CodeMixBench: Evaluating Code-Mixing Capabilities of LLMs Across 18 Languages

Yilun Yang
NUC
jeromeyluck@gmail.com

Yekun Chai*
ETH Zurich
yechai@ethz.ch

Abstract

Code-mixing, the practice of switching between languages within a conversation, poses unique challenges for traditional NLP. Existing benchmarks are limited by their narrow language pairs and tasks, failing to adequately assess large language models' (LLMs) code-mixing abilities. Despite the recognized importance of code-mixing for multilingual users, research on LLMs in this context remains sparse. Additionally, current techniques for synthesizing code-mixed data are underdeveloped to generate code-mixing. In response, we introduce CodeMixBench, a comprehensive benchmark covering eight tasks, including three specific to LLMs and five traditional NLP tasks, and 18 languages across seven language families. We also propose a new method for generating large-scale synthetic code-mixed texts by combining word substitution with GPT-4 prompting. Our evaluation reveals consistent underperformance of LLMs on code-mixed datasets involving different language families. Enhancements in training data size, model scale, and few-shot learning could improve their performance. The code and dataset are available at https://github.com/Jeromeyluck/CodeMixBench.

1 Introduction

Code-mixing is a linguistic phenomenon where multilingual speakers switch or mix two or more languages within a single utterance or conversation. This typically occurs due to a lack of suitable vocabulary or expressions in one language, the presence of untranslatable terms, or contextual factors such as interlocutors, situational context, messages, attitudes, and emotions (Kim, 2006). With the global rise of social media, there has been a substantial increase in code-mixed content (Rijhwani et al., 2017), prompting extensive interest from linguists and NLP researchers (Winata et al., 2023). However, several key issues remain unresolved.

Existing studies are difficult to compare directly because they focus on different downstream tasks and language pairs. To address this issue, LinCE (Aguilar et al., 2020) and GLUECoS (Khanuja et al., 2020b) introduced two benchmarks, but they only cover a limited number of language pairs and traditional NLP tasks. LinCE addresses four language pairs and five traditional NLP tasks, including language identification (LID), part-of-speech tagging (POS), named entity recognition (NER), sentiment analysis (SA), and machine translation (MT), while GLUECoS covers only two language pairs and six traditional NLP tasks, *i.e.*, LID, POS, NER, SA, question answering (QA), and natural language inference (NLI). These traditional NLP tasks are insufficient to evaluate LLM performance comprehensively.

Despite strong multilingual performance on various benchmarks, LLMs' capabilities with code-mixing remain underexplored. Limited studies suggest that LLMs often perform worse than smaller, fine-tuned models on code-mixing tasks (Zhang et al., 2023), and multilingual users prefer chatbots that handle code-mixing well (Bawa et al., 2020). Thus, incorporating code-mixing into LLM evaluation is crucial.

Creating new code-mixed datasets for LLMs involves using synthesis techniques. Some studies (Bhat et al., 2016; Pratapa et al., 2018) focused on generating synthetic code-mixed data to solve the scarcity of code-mixed data, using methods based on the Equivalence Constraint theory (POPLACK, 1980), a linguistic theory that restricts the occurrences of code-mixing. However, the quality of these outputs heavily depends on the performance of word alignment and syntactic parsing tools. Recent efforts to generate code-mixed text using data-driven models still face challenges related to dataset size, quality, or linguistic diversity (Yang et al., 2020; Hsu et al., 2023). Also, initial attempts to use LLMs for generating code-mixed data did not

*Corresponding author.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 2139–2169
November 4-9, 2025

fully leverage their instruction-following capabilities (Yong et al., 2023).

In response to these issues, we introduce the CodeMixBench, a code-mixing evaluation benchmark including eight tasks–three for evaluating LLMs (knowledge reasoning, mathematical reasoning, and truthfulness) and five for traditional NLP tasks (LID, POS, NER, SA, and MT). They span 18 languages from seven language families, covering high-resource, medium-resource, and low-resource languages. Our benchmark largely expands language pair and task coverage compared to LinCE and GLUECoS (Appendix A). We also propose a novel synthetic code-mixing approach using word substitution within GPT-4 prompting to generate large-scale code-mixed texts from parallel corpora.

Our contributions are summarized as follows:

1. We present CodeMixBench, the first comprehensive benchmark for evaluating the performance of LLMs on multilingual code-mixing. We have synthesized 22 datasets and, through extensive research, compiled 30 open-source code-mixed datasets to integrate into our benchmark. In total, the benchmark encompasses eight tasks and 18 languages from seven language families (§3).
2. We propose a novel pipeline for large-scale synthesis of multilingual code-mixing data, integrating word substitution with LLM prompts for the first time. The synthetic results validate the efficiency of our approach in generating substantial multilingual code-mixed data. (§3.2).
3. We evaluate three families of LLMs on CodeMixBench, revealing consistent underperformance across all models on code-mixing datasets involving language pairs from different language families. However, enhancements in training data size, model scale, post-training, and few-shot learning can improve LLM performance on code-mixing datasets (§4).

2 Related Work

Code-Mixing Challenge Early research employed linguistic rules and statistical methods (Li and Fung, 2012, 2014; Bhat et al., 2016; Rijhwani et al., 2017) for code-mixing modeling. Subsequently, research shifted towards neural network models like RNNs and LSTMs (Adel et al., 2013b,a; Wang et al., 2018; Winata et al., 2018), and more recently towards pre-trained language models such as mBERT and XLM-R (Winata et al., 2021; Malmasi et al., 2022; Pérez et al., 2022). These methodologies have been applied to various code-mixing-related downstream tasks, including language identification (Solorio et al., 2014; Molina et al., 2016), named entity recognition (Aguilar et al., 2018), part-of-speech tagging (Singh et al., 2018b; Soto and Hirschberg, 2018), sentiment analysis (Patra et al., 2018; Patwa et al., 2020), machine translation (Srivastava and Singh, 2020; Chen et al., 2022), natural language inference (Khanuja et al., 2020a), question answering (Chandu et al., 2018), and multilingual code generation (Chai et al., 2023a; Peng et al., 2024). Benchmarks, such as GLUECoS (Khanuja et al., 2020b) and LinCE (Aguilar et al., 2020) primarily focus on traditional NLP tasks and are restricted to a limited number of languages. Recent research by Zhang et al. (2023) on the performance of multilingual LLMs in code-switching contexts indicates that, despite their strong capabilities across various monolingual tasks, they still yield inferior performance compared to fine-tuned smaller models.

Synthesis of Code-Mixed Data Early research synthesized code-mixed data based on linguistic rules. Following the EC theory, (Bhat et al., 2016; Pratapa et al., 2018) utilized word alignment tools and syntactic parsers to enable the structural substitution and integration of lexical elements within aligned parse trees. Subsequently, researchers trained generative models to produce code-mixed data, such as a sequence-to-sequence model with a Pointer-Generator (Winata et al., 2019; Gupta et al., 2020), Generative Adversarial Networks (Chang et al., 2019; Chai et al., 2021, 2023b), and Variational AutoEncoders (Samanta et al., 2019b). An increasing number of works (Samanta et al., 2019a; Yang et al., 2020; Arora et al., 2023; Hsu et al., 2023) focused on extending pre-trained models for code-mixed data generation. Yong et al. (2023) examined the ability of LLMs to generate code-mixed text in Southeast Asian languages. Instead of using LLMs to directly generate code-mixed text, we revisit the EC theory and integrate its core principles into the prompt. Based on parallel corpora, we instruct the LLM to replace lexical elements between parallel sentences, thereby generating grammatically coherent code-mixed text.

3 CodeMixBench

3.1 Overview

To evaluate LLMs' comprehension of multilingual code-mixed texts, we introduce CodeMixBench, a

Family	Language	ISO code	Pop. (M)	CC (%)
Germanic	English	es	1456	45.51
	German	de	133	5.263
	Dutch	nl	30	1.910
	Frisian	fy	0.6	\
Sino-Tibetan	Chinese	zh	1138	4.423
	Hokkien	hok	50	\
Romance	Spanish	es	559	4.594
	French	fr	310	4.307
Afro-Asiatic	Arabic	ar	380	0.617
	MSA	msa	330	\
	EA	ea	103	\
Indo-Aryan	Hindi	hi	610	0.185
	Bengali	bn	273	0.106
	Marathi	mr	99	0.024
	Nepali	ne	32	0.044
Dravidian	Tamil	ta	87	0.042
	Malayalam	ml	37	0.022
Tupian	Guarani	gn	6.5	\

Table 1: **Statistics of 18 languages from 7 families**. Each language is assigned a unique code in this paper based on the ISO 639. The *Pop.* indicates the population in millions of speakers. The *CC* indicates ratios of languages in the CommomCrawl. The *MSA* and *EA* stand for Modern Standard Arabic and Egyptian Arabic.

benchmark comprising eight tasks across 18 languages. Table 1 details the speaker population and resource ratio on CommonCrawl[1] for each language, identified by their ISO 639 codes. The chosen languages exhibit diversity in language families, resource availability, and speaker populations. Motivated by Bang et al. (2023); Lai et al. (2023a,b), five languages (zh, es, fr, de, nl) are categorized as high-resource (*CC* >1%), three (ar, hi, bn) as mid-resource (0.1%-1%), and four (mr, ne, ta, ml) as low-resource (<0.01%).

Our benchmark comprises synthesized datasets targeting knowledge reasoning, mathematical reasoning, and truthfulness tasks, along with LID, POS, NER, SA, and MT tasks, which have been adapted from open-source studies. For knowledge reasoning, we developed the code-mixed MMLU (CM-MMLU) based on the MMLU test set (Hendrycks et al., 2021), featuring multiple-choice questions from 57 subjects to assess the model's comprehensive knowledge reasoning abilities. For mathematical reasoning, we created the code-mixed GSM8K (CM-GSM8K), derived from the GSM8K test set (Cobbe et al., 2021), which evaluates mathematical reasoning capabilities with each question including step-by-step solutions. For truthfulness assessment, we constructed the code-mixed TruthfulQA (CM-TruthfulQA) using 817 multiple-choice questions from the TruthfulQA test set (Lin et al., 2022). Details of the collected datasets are provided in Appendix B.

Figure 1 demonstrates the entire process of constructing our synthetic dataset, including a real example. The original datasets undergo three phases to be transformed into code-mixed datasets: First, collecting existing multilingual parallel corpora or constructing them via translation (detailed in Section 3.2). Second, instructing GPT to generate code-mixed datasets in various language pairs based on the parallel corpus (detailed in Section 3.3). Third, evaluating and filtering the synthetic dataset at word-level, semantic-level, and human-level (detailed in Section 3.4). We finally synthesized 11 code-mixed language pairs for CM-MMLU with 12,156 question-option-answer combinations, 4 pairs for CM-TruthfulQA with 3,122 multiple-choice instances, 4 pairs for CM-GSM8K with 4,367 math problems, and 3 pairs for MT with 2,711 code-mixed sentences. The datasets encompass 12 languages from six families: Germanic (en, de, nl), Romance (es, fr), Sino-Tibetan (zh), Afro-Asiatic (ar), Indo-Aryan (hi, bn, mr, ne), and Dravidian (ta). Linguistic diversity enables assessing the impact of multilingual code-switching on model performance. Detailed statistics for the synthetic datasets are provided in Appendix H.

3.2 Parallel Corpus Construction

In first phase, we construct four parallel corpora for synthesizing code-mixed datasets. Using the multilingual MMLU test set from Opaki (Lai et al., 2023b), we develop a parallel corpus of 4,018 multiple-choice questions, each available in 12 languages (en, zh, es, fr, ar, de, nl, hi, bn, mr, ne, ta). Additionally, we utilized GPT-4 Turbo to translate the English-only GSM8K and TruthfulQA datasets into four languages (zh, es, hi, ar), resulting in two parallel corpora with 1319 and 817 samples, respectively. To enhance linguistic diversity in machine translation tasks, we extracted a 4,344-sample parallel corpus (en, zh, es, ar) from the TED2013 dataset in OPUS (Tiedemann, 2012).

3.3 Instruction Synthesis

In second stage, we instruct GPT-4 Turbo to synthesize code-mixed sentences based on the parallel cor-

[1]https://commoncrawl.github.io/cc-crawl-statistics/plots/languages.html

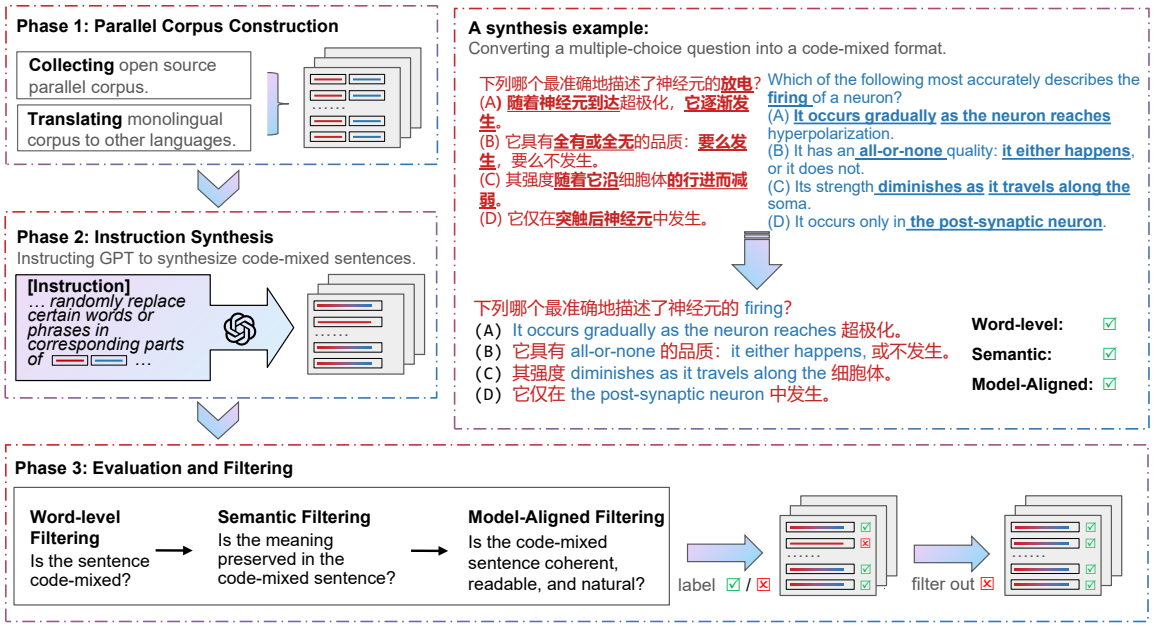


Figure 1: **Illustration of the synthesis pipeline.**

pora. Code-mixing appears as a random alternation between and within sentences, but it is actually constrained by linguistic factors. POPLACK (1980) states code-mixing happens where the grammatical structures of both languages align. By ensuring that each language fragment is syntactically correct according to its own rules and that switches occur at structurally compatible points, word substitution between parallel corpus helps create coherent mixed-language sentences. Based on this idea, we devise a prompt (shown in Appendix F.1) for GPT-4 Turbo to randomly select and replace words or phrases in equivalent places where the surface structures of two sentences align.

This method effectively embeds one language into another, implementing intra-sentential and inter-sentential code-mixing. Furthermore, we prompt GPT-4 Turbo to respond with the chosen words and their corresponding parts in another language.

3.4 Evaluation and Filtering

We implement a series of evaluation and filtering processes for the generated data.

Word-Level Filtering We use the Multilingual Index (M-index) (Barnett et al., 2000) and the Probability of Switching (I-index) (Guzmán et al., 2017) as word-level evaluation metrics. Based on word-level language tagging (annotation strategy details in Appendix C), we calculated the M-index and I-index for each code-mixed text. Two code-mixing metrics are defined as:

$$M\text{-index} = \frac{1 - \sum p_j^2}{(k-1)\sum p_j^2}$$

where p_j represents the proportion of the j-th category. k is the total number of language categories;

$$I\text{-index} = \frac{\sum_{1 \le i \le n-1} S(i, i+1)}{n-1}$$

where $S(i, i+1) = 1$ if the i-th and $i+1$-th tokens of a sentence belongs to different languages; otherwise, $S(i, i+1) = 0$. n represents the total number of tokens in a sentence.

The M-index ranges from 0 (monolingual text) to 1 (perfectly balanced code-mixed text with equal contributions from each language). Similarly, the I-index ranges from 0 (monolingual text) to 1 (optimal code-mixed text with alternating tokens from different languages). To ensure dataset quality, we set the thresholds for the M-index and I-index to 0.1 to filter out monolingual sentences and those with low mixing or switching frequencies.

Semantic Filtering To ensure the semantic consistency between the generated text and the original text, we computed the sentence similarity metrics for both texts. Additionally, We evaluated sentence similarity across two parallel corpora to assess the quality of the original parallel texts. First, we used LaBSE (Feng et al., 2022) (Appendix D) to project the original two monolingual texts (*L1*, *L2*) from the parallel corpus and the synthesized code-mixed text (*CM*) into a common vector space. Subsequently, we calculate the pairwise cosine similarities among these three texts (*CM*, *L1*, *L2*), resulting

in three similarity scores ranging from 0 to 1. The score between *CM* and *L1/L2* partially reflects the synthesis quality of our method, while the score between *L1* and *L2* indicates the translation quality of the original parallel corpus. We determine that a similarity score below 0.8 suggests potential issues in the synthesis result or parallel corpus, necessitating the exclusion of such samples.

Model-Aligned Filtering To ensure the naturalness, coherence, and readability of synthesized sentences, we employ a highly human-aligned GPT-4 Turbo model (Appendix E) for automated evaluation. We prompt the model to assess synthetic results on naturalness, coherence, and readability, assigning scores to each criterion. Each criterion is rated on a scale from 1 to 3 (poor, fair, good), with detailed definitions provided for each level, shown in Appendix F.2. We filter out synthesized sentences if any score equals 1, indicating deficiencies in naturalness, coherence, or readability.

4 Experiments

4.1 Experiment Setup

Evaluation Settings For **CM-MMLU** and **CM-TruthfulQA**, we prompt models to select the correct option for multiple-choice questions. We use chain-of-thought (CoT) evaluation for **CM-GSM8K** task and parsed the model's response using regular regex to obtain the final solution. We report accuracy as the evaluation metric. For above three tasks, we also provide the model performance of English-only evaluation (*en only*) for reference. For **LID, POS, NER, and SA** tasks, we prompt the models to generate the answers. Specifically, we provide the LLMs with all possible tags in the prompt and instruct models to generate in JSON format. In the **MT** task, we instructed models to translate code-mixed sentences. We use accuracy for LID, POS, NER, and SA tasks, and the BLEU score for MT assessment. All evaluations are under one-shot settings. We present the prompts for all 8 tasks in the Appendix G.

Models We selected LLMs from three different families for the comparison evaluation. For the GPT family, we evaluated GPT-3.5 Turbo-instruct, GPT-3.5 Turbo, GPT-4 Turbo (OpenAI et al., 2024) and GPT-4o. For the LLaMA family, we evaluated LLaMA2-Chat (7B, 13B, 70B) (Touvron et al., 2023), LLaMA3-Base (8B), and LLaMA3-Instruct (8B, 70B). For the Mistral family, we evaluated Mistral 7B (Jiang et al., 2023), Mixtral 8x7B (Jiang et al., 2024), and Mixtral 8x22B. We set the top-p to 0.95 and temperature to 0.8 for GPT, and used greedy decoding for LLaMA and Mistral models.

4.2 Main Results

Table 2 presents the experimental results of the selected models across the CM-MMLU, CM-GSM8K, and CM-TruthfulQA. Due to space constraints, the performance of the GPT family on LID, POS, NER, SA, and MT tasks is detailed in Appendix I, with visualizations provided in Figure 2.

Larger models excel on CodeMixBench In Table 2, GPT-4o achieves the highest scores across all language pairs in the CM-MMLU task, while GPT-4 Turbo attains the highest scores for each language pair in the CM-GSM8K and CM-TruthfulQA tasks. This suggests that GPT-4o excels in comprehensive knowledge reasoning, whereas GPT-4 Turbo is superior in mathematical reasoning and truthfulness. Additionally, within the LLaMA2, LLaMA3, and Mistral model families, the highest scores across all datasets consistently come from the largest models. Therefore, increasing model size enhances performance on multilingual code-mixed datasets.

GPT-3.5-Turbo-Instruct vs. GPT-3.5 Turbo In Table 2, GPT-3.5 Turbo outperforms GPT-3.5-Turbo-Instruct by an average of 2.07 points in CM-MMLU, 14.54 points in CM-GSM8K, and 7.23 points in CM-TruthfulQA. Table 8 in the Appendix I shows GPT-3.5 Turbo scored higher on LID (+9.51%), POS (+1.68%), NER (+10.99%), and MT (+1.07%), but was 14.98 points lower on SA. This may be due to differing focuses during instruction tuning, with GPT-3.5 Turbo emphasizing conversational completion and GPT-3.5-Turbo-Instruct focusing on instruction completion, leading to different training corpora. Thus, GPT-3.5 Turbo excelled over GPT-3.5-Turbo-Instruct in all CodeMixBench tasks except SA.

LLaMA3-8B-Base vs. LLaMA3-8B-Instruct In Table 2, LLaMA3-8B-Instruct performs comparably to LLaMA3-8B-Base on CM-MMLU and CM-TruthfulQA but outperforms it by 7.73 points on CM-GSM8K. This is likely due to the increased complexity of the mathematical reasoning required by CM-GSM8K. The improved performance on CM-GSM8K can be attributed to high-quality prompts during continued post-training stages, including supervised fine-tuning and alignment tun-

	GPT -Instruct	GPT			LLaMA2 -Chat			LLaMA3 -Base	LLaMA3 -Instruct		Mistral & Mixtral		
	3.5-T	3.5-T	4-T	4o	7b	13b	70b	8b	8b	70b	7b	8x7b	8x22b
						CM-MMLU							
en only	64.90	66.30	83.10	**85.60**	38.00	47.80	**61.50**	63.30	65.60	**77.20**	55.3	67.30	**75.50**
zh-en	60.99	60.81	79.08	**82.97**	30.80	35.92	**46.78**	56.31	57.63	**73.79**	47.40	59.84	**66.64**
hi-en	53.32	55.37	77.83	**82.13**	29.00	30.96	**44.63**	53.61	56.93	**74.61**	40.82	52.44	**59.67**
bn-en	46.32	47.49	72.26	**78.28**	25.49	29.71	**39.23**	46.50	49.01	**69.75**	38.60	48.11	**55.03**
mr-en	46.95	49.67	72.26	**77.98**	29.05	30.27	**38.71**	50.89	51.55	**67.29**	37.49	48.27	**55.39**
ne-en	46.70	48.78	72.78	**76.70**	25.91	29.39	**38.26**	47.22	49.22	**66.52**	34.61	45.74	**55.13**
es-en	65.01	69.20	81.24	**86.30**	32.37	42.67	**59.95**	61.78	54.98	**79.67**	53.93	69.28	**74.17**
fr-en	67.21	68.83	81.21	**85.28**	34.78	43.45	**57.54**	60.79	64.32	**78.50**	56.55	69.65	**73.98**
ar-en	53.94	56.45	77.06	**80.35**	25.71	30.04	**40.17**	51.17	46.76	**71.86**	37.32	51.52	**59.83**
ta-en	44.03	45.75	64.09	**70.77**	26.65	32.09	**39.06**	46.61	48.42	**62.18**	38.87	47.18	**52.34**
nl-en	66.08	67.14	82.64	**85.37**	32.60	42.73	**56.21**	61.32	62.11	**79.30**	53.74	68.55	**71.98**
de-en	67.63	68.46	80.71	**84.60**	34.32	42.21	**57.98**	59.74	63.91	**77.18**	54.08	66.23	**72.54**
Average	56.20	58.27	76.47	**80.97**	29.70	35.40	**47.14**	54.18	54.99	**72.79**	44.85	56.98	**63.34**
						CM-GSM8K							
en only	66.55	80.05	**95.23**	92.50	26.21	35.83	**58.78**	77.41	80.23	**93.91**	45.28	64.34	**87.29**
zh-en	57.54	73.73	**92.11**	90.61	21.98	28.97	**47.95**	67.73	76.32	**90.61**	40.06	59.34	**83.72**
hi-en	54.63	67.42	**93.60**	89.57	17.72	23.92	**40.16**	68.01	75.89	**90.45**	33.46	54.04	**82.19**
es-en	63.20	77.23	**93.91**	90.91	19.33	31.95	**53.22**	71.23	75.99	**92.41**	43.25	63.90	**84.38**
ar-en	57.20	72.36	**94.05**	90.12	14.88	21.31	**37.91**	65.16	74.86	**88.96**	33.49	51.92	**78.21**
Average	58.14	72.68	**93.42**	90.30	18.47	26.54	**44.81**	68.03	75.76	**90.61**	37.57	57.30	**82.12**
						CM-TruthfulQA							
en only	57.16	64.26	**83.72**	81.76	22.03	25.21	**43.82**	47.25	46.76	**70.87**	53.24	66.46	**73.93**
zh-en	46.43	54.09	**79.25**	77.56	18.42	24.64	**33.33**	45.53	44.36	**67.83**	48.12	56.42	**63.68**
hi-en	39.37	48.08	**81.11**	78.47	19.82	21.80	**29.99**	41.88	42.93	**66.31**	40.55	51.12	**58.52**
es-en	46.43	55.07	**81.10**	77.85	21.65	25.78	**36.80**	46.06	44.31	**68.84**	48.31	58.57	**66.46**
ar-en	46.54	50.44	**80.50**	76.48	20.63	20.88	**28.55**	42.26	42.64	**66.67**	40.88	47.67	**59.25**
Average	44.69	51.92	**80.49**	77.59	20.13	23.28	**32.17**	43.93	43.56	**67.41**	44.47	53.45	**61.98**

Table 2: **One-shot accuracy of selected models on CM-MMLU, CM-GSM8K and CM-TruthfulQA.** Where 3.5-T indicates GPT-3.5 Turbo, and 4-T indicates GPT-4 Turbo. The *en only* stands for a dataset we randomly sample from the test set of the original dataset in English. To be compared with other code-mixed datasets, the *en only* datasets for CM-MMLU, CM-GSM8K, and CM-TuthfulQA contain 1000, 1133, and 817 English instances each. The *Average* represents the mean score of each model across various datasets (excluding *en only* dataset) from a given task. For each model family, the scores of the top-performing models are highlighted in bold.

ing, followed by the pre-training of LLaMA3.

LLaMA2 vs. LLaMA3 Table 2 shows that LLaMA3-8B outperforms LLaMA2-7B-Chat with average gains of 25.29, 57.29, and 23.43 points on CM-MMLU, CM-GSM8K, and CM-TruthfulQA, respectively. Additionally, LLaMA3-70B surpasses LLaMA2-70B with improvements of 25.75, 45.80, and 35.24 points on the same benchmarks. These enhancements may be due to the training dataset for LLaMA3 containing over 15T tokens, a size seven times larger than that used for LLaMA2.

Mistral 7B vs. Mixtral 8x7B We also observe from Table 2 that Mixtral 8x7B outperforms Mistral 7B by 12.14, 19.73, and 8.98 points on CM-MMLU, CM-GSM8K, and CM-TruthfulQA, correspondingly. This improvement is likely due to the scaling of model parameters in Mixture of Experts (MoE) architecture and the substantial increase in multilingual training compute for Mixtral 8x7B.

4.3 Analysis across Languages

Figure 3 illustrates the accuracy variations of LLMs from three families on the CM-MMLU, CM-GSM8K, and CM-TruthfulQA tasks across different language pairs.

Cross-family code-mixing can impair the performance of LLMs. Figure 3 shows significant fluctuations in *zh-en*, *hi-en*, *bn-en*, *mr-en*, *ne-en*, *ar-en*, and *ta-en* language pairs, while *es-en*, *fr-en*, *de-en*, and *nl-en* pairs perform similarly to English-only scenario. This similarity may be attributed to English, German, Dutch, Spanish, and French having similar word order features according to WALS (Dryer and Haspelmath, 2013), along with their common Indo-European family and geographic proximity. Therefore, code-mixing between languages with substantial linguistic differences can

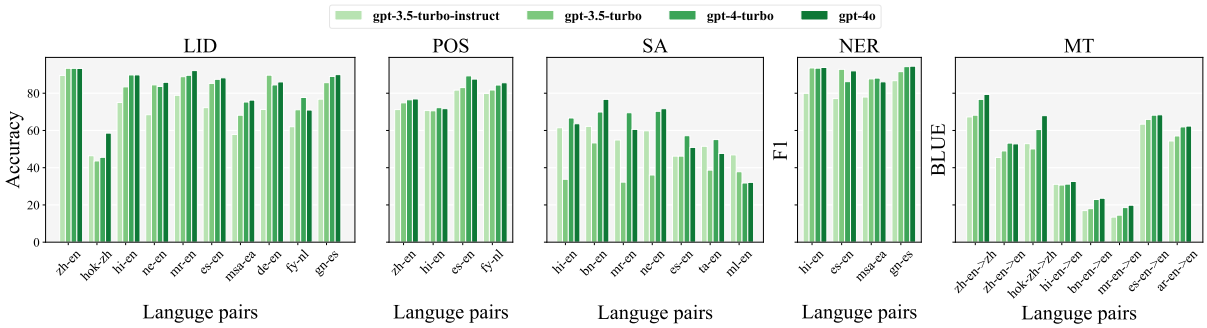


Figure 2: **One-shot accuracy versus language pairs for GPT models on LID, POS, SA, NER and MT.**

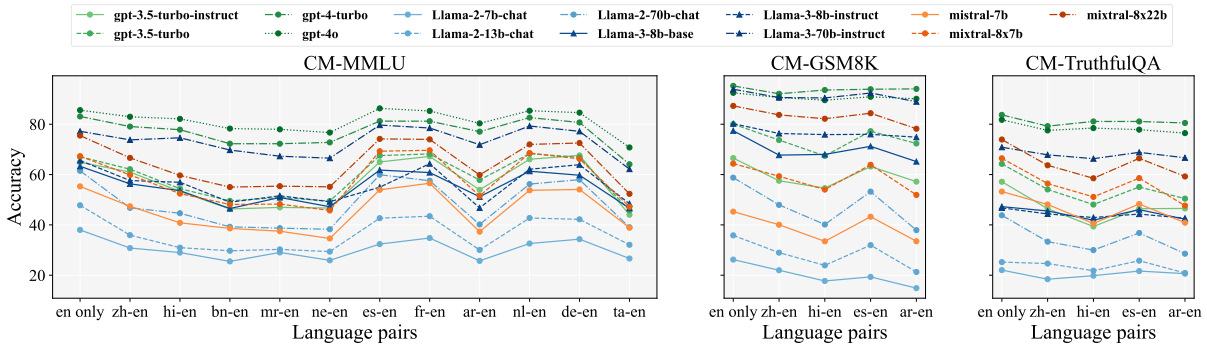


Figure 3: **Accuracy versus language pairs for models on CM-MMLU, CM-GSM8K and CM-TruthfulQA.**

significantly hinder the performance of LLMs.

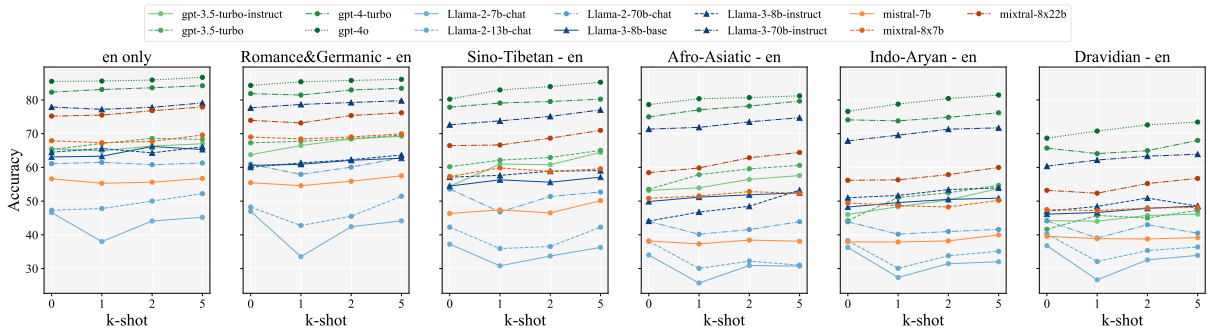
Models exhibit consistent fluctuation patterns across different code-mixed language pairs. Figure 3 reveals a notable trend: despite originating from three distinct institutions, the models display parallel accuracy fluctuations across different language pairs for the three tasks. For CM-MMLU, most models show a decline in accuracy from *en only* to *ne-en*, followed by a rebound for *es-en* and *fr-en*. This uniform impact on performance likely results from overlapping training data sourced from the internet, commonly used by three organizations during model training.

More low-resource data improves code-mixing comprehension. Analyzing Figure 3 and Table 2, the decrease for high-resource language and English code-mixing (*zh-en*) was 2.63 points compared to English-only datasets. Medium-resource language code-mixing (*hi-en*, *ar-en*, *bn-en*) showed declines of 3.47, 5.25, and 7.32 points, respectively, while low-resource language mixtures (*mr-en*, *ne-en*, *ta-en*) experienced more substantial drops of 7.62, 8.9, and 14.83 points. This indicates the model has a better understanding of code-mixed data involving high-resource languages and English. Consequently, increasing training on low-resource language corpora could improve the model's comprehension of code-mixed data involving these languages.

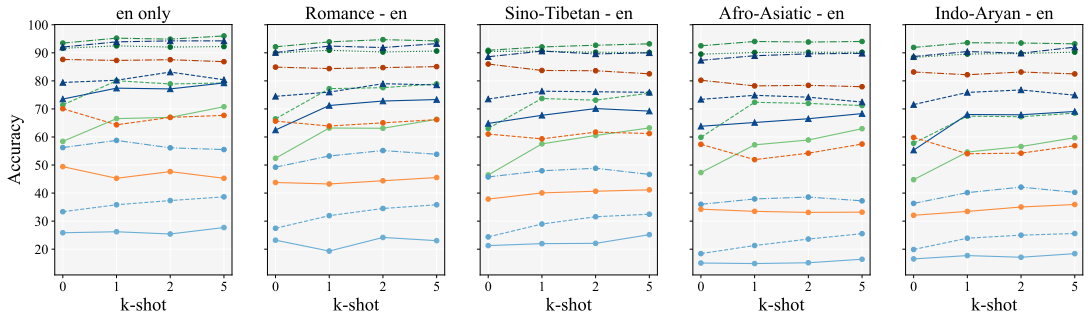
4.4 K-shot Analysis

To further investigate the impact of varying quantities of code-mixed examples on model performance, we conducted k-shot evaluations ($k \in \{0, 1, 2, 5\}$) on the CM-MMLU, CM-GSM8K, and CM-TruthfulQA datasets. English-only (*en only*) served as a control group, allowing us to compare performance trends between the *en only* and various code-mixed scenarios across different language families. Results were averaged by language family and visualized in Figure 4, with full results in Appendix J. Figure 4 shows that models like GPT-4 Turbo, GPT-4o, and LLaMA3-70B-Instruct, which have higher average accuracy scores, maintain more stable k-shot accuracy trends as k increases. This indicates their robust multilingual and few-shot learning abilities. In contrast, other models often experience sudden drops in accuracy for certain language pairs as k increases.

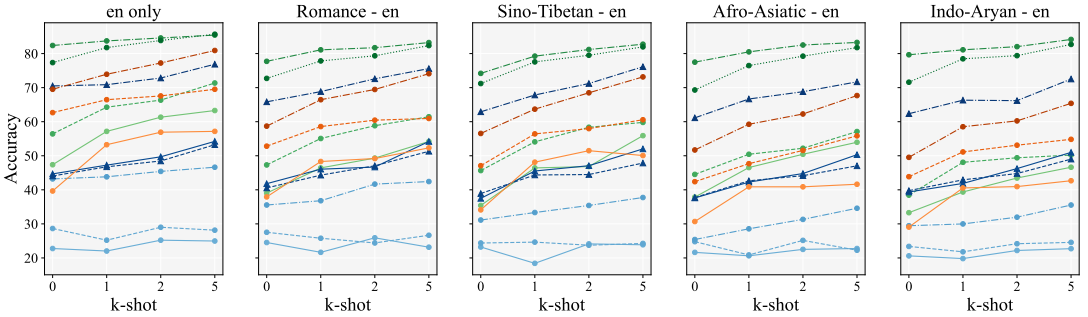
Advanced models excel at few-shot learning on knowledge and truthfulness reasoning. Figure 4(a) indicates that the accuracy of GPT models, LLaMA3, and Mistral generally increases with higher k on the CM-MMLU, whereas LLaMA2 models show a significant drop at one-shot before



(a) CM-MMLU



(b) CM-GSM8K



(c) CM-TruthfulQA

Figure 4: **Accuracy of k-shot evaluation for three model families on three tasks.**

recovering. LLaMA2-13B-Chat and LLaMA2-70B-Chat demonstrate a positive correlation between accuracy and k values in *en only* datasets, indicating their few-shot learning capabilities. In contrast, for code-mixed datasets, one-shot and two-shot accuracies are lower than zero-shot, with even five-shot performance lagging behind zero-shot for *Sino-Tibetan - en*, *Afro-Asiatic - en*, *Indo-Aryan - en*, and *Dravidian - en*. This suggests code-mixing hinders the one-shot and two-shot learning capabilities of these models, though performance can gradually recover at five-shot. Also in Figure 4(c), except for LLaMA2-7B-Chat and LLaMA2-13B-Chat, all models' accuracy scores increase with k on CM-TruthfulQA. In summary, few-shot learning is effective for all selected models except LLaMA2 in knowledge and truthfulness reasoning.

Few-shot learning minimally enhances mathematical reasoning. In Figure 4(b), the model's k-shot accuracy on the CM-GSM8K task shows less variability compared to the other tasks, which we attribute to the higher complexity of CM-GSM8K relative to CM-MMLU and CM-TruthfulQA, posing greater challenges to a model's few-shot learning. However, GPT-3.5 Turbo, GPT-3.5-Turbo-Instruct, and LLaMA3-8B-Base exhibit significant accuracy improvements from zero-shot to one-shot. This is because these models initially fail to follow the required response format in zero-shot prompts, causing incorrect answers, while one-shot improves these models' output format adherence. Besides, Mixtral-7x22b demonstrates a decrease in accuracy as k increases across all datasets, indicating its inadequate few-shot learning capability on CM-GSM8K. Overall, in the CM-GSM8K task,

few-shot learning provides limited enhancement in mathematical reasoning for the GPT and LLaMA family and may negatively affect Mistral models.

-

5 Conclusion

This study introduces CodeMixBench, a comprehensive benchmark for evaluating code-mixing performance in LLMs, spanning eight tasks and 18 languages. We also adopt GPT-4 Turbo for constructing synthetic code-mixed data to address data scarcity issues. Our findings show that while code-mixing challenges LLMs performance, improvements can be achieved through larger pre-training datasets, increased model scales, and few-shot learning. In the future, CodeMixBench holds great promise for evaluating the code-mixing capabilities of LLMs and inspiring further research in this area.

Limitations

We introduce CodeMixBench, a collection of 22 synthetic datasets and 30 open-source datasets, each with potential quality issues. Our synthesis method generates large-scale code-mixed datasets with detailed filtering, but unexpected quality problems can still occur. Furthermore, our benchmark includes 18 languages, making it challenging to maintain consistent quality control. Furthermore, it could be promising to evaluate and mitigate the potential bias in code-mixing scenarios (Ravfogel et al., 2020; Peng et al., 2025).

Acknowledgments

We would like to thank all anonymous reviewers for their insightful comments and feedback.

References

Heike Adel, Ngoc Thang Vu, Franziska Kraus, Tim Schlippe, Haizhou Li, and Tanja Schultz. 2013a. Recurrent neural network language modeling for code switching conversational speech. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8411–8415.

Heike Adel, Ngoc Thang Vu, and Tanja Schultz. 2013b. Combination of Recurrent Neural Networks and Factored Language Models for Code-Switching Language Modeling. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 206–211, Sofia, Bulgaria. Association for Computational Linguistics.

Gustavo Aguilar, Fahad AlGhamdi, Victor Soto, Mona Diab, Julia Hirschberg, and Thamar Solorio. 2018. Named Entity Recognition on Code-Switched Data: Overview of the CALCS 2018 Shared Task. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 138–147, Melbourne, Australia. Association for Computational Linguistics.

Gustavo Aguilar, Sudipta Kar, and Thamar Solorio. 2020. LinCE: A Centralized Benchmark for Linguistic Code-switching Evaluation.

Gaurav Arora, Srujana Merugu, and Vivek Sembium. 2023. CoMix: Guide Transformers to Code-Mix using POS structure and Phonetics. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7985–8002, Toronto, Canada. Association for Computational Linguistics.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–718, Nusa Dua, Bali. Association for Computational Linguistics.

Ruthanna Barnett, Eva Codó, Eva Eppler, Montse Forcadell, Penelope Gardner-Chloros, Roeland van Hout, Melissa Moyer, Maria Carme Torras, Maria Teresa Turell, Mark Sebba, Marianne Starren, and Sietse Wensing. 2000. The lides coding manual: A document for preparing and analyzing language interaction data version 1.1—july, 1999. *International Journal of Bilingualism*, 4(2):131–132.

Anshul Bawa, Pranav Khadpe, Pratik Joshi, Kalika Bali, and Monojit Choudhury. 2020. Do multilingual users prefer chat-bots that code-mix? let's nudge and find out! *Proc. ACM Hum.-Comput. Interact.*, 4(CSCW1).

Gayatri Bhat, Monojit Choudhury, and Kalika Bali. 2016. Grammatical Constraints on Intra-sentential Code-Switching: From Theories to Working Models.

Anouck Braggaar and Rob van der Goot. 2021. Challenges in Annotating and Parsing Spoken, Code-switched, Frisian-Dutch Data. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 50–58, Kyiv, Ukraine. Association for Computational Linguistics.

Jesús Calvillo, Le Fang, Jeremy Cole, and David Reitter. 2020. Surprisal Predicts Code-Switching in Chinese-English Bilingual Text. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4029–4039, Online. Association for Computational Linguistics.

Yekun Chai, Shuohuan Wang, Chao Pang, Yu Sun, Hao Tian, and Hua Wu. 2023a. ERNIE-code: Beyond English-centric cross-lingual pretraining for programming languages. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10628–10650, Toronto, Canada. Association for Computational Linguistics.

Yekun Chai, Qiyue Yin, and Junge Zhang. 2023b. Improved training of mixture-of-experts language gans. In *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2023, Rhodes Island, Greece, June 4-10, 2023*, pages 1–5. IEEE.

Yekun Chai, Haidong Zhang, Qiyue Yin, and Junge Zhang. 2021. Counter-contrastive learning for language gans. In *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 4834–4839. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Navya Jose, Shardul Suryawanshi, Elizabeth Sherly, and John Philip McCrae. 2020a. A Sentiment Analysis Dataset for Code-Mixed Malayalam-English. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced Languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 177–184, Marseille, France. European Language Resources association.

Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, and John Philip McCrae. 2020b. Corpus creation for sentiment analysis in code-mixed Tamil-English text. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 202–210, Marseille, France. European Language Resources association.

Khyathi Chandu, Thomas Manzini, Sumeet Singh, and Alan W. Black. 2018. Language Informed Modeling of Code-Switched Text. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 92–97, Melbourne, Australia. Association for Computational Linguistics.

Ching-Ting Chang, Shun-Po Chuang, and Hung-Yi Lee. 2019. Code-Switching Sentence Generation by Generative Adversarial Networks and its Application to Data Augmentation. In *Proc. Interspeech 2019*, pages 554–558.

Tanmay Chavan, Omkar Gokhale, Aditya Kane, Shantanu Patankar, and Raviraj Joshi. 2023. My Boli: Code-mixed Marathi-English Corpora, Pretrained Language Models and Evaluation Benchmarks. In *Findings of the Association for Computational Linguistics: IJCNLP-AACL 2023 (Findings)*, pages 242–249, Nusa Dua, Bali. Association for Computational Linguistics.

Shuguang Chen, Gustavo Aguilar, Anirudh Srinivasan, Mona Diab, and Thamar Solorio. 2022. CALCS 2021 Shared Task: Machine Translation for Code-Switched Data.

Luis Chiruzzo, Marvin Agüero-Torales, Gustavo Giménez-Lugo, Aldo Alvarez, Yliana Rodríguez, Santiago Góngora, and Thamar Solorio. 2023. Overview of GUA-SPA at IberLEF 2023: Guarani-Spanish Code Switching Analysis. *Procesamiento del Lenguaje Natural*, 71(0):321–328.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training Verifiers to Solve Math Word Problems.

Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online (v2020.4)*. Zenodo.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT Sentence Embedding.

Deepak Gupta, Asif Ekbal, and Pushpak Bhattacharyya. 2020. A Semi-supervised Approach to Generate the Code-Mixed Text using Pre-trained Encoder and Transfer Learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2267–2280, Online. Association for Computational Linguistics.

Gualberto A. Guzmán, Joseph Ricard, Jacqueline Serigos, Barbara E. Bullock, and Almeida Jacqueline Toribio. 2017. Metrics for modeling code-switching across corpora. In *Interspeech*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring Massive Multitask Language Understanding.

I-Hung Hsu, Avik Ray, Shubham Garg, Nanyun Peng, and Jing Huang. 2023. Code-Switched Text Synthesis in Unseen Language Pairs. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5137–5151, Toronto, Canada. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian,

Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of Experts.

Simran Khanuja, Sandipan Dandapat, Sunayana Sitaram, and Monojit Choudhury. 2020a. A New Dataset for Natural Language Inference from Code-mixed Conversations.

Simran Khanuja, Sandipan Dandapat, Anirudh Srinivasan, Sunayana Sitaram, and Monojit Choudhury. 2020b. GLUECoS : An Evaluation Benchmark for Code-Switched NLP.

Eunhee Kim. 2006. Reasons and motivations for code-mixing and code-switching. *Issues in EFL*, 4(1):43–61.

Viet Lai, Nghia Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Nguyen. 2023a. ChatGPT beyond English: Towards a comprehensive evaluation of large language models in multilingual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13171–13189, Singapore. Association for Computational Linguistics.

Viet Dac Lai, Chien Van Nguyen, Nghia Trung Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2023b. Okapi: Instruction-tuned Large Language Models in Multiple Languages with Reinforcement Learning from Human Feedback.

Ying Li and Pascale Fung. 2012. Code-Switch Language Model with Inversion Constraints for Mixed Language Speech Recognition. In *Proceedings of COLING 2012*, pages 1671–1680, Mumbai, India. The COLING 2012 Organizing Committee.

Ying Li and Pascale Fung. 2014. Language Modeling with Functional Head Constraint for Code Switching Speech Recognition. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 907–916, Doha, Qatar. Association for Computational Linguistics.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring How Models Mimic Human Falsehoods.

Sin-En Lu, Bo-Han Lu, Chao-Yi Lu, and Richard Tzong-Han Tsai. 2022. Exploring Methods for Building Dialects-Mandarin Code-Mixing Corpora: A Case Study in Taiwanese Hokkien. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6287–6305, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022. MultiCoNER: A Large-scale Multilingual Dataset for Complex Named Entity Recognition. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3798–3809, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Deepthi Mave, Suraj Maharjan, and Thamar Solorio. 2018. Language Identification and Analysis of Code-Switched Social Media Text. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 51–61, Melbourne, Australia. Association for Computational Linguistics.

Giovanni Molina, Fahad AlGhamdi, Mahmoud Ghoneim, Abdelati Hawwari, Nicolas Rey-Villamizar, Mona Diab, and Thamar Solorio. 2016. Overview for the Second Shared Task on Language Identification in Code-Switched Data. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 40–49, Austin, Texas. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob

Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, C. J. Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. GPT-4 Technical Report.

Niraj Pahari and Kazutaka Shimada. 2023. Language Preference for Expression of Sentiment for Nepali-English Bilingual Speakers on Social Media. In *Sixth Workshop on Computational Approaches to Linguistic Code-Switching*.

Braja Gopal Patra, Dipankar Das, and Amitava Das. 2018. Sentiment Analysis of Code-Mixed Indian Languages: An Overview of SAIL_Code-Mixed Shared Task @ICON-2017.

Parth Patwa, Gustavo Aguilar, Sudipta Kar, Suraj Pandey, Srinivas PYKL, Björn Gambäck, Tanmoy Chakraborty, Thamar Solorio, and Amitava Das. 2020. SemEval-2020 Task 9: Overview of Sentiment Analysis of Code-Mixed Tweets.

Qiwei Peng, Yekun Chai, and Xuhong Li. 2024. HumanEval-XL: A multilingual code generation benchmark for cross-lingual natural language generalization. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8383–8394, Torino, Italia. ELRA and ICCL.

Qiwei Peng, Guimin Hu, Yekun Chai, and Anders Søgaard. 2025. Debiasing multilingual llms in cross-lingual latent space. *arXiv preprint arXiv:2508.17948*.

Juan Manuel Pérez, Damián Ariel Furman, Laura Alonso Alemany, and Franco M. Luque. 2022. RoBERTuito: A pre-trained language model for social media text in Spanish. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 7235–7243, Marseille, France. European Language Resources Association.

SHANA POPLACK. 1980. Sometimes i'll start a sentence in spanish y termino en espaÑol: toward a typology of code-switching1. *Linguistics*, 18(7-8):581–618.

Adithya Pratapa, Gayatri Bhat, Monojit Choudhury, Sunayana Sitaram, Sandipan Dandapat, and Kalika Bali. 2018. Language Modeling for Code-Mixing: The Role of Linguistic Theory based Synthetic Data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1543–1553, Melbourne, Australia. Association for Computational Linguistics.

Md Nishat Raihan, Umma Tanmoy, Anika Binte Islam, Kai North, Tharindu Ranasinghe, Antonios Anastasopoulos, and Marcos Zampieri. 2023. Offensive Language Identification in Transliterated and Code-Mixed Bangla. In *Proceedings of the First Workshop on Bangla Language Processing (BLP-2023)*, pages 1–6, Singapore. Association for Computational Linguistics.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null it out: Guarding protected attributes by iterative nullspace projection. *arXiv preprint arXiv:2004.07667*.

Shruti Rijhwani, Royal Sequiera, Monojit Choudhury, Kalika Bali, and Chandra Shekhar Maddila. 2017. Estimating Code-Switching on Twitter with a Novel Generalized Word-Level Language Detection Technique. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1971–1982, Vancouver, Canada. Association for Computational Linguistics.

Bidisha Samanta, Niloy Ganguly, and Soumen Chakrabarti. 2019a. Improved Sentiment Detection via Label Transfer from Monolingual to Synthetic Code-Switched Text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3528–3537, Florence, Italy. Association for Computational Linguistics.

Bidisha Samanta, Sharmila Reddy, Hussain Jagirdar, Niloy Ganguly, and Soumen Chakrabarti. 2019b. A Deep Generative Model for Code-Switched Text.

Royal Sequiera, Monojit Choudhury, and Kalika Bali. 2015. POS Tagging of Hindi-English Code Mixed Text from Social Media: Some Machine Learning

Experiments. In *Proceedings of the 12th International Conference on Natural Language Processing*, pages 237–246, Trivandrum, India. NLP Association of India.

Kushagra Singh, Indira Sen, and Ponnurangam Kumaraguru. 2018a. Language Identification and Named Entity Recognition in Hinglish Code Mixed Tweets. In *Proceedings of ACL 2018, Student Research Workshop*, pages 52–58, Melbourne, Australia. Association for Computational Linguistics.

Kushagra Singh, Indira Sen, and Ponnurangam Kumaraguru. 2018b. A Twitter Corpus for Hindi-English Code Mixed POS Tagging. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 12–17, Melbourne, Australia. Association for Computational Linguistics.

Thamar Solorio, Elizabeth Blair, Suraj Maharjan, Steven Bethard, Mona Diab, Mahmoud Ghoneim, Abdelati Hawwari, Fahad AlGhamdi, Julia Hirschberg, Alison Chang, and Pascale Fung. 2014. Overview for the First Shared Task on Language Identification in Code-Switched Data. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 62–72, Doha, Qatar. Association for Computational Linguistics.

Victor Soto and Julia Hirschberg. 2017. Crowdsourcing Universal Part-of-Speech Tags for Code-Switching. In *Interspeech 2017*, pages 77–81. ISCA.

Victor Soto and Julia Hirschberg. 2018. Joint Part-of-Speech and Language ID Tagging for Code-Switched Data. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 1–10, Melbourne, Australia. Association for Computational Linguistics.

Vivek Srivastava and Mayank Singh. 2020. PHINC: A Parallel Hinglish Social Media Code-Mixed Corpus for Machine Translation. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 41–49, Online. Association for Computational Linguistics.

Igor Sterner and Simone Teufel. 2023. TongueSwitcher: Fine-Grained Identification of German-English Code-Switching. In *Proceedings of the 6th Workshop on Computational Approaches to Linguistic Code-Switching*, pages 1–13, Singapore. Association for Computational Linguistics.

Jörg Tiedemann. 2012. Parallel Data, Tools and Interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models.

Aditya Vavre, Abhirut Gupta, and Sunita Sarawagi. 2022. Adapting Multilingual Models for Code-Mixed Translation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7133–7141, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Changhan Wang, Kyunghyun Cho, and Douwe Kiela. 2018. Code-Switched Named Entity Recognition with Embedding Attention. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 154–158, Melbourne, Australia. Association for Computational Linguistics.

Genta Winata, Alham Fikri Aji, Zheng Xin Yong, and Thamar Solorio. 2023. The Decades Progress on Code-Switching Research in NLP: A Systematic Survey on Trends and Challenges. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 2936–2978, Toronto, Canada. Association for Computational Linguistics.

Genta Indra Winata, Samuel Cahyawijaya, Zihan Liu, Zhaojiang Lin, Andrea Madotto, and Pascale Fung. 2021. Are Multilingual Models Effective in Code-Switching? In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 142–153, Online. Association for Computational Linguistics.

Genta Indra Winata, Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. 2019. Code-Switched Language Models Using Neural Based Synthetic Data from Parallel Sentences. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 271–280, Hong Kong, China. Association for Computational Linguistics.

Genta Indra Winata, Chien-Sheng Wu, Andrea Madotto, and Pascale Fung. 2018. Bilingual Character Representation for Efficiently Addressing Out-of-Vocabulary Words in Code-Switching Named Entity Recognition. In *Proceedings of the Third Workshop*

on Computational Approaches to Linguistic Code-Switching, pages 110–114, Melbourne, Australia. Association for Computational Linguistics.

Jian Yang, Shuming Ma, Dongdong Zhang, ShuangZhi Wu, Zhoujun Li, and Ming Zhou. 2020. Alternating Language Modeling for Cross-Lingual Pre-Training. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9386–9393.

Zheng Xin Yong, Ruochen Zhang, Jessica Forde, Skyler Wang, Arjun Subramonian, Holy Lovenia, Samuel Cahyawijaya, Genta Winata, Lintang Sutawika, Jan Christian Blaise Cruz, Yin Lin Tan, Long Phan, Long Phan, Rowena Garcia, Thamar Solorio, and Alham Aji. 2023. Prompting Multilingual Large Language Models to Generate Code-Mixed Texts: The Case of South East Asian Languages. In *Proceedings of the 6th Workshop on Computational Approaches to Linguistic Code-Switching*, pages 43–63, Singapore. Association for Computational Linguistics.

Ruochen Zhang, Samuel Cahyawijaya, Jan Christian Blaise Cruz, Genta Winata, and Alham Aji. 2023. Multilingual Large Language Models Are Not (Yet) Code-Switchers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12567–12582, Singapore. Association for Computational Linguistics.

A CodeMixBench vs. other benchmarks

As shown in Table 3, LinCE includes four language pairs and five NLP tasks: Language Identification (LID), Part of Speech (POS), Named Entity Recognition (NER), Sentiment Analysis (SA), and Machine Translation (MT). In contrast, GLUECoS covers two language pairs, lacks the MT task, but adds Question Answering (QA) and Natural Language Inference (NLI). Our review of recent code-mixing studies indicates that research extends beyond the language pairs used in LinCE and GLUECoS. Therefore, we expanded to 16 language pairs and introduced tasks better suited for evaluating LLMs, such as Multi-Choice, Math, and Truthfulness, resulting in a total of eight tasks.

B Collected Datasets

In Table 4, we selected and reconstructed 30 datasets from existing open-source projects. To comprehensively evaluate the performance of large models on code-mixing, we aimed to encompass a diverse range of language families and tasks, prioritizing manually annotated datasets. Ultimately, we cover traditional NLP tasks such as Language Identification (LID), Named Entity Recognition (NER), Part-of-Speech tagging (POS), Sentiment Analysis (SA), and Machine Translation (MT), and cover 16 languages from seven language families: Germanic (en, de, nl, fy), Sino-Tibetan (zh, hok), Romance (es), Afro-Asiatic (msa, ea), Indo-Aryan (hi, bn, ne, mr), Dravidian (ta, ml), and Tupian (gn).

B.1 Datasets of LID Task

zh-en Calvillo et al. (2020) collected data from the Chinese Students and Scholars Association Bulletin Board Systems (CSSA BBS) of Pennsylvania State University, Carnegie Mellon University, and the University of Pittsburgh. The dataset consists of posts from bilingual Chinese-English speakers who have studied in the US for several years. The dataset includes 3,022 samples, totalling 37,064 tokens, with 25,092 Chinese tokens, 7,228 English tokens, and 4,744 punctuation tokens.

hok-zh Lu et al. (2022) utilized a rule-based approach to synthesize parallel corpora into a Hokkien-Mandarin code-mixed corpus, ensuring dataset quality through subsequent post-processing steps. The parallel corpora are derived from iCorpus and the Ministry of Education's Taiwanese Southern Min Dictionary (MoeDict). The test set comprises 3,800 code-mixed sentences and 44,022 tokens, with the distribution as follows: 30,941 Hokkien tokens, and 13,081 Mandarin tokens.

hi-en LinCE (Aguilar et al., 2020) constructed the Hindi-English dataset based on Mave et al. (2018) and ICON 2016 competition Sequiera et al. (2015). We utilized the development set, comprising 744 social media posts from Twitter and Facebook, with the following token distribution: Hindi (8,997), English (3,306), language-independent tokens (2,231), mixed (5), named entities (875), unknown (2), foreign words (29), ambiguous (1).

ne-en The Nepali-English corpus, originally introduced by 2014 CALCS (Computational Approaches to Linguistic Code-Switching) workshop (Solorio et al., 2014), has been restructured by LinCE. We use the development set, comprising 1,332 tweets, with the following token distributions: Nepali (5,649), English (8,417), named entities (514), mixed (17), and ambiguous (13).

mr-en We selected the test set from the MeLID dataset developed by Chavan et al. (2023), which includes 1,340 Marathi-English code-switched tweets annotated by four native Marathi speakers. It contains 11,485 Marathi tokens, 2,925 English tokens, and 1,535 tokens from other categories,

Language Pairs	LID	POS	NER	SA	MT	QA	NLI	Multi-Choice	Math	Truthfulness
					LinCE					
Spanish-English	✓	✓	✓	✓	✓	-	-	-	-	-
Hindi-English	✓	✓	✓	-	✓	-	-	-	-	-
Nepali-English	✓	-	-	-	-	-	-	-	-	-
MS Arabic-Egyptian Arabic	✓	-	✓	-	✓	-	-	-	-	-
					GLUECoS					
Spanish-English	✓	✓	✓	✓	-	-	-	-	-	-
Hindi-English	✓	✓	✓	✓	-	✓	✓	-	-	-
					CodeMixBench					
Spanish-English	✓	✓	✓	✓	✓	-	-	✓	✓	✓
Hindi-English	✓	✓	✓	✓	✓	-	-	✓	✓	✓
Nepali-English	✓	-	-	✓	-	-	-	✓	-	-
MS Arabic-Egyptian Arabic	✓	-	✓	-	-	-	-	-	-	-
Arabic-English	-	-	-	-	✓	-	-	✓	✓	✓
Chinese-English	✓	✓	-	-	✓	-	-	✓	✓	✓
Bengali-English	-	-	-	✓	✓	-	-	✓	-	-
Marathi-English	✓	-	-	-	✓	-	-	✓	-	-
Tamil-English	-	-	-	✓	-	-	-	✓	-	-
Malayalam-English	-	-	-	✓	-	-	-	-	-	-
French-English	-	-	-	-	-	-	-	✓	-	-
Dutch-English	-	-	-	-	-	-	-	✓	-	-
German-English	✓	-	-	-	-	-	-	✓	-	-
Frisian-Dutch	✓	✓	-	-	-	-	-	-	-	-
Hokkien-Chinese	✓	-	-	-	✓	-	-	-	-	-
Guarani-Spanish	✓	-	✓	-	-	-	-	-	-	-

Table 3: **Overview of the CodeMixbench language pairs and tasks compared to LinCE and GLUECoS.**

intended to facilitate research in language identification tasks.

es-en The Spanish-English corpus was obtained from the 2016 CALCS workshop (Molina et al., 2016). LinCE provided new splits for this corpus, and we employ the development set, which comprises 3,332 tweets and 40,391 tokens. The token distribution in the development set is as follows: English tokens (16,712), Spanish tokens (14,955), language-independent tokens (7,830), tokens mixed in English and Spanish (6), named entities (815), unknown (32), foreign words (2), and ambiguous (39).

msa-ea LinCE restructured the Modern Standard Arabic (MSA)-Egyptian Arabic (EA) corpus from the 2016 CALCS workshop (Molina et al., 2016). We choose the development set, comprising 1,332 tweets, with the following token distributions: MSA (13,317), EA (4,100), language-independent tokens (1,707), named entities (2,688), mixed (2), and ambiguous (164).

de-en The TONGUESWITCHER (Sterner and Teufel, 2023) project offers a substantial corpus of 25.6 million German-English code-switched tweets, annotated using both rule-based and neural network methods. We use the test set of the dataset which contains 1,252 tweets and 37,511 tokens, We utilize the test set from the dataset, comprising 1,252 tweets and 37,511 tokens: 34,190 in German, 3,175 in English, and 146 in German-English code-switching.

fy-nl This dataset originated from broadcasts by Omrop Fryslân (Frisian Broadcasting Company), comprising approximately 18.5 hours of spontaneous interviews. Braggaar and van der Goot (2021) randomly selected and annotated 400 utterances with LID tags. Among these, 67.8% of the words are in Frisian, 26.1% are in Dutch, and the rest comprise a mix of Frisian-Dutch, hesitation markers (e.g., "eh"), or other languages. We use the test subset of the dataset, comprising 280 samples, with the following token counts: Dutch (378), Frisian (1,955), Frisian-Dutch (15), and other tokens (8).

gn-es Chiruzzo et al. (2023) built this dataset from news articles and tweets. It consists of approximately 25,000 tokens and is annotated in two stages by six annotators proficient in Spanish and with some knowledge of Guarani. We utilized the test set comprising 180 sentences and 2,857 tokens,

	Languages	Size	All Tokens	M-index	I-index	Source
LID	zh-en	3,022	37,064	0.538	0.399	Calvillo et al. (2020)
	hok-zh	3,800	44,022	0.557	0.173	Lu et al. (2022)
	hi-en	744	15,446	0.224	0.137	Mave et al. (2018)
	ne-en	1,332	19,273	0.388	0.220	Solorio et al. (2014)
	mr-en	1,340	15,945	0.347	0.241	Chavan et al. (2023)
	es-en	1,133	40,391	0.160	0.077	Molina et al. (2016)
	msa-ea	1,116	21,978	0.073	0.031	Molina et al. (2016)
	de-en	1,252	37,511	0.232	0.077	Sterner and Teufel (2023)
	fy-nl	250	2,356	0.381	0.278	Braggaar and van der Goot (2021)
	gn-es	180	2,857	0.558	0.327	Chiruzzo et al. (2023)
POS	zh-en	2,909	35,600	-	-	Calvillo et al. (2020)
	hi-en	160	3,476	-	-	Singh et al. (2018b)
	es-en	1,000	7,712	-	-	Soto and Hirschberg (2017)
	fy-nl	250	2,356	0.381	0.278	Braggaar and van der Goot (2021)
NER	hi-en	314	5,364	-	-	Singh et al. (2018a)
	es-en	1,000	12,139	-	-	Aguilar et al. (2018)
	msa-ea	1,122	22,742	-	-	Aguilar et al. (2018)
	gn-es	180	2,857	0.558	0.327	Chiruzzo et al. (2023)
SA	hi-en	1,261	-	-	-	Patra et al. (2018)
	bn-en	1,000	-	-	-	Raihan et al. (2023)
	mr-en	1,250	-	-	-	Chavan et al. (2023)
	ne-en	1,070	-	-	-	Pahari and Shimada (2023)
	es-en	1,859	28,202	-	-	Patwa et al. (2020)
	ta-en	3,049	-	-	-	Chakravarthi et al. (2020b)
	ml-en	1,171	-	-	-	Chakravarthi et al. (2020a)
MT	zh-en->zh	3,022	37,064	0.538	0.399	Calvillo et al. (2020)
	hok-zh->zh	3,800	44,022	0.557	0.173	Lu et al. (2022)
	hi-en->en	942	11,849	0.90	0.53	Chen et al. (2022)
	bn-en->en	2,000	-	-	-	Vavre et al. (2022)
	mr-en->en	2,000	-	-	-	Vavre et al. (2022)

Table 4: **The statistics of collected datasets.**

categorized as follows: 1,193 Guarani tokens, 815 Spanish tokens, 47 mixed-language tokens, 8 foreign words, 331 named entities, and 463 tokens classified as other.

B.2 Datasets of POS Task

zh-en In the zh-en dataset in the LID task, we introduced the dataset built by Calvillo et al. (2020), which is based on the CSSA BBS. They employed the Stanford Parser to obtain POS tags for code-mixed sentences. We selected a dataset consisting of 2,909 sentences and 35,600 tokens. The distribution of POS tags is as follows: NN (9,990), PU (5,880), VC (604), CD (1,691), M (1,207), JJ (710), P (736), MSP (41), VV (5,331), VA (924), VE (716), DEG (677), CC (461), AD (3,236), PN (788), DT (493), NT (273), LC (324), DEC (447), SP (250), OD (67), NR (359), ETC (60), CS (87), AS (180), DER (11), SB (10), BA (16), URL (1), DEV (13), IJ (15), and LB (2).

hi-en LinCE proposed standard splits for a dataset comprising 1,489 tweets (33,010 tokens) annotated with POS tags (Singh et al., 2018b). We select a development set of 160 tweets, with the following token counts per POS category:

- X (790) for all other categories such as abbreviations or foreign words.
- VERB (669) is used for verbs.

- NOUN (516) is used for nouns.
- ADP (346) is used for prepositions and postpositions.
- PROPN (271) is used for proper nouns.
- ADJ (170) is used for adjectives.
- PRON (159) is used for pronouns.
- PART (145) is used for particles.
- DET (116) is used for determiners and articles.
- ADV (100) is used for adverbs.
- CONJ (77) is used for coordinating conjunctions. This is represented by 'CCONJ' in the universal POS tagset.
- PART_NEG (43) is used for indicating negation.
- PRON_WH (39) is used for interrogative pronouns (like where, why, etc.).
- NUM (35) is used for numerals.

es-en The Spanish-English dataset is derived from the Miami Bangor corpus (Soto and Hirschberg, 2017). LinCE stratified the dataset into training (27,893 sentences), development (4,298 sentences), and testing (10,720 sentences) sets. From the development set, 1,000 samples (totalling 7,712 tokens) were randomly selected. The token counts per part-of-speech tag are as follows: VERB (1,262), PUNCT (1,234), PRON (1,189), NOUN (676), DET (552), ADV (498), ADP (472), INTJ (362), CONJ (278), ADJ (254), AUX (243), SCONJ (238), PART (165), PROPN (150), NUM (86), and UNK (53).

fy-nl Braggaar and van der Goot (2021) also annotated 400 broadcast utterances with POS tags. We utilize the test set, which includes 280 samples. The token counts for each POS tag in this subset are as follows: NOUN (310), ADP (288), PRON (285), ADV (284), VERB (263), DET (232), PROPN (154), ADJ (142), AUX (111), INTJ (105), CCONJ (101), SCONJ (41), and NUM (40).

B.3 Datasets of NER Task

hi-en Singh et al. (2018a) developed a dataset of 2,079 tweets annotated by three linguistic experts, and subsequently splits by LinCE. From this dataset, we selected a development set of 314 tweets, comprising 5,364 tokens. The token distribution includes 4,789 O tokens, 61 B-ORGANISATION tokens, 19 I-ORGANISATION tokens, 254 B-PERSON tokens, 112 I-PERSON tokens, 105 B-PLACE tokens, and 24 I-PLACE tokens.

es-en The Spanish-English corpus, introduced at the 2018 CALCS workshop (Aguilar et al., 2018) for NER, was used fairly splited by LinCE. We randomly sample 1,000 instances from the development set, comprising a total of 12,139 tokens. The distribution of entity tokens is as follows: 11,834 O tokens, 82 B-PER tokens, 25 I-PER tokens, 21 B-PROD tokens, 3 I-PROD tokens, 47 B-LOC tokens, 18 I-LOC tokens, 7 B-TIME tokens, 4 I-TIME tokens, 12 B-ORG tokens, 13 I-ORG tokens, 5 B-EVENT tokens, 7 I-EVENT tokens, 14 B-TITLE tokens, 18 I-TITLE tokens, 14 B-GROUP tokens, 7 I-GROUP tokens, 7 B-OTHER tokens, and 1 I-OTHER token.

msa-ea This MSA-EA corpus was also introduced at the 2018 CALCS workshop. We utilized the development set, comprising 1122 samples with a total of 22742 tokens. The token distribution is as follows: O tokens: 20,031, B-PER tokens: 698, I-PER tokens: 415, B-GROUP tokens: 191, I-GROUP tokens: 112, B-LOC tokens: 358, I-LOC tokens: 116, B-PROD tokens: 55, I-PROD tokens: 26, B-ORG tokens: 149, I-ORG tokens: 114, B-TITLE tokens: 115, I-TITLE tokens: 143, B-EVENT tokens: 69, I-EVENT tokens: 52, B-TIME tokens: 61, I-TIME tokens: 18, B-OTHER tokens: 17, and I-OTHER tokens: 2.

gn-es In LID task, we introduce the Guarani-Spanish dataset constructed by Chiruzzo et al. (2023). In addition to LID labels, this dataset contains manually annotated NER tags. We selected the test set, which comprises the following token counts: 2526 overall tokens, 81 B-PER tokens, 89 B-ORG tokens, 34 I-PER tokens, 33 B-LOC tokens, 21 I-LOC tokens, and 73 I-ORG tokens.

B.4 Datasets of SA Task

hi-en Patra et al. (2018) built this Hindi-English dataset, derived from the social media platform

Twitter, has been manually annotated for sentiment, encompassing positive, negative, and neutral labels. For our study, we utilized a test set comprising 1,261 samples, which includes 385 positive, 290 negative, and 586 neutral instances.

bn-en The TB-OLID dataset (Raihan et al., 2023), designed for offensive language detection in code-mixed texts, comprises 5,000 Facebook comments, with English constituting 38.42% of the content. All comments are manually annotated. For our benchmark, we utilized the test subset of 1,000 instances, consisting of 573 non-offensive and 427 offensive comments.

mr-en Chavan et al. (2023) also provided a Marathi-English dataset with manually annotated sentiment labels. We selected the test set containing 1,250 instances, distributed as 417 positive, 417 negative, and 416 neutral samples.

ne-en The dataset consists of code-switched Nepali-English comments from YouTube, intended for sentiment analysis with manually annotated labels (Pahari and Shimada, 2023). The test set we used includes 1,070 samples, distributed as follows: 346 Positive, 359 Negative, and 365 Neutral.

es-en We used the development set from the Spanish-English corpus provided in the SentiMix competition (Patwa et al., 2020), partitioned by LinCE. This set includes 1,859 instances, categorized as follows: 1,037 Positive, 305 Negative, and 517 Neutral.

ta-en The TamilMixSentiment (Chakravarthi et al., 2020b) dataset consists of manually annotated Tamil-English code-mixed comments from YouTube. The test set, which we utilized, comprises 3,049 instances with the following distribution: 2,075 Positive, 424 Negative, 173 Neutral, and 377 Mixed feelings.

ml-en Chakravarthi et al. (2020a) curated this Malayalam-English dataset from comments on 2019 Malayalam movie trailers on YouTube, with sentiment annotations performed by at least three trained annotators. We employed their test set, which includes 1171 instances: 565 Positive, 138 Negative, 398 Neutral, and 70 Mixed feelings.

B.5 Datasets of MT Task

zh-en → zh Calvillo et al. (2020) employed five bilingual Chinese-English speakers to translate 3,022 sentences from the previously introduced zh-en dataset in the LID task into Chinese. These translators, international Chinese undergraduates, match the language proficiency and cultural background of the CSSA BBS forum users.

hok-zh → zh Given that the Hokkien-Mandarin dataset is synthesized from parallel corpora (Lu et al., 2022), it allows for the straightforward construction of a translation task utilizing both the synthesized data and the original data. As a result, we have developed a dataset comprising 3,800 samples, facilitating the translation of Hokkien-Mandarin into Mandarin.

hi-en → en Chen et al. (2022) created a translation task from English to Hinglish at the 2021 CALCS workshop, using a subset of the CMU Document Grounded Conversations dataset. We utilized its development set and converted it into a Hinglish-to-English translation task, comprising 942 instances.

bn-en → en Vavre et al. (2022) proposed a dataset for translating Bengali-English texts to English, sourced from the Spoken Tutorial project. This dataset includes transcriptions from video lectures collected from the Spoken Tutorial educational website, as well as parallel sentences from the Samanantar project and other sources. On average, each sentence contains 11.32 Bengali tokens and 13.31 English tokens. We selected the ST-Hard subset for testing, which comprises 2000 sentences where the baseline model performed the poorest.

mr-en → en Vavre et al. (2022) also introduced a Marathi-English code-mixed to English translation task, sourced from the Spoken Tutorial project. Each sentence in this dataset averages 11.32 Marathi tokens and 13.00 English tokens. We similarly selected the ST-Hard subset, containing 2000 sentences.

C Automatic LID Annotation

Our method for word-level Language Identification annotation is simple and effective, utilizing the GPT-4 Turbo model without relying on extra dictionaries. Based on the parallel sentences (*L1*, *L2*), we instruct the model to replace tokens from *L1* with corresponding tokens from *L2*, to synthesize code-mixed sentences (*CM*). We identify tokens' LID tags as follows: tokens from *L1* not present in *L2* are marked as the first language, tokens from *L2* not present in *L1* are marked as the second lan-

Language Pair	CM & L1	CM & L2	L1 & L2
zh-en	0.958	0.936	0.914
hi-en	0.951	0.918	0.887
bn-en	0.938	0.907	0.883
mr-en	0.910	0.894	0.854
ne-en	0.929	0.905	0.879
es-en	0.972	0.967	0.939
fr-en	0.974	0.962	0.935
ar-en	0.956	0.941	0.911
ta-en	0.892	0.873	0.838
nl-en	0.967	0.952	0.923
de-en	0.968	0.945	0.914

Table 5: **LaBSE scores for synthetic code-mixed data across different language pairs.** $L1$ indicates non-English languages. $L2$ denotes English. CM denotes synthesized code-mixing data.

guage, and if tokens belonging to both *L1* and *L2* we consider this token to be language-independent and mark it as "other". This approach is particularly effective for languages with distinct character sets. However, for languages sharing the same script, such as English and French, this method may inaccurately label shared tokens as "other". To resolve this issue, we also instruct the model to return all the replaced tokens, forming set *X*. If a token in the code-mixed sentence comes from *X*, we mark it as the second language. This automatic annotation technique is suitable for large-scale multi-language annotation tasks. We designed regular expressions to tokenize sentences into words, and for Chinese text, we use the Jieba[2] tokenizer.

D Semantic Filtering using LaBSE

The Language-agnostic BERT Sentence Encoder (LaBSE) is a BERT-based model trained for sentence embeddings in 109 languages. As shown in Table 5, LaBSE scores are high because GPT generated code-mixed sentences by replacing corresponding parts in parallel sentences, maintaining their original structure with minor linguistic changes. Our experiments demonstrated LaBSE's stability in computing semantic similarity scores for code-mixed sentences. We also sampled 20 examples from each of the 11 language pairs in CM-MMLU and manually verified LaBSE's evaluation of the synthetic data. Our manual reviews align closely with LaBSE's high scores, likely because the synthetic data was generated using simple word substitution by a powerful GPT model, which minimally impacted the source text's semantics. Consequently, we used LaBSE for batch evaluation of synthetic data quality.

[2]https://github.com/fxsjy/jieba

E Model Aligned Filtering using GPT-4

We employed the robust GPT-4 turbo, incorporating detailed scoring guidelines and the Chain-of-Thought (CoT) methodology within the prompt (see Appendix F.2) to guide the model in performing analysis before assigning a final score, thereby enhancing the reliability of the assessment. We use the Mean Absolute Percentage Error (MAPE) formula to compute the differences between GPT and human scores across three dimensions: coherence, naturalness, and readability, where n equals 3.

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\text{GPT\_score}_i - \text{human\_score}_i}{\text{human\_score}_i}\right|$$

$$\text{Agreement} = 1 - \text{MAPE}$$

We sampled 20 instances from each language pair in the CM-MMLU dataset and manually reviewed the model-aligned evaluation results, achieving a 91.4% average agreement rate. Table 6 displays four randomly selected examples. While GPT cannot fully replace human evaluators, it can process large volumes of data in batches and achieve a high degree of consistency with human assessments.

F Prompts for Building CodeMixBench

F.1 Synthesis Prompt

```
You will receive a pair of parallel mo-
nolingual sentences. Randomly replace
certain words or phrases in the first
sentence with corresponding parts from
the second sentence to synthesize them
into a codemixed sentence. Finally, ou-
tput the code-mixed sentence and words
or phrases you have replaced in the
following format:

Code-mixed sentence: . . .
Replaced parts:
<<<word/phrase-->word/phrase>>>,
<<<word/phrase-->word/phrase>>>, . . .
```

F.2 Model-Aligned Filtering Prompt

```
You will be presented with a code-mixed
sentence. Your task is to evaluate the
sentence based on three separate met-
rics. Assuming the readers are people
familiar with each language in the se-
```

Index	Code-Mix	Chain of Thought Analysis	GPT-4	Human
1	_____ 在向目标受众传递信息方面非常 effective, 因为它可以以大多数其他 media 无法做到的方式进行 explanation。(A): Digital 媒体。(B): 印刷媒体。(C): Broadcast 媒体。(D): 户外 media。	The multi-choice question integrates English and Chinese in a manner that maintains logical flow and coherence, especially for readers familiar with both languages. The use of English words such as "effective," "media," and "explanation" within a predominantly Chinese sentence structure does not disrupt the overall understanding. This indicates a good level of coherence. The naturalness of the code-mixing might vary depending on the reader's fluency in both languages, but given the context of media and communication, the use of English terms feels appropriate and mirrors real-world usage in bilingual or multilingual environments. Therefore, the naturalness is fairly high. The readability is also good, as the structure of the question follows a logical pattern, and the code-mixing does not obscure the meaning. The bilingual reader should find the question accessible, with the English terms likely being familiar concepts within the context of media studies.	3, 2, 3	3, 2, 3
2	在膳食脂肪酸类中，which have the most beneficial effect on plasma LDL cholesterol levels?(A): n-3和n-6 polyunsaturated fatty acids (B): 饱和脂肪酸和trans unsaturated fatty acids (C): Monounsaturated fatty acids (D): Both a和c	The multi-choice question integrates English and Chinese in a manner that maintains logical flow and coherence, especially for readers familiar with both languages. The terminology related to dietary fats and cholesterol levels is presented in a way that leverages English scientific terms within a Chinese sentence structure, which is a common practice in scientific communities with bilingual speakers. This integration feels natural to those accustomed to discussing specialized topics in a code-mixed language environment. The readability is somewhat impacted by the switch between languages, especially for readers who may not be equally proficient in both, but the overall structure and context help in understanding the question's intent.	3, 3, 2	3, 3, 3
3	定价方法根据顾客所感知的价值来设定价格的是: (A): Cost-oriented approach。 (B): 需求导向的方法。 (C): Competitor-oriented approach。 (D): 价值导向的方法。	The multi-choice question integrates English and Chinese languages, focusing on pricing strategies. The terms "Cost-oriented approach," "Competitor-oriented approach," and the Chinese equivalents for "demand-oriented method" and "value-oriented method" are mixed, targeting an audience familiar with business terminology in both languages. The question maintains coherence by clearly presenting the options related to pricing methods, despite the language switch. The naturalness is somewhat affected by the direct mixing of English and Chinese terms, which might not be standard practice in a purely academic or professional setting but could be natural in a bilingual educational context. The readability is fairly high, as the structure of the question is straightforward, and the bilingual reader can easily understand the options provided, even though they switch between languages.	2, 2, 2	3, 2 ,3
4	语句1 | 传统机器学习结果假定训练集和测试集是独立且同分布的。 语句2 | In 2017, COCO models were usually pretrained on ImageNet. (A): 正确，True (B): 错误，False (C): 正确，False (D): 错误，True	The multi-choice question integrates two languages (Chinese and English) to discuss a concept related to machine learning and model pretraining. The transition between the two languages is smooth, given the context is academic and technical, where English terms are commonly used in non-English speaking countries due to the global nature of the field. The statements are connected logically, discussing assumptions in machine learning and practices in model pretraining, which are relevant to each other. The use of both languages seems intentional and reflects a natural way of discussing specialized topics in a bilingual setting. The readability is high, assuming the audience is familiar with both languages and the technical terms used. The options provided mix languages for "correct" and "incorrect", which might slightly challenge readability but remains within the context of code-mixing practices.	3, 3, 2	3, 3, 2

Table 6: **Four sampling validation examples in Model-Aligned Evaluation.** The scores in both the GPT-4 and Human columns are arranged in the order of Coherence, Naturalness, and Readability.

ntence.

Evaluation Criteria:
Coherence (1-3): Assesses how well the sentence elements are connected and flow together, considering the mixing of languages.
1: Poor. The sentence lacks logical flow or connection between its parts, making it hard to understand.
2: Fair. The sentence has some logical connections between its parts, but the flow might be interrupted by awkward language mixing.
3: Good. The sentence demonstrates a clear and logical connection between its parts, with the mixing of languages not hindering understanding.

Naturalness (1-3): Evaluate the sentence for its natural-sounding language use and integration of the code-mixed elements.
1: Poor. The sentence sounds unnatural or forced, with the mixing of languages seeming out of place.
2: Fair. The sentence sounds somewhat natural, though the integration of different languages can occasionally feel awkward.
3: Good. The sentence sounds natural and the mixing of languages appears seamless and intentional.

Readability (1-3): Measures how easy it is to read and understand the sentence, considering the impact of code-mixing on readability.
1: Poor. The sentence is difficult to read, with the mixing of languages significantly hindering comprehension.
2: Fair. The sentence is readable, though the reader may need to pause to understand the mixed languages.
3: Good. The sentence is easy to read, with the code-mixing enhancing or not detracting from the ability to understand the content.

Output your evaluation following this format:
Concise and refined evaluation analysis: . . .
Scores (only scores): coherence score, nat-uralness score, readability score.

G Prompts of Experiment

G.1 Prompt of LID, POS, NER Task

You are a smart and intelligent [INPUT TASK] system. You will receive a tokenized sentence code-mixed with [INPUT FIRST LANGUAGE] and [INPUT SECOND LANGUAGE]. Label each token in the tokenized sentence based on the categories: [tag_1, tag_2, . . . , tag_k]

You must tag every token in the tokenized sentence in order, without skipping or missing any token for any reason. Fill in this JSON format: [{specific token_1: tag_k}, {specific token_2: tag_k}].

Please refer to the example:
Tokenized sentence: [INPUT A CODE-MIXED SENTENCE]
Your answer: [JSON FORMAT].

Tokenized sentence: [INPUT A CODE-MIXED SENTENCE];
Your answer:

G.2 Prompt of SA Task

You are a smart and intelligent sentiment analysis (SA) system. I will give you a code-mixed sentence that has been mixed with [INPUT FIRST LANGUAGE] and [INPUT SECOND LANGUAGE]. Assign the appropriate label from: [tag_1, tag_2, . . . , tag_k].

Please refer to the example:
Sentence: [INPUT A CODE-MIXED SENTENCE]
Your answer: [INPUT A TAG]

Sentence: [INPUT A CODE-MIXED SENTENCE]
Your answer:

G.3 Prompt of MT Task

You will receive a sentence code-mixed with [INPUT FIRST LANGUAGE] and [INPUT SECOND LANGUAGE]. Translate the given sentence into [INPUT TARGET LANGUAGE].

```
Please refer to the example:
Sentence: [INPUT A CODE-MIXED SENTENCE]
Your answer: [INPUT TARGET SENTENCE]

Sentence: [INPUT A CODE-MIXED SENTENCE]
Your answer:
```

G.4 Prompt of CM-MMLU Task

```
You are a system possessing knowledge
in all subjects. You are skilled at
selecting the correct answer based on
multiple-choice questions. Do not in-
clude explanations in your answer.

(k-shot setting here)
Question: [INPUT MULTIPLE-CHOICE QUE-
STION]
Answer: [INPUT ANSWER]
...

Question: [INPUT MULTIPLE-CHOICE QUE-
STION]
Answer:
```

G.5 Prompt of CM-GSM8K Task

```
You are skilled at solving mathemati-
cal problems. Output the solution and
final answer for the next problem.
The solution should include the entire
process of calculating the final ans-
wer. The final answer to the problem
is just one definite numerical value.
Don't output the problem.
Output in this format:
Solution:
Final answer: (one definite numerical
value)

(k-shot setting here)
Problem: [INPUT MATH PROBLEM]
Solution: [INPUT CoT SOLUTION]
Final answer: [INPUT FINAL ANSWER]
...

Problem: [INPUT MATH PROBLEM]
Solution:
Final answer:
```

G.6 Prompt of CM-TruthfulQA Task

```
You are skilled at selecting the cor-
rect answer based on multiple-choice
questions. Do not include explanati-
ons in your answer.

(k-shot setting here)
Question: [INPUT MULTIPLE-CHOICE QUE-
STION]
Answer: [INPUT ANSWER]
...

Question: [INPUT MULTIPLE-CHOICE QUE-
STION]
Answer:
```

H Statistics of Synthetic Datasets

We synthesize CM-MMLU (11 language pairs), CM-GSM8K (4 pairs), CM-TruthfulQA (4 pairs), and MT tasks (3 pairs), detailed in Table 7. We observe that each dataset contains an average of 1,016 samples, with token counts of 24,543, 19,897, and 3,330 for two languages and language-independent tokens (i.e. punctuation, numerals, and formulas), respectively. Both Semantic and Model-Aligned evaluations show high scores. The weighted average M-index across 22 datasets is 0.81, indicating a balanced proportion of the two languages within the text. The average I-index of 0.25 meets our expectations, as a high I-index would not represent realistic code-mixing. Imagining a sentence code-mixed with Chinese and English like "我们 will 走 very 长的 journey, 所以 we 得 bring 足够的 food 和 water" (We will take a very long journey, so we need to bring enough food and water). The sentence has both the M-index and the I-index equal to 1 but is difficult to read and appears unrealistic. For the single dataset, we analyzed the distributions of the M-index and I-index metrics within the dataset. One dataset (*es-en* of CM-MMLU) is illustrated in Figure 5 and others are shown in Figure 6. In summary, our statistical analysis indicates that our synthesized dataset demonstrates sufficient code-mixing between pairs of languages while preserving coherence, naturalness, readability, and a high degree of similarity to the original task sentences. We spent a total cost of $718.45 to construct these datasets.

	Lang.	Size	L1 / L2 / Other tokens	Word-Level		Semantic			Model-Aligned		
				M	I	sim1	sim2	sim3	Co.	Na.	Re.
MMLU	zh-en	1133	32510 / 17765 / 2646	0.75	0.22	0.96	0.94	0.92	2.89	2.52	2.48
	es-en	1146	26303 / 30492 / 3652	0.87	0.31	0.97	0.97	0.94	2.89	2.62	2.56
	fr-en	1107	29412 / 27589 / 3549	0.86	0.27	0.97	0.96	0.94	2.80	2.49	2.42
	de-en	1078	27856 / 22163 / 3701	0.85	0.28	0.97	0.95	0.91	2.79	2.44	2.39
	nl-en	1135	28992 / 26243 / 3551	0.87	0.31	0.97	0.95	0.92	2.85	2.52	2.48
	ar-en	1155	26977 / 18815 / 3346	0.78	0.22	0.96	0.94	0.92	2.85	2.54	2.45
	hi-en	1024	30767 / 19174 / 3417	0.77	0.25	0.95	0.92	0.89	2.93	2.74	2.55
	bn-en	1114	23912 / 22680 / 3667	0.82	0.25	0.93	0.91	0.87	2.86	2.63	2.50
	mr-en	1067	21402 / 21956 / 4380	0.84	0.24	0.93	0.91	0.86	2.81	2.57	2.45
	ne-en	1150	26268 / 21434 / 3737	0.82	0.25	0.93	0.91	0.87	2.83	2.58	2.44
	ta-en	1047	18477 / 23521 / 5570	0.81	0.23	0.97	0.98	0.94	2.76	2.57	2.44
GSM8K	zh-en	825	22244 / 15934 / 3036	0.77	0.19	0.96	0.94	0.92	2.46	2.18	2.21
	es-en	1231	24208 / 26113 / 5902	0.86	0.34	0.98	0.97	0.95	2.44	2.20	2.19
	ar-en	1141	23506 / 20578 / 5229	0.84	0.23	0.96	0.94	0.92	2.26	2.12	2.09
	hi-en	1170	28128 / 22778 / 6285	0.80	0.26	0.96	0.93	0.91	2.51	2.26	2.26
TruthfulQA	zh-en	771	30461 / 15663 / 1589	0.72	0.20	0.97	0.94	0.92	2.80	2.36	2.42
	es-en	799	24467 / 20517 / 1953	0.85	0.31	0.98	0.96	0.94	2.74	2.38	2.36
	ar-en	795	23311 / 16260 / 2810	0.82	0.24	0.97	0.95	0.93	2.77	2.43	2.35
	hi-en	757	28447 / 16764 / 2206	0.75	0.25	0.97	0.93	0.90	2.87	2.67	2.47
MT	zh-en	850	15934 / 9763 / 825	0.78	0.17	0.92	0.89	0.85	2.60	2.44	2.31
	es-en	1059	15047 / 13006 / 1155	0.86	0.29	0.95	0.93	0.89	2.59	2.46	2.29
	ar-en	802	11319 / 8527 / 1063	0.85	0.22	0.93	0.91	0.87	2.50	2.37	2.19
Average		1016	24543 / 19897 / 3330	0.81	0.25	0.96	0.94	0.91	2.72	2.46	2.38

Table 7: **The statistics of synthesized datasets.** The column $Lang.$ indicates the two languages code-mixed in the dataset. The column $Size$ indicates the size of the dataset. The column $L1/L2/Other\ tokens$ shows token counts for the first language, the second language, and other language-independent tokens. In the column $Word\text{-}Level$, M indicates the M-index, and I indicates the I-index. In the column $Semantic$, $sim1$ represents the similarity between code-mixed text and the monolingual text in the first language, $sim2$ the similarity with the text in the second language, and $sim3$ the similarity between the monolingual texts in the first and second languages. In the $Model\text{-}Aligned$ column, $Co.$, $Na.$, and $Re.$ denote coherence, naturalness, and readability respectively.

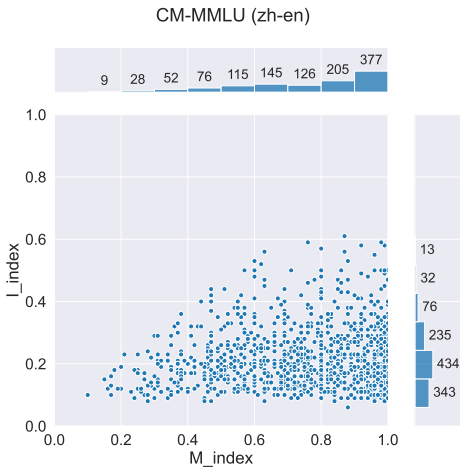


Figure 5: **The distribution of 1133 samples in the code-mixed (zh-en) MMLU.** Two histograms are added around the scatter plot in this figure. The scatter plot displays the M-index and I-index for each sample. The histograms represent the distributions of two metrics.

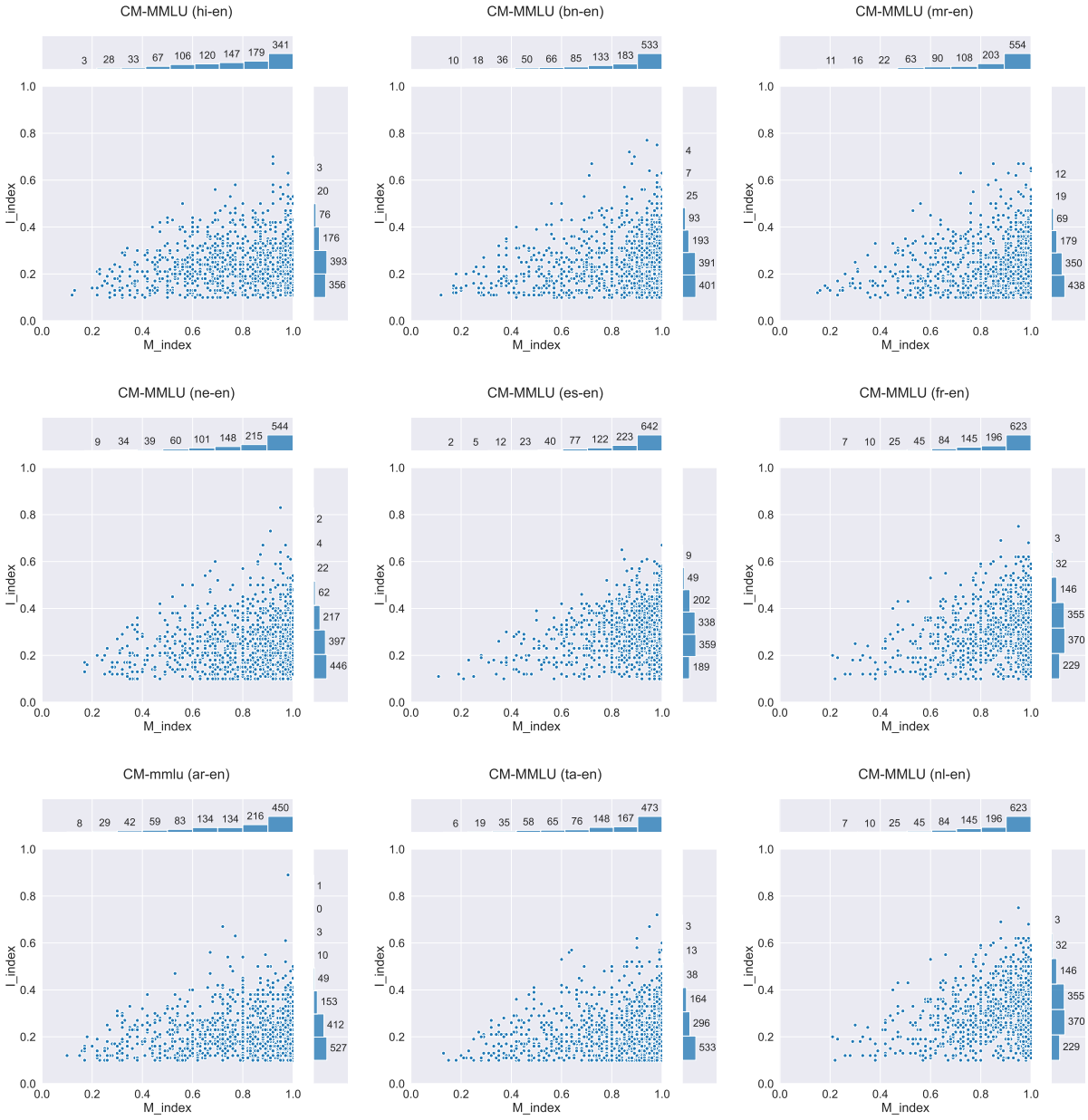


Figure 6: **Distribution of Additional synthetic code-mixing datasets in CodeMixBench.**

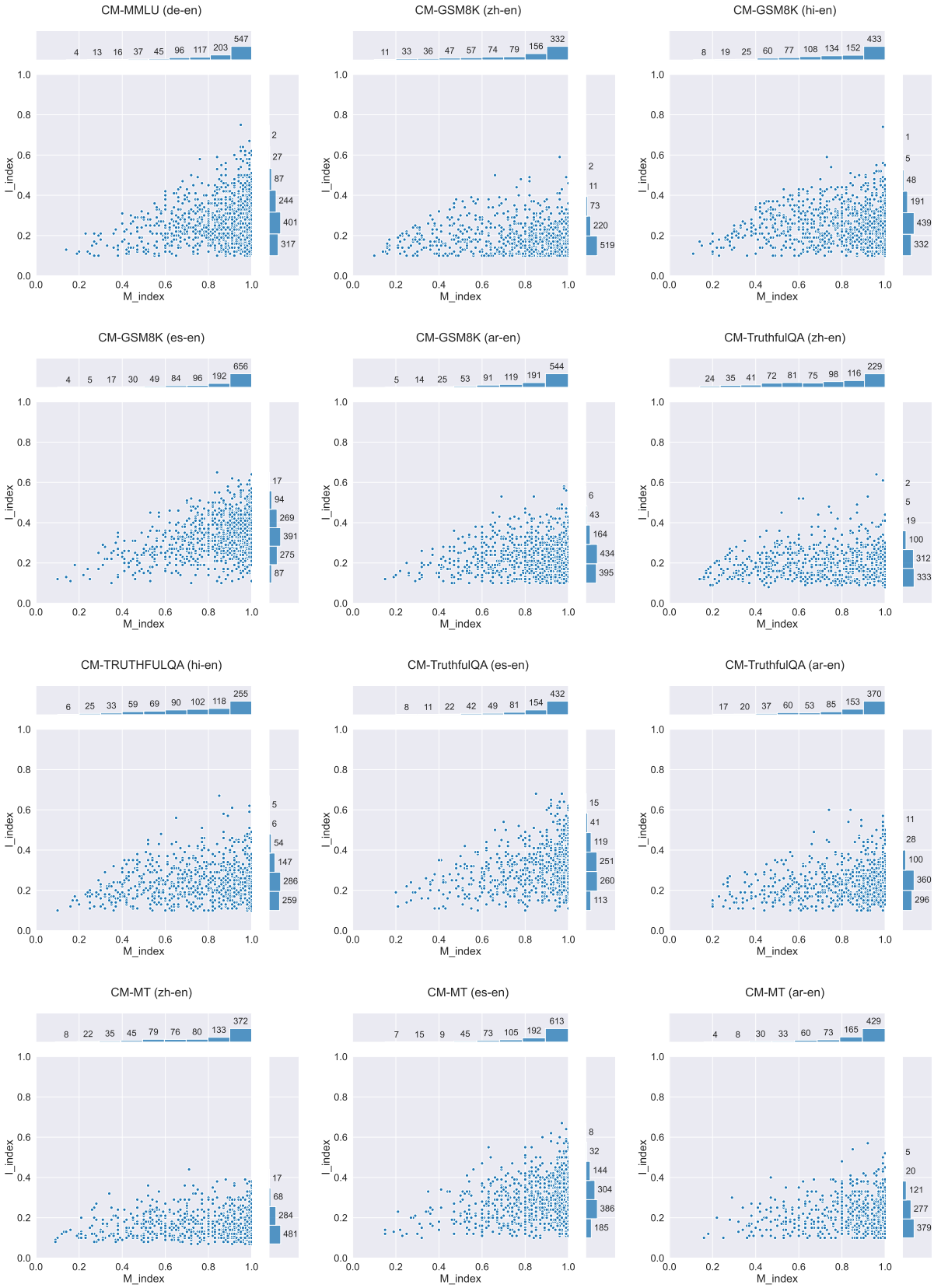


Figure 6: **Distribution of Additional synthetic code-mixing datasets in CodeMixBench.**

I Experiment Results of Collected Datasets

	GPT-3.5-Turbo-Instruct	GPT-3.5-Turbo	GPT-4-Turbo	GPT-4o
	Language Identification (Accuracy)			
zh-en	89.57	**93.38**	93.35	93.31
hok-en	46.43	43.62	45.58	**58.57**
hi-en	75.03	83.41	89.81	**89.84**
ne-en	68.46	84.47	83.63	**85.87**
mr-en	78.87	88.88	89.63	**92.15**
es-en	72.26	85.28	87.47	**88.26**
msa-ea	57.86	68.18	75.26	**76.30**
de-en	71.27	**89.70**	84.45	86.06
fy-nl	62.02	71.11	**77.72**	70.97
gn-es	76.82	85.67	89.02	**89.99**
Average	69.86	79.37	81.59	**83.13**
	Part Of Speech (Accuracy)			
zh-en	71.21	74.83	76.47	**76.91**
hi-en	70.69	70.56	**72.23**	71.70
es-en	81.68	83.02	**89.32**	87.58
fy-nl	79.84	81.73	84.39	**85.62**
Average	75.85	77.53	**80.60**	80.45
	Named Entity Recognition (F1)			
hi-en	79.92	93.56	93.45	**93.82**
es-en	77.12	**92.84**	86.21	92.00
msa-ea	77.95	87.70	**88.12**	86.11
gn-es	86.74	91.59	94.28	**94.51**
Average	80.43	91.42	90.51	**91.61**
	Sentiment Analysis (Accuracy)			
hi-en	61.46	33.78	**66.69**	63.60
bn-en	62.20	53.30	69.90	**76.70**
mr-en	54.88	32.24	**69.52**	60.56
ne-en	59.81	36.07	70.28	**71.68**
es-en	46.21	46.21	**57.18**	50.89
ta-en	51.49	38.70	**55.10**	47.65
ml-en	46.88	37.83	31.77	32.11
Average	54.71	39.73	**60.06**	57.60
	Machine Translation (BLUE)			
zh-en → zh	67.28	68.19	76.69	**79.35**
zh-en → en*	45.47	49.00	**53.21**	52.78
hok-zh → zh	52.92	50.08	60.48	**67.95**
hi-en → en	31.08	30.68	31.17	**32.61**
bn-en → en	16.96	17.99	22.91	**23.59**
mr-en → en	13.46	14.51	18.57	**19.84**
es-en → en*	63.38	65.94	68.20	**68.40**
ar-en → en*	54.35	57.04	61.90	**62.35**
Average	43.11	44.18	49.14	**50.86**

Table 8: **One-shot evaluation of GPT models on LID, POS, NER, SA and MT.** The *Average* represents the mean score of each model across various datasets from a given task. For each model family, the scores of the top-performing models are highlighted in bold. "*" indicates the datasets we synthesized.

J Results of K-shot Experiments across Language Pairs

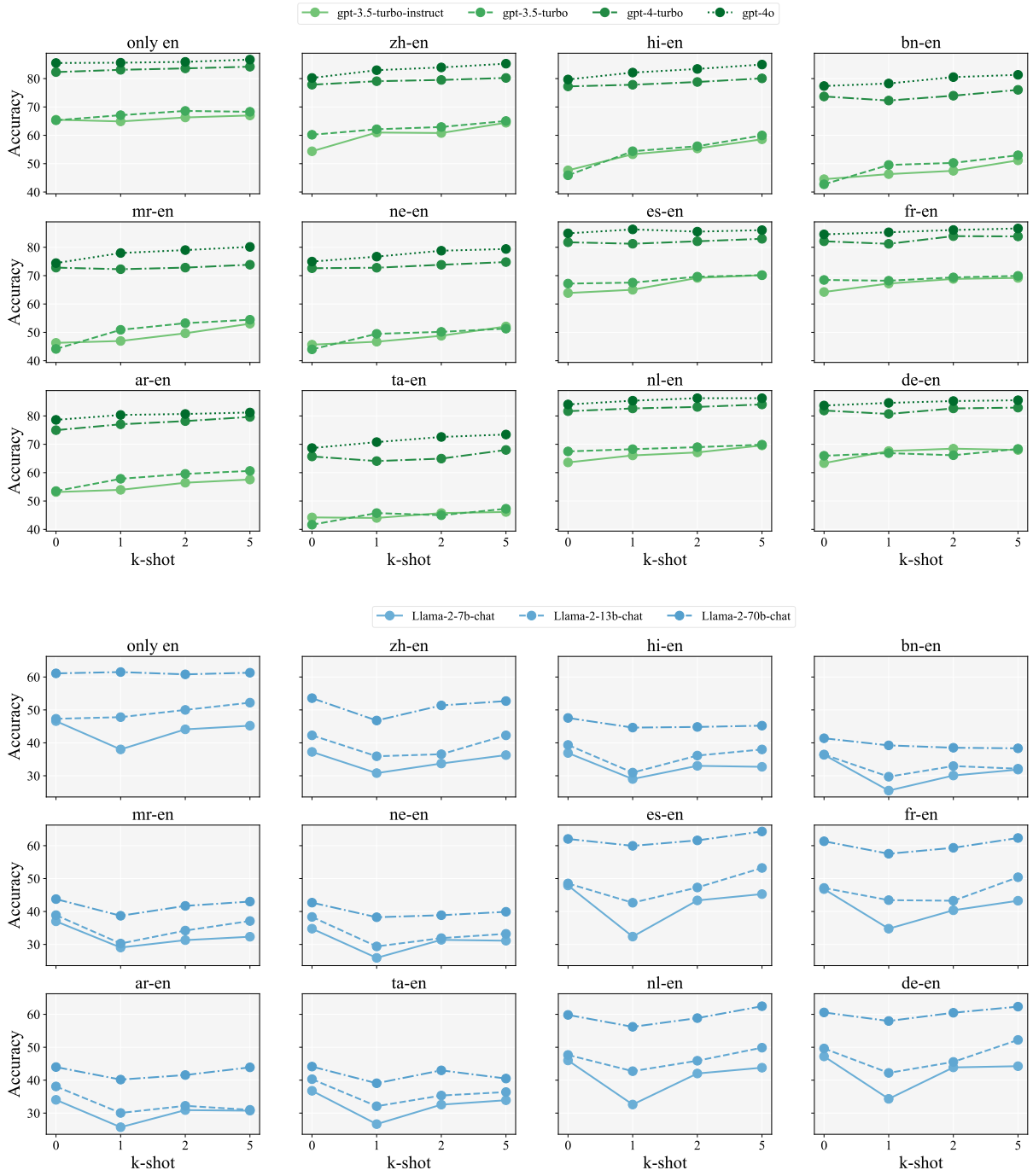


Figure 7: **Accuracy of K-shot evaluation across three model families on CM-MMLU.**

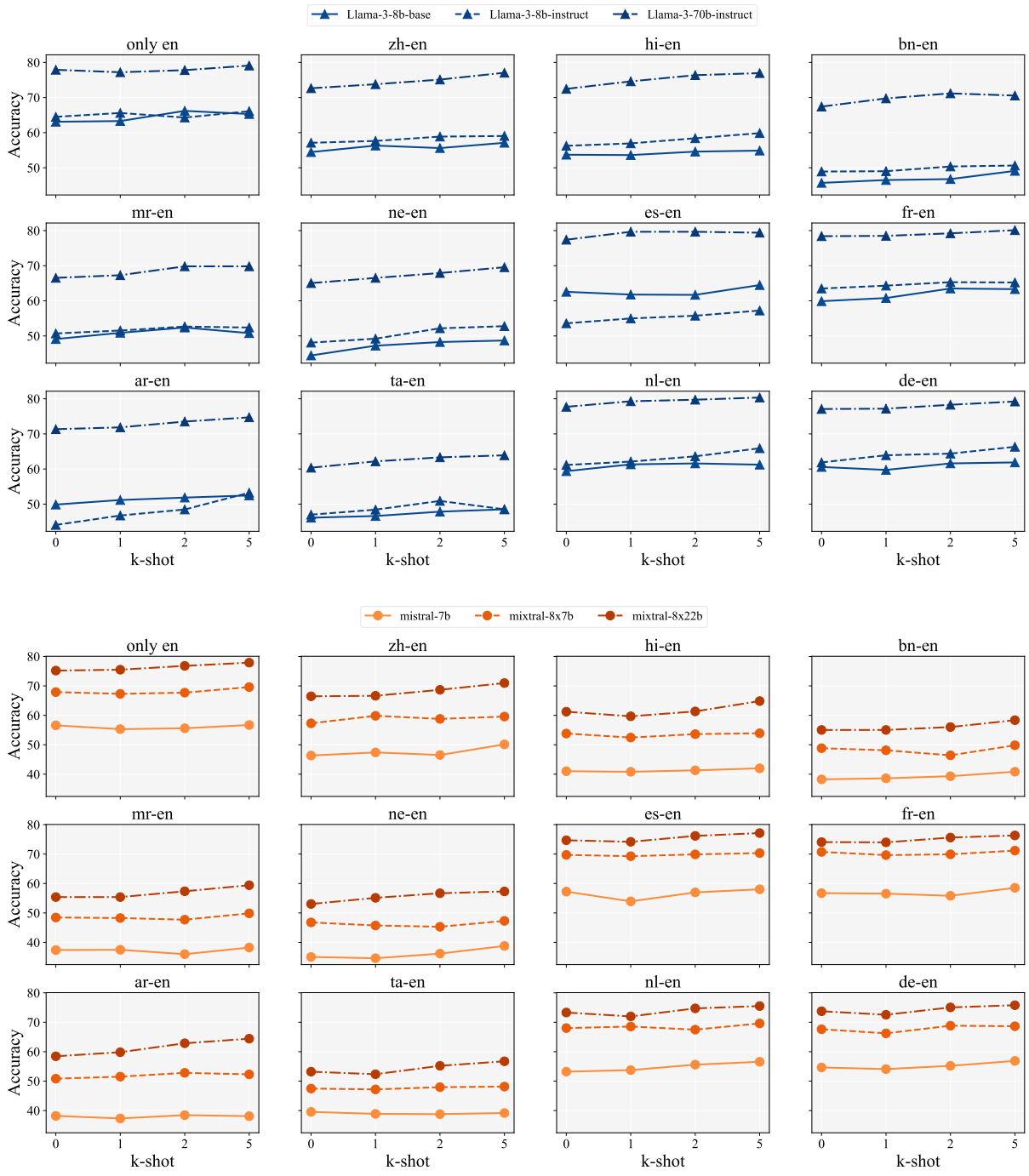


Figure 7: **Accuracy of K-shot evaluation across three model families on CM-MMLU.**

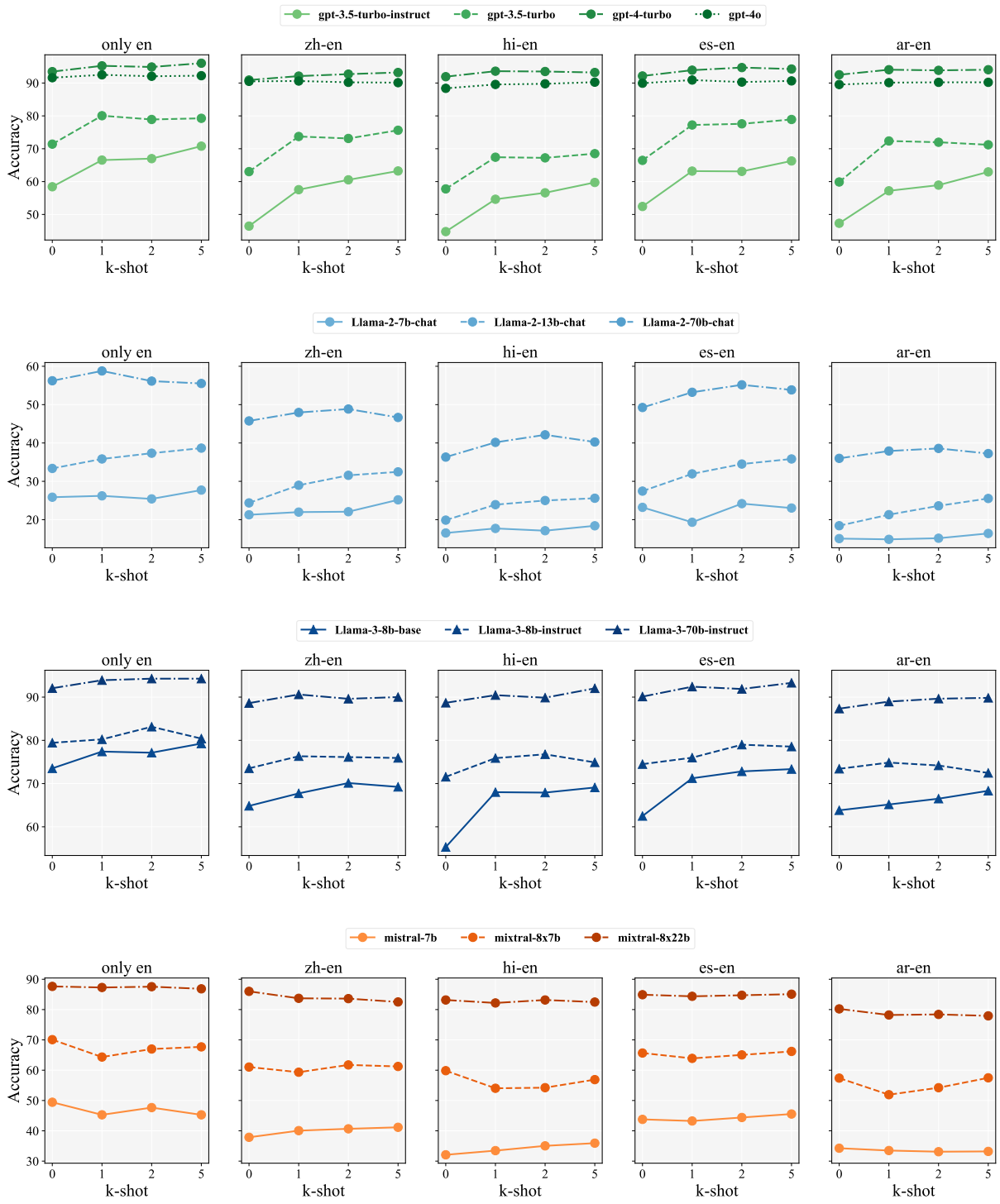


Figure 8: **Accuracy of K-shot evaluation across three model families on CM-GSM8K.**

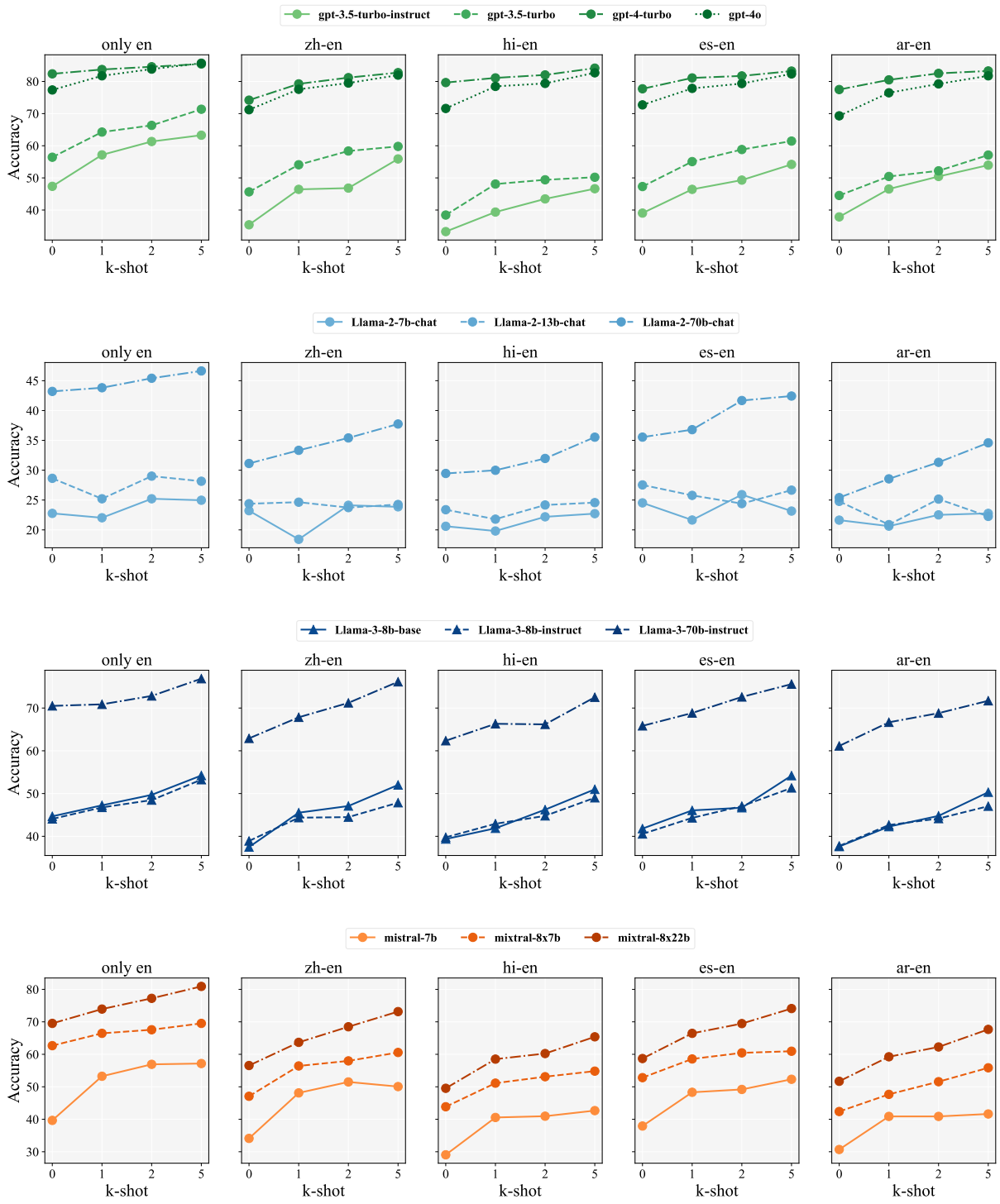


Figure 9: **Accuracy of K-shot evaluation across three model families on CM-TruthfulQA.**