
Warning: API call failed in dialogue: PC_goal_2-50_v2###10456 => turn #0
Résultats de la recherche dans la base de données : le nombre de correspondances réussies est de 1.
marque: Lenovo
usage: ordinateur portable léger
taille de l'écran: 14 pouces
capacité de mémoire: 8 Go
modèle de CPU: Intel Core i5-8250U
...
Diff Information:
Ground Truth: (PC, taille de l'écran, 14 pouces)
Current Constraints: (PC, taille de l'écran, 15 pouces)
PC
marque: Lenovo, Dell, ...
usage: léger, jeu, ...
capacité de mémoire: 8 Go, 16 Go, ...
taille de l'écran: 14 pouces, 15 pouces, ...
...
...
...
Target Annotations
Replace incorrect value "15 pouces" with ground-truth value "14 pouces"
Bonjour, je souhaite acheter un ordinateur portable léger de 14 pouces de marque Lenovo, de préférence avec une capacité mémoire de 8 Go.
Source
Target
"Lenovo": [[10, 11]]
"14-inch": [[6, 7]]
"8GB": [[17, 18]]
"lightweight laptop": [[7, 9]]
"Lenovo": [[13, 14]]
"14 pouces": [[9, 11]]
"8 Go": [[21, 23]]
"ordinateur portable léger": [[5, 8]]
"Lenovo": "Lenovo"
"14-inch": "14 pouces"
"8GB": "8 Go"
"lightweight laptop": "ordinateur portable léger"
("Inform", "PC", "marque", "Lenovo")
("Inform", "PC", "taille de l'écran", "14 pouces")
("Inform", "PC", "capacité de mémoire", "8 Go")
("Request", "PC", "nom du produit", "")
("Greeting", "Général", "", "")
("Inform", "PC", "usage", "ordinateur portable léger")
"PC-marque": "Lenovo"
"PC-capacité de mémoire": "8 Go"
"PC-taille de l'écran": "14 pouces"
"PC-usage": "ordinateur portable léger"
Revised Value Mappings after API-level Checking
Revised Annotations after API Checking Process

Figure 4: The entity and API checking process of X-RiSAWOZ from English to French.