MLT-DR: Multi-Lingual/Task Demonstration Retrieval An Attempt towards Generalized Retriever for In-Context Learning

Kazuma Hashimoto Arjun Reddy Akula Karthik Raman Michael Bendersky
Google DeepMind, Mountain View
{kazumah,arjunakula,karthikraman,bemike}@google.com

Abstract

This paper presents Multi-Lingual/Task Demonstration Retrieval (MLT-DR) for in-context learning with Large Language Models (LLMs). Our goal is to investigate how dense demonstration retrieval models are generalized across languages and tasks. We first convert 81 tasks into a common format, covering various languages, task types, and domains. For 8 English-based tasks among them, we use machine translation to create synthetic multi/cross-lingual tasks, by translating the examples into non-English languages to explicitly cover more than 130 languages. We then use an instruction-tuned LLM to estimate utility of demonstrations for all the tasks to train the demonstration retrieval models. In our experiments, we show an interesting counterintuitive observation; to compute embeddings of demonstrations, using both the input and ground-truth output hurts the generalization ability of the retriever on unseen tasks whose output space is quite different from those in the seen task set. We also examine that our retriever robustly works even with LLMs that we did not touch during the development of the models.

1 Introduction

In-Context Learning (ICL) is an emergent strategy to make Large Language Models (LLMs) perform a task by showing its instruction and *demonstrations* (i.e., input-output pairs) without fine-tuning the LLMs (Brown et al., 2020; Zhao et al., 2021). A crucial research question in this line of work is how to select demonstrations for a new test input. A well-studied approach is to use a general or task-specific text encoder to retrieve demonstrations whose inputs are similar to the test input (Liu et al., 2022). Furthermore, such a text retriever can be effectively fine-tuned by estimating the utility of the demonstrations for a specific LLM (Rubin et al., 2022; Luo et al., 2023).

Li et al. (2023) and Wang et al. (2023) have made progress towards fine-tuning a single demonstration retriever for multiple tasks. They have even shown that the multi-task demonstration retrievers can be generalized on *unseen* datasets (that are *not* used in fine-tuning the retrievers). The key factor is that the unseen datasets share the output formats with those used in the fine-tuning.[1] What is the boundary of the generalization ability?

As an attempt to answer this question, we investigate capabilities of Multi-Lingual/Task Demonstration Retrieval (MLT-DR). We first collect 81 tasks from publicly available datasets,[2] covering diverse languages, task types, and domains. We apply a data augmentation technique to generate synthetic multi/cross-lingual tasks for 8 English-based tasks to improve the generalization ability on low-resource languages, by using machine translation for more than 130 languages. We then fine-tune a general multi-lingual text retriever with feedbacks from an LLM and evaluate fine-tuned models both on seen and unseen tasks.

The findings in our experiments are summarized as follows:

- A counterintuitive finding is that using both the input and ground-truth output to compute demonstration embeddings hurts the generalization ability on unseen tasks, especially when the output spaces are semantically non-trivial.
- The simple translation-based data augmentation helps preserve the generalization ability for low-resource languages (and cross-lingual ICL).

[1] Sentiment classification in a different domain, natural language inference in a different input style, code summarization for different programming languages, etc.

[2] We use the two terms, "tasks" and "datasets," interchangeably as in Wang et al. (2023).

- The fine-tuned retriever can be used for unseen LLMs, and thus we believe that our retriever will serve as a baseline, a building block to be combined with various techniques, starting points to try further fine-tuning, etc. for future research.

2 Multi-Task Demonstration Retrieval

A multi-task demonstration retriever R is designed to estimates $s(d|x, t)$, a utility score of a demonstration d given an input x and its corresponding task t (Li et al., 2023; Wang et al., 2023). It is a common practice to model this as a dense retrieval model (Karpukhin et al., 2020):

$$s(d|x, t) = E_q(x, t) \cdot E_c(d, t), \quad (1)$$

where E_q is an encoder model for the query input, and E_c for the demonstration candidate. We fine-tune a general dense retrieval model R_0; for our primary research question, we assume that R_0 can handle many languages and domains in diverse text formats (like mT5 (Xue et al., 2021)) and is trained by a general task-agnostic text retrieval objective (like Izacard et al. (2021)).

Contrastive Learning The dense retriever model is usually fine-tuned with contrastive learning (Karpukhin et al., 2020). The previous studies used various forms of contrastive learning; for example, Wang et al. (2023) used a combination of cross-attention and dense-retrieval models with a knowledge distillation technique. In this work, we follow a simple and well-established formulation in Yang et al. (2019). To do this, we construct a query set $\mathcal{Q}_t$ and a demonstration candidate set $\mathcal{C}_t$, by splitting the original training set of the task.

Sampling candidates We first sample demonstration candidates (from $\mathcal{C}_t$) for a query input $x \in \mathcal{Q}_t$, by combining two types:

- retrieval-based candidates and

- random candidates.

ℓ candidates are given by the baseline retriever R_0, and m candidates by random sampling, resulting in $(\ell + m)|\mathcal{Q}_t|$ query-candidate pairs for the task t. $(\ell, m) = (10, 10)$ is the default setting, except that we use $(\ell, m) = (50, 50)$ for very small datasets.

Scoring candidates Next, we annotate the usefulness of a candidate d to perform the task t for x. The usefulness is scored by using an LLM:

$$u(d|x, y, t), \quad (2)$$

where y is a gold output of x. We employ the incremental utility function in Hashimoto et al. (2024), where the scores are in the range of $[0.0, 1.0]$;

- $u(d|x, y, t) = 0.5$ means that d does not affect the LLM's prediction,

- $u(d|x, y, t) > 0.5$ means a positive effect, and

- $u(d|x, y, t) < 0.5$ means a negative effect.

The utility scores are annotated in a task-specific fashion as described in Appendix A.1. We use the utility scores to select *positive* and *hard negative* candidates for the contrastive learning.

Positive candidates For x, a positive candidate d_p satisfies

$$u(d_\mathrm{p}|x, y, t) \geq 0.5 + \delta_1, \quad (3)$$

where $\delta_1 \in (0.0, 0.5]$ is a margin to ensure the quality of d_p. The larger the margin value is, the more significant the contribution of d_p is. However, there is a trade-off; a large margin value reduces the number of the training examples we can use. We have tried different values in the development of our framework, and we empirically set $\delta_1 = 0.05$.

Hard negative candidates We pair d_p with a set of hard negative candidates $\{d_\mathrm{n}\}$, such that they satisfy

$$u(d_\mathrm{p}|x, y, t) - u(d_\mathrm{n}|x, y, t) \geq \delta_2, \quad (4)$$

where $\delta_2 \in [0.0, 1.0]$ is another margin to ensure the quality difference between the positive and hard negatives; we empirically set $\delta_2 = 0.1$.

Multi-task fine-tuning Consequently, we have a set of the tuples

$$(x, d_\mathrm{p}, \{d_\mathrm{n}\}) \quad (5)$$

for the task. Then the baseline retriever R_0 is fine-tuned to satisfy $s(d_\mathrm{p}|x, t) > s(d_\mathrm{n}|x, t)$ by the contrastive learning. The fine-tuning process is done by mixing the tuples from all the tasks we use for the retriever training.

3 The Role of Ground-Truth Outputs

There are two major dimensions in the design of the demonstration retriever in Section 2: what texts are fed into

1) the query encoder E_q and

2) the candidate encoder E_c.

The former is relatively straightforward; we can concatenate a task instruction of t and the query text: $[\text{Instruction}(t); x]$ as done in Li et al. (2023) and also in task-aware retrievers (Asai et al., 2023; Su et al., 2023).

For the candidate encoder, we find a standard practice in the previous studies (Rubin et al., 2022; Li et al., 2023; Luo et al., 2023; Wang et al., 2023); they concatenate the input and ground-truth output of the demonstration:

$$[\text{Instruction}(t); d_{\text{in}}; d_{\text{out}}],$$

where the instruction is used optionally for the multi-task learning cases. We may think that this is a natural and reasonable design; however, we cast doubt on this from a view point of the generalization ability on unseen tasks.

Diversity in the output space Let's think about tasks whose outputs are specifically designed for them. Classification is considered to be the most representative one. For some datasets, the output space is limited and not ambiguous:

- {"positive", "negative", "neutral"} in sentiment classification,
- {"entailment", "contradiction", "neutral"} in natural language inference, and
- {"sports", "music", ...} in topic classification.

For others, we see diverse, unlimited, and domain-specific labels: intent classification, relation classification, etc. It is often the case that such class labels are represented with simple words or short phrases, and they are not always comprehensive even for humans. Other example tasks are slot labeling and named entity recognition, where slot/entity labels can be arbitrary strings, and the output format can be designed in various ways (Raman et al., 2022). Is the candidate encoder robust in the diverse output space?

To answer this question, we compare the following three designs for the demonstration representations by the candidate encoder:

- STD: $[\text{Instruction}(t); d_{\text{in}}; d_{\text{out}}]$,
- DESC: $[\text{Instruction}(t); d_{\text{in}}; \text{Description}(d_{\text{out}})]$,
- NO: $[\text{Instruction}(t); d_{\text{in}}]$.

STD is the standard approach in the previous work as mentioned above.

DESC is to replace d_{out} with its description, $\text{Description}(d_{\text{out}})$, to explain the meaning of the output (Rastogi et al., 2020; Gao et al., 2023b). We apply DESC to tasks with symbolic outputs (e.g., classification), and manually give a description for each output candidate. For example, in the DDI13 relation extraction task, we adapt the original definitions of the relation labels in the dataset paper (Herrero-Zazo et al., 2013); if we cannot find definitions even in the dataset papers, we refer to training examples to come up with the descriptions.

NO removes the use of d_{out}, which is *counterintuitive* against the common practice. During the development of DESC, we have observed that it is not trivial to provide comprehensive descriptions, and the actual examples themselves clearly tell us the meaning of the output space (Simard et al., 1992; Zhang et al., 2020). This motivates us to investigate NO solely based on the input representations.

4 Experimental Settings

4.1 LLM and Retriever

We use Flan-PaLM2 (S) (Google et al., 2023) as our main LLM, and follow the prompt design in Gao et al. (2023a). As the baseline (multi-lingual) retriever R_0, we use the t5x-retrieval code base (Ni et al., 2022) to fine-tune mT5 large (Xue et al., 2021) with a general text retrieval objective in Izacard et al. (2021) on the mC4 corpus (Xue et al., 2021). The retriever has 565M model parameters.

4.2 Tasks

Seen tasks To fine-tune our retrievers, we collect NLP tasks in diverse languages and domains from publicly available resources like Flan-v1 (Wei et al., 2021), MTEB (Muennighoff et al., 2023), those used in Li et al. (2023), and others, resulting in 81 tasks in total. The complete list of them is summarized in Table 1. For each task, we manually write a long task instruction to construct the prompt for the LLM, and a short task instruction (i.e., Instruction(t)) for the retriever.

No.	Name	Type	Languages	Source	Scoring	$\lvert\mathcal{Q}_t\rvert$	$\lvert\mathcal{C}_t\rvert$
01	WMT14 en→fr (Bojar et al., 2014)	Machine translation	en, fr	Link	GLEU	100,000	30,059,732
02	WMT14 fr→en (Bojar et al., 2014)	Machine translation	en, fr	Link	GLEU	100,000	30,059,732
03	WMT16 en→de (Bojar et al., 2016)	Machine translation	de, en	Link	GLEU	60,000	4,143,251
04	WMT16 de→en (Bojar et al., 2016)	Machine translation	de, en	Link	GLEU	60,000	4,143,251
05	WMT16 en→ru (Bojar et al., 2016)	Machine translation	en, ru	Link	GLEU	30,000	2,296,592
06	WMT16 ru→en (Bojar et al., 2016)	Machine translation	en, ru	Link	GLEU	30,000	2,296,592
07	ANLI r1 (Nie et al., 2020)	Natural language inference	en [+MT]	Link	Probability	8,473	8,473
08	ANLI r2 (Nie et al., 2020)	Natural language inference	en	Link	Probability	22,730	22,730
09	ANLI r3 (Nie et al., 2020)	Natural language inference	en	Link	Probability	30,000	70,459
10	QNLI (Rajpurkar et al., 2018)	Natural language inference	en	Link	Probability	30,000	74,543
11	MNLI (Williams et al., 2018)	Natural language inference	en	Link	Probability	30,000	100,000
12	WNLI (Levesque et al., 2012a)	Natural language inference	en	Link	Probability	317	318
13	MRPC (Dolan and Brockett, 2005)	Paraphrase identification	en	Link	Probability	200	3,268
14	PAWS (Zhang et al., 2019)	Paraphrase identification	en	Link	Probability	30,000	19,401
15	Tatoeba (Artetxe and Schwenk, 2019)	Translation identification	sqi, fry, kur, tur, ...	Link	Probability	30,000	177,554
16	IMDB (Maas et al., 2011)	Sentiment classification	en	Link	Probability	12,400	12,400
17	SST2 (Socher et al., 2013)	Sentiment classification	en	Link	Probability	30,000	37,149
18	Yelp (Fast.AI)	Sentiment classification	en	Link	Probability	30,000	100,000
19	Tweet Sentiment Extraction (Kaggle)	Sentiment classification	en [+MT]	Link	Probability	10,000	17,281
20	AfriSenti (Muhammad et al., 2023a)	Sentiment classification	amh, hau, ibo, ...	Link	Probability	30,000	33,685
21	TweetEval-emoji (Barbieri et al., 2018)	Emoji classification	en	Link	Probability	20,000	25,000
22	TweetEval-emotion (Mohammad et al., 2018)	Emotion classification	en	Link	Probability	1,600	1,657
23	DialogEmotion (Kumar et al., 2024)	Multi-speaker emotion classification	en, hi	Link	F1	700	799
24	Massive-intent (FitzGerald et al., 2022)	Dialog intent classification	af, am, ar, az, ...	Link	Probability	30,000	100,000
25	MTOP-domain (Li et al., 2021)	Dialog domain classification	de, en, es, fr, ...	Link	Probability	30,000	43,928
26	MTOP-intent (Li et al., 2021)	Dialog intent classification	de, en, es, fr, ...	Link	Probability	30,000	43,928
27	ATIS-intent (Price, 1990)	Multi-label dialog intent classification	en	Link	F1	2,000	2,189
28	E2ENLG-reversed (Dušek et al., 2019)	Semantic parsing (text to dict)	en	Link	F1	16,662	16663
29	WikiSQL (Zhong et al., 2017)	Semantic parsing (text/table to SQL)	en	Link	GLEU	20,000	36,355
30	BC5CDR (Li et al., 2016)	Named entity recognition (biomedical)	en	Link	F1	2,000	2,560
31	BioNLP13PC (Ohta et al., 2013)	Named entity recognition (biomedical)	en	Link	F1	1,000	1,499
32	JNLPBA (Huang et al., 2020)	Named entity recognition (biomedical)	en	Link	F1	9,000	9,346
33	MultiCoNER2 (Fetahu et al., 2023)	Named entity recognition	de, fa, fr, ...	Link	F1	30,000	140,824
34	CoNLL2003 (Tjong Kim Sang and De Meulder, 2003)	Named entity recognition	en	Link	F1	7,000	7,041
35	MTOP-slot (Li et al., 2021)	Dialog slot labeling	en, fr, hi	Link	F1	19,000	19,811
36	SNIPS-slot (Coucke et al., 2018)	Dialog slot labeling	en	Link	F1	6,000	7,084
37	ATIS-slot (Price, 1990)	Dialog slot labeling	en	Link	F1	2,000	2,478
38	SemRel (Hendrickx et al., 2010)	Relation classification (nominals)	en [+MT]	Link	Probability	3,800	4,000
39	DDI13 (Herrero-Zazo et al., 2013)	Relation classification (drugs)	en	Link	Probability	8,000	10,779
40	ChemProt (Islamaj Doğan et al., 2019)	Relation classification (chemical and protein)	en	Link	Probability	9,000	10,460
41	WordSeg (Bañón et al., 2020)	Word segmentation	en	Link	GLEU	30,000	100,000
42	FixPunct (Bañón et al., 2020)	Punctuation fix	en	Link	GLEU	30,000	100,000
43	CoLA (Warstadt et al., 2019)	Linguistic acceptability judgment	en	Link	Probability	4,175	4,176
44	CoNLL2000 (Tjong Kim Sang and Buchholz, 2000)	Syntactic phrase chunking	en	Link	F1	4,000	4,936
45	Pronoun (Rahman and Ng, 2012)	Coreference resolution	en	Link	Probability	561	561
46	WSC (Levesque et al., 2012b)	Coreference resolution	en	Link	Probability	252	252
47	WinoGrande (Sakaguchi et al., 2019)	Sentence completion	en	Link	Probability	20,099	20,099
48	WiC (Pilehvar and Camacho-Collados, 2019)	Word sense disambiguation	en	Link	Probability	2,614	2,614
49	Python (Lu et al., 2021)	Code summarization	en	Link	GLEU	30,000	100,000
50	Java (Lu et al., 2021)	Code summarization	en	Link	GLEU	30,000	100,000
51	Go (Lu et al., 2021)	Code summarization	en	Link	GLEU	30,000	100,000
52	PHP (Lu et al., 2021)	Code summarization	en	Link	GLEU	30,000	100,000
53	Gigaword (Napoles et al., 2012)	Text summarization	en	Link	GLEU	30,000	100,000
54	SAMSum (Gliwa et al., 2019)	Dialog summarization	en	Link	GLEU	7,366	7,366
55	iDebate (Wang and Ling, 2016)	Debate summarization	en [+MT]	Link	GLEU	859	800
56	MultiHateCheck (Röttger et al., 2022)	Hate speech detection/classification	en, fr, hi, it, ...	Link	Probability	20,055	20,055
57	Toxic (Muennighoff et al., 2023)	Toxic text detection	en	Link	Probability	24,900	24,900
58	Countfact (O'Neill et al., 2021)	Counterfactual review detection	de, en, ja	Link	Probability	7,500	7,718
59	Irony (Van Hee et al., 2018)	Irony detection	en	Link	Probability	1,400	1,462
60	Offensive (Zampieri et al., 2019)	Offensive text detection	en	Link	Probability	5,000	6,916
61	Sarcasm (Abu Farha et al., 2022)	Sarcasm detection	ar, en	Link	Probability	2,500	3,414
62	SQuAD2 (Rajpurkar et al., 2018)	Reading comprehension	en	Link	GLEU	30,000	100,119
63	BoolQ (Clark et al., 2019)	Reading comprehension	en [+MT]	Link	Probability	4,613	4,614
64	DROP (Dua et al., 2019)	Reading comprehension (numerical)	en	Link	Probability	29,635	46,621
65	OpenbookQA (Mihaylov et al., 2018)	Reading comprehension	en	Link	Probability	2,478	2,478
66	Cosmos (Huang et al., 2019)	Reading comprehension (common sense)	en	Link	Probability	12,531	12,531
67	SciDocs (Cohan et al., 2020)	Relevance, re-ranking	en	Link	Probability	30,000	99,159
68	HotpotQA (Yang et al., 2018)	Relevance, re-ranking	en	Link	F1	30,000	60,447
69	AI2 ARC-easy (Clark et al., 2018)	Closed-book question answering	en	Link	Probability	1,025	1,026
70	AI2 ARC-challenge (Clark et al., 2018)	Closed-book question answering	en	Link	Probability	459	460
71	TriviaQA (Joshi et al., 2017)	Closed-book question answering	en	Link	Probability	30,000	108,184
72	Math (Saxton et al., 2019)	Math question answering	en	Link	Probability	30,000	100,000
73	CommonGen (Lin et al., 2020)	Constrained text generation (common sense)	en	Link	GLEU	30,000	37,189
74	SNLI-en (Bowman et al., 2015)	Constrained text generation (entailment)	en	Link	GLEU	10,112	33,106
75	PIQA-qgen (Bisk et al., 2019)	Question/query generation	en [+MT]	Link	GLEU	7,956	7,957
76	arXiv (Muennighoff et al., 2023)	Multi-label topic/category classification	en	Link	F1	30,000	69,113
77	medRxiv (Muennighoff et al., 2023)	Topic/category classification	en	Link	Probability	5,000	16,229
78	DBpedia (Lehmann et al., 2014)	Topic/category classification	en [+MT]	Link	Probability	5,000	5,000
79	Yahoo (Zhang et al., 2015)	Topic/category classification	en	Link	Probability	14,575	14,575
80	AG news (Zhang et al., 2015)	Topic/category classification	en	Link	Probability	30,000	89,800
81	TREC (Li and Roth, 2002)	Topic/category classification	en [+MT]	Link	Probability	2,626	2,626

Table 1: The list of the 81 tasks used as *seen* tasks. "[+MT]" in the Languages column means that the dataset is used for the data augmentation described in Section 5.4.

Name	Type	Notes
AfriSenti Zero (Muhammad et al., 2023b)	Sentiment classification (positive, negative, neutral)	Two **held-out African languages** are targeted, while 12 other African languages are used in a seen sentiment classification task (AfriSenti).
GoEmotions (Demszky et al., 2020)	Multi-label emotion classification (28 classes)	This is a **multi-label fine-grained** task, while a 4-way (single-class) classification task (TweetEval-emotion) is included in the seen tasks.
CLINC150 (Larson et al., 2019)	Dialog intent classification (150 classes)	Similar tasks (ATIS/MTOP/Massive-intent) are included in the seen tasks, and this is another task with multi-domain **fine-grained** classes.
Orcas-I (Alexander et al., 2022)	Search query intent classification (5 classes)	This is different from those in the seen tasks; the search queries are not always comprehensive and thus rely on **retrieval augmentation**.
MIT-R (Dataset link)	Dialog slot labeling (8 slot types)	Similar tasks (ATIS/MTOP/SNIPS-slot, E2ENLG-reversed) are used in the seen tasks, and this is expected to be the **easiest** unseen task.
SSENT (Barnes et al., 2022)	Polar expression extraction (positive, negative)	The task format is similar to that of MIT-R, but focuses on **polar (positive and negative) expressions** of hotel reviews in **Spanish**.
XML-MT (Hashimoto et al., 2019)	Machine translation (en→ja, en→fi)	Machine translation tasks (WMT14/16) are included in the seen tasks, but this focuses on **two other language pairs** and **XML-tagged texts**.

Table 2: Tasks for the *unseen* task evaluation. “Notes” explain what aspects we focus on in the evaluation.

	AfriSenti (46.30)	DDI13 (18.18)	ATIS-intent (35.49)	MTOP-intent (48.46)
R_0	49.24 51.39 52.78 54.98	19.92 23.59 25.52 28.8	70.31 87.16 91.74 95.48	84.22 88.55 90.55 92.55
R_{STD}	+1.24 +2.75 +4.84 +7.29	+8.42 +11.13 +14.90 +14.87	+4.11 +2.79 +3.87 +2.27	+8.10 +5.67 +4.53 +3.11
R_{DESC}	+1.28 +3.12 +5.12 +8.03	+5.56 +10.39 +15.67 +15.11	+5.60 +2.41 +3.88 +2.65	+7.86 +5.46 +4.48 +2.92
R_{NO}	+1.43 +3.07 +4.97 +7.74	+7.46 +11.06 +12.89 +16.14	+6.61 +3.24 +3.87 +2.87	+8.32 +6.07 +4.97 +3.50
	Countfact (26.48)	Offensive (53.44)	BC5CDR (2.70)	PHP (3.00)
R_0	41.44 48.80 55.28 63.37	61.15 65.14 63.98 63.76	37.44 55.14 60.45 63.28	13.61 14.44 13.82 11.00
R_{STD}	+5.34 +9.47 +9.79 +6.90	+1.26 +2.21 +3.46 +1.99	+7.87 +4.21 +1.49 -1.08	+1.68 +1.39 +1.54 +0.55
R_{DESC}	+4.92 +9.48 +9.81 +4.79	+0.72 +1.80 +4.00 +1.32	+7.76 +4.01 +2.01 -0.83	+1.75 +1.54 +1.51 +1.38
R_{NO}	+4.01 +8.92 +10.27 +10.44	+0.73 +2.89 +4.44 +3.66	+7.26 +4.41 +2.55 +0.49	+1.42 +1.20 +1.09 +0.28

Table 3: Seen task results. The four numbers in the R_0 rows correspond to the scores by 1,3,5,10-shot ICL with the baseline retriever R_0. The rest of the rows show the absolute improvements by using the fine-tuned retrievers (R_{STD}, R_{DESC}, and R_{NO}) based on the three types of the demonstration representations. The score next to the task name reports the LLM’s zero-shot performance to know its knowledge about the task without any demonstrations.

Unseen tasks To evaluate the generalization ability of the demonstration retrievers from diverse angles, we use the tasks summarized in Table 2. The “Notes” in the table explain what kinds of unseen aspects we would like to test with the retrievers. For each task, we use the whole training set to construct the candidate set $\mathcal{C}_t$; the AfriSenti Zero task does not have any training examples, and we use the AfriSenti task for the candidate set (i.e., a cross-lingual ICL setting). We describe more details in Appendix B.

5 Results

We evaluate the retrievers based on k-shot ICL with $k \in \{1, 3, 5, 10\}$. Unless otherwise stated, we simply use the top-k retrieved demonstrations to construct the prompts for the LLM. All the evaluation scores are in the range of [0, 100], and Appendix C describes the metric for each task.

5.1 Evaluation on Seen Tasks

We first confirm the effectiveness of the fine-tuned retrievers on the seen tasks as in the previous studies (Li et al., 2023; Wang et al., 2023). We use a sentiment classification task in 12 African languages (AfriSenti), a relation extraction task in the biomedical domain (DDI13), two (single/multi-label) dialog intent classification tasks (ATIS/MTOP-intent), two binary (counterfactual/offensive) detection tasks (Countfact, Offensive), a named entity recognition task in the biomedical domain (BC5CDR), and a code summarization task (PHP).

Table 3 shows the results. It is consistent with the previous work that the fine-tuned retrievers perform significantly better than the baseline retriever. We hypothesized that the three types of the fine-tuned retrievers perform similarly on the seen tasks, and it is true in most of the cases. Overall, we did not observe the potential advantage of R_{DESC} in the results.

However, we sometimes see nontrivial gains by R_{NO}, for example, in the COUNTFACT result. This is presumably because using the output labels is severely affected by overfitting. It is also interesting to see that R_{NO} works well even on tasks with more complex output space like BC5CDR.

	AfriSenti Zero (39.43)				GoEmotions (27.92)				CLINC150 (70.58)				Orcas-I (42.00)			
R_0	40.50	41.48	41.92	42.97	27.19	29.05	30.66	32.36	91.36	93.53	94.24	95.87	46.30	48.70	51.00	54.30
R_{STD}	-0.51	-0.54	-0.03	-1.37	+0.52	+0.34	-0.48	-1.31	-1.34	-1.60	-1.62	-1.96	-0.90	-1.20	-3.50	-6.00
R_{DESC}	-1.00	-0.27	-0.32	-1.81	+0.53	+0.53	-0.04	+0.74	-0.69	-1.31	-1.08	-2.11	+1.40	+0.90	+0.50	-0.30
R_{NO}	-0.41	-1.32	-1.25	-0.44	+0.34	+0.61	-0.05	-0.09	+2.35	+2.14	+1.78	+0.40	+0.70	+0.50	-1.00	-0.80
	MIT-R (1.09)				SSENT (7.38)				XML-MT enja (37.71)				XML-MT enfi (23.56)			
R_0	40.14	49.34	54.54	60.46	24.66	27.52	30.33	27.32	52.10	55.54	56.19	56.08	36.43	39.00	39.86	40.00
R_{STD}	+6.44	+6.10	+4.68	+1.83	+3.21	+3.02	-0.21	-2.10	+0.36	+0.93	+0.31	+0.55	-0.23	+0.26	+0.08	-0.43
R_{DESC}	+5.63	+5.18	+3.98	+1.78	+3.95	+4.03	+1.38	+1.38	+0.52	+0.57	+1.08	+0.28	-0.06	-0.03	+0.56	-0.22
R_{NO}	+5.19	+5.88	+3.99	+2.26	+0.66	+1.35	-1.16	+0.44	+0.85	+0.06	+0.92	+0.02	+0.84	+0.72	+0.60	-2.32

Table 4: Unseen task results with Flan-PaLM 2. The structure of this table is analogous to that of Table 3.

	AfriSenti Zero (44.48)				GoEmotions (28.26)				CLINC150 (92.62)				Orcas-I (49.10)			
R_0	55.83	55.81	54.42	54.03	31.61	33.50	35.57	37.97	96.22	97.22	97.51	97.73	59.00	60.90	61.90	65.4
R_{NO}	-0.75	-2.61	-3.00	-3.37	-0.33	+0.34	-0.12	+0.10	+0.54	+0.85	+0.56	+0.56	-0.90	+0.50	+0.80	-0.30
	MIT-R (8.60)				SSENT (22.40)				XML-MT enja (27.94)				XML-MT enfi (24.16)			
R_0	64.93	68.45	72.85	75.25	44.96	50.34	52.22	53.91	58.45	62.51	63.10	63.94	42.90	45.47	45.90	47.34
R_{NO}	+3.48	+2.98	+2.23	+1.65	+0.93	+1.49	+1.05	+3.68	+1.37	+0.58	+1.01	+0.97	+0.57	-0.07	-0.40	+0.23

Table 5: Unseen task results with Gemini 1.5 Pro. The structure of this table is analogous to that of Table 3.

5.2 Evaluation on Unseen Tasks

We then evaluate the retrievers on the unseen tasks. Table 4 shows the results, and below we summarize the key points.

- All the fine-tuned retrievers perform worse than R_0 on AfriSenti Zero. We hypothesize that "catastrophic forgetting" is caused by the fact that the two zero-shot languages (Oromo and Tigrinya) are never observed in the retriever fine-tuning process.

- It is surprising to see that R_{STD} performs significantly worse than R_0 on fine-grained classification tasks whose labels are not easy to interpret. Especially, it fails on CLINC150, even when we have successful results on the intent classification tasks in Table 3. In contrast, R_{NO} provides more robust results.

- It matches our expectation that all the fine-tuned retrievers perform well on MIT-R as explained in Table 2.

- Overall, the effects of using R_{DESC} are not conclusive. We see the potential benefit on Orcas-I (whose label descriptions are helpful even for humans) and SSENT, while it does not help on CLINC150. It is possible that the provided label descriptions are not good enough, but this nontrivial process itself indicates that R_{DESC} would not be the best way.

	Natural Instructions (25.28)			
R_0	26.59	27.08	26.95	27.04
R_{NO}	+0.26	+0.49	+0.81	+0.37

Table 6: Natural Instructions results.

- Based on the SSENT results, using the task output would be effective for some tasks. An interesting future work is to consider how to strike a balance between R_{STD} and R_{NO}.

More unseen tasks We further perform evaluation on 20 unseen text generation tasks from Super Natural Instructions (Wang et al., 2022) to test the robustness of the demonstration retriever. The tasks include machine translation, text summarization, question answering, paraphrase generation, etc, and the datasets are *not* used in fine-tuning our retrievers. Table 6 shows the average scores across all the tasks, and we can see some gains by using R_{NO}. The size of the training set for a task is limited to around 6,000 examples in Super Natural Instructions, and thus this might not be the best setup for ICL; still, our retriever shows the robust results.

Transfer ability Following the previous work (Li et al., 2023; Wang et al., 2023), we test how R_{NO} works with another LLM, Gemini 1.5 Pro (Reid et al., 2024). It should be noted that we have never touched the new LLM until we perform the final test evaluation. Table 5 shows the results, and we can see consistent trends. Gemini

	ATIS-intent			COUNTFACT		
R_0	87.16	91.74	95.48	48.80	55.28	63.37
R_{NO}	+3.24	+3.87	+2.87	+8.92	+10.27	+10.44
+cov.	+4.85	+4.34	+2.14	+11.13	+12.65	+11.57
	AfriSenti Zero			SSENT		
R_0	41.48	41.92	42.97	27.52	30.33	27.32
R_{NO}	-1.32	-1.25	-0.44	+1.35	-1.16	+0.44
+cov.	-1.12	-0.05	-0.20	+3.07	+1.19	+1.54

Table 7: Coverage-based selection results. $k = 1$ is not affected by this method, and we only show the scores with $k = 3, 5, 10$.

	AfriSenti Zero (39.43)			
R_0	40.50	41.48	41.92	42.97
R_{NO}	-0.41	-1.32	-1.25	-0.44
R_{NO}+MT	+0.15	+0.39	+0.49	+1.29
	ATIS-intent hi,tr (29.67)			
R_0	62.18	79.09	84.39	89.26
R_{NO}	+3.11	+2.44	+2.57	+1.27
R_{NO}+MT	+5.72	+3.82	+3.02	+2.47

Table 8: Cross-lingual ICL results with Flan-PaLM 2.

1.5 pro achieves much better baseline scores than those of Flan-PaLM 2 (S), but still R_{NO} helps. It is encouraging that our fine-tuned retriever works well even for this much stronger LLM.

5.3 Compatibility with Existing Methods

We discuss the potential of using R_{NO} as a basic building block in diverse scenarios for future work. In other words, we do not intend to claim that our retriever should be always used alone, and instead we believe that our retriever can be used along with existing methods.

For example, we consider the coverage-based demonstration selection method in Gupta et al. (2023), and we apply their "cosine" method to the top-retrieved candidates by R_{NO}. Table 7 shows the results, and the method works well with our retriever.

Other possible future directions are using our retriever for sequential selection models (Scarlatos and Lan, 2024; Liu et al., 2024), continual learning with more tasks and languages, and explicit adaptation to other LLMs.

5.4 Improved Language Coverage by Machine Translation

We have observed that the fine-tuning process degrades the generalization ability of the retriever on unseen languages. Our seen task set covers various languages as shown in Table 1, but still, English is dominant. How can we make our retriever more robust from this viewpoint? One solution is to add more and more tasks in many languages, but it is not a trivial effort.

To this end, we consider using machine translation for data augmentation as in the common practice (Balahur and Turchi, 2014; Lee et al., 2018). We describe our process below:

1. Select 8 tasks (~10% of the whole) from the seen task list in Table 1: ANLI r1, Tweet Sentiment Extraction, SemRel, iDebate, BoolQ, PIQA-qgen, DBpedia, and TREC; all the selected tasks are originally in English.

2. Use Google Translate[3] to translate the examples in the query set $\mathcal{Q}_t$ and the candidate set $\mathcal{C}_t$ for the selected task; for each example in $\mathcal{Q}_t$, we randomly sample a target languages (b ($> a$) for $\mathcal{C}_t$), and consequently we have multi-lingual query and candidate sets.[4]

3. Add the multilingual version of the 8 tasks to the seen task list; note that the new tasks are separated from the original English ones, and the utility estimation for the retriever fine-tuning is done solely within the synthetic data.

By this, the demonstration retriever is **exposed to more than 130 languages** during the fine-tuning.

We revisit the evaluation on AfriSenti Zero; this is considered to be a cross-lingual ICL evaluation, in that the languages in the query set and the candidate set are different. We add another cross-lingual ICL evaluation with the Hindi and Turkish variants of the ATIS-intent task, where we use the original English ATIS-intent for the candidate set.

Table 8 shows the results, and we can see that R_{NO} with the data augmentation (R_{NO}+MT) performs the best. Hindi and Turkish are included in seen tasks (e.g., Massive-intent), but still the data augmentation helps. Note that using the synthetic data does not degrade the retriever's performance on other tasks.

In our checkpoint release, we will also provide a model that is based on even more languages for the data augmentation. The model covers more than 230 languages.[5]

[3] As of early June 2024, 132 non-English languages are supported at https://cloud.google.com/translate/docs/languages.

[4] In Appendix A.3, we describe details of this process

[5] https://support.google.com/translate/answer/15139004

6 Conclusion

We have presented our multi-lingual and multi-task demonstration retriever for in-context learning with LLMs. We showed the counterintuitive finding to improve the generalization ability of the demonstration representations, and improved multi/cross-lingual performance of the retriever by the translation-based data augmentation. We believe that our released models will be useful for future work.

Limitations

Task coverage We did our best to collect as diverse tasks as possible. However, we would be able to find new tasks where our retriever does not work well. Our future effort will be to improve the task coverage or seek the use of instruction-tuned LLMs themselves (Gemini, GPT, Llama, etc.) as a retriever to leverage their generalization ability.

Short task instruction We assume the use of the short task instruction for our retriever. To handle new tasks that are quite different from those in our task set, we may need to come up with new short task instructions. In such a case, we suggest that the users refer to the complete list (in Appendix A.2) of all the instructions we used, to design the new instructions.

Translation error in data augmentation No machine translation systems (including Google Translate we used in our experiments) are perfect, and thus we expect that translation errors exist in our synthetic multi-lingual tasks. To avoid the potential negative effects by the translation errors, we did not use the synthetic data for validation and evaluation to test our retriever's quality.

References

Ibrahim Abu Farha, Silviu Vlad Oprea, Steven Wilson, and Walid Magdy. 2022. SemEval-2022 Task 6: iSarcasmEval, Intended Sarcasm Detection in English and Arabic. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 802–814.

Daria Alexander, Wojciech Kusa, and Arjen P. de Vries. 2022. ORCAS-I: Queries Annotated with Intent using Weak Supervision. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 3057–3066, New York, NY, USA. Association for Computing Machinery.

Mikel Artetxe and Holger Schwenk. 2019. Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Akari Asai, Timo Schick, Patrick Lewis, Xilun Chen, Gautier Izacard, Sebastian Riedel, Hannaneh Hajishirzi, and Wen-tau Yih. 2023. Task-aware Retrieval with Instructions. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3650–3675.

Alexandra Balahur and Marco Turchi. 2014. Comparative experiments using supervised learning and machine translation for multilingual sentiment analysis. *Computer Speech Language*, 28:56–75.

Marta Bañón, Pinzhen Chen, Barry Haddow, Kenneth Heafield, Hieu Hoang, Miquel Esplà-Gomis, Mikel L. Forcada, Amir Kamran, Faheem Kirefu, Philipp Koehn, Sergio Ortiz Rojas, Leopoldo Pla Sempere, Gema Ramírez-Sánchez, Elsa Sarrías, Marek Strelec, Brian Thompson, William Waites, Dion Wiggins, and Jaume Zaragoza. 2020. ParaCrawl: Web-Scale Acquisition of Parallel Corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4555–4567, Online. Association for Computational Linguistics.

Francesco Barbieri, Jose Camacho-Collados, Francesco Ronzano, Luis Espinosa-Anke, Miguel Ballesteros, Valerio Basile, Viviana Patti, and Horacio Saggion. 2018. Semeval 2018 task 2: Multilingual emoji prediction. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 24–33.

Jeremy Barnes, Laura Ana Maria Oberländer, Enrica Troiano, Andrey Kutuzov, Jan Buchmann, Rodrigo Agerri, Lilja Øvrelid, and Erik Velldal. 2022. SemEval-2022 Task 10: Structured Sentiment Analysis. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2019. PIQA: Reasoning about Physical Commonsense in Natural Language.

Ondřej Bojar, Christian Buck, Rajen Chatterjee, Christian Federmann, Liane Guillou, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Aurélie Névéol, Mariana Neves, Pavel Pecina, Martin Popel, Philipp Koehn, Christof Monz, Matteo Negri, Matt Post, Lucia Specia, Karin Verspoor, Jörg Tiedemann, and Marco Turchi, editors. 2016. *Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers*. Association for Computational Linguistics, Berlin, Germany.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, and Lucia Specia, editors. 2014. *Proceedings* *of the Ninth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Baltimore, Maryland, USA.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the Surprising Difficulty of Natural Yes/No Questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S. Weld. 2020. SPECTER: Document-level Representation Learning using Citation-informed Transformers.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. GoEmotions: A dataset of fine-grained emotions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4040–4054, Online. Association for Computational Linguistics.

William B. Dolan and Chris Brockett. 2005. Automatically Constructing a Corpus of Sentential Paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. DROP: A Reading Comprehension Benchmark Requiring Discrete Reasoning Over Paragraphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2368–2378, Minneapolis, Minnesota. Association for Computational Linguistics.

Ondřej Dušek, David M. Howcroft, and Verena Rieser. 2019. Semantic Noise Matters for Neural Natural Language Generation. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 421–426, Tokyo, Japan. Association for Computational Linguistics.

Fast.AI. Yelp sentiment classification dataset.

Besnik Fetahu, Sudipta Kar, Zhiyu Chen, Oleg Rokhlenko, and Shervin Malmasi. 2023. SemEval-2023 Task 2: Fine-grained Multilingual Named Entity Recognition (MultiCoNER 2). In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2247–2265, Toronto, Canada. Association for Computational Linguistics.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2022. MASSIVE: A 1M-Example Multilingual Natural Language Understanding Dataset with 51 Typologically-Diverse Languages.

Lingyu Gao, Aditi Chaudhary, Krishna Srinivasan, Kazuma Hashimoto, Karthik Raman, and Michael Bendersky. 2023a. Ambiguity-Aware In-Context Learning with Large Language Models. *arXiv preprint cs.CL 2309.07900*.

Lingyu Gao, Debanjan Ghosh, and Kevin Gimpel. 2023b. The Benefits of Label-Description Training for Zero-Shot Text Classification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13823–13844, Singapore. Association for Computational Linguistics.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. SAMSum Corpus: A Human-annotated Dialogue Dataset for Abstractive Summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79, Hong Kong, China. Association for Computational Linguistics.

Google, Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre

Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyeontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou, Denny Zhou, Slav Petrov, and Yonghui Wu. 2023. PaLM 2 Technical Report.

Shivanshu Gupta, Matt Gardner, and Sameer Singh. 2023. Coverage-based Example Selection for In-Context Learning. *arXiv preprint cs.CL 2305.14907*.

Kazuma Hashimoto, Raffaella Buschiazzo, James Bradbury, Teresa Marshall, Richard Socher, and Caiming Xiong. 2019. A High-Quality Multilingual Dataset for Structured Documentation Translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 116–127.

Kazuma Hashimoto, Karthik Raman, and Michael Bendersky. 2024. Take One Step at a Time to Know Incremental Utility of Demonstration: An Analysis on Reranking for Few-Shot In-Context Learning. In *NAACL 2024*.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations between Pairs of Nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden. Association for Computational Linguistics.

María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. 2013. The DDI corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of Biomedical Informatics*, 46(5):914–920.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos QA: Machine Reading Comprehension with Contextual Commonsense Reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2391–2401, Hong Kong, China. Association for Computational Linguistics.

Ming-Siang Huang, Po-Ting Lai, Pei-Yen Lin, Yu-Ting You, Richard Tzong-Han Tsai, and Wen-Lian Hsu. 2020. Biomedical named entity recognition and linking datasets: survey and our recent development. *Briefings in Bioinformatics*, 21(6):2219–2238.

Rezarta Islamaj Doğan, Sun Kim, Andrew Chatr-aryamontri, Chih-Hsuan Wei, Donald C Comeau, Rui Antunes, Sérgio Matos, Qingyu Chen, Aparna Elangovan, Nagesh C Panyam, Karin Verspoor, Hongfang Liu, Yanshan Wang, Zhuang Liu, Berna Altınel, Zehra Melce Hüsünbeyi, Arzucan Özgür, Aris Fergadis, Chen-Kai Wang, Hong-Jie Dai, Tung Tran, Ramakanth Kavuluru, Ling Luo, Albert Steppi, Jinfeng Zhang, Jinchan Qu, and Zhiyong Lu. 2019. Overview of the BioCreative VI Precision Medicine Track: mining protein interactions and mutations for precision medicine. *Database*, 2019:bay147.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Kaggle. Tweet Sentiment Extraction.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781.

Shivani Kumar, Md Shad Akhtar, Erik Cambria, and Tanmoy Chakraborty. 2024. SemEval 2024 – Task

10: Emotion Discovery and Reasoning its Flip in Conversation (EDiReF).

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. 2019. An Evaluation Dataset for Intent Classification and Out-of-Scope Prediction. 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 1311–1316.

Kyungjae Lee, Kyoungho Yoon, Sunghyun Park, and Seung-won Hwang. 2018. Semi-supervised Training Data Generation for Multilingual Question Answering. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, and Christian Bizer. 2014. DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web Journal*, 6.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012a. The Winograd schema challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'12, page 552–561. AAAI Press.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012b. The Winograd schema challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'12, page 552–561. AAAI Press.

Haoran Li, Abhinav Arora, Shuohui Chen, Anchit Gupta, Sonal Gupta, and Yashar Mehdad. 2021. 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.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. 2016. BioCreative V CDR task corpus: a resource for chemical disease relation extraction. *Database: The Journal of Biological Databases and Curation*, 2016.

Xiaonan Li, Kai Lv, Hang Yan, Tianyang Lin, Wei Zhu, Yuan Ni, Guotong Xie, Xiaoling Wang, and Xipeng Qiu. 2023. Unified Demonstration Retriever for In-Context Learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4644–4668.

Xin Li and Dan Roth. 2002. Learning Question Classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. 2020. CommonGen: A Constrained Text Generation Challenge for Generative Commonsense Reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1823–1840, Online. Association for Computational Linguistics.

Haoyu Liu, Jianfeng Liu, Shaohan Huang, Yuefeng Zhan, Hao Sun, Weiwei Deng, Furu Wei, and Qi Zhang. 2024. Se^2: Sequential Example Selection for In-Context Learning.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What Makes Good In-Context Examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114.

Shuai Lu, Daya Guo, Shuo Ren, Junjie Huang, Alexey Svyatkovskiy, Ambrosio Blanco, Colin Clement, Dawn Drain, Daxin Jiang, Duyu Tang, Ge Li, Lidong Zhou, Linjun Shou, Long Zhou, Michele Tufano, MING GONG, Ming Zhou, Nan Duan, Neel Sundaresan, Shao Kun Deng, Shengyu Fu, and Shujie LIU. 2021. CodeXGLUE: A Machine Learning Benchmark Dataset for Code Understanding and Generation. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Man Luo, Xin Xu, Zhuyun Dai, Panupong Pasupat, Mehran Kazemi, Chitta Baral, Vaiva Imbrasaite, and Vincent Y Zhao. 2023. Dr.ICL: Demonstration-Retrieved In-context Learning. *arXiv preprint cs.CL 2305.14128*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a Suit of Armor Conduct Electricity? A New Dataset for Open Book Question Answering.

Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. Semeval-2018 task 1: Affect in tweets. In *Proceedings of the 12th international workshop on semantic evaluation*, pages 1–17.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. MTEB: Massive Text Embedding Benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.

Shamsuddeen Muhammad, Idris Abdulmumin, Abinew Ayele, Nedjma Ousidhoum, David Adelani, Seid Yimam, Ibrahim Ahmad, Meriem Beloucif, Saif Mohammad, Sebastian Ruder, Oumaima Hourrane, Alipio Jorge, Pavel Brazdil, Felermino Ali, Davis David, Salomey Osei, Bello Shehu-Bello, Falalu Lawan, Tajuddeen Gwadabe, Samuel Rutunda, Tadesse Belay, Wendimu Messelle, Hailu Balcha, Sisay Chala, Hagos Gebremichael, Bernard Opoku, and Stephen Arthur. 2023a. AfriSenti: A Twitter Sentiment Analysis Benchmark for African Languages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13968–13981, Singapore. Association for Computational Linguistics.

Shamsuddeen Hassan Muhammad, Idris Abdulmumin, Seid Muhie Yimam, David Ifeoluwa Adelani, Ibrahim Said Ahmad, Nedjma Ousidhoum, Abinew Ali Ayele, Saif Mohammad, Meriem Beloucif, and Sebastian Ruder. 2023b. SemEval-2023 task 12: Sentiment analysis for African languages (AfriSenti-SemEval). In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2319–2337, Toronto, Canada. Association for Computational Linguistics.

Courtney Napoles, Matthew Gormley, and Benjamin Van Durme. 2012. Annotated Gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, pages 95–100, Montréal, Canada. Association for Computational Linguistics.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2022. Sentence-T5: Scalable Sentence Encoders from Pre-trained Text-to-Text Models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A New Benchmark for Natural Language Understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Tomoko Ohta, Sampo Pyysalo, Rafal Rak, Andrew Rowley, Hong-Woo Chun, Sung-Jae Jung, Sung-Pil Choi, Sophia Ananiadou, and Jun'ichi Tsujii. 2013. Overview of the Pathway Curation (PC) task of BioNLP Shared Task 2013. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 67–75, Sofia, Bulgaria. Association for Computational Linguistics.

James O'Neill, Polina Rozenshtein, Ryuichi Kiryo, Motoko Kubota, and Danushka Bollegala. 2021. I Wish I Would Have Loved This One, But I Didn't – A Multilingual Dataset for Counterfactual Detection in Product Reviews.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the Word-in-Context Dataset for Evaluating Context-Sensitive Meaning Representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1267–1273, Minneapolis, Minnesota. Association for Computational Linguistics.

Patti Price. 1990. Evaluation of spoken language systems: The ATIS domain. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27, 1990.*

Altaf Rahman and Vincent Ng. 2012. Resolving Complex Cases of Definite Pronouns: The Winograd Schema Challenge. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 777–789, Jeju Island, Korea. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know What You Don't Know: Unanswerable Questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Karthik Raman, Iftekhar Naim, Jiecao Chen, Kazuma Hashimoto, Kiran Yalasangi, and Krishna Srinivasan. 2022. Transforming Sequence Tagging Into A Seq2Seq Task. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11856–11874.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8689–8696.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Paul Röttger, Haitham Seelawi, Debora Nozza, Zeerak Talat, and Bertie Vidgen. 2022. Multilingual HateCheck: Functional Tests for Multilingual Hate Speech Detection Models. In *Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH)*, pages 154–169, Seattle, Washington (Hybrid). Association for Computational Linguistics.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. WinoGrande: An Adversarial Winograd Schema Challenge at Scale.

David Saxton, Edward Grefenstette, Felix Hill, and Pushmeet Kohli. 2019. Analysing Mathematical Reasoning Abilities of Neural Models.

Alexander Scarlatos and Andrew Lan. 2024. RetICL: Sequential Retrieval of In-Context Examples with Reinforcement Learning.

Patrice Simard, Yann LeCun, and John Denker. 1992. Efficient Pattern Recognition Using a New Transformation Distance. In *Advances in Neural Information Processing Systems*, volume 5. Morgan-Kaufmann.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2023. One Embedder, Any Task: Instruction-Finetuned Text Embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121, Toronto, Canada. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Sabine Buchholz. 2000. Introduction to the CoNLL-2000 Shared Task Chunking. In *Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop*.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018. Semeval-2018 task 3: Irony detection in english tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50.

Liang Wang, Nan Yang, and Furu Wei. 2023. Learning to Retrieve In-Context Examples for Large Language Models. *arXiv preprint cs.CL 2307.07164*.

Lu Wang and Wang Ling. 2016. Neural Network-Based Abstract Generation for Opinions and Arguments. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 47–57, San Diego, California. Association for Computational Linguistics.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022. Super-NaturalInstructions:Generalization via Declarative Instructions on 1600+ Tasks. In *EMNLP*.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural Network Acceptability Judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned Language Models are Zero-Shot Learners. In *International Conference on Learning Representations*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *arXiv preprint cs.CL 1609.08144*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer.

In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498.

Yinfei Yang, Gustavo Hernandez Abrego, Steve Yuan, Mandy Guo, Qinlan Shen, Daniel Cer, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2019. Improving Multilingual Sentence Embedding using Bi-directional Dual Encoder with Additive Margin Softmax. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5370–5378. International Joint Conferences on Artificial Intelligence Organization.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86.

Jianguo Zhang, Kazuma Hashimoto, Wenhao Liu, Chien-Sheng Wu, Yao Wan, Philip Yu, Richard Socher, and Caiming Xiong. 2020. Discriminative Nearest Neighbor Few-Shot Intent Detection by Transferring Natural Language Inference. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5064–5082.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level Convolutional Networks for Text Classification. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase Adversaries from Word Scrambling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1298–1308, Minneapolis, Minnesota. Association for Computational Linguistics.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, pages 12697–12706.

Victor Zhong, Caiming Xiong, and Richard Socher. 2017. Seq2sql: Generating structured queries from natural language using reinforcement learning.

Appendix

A Seen Tasks

A.1 Task List

Table 1 summarizes the 81 tasks we used to fine-tune the demonstration retriever. We started with datasets from Flan-v1 (Wei et al., 2021), MTEB (Muennighoff et al., 2023), and those in Li et al. (2023). We then further collected more datasets whose task formats are not well covered by our initial collection. In the following, we explain how to read the table.

Name We give a task name for each of them, while the names would not exactly match with those used in previous work.

Type We briefly describe the goal of every task by commonly-used terminologies.

Languages We collect datasets that use not only English, but also other languages to make our demonstration retriever work in as many languages as possible. Note that our retriever is based on mT5 (Xue et al., 2021) for the same purpose.

Source We provide the URL where we get the dataset for each task. The “Link” works only on PDF readers.

Scoring In the “Scoring candidates” paragraph in Section 2, we use the LLM to score a demonstration’s usefulness for an input. We follow Hashimoto et al. (2024) to use different scoring functions, depending on the task types. We use the following three functions in this work:

- Probability– for tasks like single-class classification and multiple-choice selection, we use the probability value for generating the ground-truth output by the LLM: $p(y|x, t, d)$.
- F1– for tasks like text segmentation and multi-label classification, we use an F1 score by comparing the LLM’s prediction (i.e., 1-shot prediction with d) against the ground-truth output, so that we can reward partially correct predictions.
- GLEU– for other text generation tasks, we use the GLEU score (Wu et al., 2016).

A.2 Task Information

We briefly describe the information about each of the seen tasks, to mainly present our full (F) and short (S) task instructions used in our experiments. For all the data in any languages, we use the English-based instructions.

No. 01–06 For the standard machine translation tasks, we use the following task instructions:

F: The goal of this task is to translate from [language 1] to [language 2].

S: Translation: [language 1] to [language 2].

No. 07–09 For the ANLI tasks, we use the following task instructions:

F: The goal of this task is to judge if the hypothesis can be concluded, given the context. The output is "Yes", "No", or "It's impossible to say".

S: Natural language inference: context to hypothesis.

No. 10 For the QNLI task, we use the following task instructions:

F: The goal of this task is to identify if the sentence correctly answers the question. The output is yes or no.

S: Natural language inference: sentence to question.

No. 11 For the MNLI task, we use the following task instructions:

F: The goal of this task is to identify if the premise entails the hypothesis. The output is entailment, contradiction, or neutral.

S: Natural language inference: premise to hypothesis.

No. 12 For the WNLI task, we use the following task instructions:

F: The goal of this task is to identify if text2 is true or false, given text1.

S: Natural language inference: text1 to text2.

No. 13–14 For the paraphrase identification tasks, we use the following task instructions:

F: The goal of this task is to identify if sentence1 and sentence2 have the same meaning. The output is yes or no.

S: Paraphrase identification: sentence1 and sentence2.

No. 15 For the Tatoeba task, we use the following task instructions:

F: The goal of this task is to identify if sentence1 is a translation of sentence2. The output is Yes or No.

S: Translation identification: sentence1 and sentence2.

We note that we used the test set of this task, and therefore our retrievers cannot be used for Tatoeba evaluation in any ways.

No. 16–18 For the binary sentiment classification tasks, we use the following task instructions:

F: The goal of this task is to identify the sentiment given the text. The output is positive or negative.

S: Sentiment classification.

No. 19–20 For the three-way sentiment classification tasks, we use the following task instructions:

F: The goal of this task is to identify the sentiment label of the tweet. The output is positive, negative, or neutral.

S: Sentiment classification.

No. 21 For the TweetEval-emoji task, we use the following task instructions:

F: The goal of this task is to identify the emoji relevant to the tweet. The 20 possible emojis are ...

S: Emoji generation.

No. 22 For the TweetEval-emotion task, we use the following task instructions:

F: The goal of this task is to identify the emotion of the tweet. The 4 possible emotions are anger, joy, optimism, or sadness.

S: Emotion classification.

No. 23 For the DialogEmotion task, we use the following task instructions:

F: The goal of this task is to list all the speaker names who experience the specific emotion in the conversation. The output will be a #-separated list like "speaker_1#speaker_4#speaker_5".

S: Emotion detection: speakers.

No. 24 For the Massive-intent task, we use the following task instructions:

F: The goal of this task is to identify the intent label of the user's input. The list of the 60 labels is: alarm_query, alarm_remove, alarm_set, audio_volume_down, audio_volume_mute, ...

S: User input intent classification.

No. 25 For the MTOP-domain task, we use the following task instructions:

F: The goal of this task is to identify the domain of the user's input. There are 11 possible domains: alarm, calling, event, messaging, music, news, people, recipes, reminder, timer, weather.

S: User input domain classification.

No. 26 For the MTOP-intent task, we use the following task instructions:

F: The goal of this task is to identify the intent of the user's input. There are 113 possible intents: ADD_TIME_TIMER, ADD_TO_PLAYLIST_MUSIC, ...

S: User input domain classification.

No. 27 For the ATIS-intent task, we use the following task instructions:

F: The goal of this task is to identify user's intents from abbreviation, aircraft, airfare, ... If multiple intents are identified, the output will be a #-separated string: intent_1#intent_2#intent_3.

S: Multi-label intent classification.

No. 28 For the E2ENLG-reversed task, we use the following task instructions:

F: The goal of this task is to extract attributes given a text about restaurant. The list of the 8 possible attributes are area, customerRating, eatType, familyFriendly, food, name, near, or priceRange. The output is a Python dictionary like {"attribute_1": "value_1", "attribute_2": "value_2", "attribute_3": "value_3"}

S: Attribute extraction.

No. 29 For the WikiSQL task, we use the following task instructions:

F: The goal of this task is to convert the natural language question into an SQL query, based on the table.

S: Text/table to SQL generation.

No. 30 For the BC5CDR task, we use the following task instructions:

F: The goal of this task is to copy the given text by tagging entities with XML tags. There are 2 entity types: Chemical, Disease. Then the output is like "word1 <Chemical>word2 word3</Chemical> word4 <Disease>word5</Disease>".

S: Named entity extraction: biomedical.

No. 31 For the BioNLP13PC task, we use the following task instructions:

F: The goal of this task is to copy the given text by tagging entities with XML tags. There are 4 entity types: Cellular_component, Complex, Gene_or_gene_product, Simple_chemical. Then the output is like "word1 <Complex>word2 word3</Complex> word4 <Simple_chemical>word5</Simple_chemical>".

S: Named entity extraction: biomedical.

No. 32 For the JNLPBA task, we use the following task instructions:

F: The goal of this task is to copy the given text by tagging entities with XML tags. There are 5 entity types: DNA, RNA, cell_line, cell_type, protein. Then the output is like "word1 <DNA>word2 word3</DNA> word4 <protein>word5</protein>".

S: Named entity extraction: biomedical.

No. 33 For the MultiCoNER2 task, we use the following task instructions:

F: The goal of this task is to copy the given text by tagging entities with XML tags. There are 33 entity types: AerospaceManufacturer, AnatomicalStructure, ... Then the output is like "word1 <Artist>word2 word3</Artist> word4 <Drink>word5</Drink>".

S: Named entity extraction: Wikipedia.

No. 34 For the CoNLL2003 task, we use the following task instructions:

F: The goal of this task is to copy the given text by tagging entities with XML tags. There are 4 entity types: Location, Miscellaneous, Organization, Person. Then the output is like "word1 <Location>word2 word3</Location> word4 <Person>word5</Person>".

S: Named entity extraction: news.

No. 35 For the MTOP-slot task, we use the following task instructions:

F: The goal of this task is to copy the given text by tagging attributes with XML tags. There are 74 attribute types: AGE, ALARM_NAME, ... Then the output is like "word1 <AGE>word2 word3</AGE> word4 <CONTACT>word5</CONTACT>".

S: Attribute extraction.

No. 36 For the SNIPS-slot task, we use the following task instructions:

F: The goal of this task is to copy the given text by tagging attributes with XML tags. There are 39 attribute types: album, artist, best_rating, ... Then the output is like "word1 <city>word2 word3</city> word4 <country>word5</country>".

S: Attribute extraction.

No. 37 For the ATIS-slot task, we use the following task instructions:

F: The goal of this task is to copy the given text by tagging attributes with XML tags. There are 79 attribute types: aircraft_code, airline_code, ... Then the output is like "word1 <airport_code>word2 word3</airport_code> word4 word5".

S: Attribute extraction.

No. 38 For the SemRel task, we use the following task instructions:

F: The goal of this task is to identify relation between the two entities marked by <e1></e1> and <e2></e2>. The possible relations are "e1:Cause e2:Effect", "e1:Effect e2:Cause", ... If the relation type is not one of the above, the output will be "Other".

S: Relation classification: e1 and e2.

No. 39 For the DDI2013 task, we use the following task instructions:

F: The goal of this task is to identify the relation type of two drugs mentioned as @DRUG$ in the text. There are 4 relation types: advise, effect, int, mechanism. If there is no relation between the drugs, the answer is false.

S: Relation extraction: @DRUG$ and @DRUG$.

No. 40 For the ChemProt task, we use the following task instructions:

F: The goal of this task is to identify the relation of @CHEMICAL$ and @GENE$ (or just @CHEM-GENE$) in the text. The answer is true or false.

S: Relation extraction: @CHEMICAL$ and @GENE$ (or @CHEM-GENE$).

No. 41 For the WordSeg task, we use the following task instructions:

F: The goal of this task is to segment the words in the given characters. The output is like "word_1 word_2 word_3".

S: Word segmentation.

No. 42 For the FixPunct task, we use the following task instructions:

F: The goal of this task is to generate the input text with punctuation.

S: Text punctuation.

No. 43 For the CoLA task, we use the following task instructions:

F: The goal of this task is to identify if the input text is linguistically acceptable or not. The output is acceptable or unacceptable.

S: Linguistic acceptableness.

No. 44 For the CoNLL2000 task, we use the following task instructions:

F: The goal of this task is to copy the given text by tagging syntactic phrases with XML tags. There are 11 phrase types: ADJP, ADVP, CONJP, INTJ, LST, NP, PP, PRT, SBAR, UCP, VP. Then the output is like "word1 <VP>word2 word3</VP> word4 <NP>word5</NP>".

S: Syntactic phrase chunking.

No. 45 For the Pronoun task, we use the following task instructions:

F: The goal of this task is to identify what the pronoun corresponds to, given the sentence. The output is a phrase/entity in the sentence.

S: Coreference resolution: pronoun.

No. 46 For the WSC task, we use the following task instructions:

F: The goal of this task is to identify if text1 and text2 are the same in the given context. The output is yes or no.

S: Text sense equivalence: text1 and text2 in context.

No. 47 For the WinoGrande task, we use the following task instructions:

F: The goal of this task is to select one of the given options to complete the context.

S: Text completion.

No. 48 For the WiC task, we use the following task instructions:

F: The goal of this task is to identify if the specified word has the same meaning in sentence1 and sentence2. The output is yes or no.

S: Word sense equivalence: word in sentence1 and sentence2.

No. 49–52 For the code summarization tasks, we use the following task instructions:

F: The goal of this task is to write comment about the [language] code.

S: Code summarization: [language].

No. 53 For the Gigaword task, we use the following task instructions:

F: The goal of this task is to extract a text segment that summarizes the input text.

S: Text summarization.

No. 54 For the SAMSum task, we use the following task instructions:

F: The goal of this task is to summarize the dialogue.

S: Dialogue summarization.

No. 55 For the iDebate task, we use the following task instructions:

F: The goal of this task is to generate a claim about the debate topic and the arguments.

S: Claim generation.

No. 56 For the MultiHateCheck task, we use the following task instructions:

F: The goal of this task is to identify if the input text is hateful or non-hateful, and its activity type. The list of "hateful" types are derog_dehum, derog_impl, ... The list of "non-hateful" types are counter_quote, counter_ref, ... The output is "hateful:type" or "non-hateful:type".

S: Hate speech detection.

We note that we used the test set of this task, and therefore our retrievers cannot be used for MultiHateCheck evaluation in any ways.

No. 57 For the Toxic task, we use the following task instructions:

F: The goal of this task is to identify if the input text is "toxic" or "not toxic".

S: Toxic conversation detection.

No. 58 For the Countfact task, we use the following task instructions:

F: The goal of this task is to identify if the input text is counterfactual or not-counterfactual.

S: Counterfactual review detection.

No. 59 For the Irony task, we use the following task instructions:

F: The goal of this task is to identify if the input tweet is irony or not. The output is Irony or Non-irony.

S: Irony tweet detection.

No. 60 For the Offensive task, we use the following task instructions:

F: The goal of this task is to identify if the input tweet is offensive or not. The output is Offensive or Non-offensive.

S: Offensive tweet detection.

No. 61 For the Sarcasm task, we use the following task instructions:

F: The goal of this task is to identify if an input text is sarcastic or non-sarcastic.

S: Sarcastic text detection.

No. 62 For the SQuAD2 task, we use the following task instructions:

F: The goal of this task is to extract an answer phrase from the context to answer the question. If the question cannot be answered, then the output is "unanswerable".

S: Question answering.

No. 63 For the BoolQ task, we use the following task instructions:

F: The goal of this task is to answer the question, given the title and text.

S: Question answering.

No. 64 For the DROP task, we use the following task instructions:

F: The goal of this task is to answer the question, given the context.

S: Question answering.

No. 65 For the OpenbookQA task, we use the following task instructions:

F: The goal of this task is to answer the question based on the fact. The output is one of the given options.

S: Multiple-choice question answering.

No. 66 For the Cosmos task, we use the following task instructions:

F: The goal of this task is to answer the question, given the context. The output is one of the given options.

S: Multiple-choice question answering.

No. 67 For the SciDocs task, we use the following task instructions:

F: The goal of this task is to identify if the candidate title is topically "Relevant" or "Not relevant" to the query title of a scientific document.

S: Relevance: candidate title to query title.

We note that we used the test set of this task, and therefore our retrievers cannot be used for SciDocs evaluation in any ways.

No. 68 For the HotpotQA task, we use the following task instructions:

F: The goal of this task is to identify documents that are relevant to answering the question (QUESTION). The output is a #-separated list of the document IDs like "DOC_2#DOC_4".

S: Relevance: document IDs to question.

No. 69–70 For the AI2 ARC tasks, we use the following task instructions:

F: The goal of this task is to answer the question. The output is one of the given options.

S: Multiple-choice question answering.

No. 71 For the TriviaQA task, we use the following task instructions:

F: The goal of this task is to answer the question.

S: Question answering.

No. 72 For the Math task, we use the following task instructions:

F: The goal of this task is to solve the math problem.

S: Math problem solution.

No. 73 For the CommonGen task, we use the following task instructions:

F: The goal of this task is to generate a short text by using all the words in the input text.

S: Text generation: using all words.

No. 74 For the SNLI-en task, we use the following task instructions:

F: The goal of this task is to generate a text that can be entailed by the input text.

S: Text generation: entailment.

No. 75 For the PIQA-qgen task, we use the following task instructions:

F: The goal of this task is to generate a query that leads to the input text.

S: Query generation.

No. 76 For the arXiv task, we use the following task instructions:

- F: The goal of this task is to identify all the categories about the arXiv article. There are 147 categories: astro-ph, astro-ph.CO, ... The output is a list of the categories separated by # like "category_1#category_2#category_3".
- S: Multi-label category classification.

This task is based on a very large dataset, and we used a part of it (`train_0.jsonl.gz`).

No. 77 For the medRxiv task, we use the following task instructions:

- F: The goal of this task is to identify the category of the medRxiv article. There are 51 categories: addiction medicine, allergy and immunology, ...
- S: Category classification.

No. 78 For the DBpedia task, we use the following task instructions:

- F: The goal of this task is to identify the topic of the input text. The output is one of the 14 topics: Company, Educational Institution, Artist, Athlete, ...
- S: Topic classification.

No. 79 For the Yahoo task, we use the following task instructions:

- F: The goal of this task is to identify the topic about the community QA. The output is one of the 10 topics: Society & Culture, Science & Mathematics, Health, ...
- S: Topic classification.

No. 80 For the AG news task, we use the following task instructions:

- F: The goal of this task is to identify the topic of the titled text. The output is one of the 4 topics: World, Sports, Business, Science/Tech.
- S: News topic classification.

No. 81 For the TREC task, we use the following task instructions:

- F: The goal of this task is to identify what type of thing the question is asking about. The output is one of the 6 types: description, entity, abbreviation, human, numeric, location.
- S: Question topic classification.

A.3 Multi-lingual Data Augmentation

We describe details about the data augmentation presented in Section 5.4.

ANLI r1 The original input and output of this task are formatted as follows:

$x =$ context: "*context*" hypothesis: "*hypothesis*"

$y =$ Yes

We apply the translation to *context* and *hypothesis*, and keep the others in English. We set $(a, b) = (10, 20)$ for the target language sampling.

Tweet Sentiment Extraction The original input and output of this task are formatted as follows:

$x =$ *text*

$y =$ neutral

We apply the translation to *text*, and keep the others in English. We set $(a, b) = (4, 8)$ for the target language sampling.

SemRel The original input and output of this task are formatted as follows:

$x =$ ... *<e1>...</e1>* ... *<e2>...</e2>* ...

$y =$ e1:Effect e2:Cause

We apply the translation to ... *<e1>...</e1>* ... *<e2>...</e2>* ..., and keep the others in English. We filter out translated examples that result in not having the entity markers of e1 and e2. We set $(a, b) = (10, 20)$ for the target language sampling.

iDebate The original input and output of this task are formatted as follows:

$x =$ debate topic: "*debate topic*" arguments: "*arguments*"

$y =$ *claim*

We apply the translation to *debate topic*, *arguments*, and *claim*, and keep the others in English. We set $(a, b) = (20, 80)$ for the target language sampling.

BoolQ The original input and output of this task are formatted as follows:

$x =$ title: "*title*" text: "*text*" question: "*question*"

$y =$ *answer*

We apply the translation to *title*, *text*, *question*, and *answer*, and keep the others in English. We set $(a, b) = (10, 20)$ for the target language sampling.

PIQA-qgen The original input and output of this task are formatted as follows:

$x =$ *text*

$y =$ *query*

We apply the translation to *text* and *query*, and keep the others in English. We set $(a, b) = (10, 20)$ for the target language sampling.

DBpedia The original input and output of this task are formatted as follows:

$x =$ *text*

$y =$ Educational Institution

We apply the translation to *text*, and keep the others in English. We set $(a, b) = (10, 20)$ for the target language sampling.

TREC The original input and output of this task are formatted as follows:

$x =$ *text*

$y =$ human

We apply the translation to *text*, and keep the others in English. We set $(a, b) = (20, 40)$ for the target language sampling.

B Unseen Tasks

Table 2 summarized the unseen tasks we used in our experiments, and in this section we provide further details of the tasks.

AfriSenti Zero For this task, we use the following task instructions:

F: The goal of this task is to identify the sentiment label of the tweet. The output is positive, negative, or neutral.

S: Sentiment classification.

These are identical to those of the AfriSenti task.

GoEmotions For this task, we use the following task instructions:

F: The goal of this task is to identify emotions in the text from admiration, amusement, anger, ... If multiple emotions are identified, the output will be a #-separated string: emotion_1#emotion_2#emotion_3.

S: Multi-label emotion classification.

CLINC150 For this task, we use the following task instructions:

F: The goal of this task is to identify an intent given a user input. There are 150 intents: "current_location" "oil_change_when" "oil_change_how" ... Then the output is an intent label.

S: User input intent classification.

Unlike the previous work (Zhang et al., 2020; Hashimoto et al., 2024), we excluded all the out-of-scope examples from this task, and soley focus on the intent classification aspect.

Orcas-I For this task, we use the following task instructions:

F: The goal of this task is to identify the intent of the query with the search results (titles and URLs). The output is one of the 5 labels: Abstain, Factual, Transactional, Navigational, Instrumental.

S: Query intent classification.

MIT-R For this task, we use the following task instructions:

F: The goal of this task is to copy the given text by tagging attributes with XML tags. There are 8 attribute types: Amenity, Cuisine, Dish, Hours, Location, Price, Rating, Restaurant_Name. Then the output is like "word1 <Rating>word2 word3</Rating> word4 <Location>word5</Location>".

S: Attribute extraction.

SSENT For this task, we use the following task instructions:

F: The goal of this task is to copy the given text by tagging attributes with XML tags. There are 2 attribute types: Positive and Negative. Then the output is like "word1 <Negative>word2 word3</Negative> word4 <Positive>word5</Positive>".

S: Attribute extraction.

XML-MT For this task, we use the following task instructions:

F: The goal of this task is to translate an XML-tagged text from English to [target language] by preserving the XML structure. Both the input and output are like "word1 <tag-A>word2 word3</tag-A> word4 <tag-B>word5</tag-B>".

S: Translation: English to [target language].

C Evaluation Metrics

This section describes the evaluation metric used for each task in our evaluation. All the scores are in the range of [0, 100].

C.1 Seen Tasks

AfriSenti We use the label matching accuracy for this task.

DDI13 We use an F1 score based on precision and recall of the non-false classes.

ATIS-intent We use a corpus-level F1 score for the multi-label classification task.

MTOP-intent We use the label matching accuracy for this task.

Countfact We use a corpus-level F1 score based on precision and recall of the “counterfactual” class.

Offensive We use a corpus-level F1 score based on precision and recall of the “Offensive” class.

BC5CDR We use a corpus-level F1 score based on precision and recall of the labeled entities.

PHP We use a corpus-level BLEU (Papineni et al., 2002) score for this text generation task.

C.2 Unseen Tasks

AfriSenti Zero We use the label matching accuracy for this task.

GoEmotions We use a corpus-level F1 score for the multi-label classification task.

CLINC150 We use the label matching accuracy for this task.

Orcas-I We use the label matching accuracy for this task.

MIT-R We use a corpus-level F1 score based on precision and recall of the labeled attributes.

SSENT We use a corpus-level F1 score based on precision and recall of the labeled attributes.

XML-MT We use the structured BLEU metric (Hashimoto et al., 2019).