Ai4bharat Assignment
Github Code

Aakash Kumar Agarwal
MS by Research
CSE, IIT Bombay

February 2026

1 Datasets

1.1 Spoken Language

Table 1 and Figure 1 summarize the training statistics for the spoken domain corpora. The table reports overall dataset characteristics, while the figure illustrates the average sentence length in tokens for English and Hindi across different datasets. Together, they provide a concise overview of corpus size and sentence length distribution used in training.

Table 1: Training Data Statistics Summary for Spoken Datasets (Hindi-English)

Dataset	Lang	Sentences	Total Words	Avg Words	Word Std	Min/Max Words	Avg Chars	Char Std
GlobalVoices	English	1,843	30,910	16.77	12.45	0 / 87	99.10	74.49
	Hindi	1,843	32,485	17.63	13.24	0 / 84	90.35	67.57
IITB	English	1,161,358	15,149,131	13.04	15.22	0 / 1,917	74.85	86.71
	Hindi	1,161,358	16,333,361	14.06	16.54	0 / 1,380	72.52	83.24
OpenSubtitles	English	2,103,175	12,126,810	5.77	4.37	1 / 65	29.43	23.01
	Hindi	2,103,175	7,745,456	3.68	2.71	1 / 70	22.73	16.96
Samanantar	English	7,087,994	116,415,974	16.42	14.08	1 / 16,292	97.87	85.09
	Hindi	7,087,994	128,782,369	18.17	14.19	1 / 7,545	93.89	75.69
TED2020	English	33,343	531,863	15.95	11.48	0 / 301	88.03	64.11
	Hindi	33,343	587,366	17.62	12.67	0 / 413	84.01	61.26

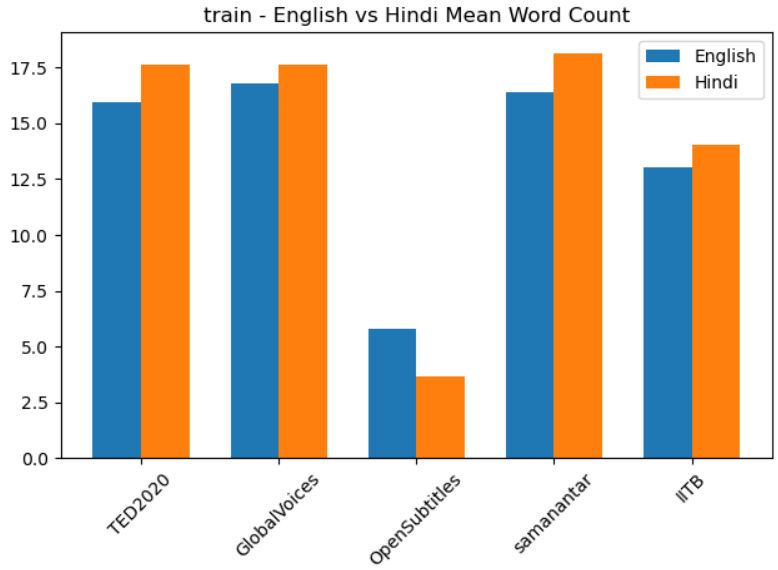


Figure 1: The plot compares average word counts per sentence in both languages.

1.1.1 IITB Corpus

The **IIT Bombay English-Hindi Parallel Corpus** [4] is a comprehensive resource developed by the Center for Indian Language Technology at IIT Bombay to support machine translation and natural language processing research. Currently in Version 3.1, the dataset comprises a training set of approximately 1.66 million parallel sentence pairs and a massive monolingual Hindi corpus containing over 45 million sentences, alongside standard development and test sets derived from WMT 2014 newswire data.
To ensure linguistic variety, the data is aggregated from diverse sources, including software localization files, subtitles, government websites, judicial documents, and dictionaries..[1]

Example 1.1: IITB Corpus

english	The British at once agreed and also made him their agent.
hindi	जैसे समय बीतता गया परिस्थितियों में भी सुधार होता गया।

1.1.2 Open Subtitles

The **OpenSubtitles** corpus [5] is a large-scale parallel dataset extracted from movie and television subtitles. It is a primary resource for training models on spoken-style language, offering approximately 2.1 million English-Hindi sentence pairs. The dataset is characterized by its colloquial nature, containing disfluencies, informal grammar, and conversational context often missing from

[1] `https://huggingface.co/datasets/cfilt/iitb-english-hindi`

formal newswire corpora.[2]

Example 1.2: Open Subtitles

english	I've been hearing you come home every night at God knows what hour.
hindi	मैं सुन रहहूँकतुम बहुत देर से रतक घर पर आतेह।।

1.1.3 Ted talks

The **TED2020** corpus [7] consists of transcripts and translations derived from TED talks. It is widely used for spoken domain adaptation as it bridges the gap between formal written text and spontaneous speech. The dataset contains roughly 33,000 English-Hindi pairs, featuring rhetorical questions, first-person narratives, and diverse topics, making it ideal for evaluating translation models on "planned speech."[3]

Example 1.3: Ted talks

english	They don't know, OK, and they're trying to get another member of The 99 to join them.
hindi	उन्हें पता भी नही है, है न। और ये कोशिश कर रहे है कि 'द ९९' में से एक और उनसे जुड जाये।।

1.1.4 Samanantar

Samanantar [6] is the largest publicly available parallel corpora collection for Indic languages. The English-Hindi portion contributes over 7 million sentence pairs to the training data, significantly scaling up available resources. It aggregates data from web crawls and existing sources, aligning them to create a massive general-domain dataset essential for training robust NMT models.[4]

Example 1.4: Samanantar

english	Modi is abusing history for political mileage
hindi	मोदी : राजनैतिक फायदे के लिये इतिहास का दुरूपयोग

1.1.5 Global Voices

The **Global Voices** corpus [11] is a smaller, high-quality parallel dataset derived from the Global Voices news website. It consists of approximately 1,800 sentence pairs of journalistic text translated by volunteers. While smaller in size, it provides high-quality, human-translated data useful for evaluating model performance on news and reportage domains.[5]

[2] `https://opus.nlpl.eu/OpenSubtitles-v2018.php`
[3] `https://opus.nlpl.eu/datasets/TED2020`
[4] `https://huggingface.co/datasets/ai4bharat/samanantar`
[5] `https://opus.nlpl.eu/GlobalVoices.php`

Example 1.5: Global Voices

english	Videos showed policemen in plainclothes confronting the peaceful protesters, pulling and tearing the signs from Nga's supporters.
hindi	कई वीडियो में सादे कपड़ों में पुलिसकर्मी को शांतिपूर्ण रूप से धरना दे रहे प्रदर्शनकारियों से तख्तियाँ को छीन कर फाड़ते देखा गया।

1.2 Code Mixed

Table 2 and Figure 2 summarize the training statistics for the spoken domain corpora. The table reports overall dataset characteristics, while the figure illustrates the average sentence length in tokens for English and Hindi across different datasets. Together, they provide a concise overview of corpus size and sentence length distribution used in training.

Table 2: Training Data Statistics Summary for Code Mixed (Hinglish-English)

Dataset	Lang	Sentences	Total Words	Avg Words	Word Std	Min/Max Words	Avg Chars	Char Std
Comi-Lingua	English	17,602	336,770	19.13	4.26	2 / 37	117.11	27.25
	Hinglish	17,602	359,117	20.40	3.72	4 / 50	121.70	25.41
English-Hinglish TOP	English	176,596	1,368,372	7.75	3.49	1 / 51	38.87	16.82
	Hinglish	176,596	1,517,052	8.59	3.70	1 / 214	44.30	19.07
Hinge	English	1,383	23,629	17.09	13.06	3 / 121	95.49	74.82
	Hinglish	1,383	23,906	17.29	12.77	3 / 119	96.90	72.32
HINMIX	English	4,200,000	74,617,498	17.77	11.72	1 / 520	104.22	69.57
	Hinglish	4,200,000	76,637,906	18.25	11.53	2 / 165	107.00	68.46
LinCE	English	8,060	87,603	10.87	10.08	1 / 247	56.15	56.13
	Hinglish	8,060	93,025	11.54	11.02	1 / 341	61.06	60.13
PHINC	English	9,616	118,427	12.32	6.70	1 / 40	74.07	38.35
	Hinglish	9,616	127,757	13.29	6.99	1 / 40	76.73	37.09

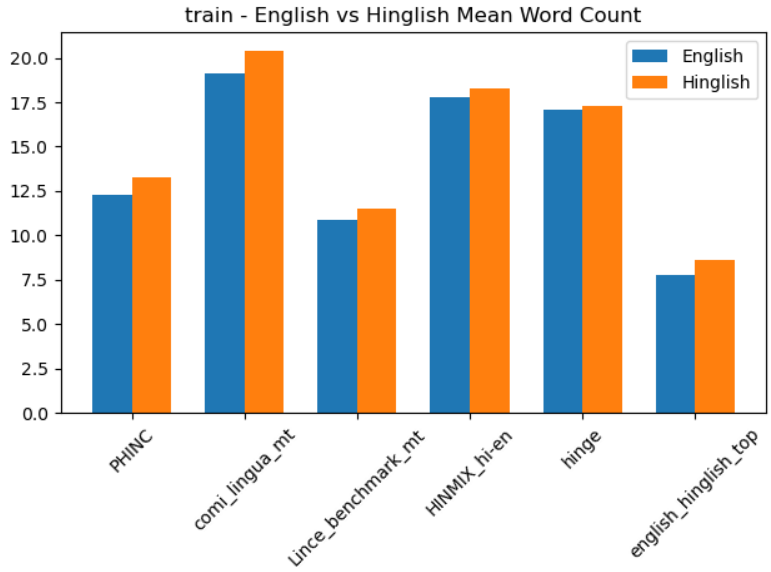


Figure 2: The plot compares average word counts per sentence in both languages.

1.2.1 COMI-LINGUA

COMI-LINGUA (COde-MIxing and LINGuistic Insights on Natural Hinglish Usage and Annotation) [8] is a large-scale, expert-annotated dataset designed for Hindi-English code-mixed text (Hinglish). It was developed by the Lingo Research Group at IIT Gandhinagar to address the lack of high-quality, diverse, and manually verified data for multilingual NLP tasks like Machine Translation (MT) which has Parallel translation of sentences in Romanized Hindi, Devanagari Hindi, and English languages. Initial translation predictions were generated using the Llama 3.3 LLM, which annotators then refined and corrected.[6] for Machine Translation (MT): Parallel translation of sentences in Romanized Hindi and Devanagari Hindi and English languages. Initial translation predictions were generated using the Llama 3.3 LLM, which annotators then refined and corrected.

Example 1.6: COMI-LINGUA

hinglish	भारत में भी green growth, climate resilient infrastructure और ग्रीन transition पर विशेष रूप से बल दिया जा रहा है।
english	In India too, special emphasis is being given to green growth, climate resilient infrastructure, and green transition.
romanized hindi	Bharat mein bhi green growth, climate resilient infrastructure aur green transition par vishesh roop se bal diya ja raha hai.
devnagari hindi	भारत में भी हरित विकास, जलवायु सहनशील आधारिक संरचना और हरित संक्रमण पर विशेष रूप से बल दिया जा रहा है।

[6] https://huggingface.co/datasets/LingoIITGN/COMI-LINGUA

1.2.2 English-Hinglish TOP

The **English-Hinglish TOP** dataset [2] is a code-mixed adaptation of the Task-Oriented Parsing (TOP) dataset. It contains approximately 176,000 sentences focused on navigation, event planning, and reminder queries (e.g., "Find me a route to..."). This dataset is critical for evaluating semantic parsing and intent recognition in code-mixed environments where users often switch languages while issuing voice commands.[7]

Example 1.7: English-Hinglish TOP

english	What is the UV rating today?
hinglish	Today ki UV rating kya hai?
english parse	`[IN:GET_WEATHER What is the [SL:WEATHER_ATTRIBUTE UV rating ] [SL:DATE_TIME today ] ? ]`
hinglish parse	`[IN:GET_WEATHER [SL:DATE_TIME Today ] ki [SL:WEATHER_ATTRIBUTE UV rating ] kya hai? ]`
domain	weather
generated by	human

1.2.3 HinGE

HinGE (Hindi-English Code-Mixed Generation and Evaluation) [10] is a dataset designed to evaluate the naturalness of Hindi-English code-mixed text generation. It contains approximately 1,300 human-produced Hinglish sentences aligned with their corresponding English and Hindi counterparts. The dataset is manually annotated by five annotators and is intended for natural language generation tasks involving code-mixing. It provides a benchmark for assessing the quality of generated Hinglish text and for distinguishing synthetic code-switching from naturally occurring human patterns.[8]
The dataset includes the following fields:

- **English, Hindi**: Parallel source sentences drawn from the IITB English-Hindi parallel corpus.
- **Human-generated Hinglish**: A set of Hinglish sentences created by human annotators.
- **WAC**: Hinglish sentence generated using the WAC algorithm.
- **WAC rating1, WAC rating2**: Quality scores assigned to the WAC-generated sentence by annotators, each ranging from 1 to 10.
- **PAC**: Hinglish sentence generated using the PAC algorithm.
- **PAC rating1, PAC rating2**: Quality scores assigned to the PAC-generated sentence by annotators, each ranging from 1 to 10.

[7] `https://github.com/google-research-datasets/Hinglish-TOP-Dataset`
[8] `https://github.com/Vivek-Iist/HinGE`

Example 1.8: HinGE

english	It was presented to the Legislative Council in 1856 and was passed in 1860.
hindi	इसे 1856 में विधायी परिषद के समक्ष प्रस्तुत किया गया और 1860 में पारित किया गया।
human-generated hinglish	• Ise 1856 mein legislative council ke samaksh prastut kiya gya and 1860 mein paarit kiya gya. • Ise 1856 mein vidhai parishad ko present kiya and 1860 mein pass kiya. • It was presented to vidhayi parishad in 1856 aur 1860 me parit kiya gaya. • Ise 1856 me legislative council ke samaksh present kiya gaya aur 1860 me pass kiya. • 1856 me it was presented to the legislative council aur 1860 me it was passed. • 1856 me ise legislative council ke samaksh prensent kiya gaya aur 1860 me parit kiya gaya.
WAC	ise 1856 men legislative council ke samaksh prastut kiya gaya aur 1860 men parit kiya gaya.
WAC rating	9, 6
PAC	ise 1856 men legislative council ke samaksh prastut kiya gaya aur 1860 men parit kiya gaya.
PAC rating	9, 8

1.2.4 HINMIX

HINMIX [3] is a massive synthetic parallel corpus containing approximately 4.2 million Hinglish-English sentence pairs. It was generated to overcome the data scarcity in code-mixed translation by using transliteration and synthetic code-mixing pipelines. This large-scale noisy dataset helps models learn robust representations of Romanized Hindi and code-switching grammar.[9]
The authors construct this large synthetic Hinglish-English dataset by leveraging a bilingual Hindi-English corpus and systematically generating multiple code-mixed and transliterated variants. The dataset is divided into train, validation, and test splits. It contains the following subsets:

- **Hi**: Hindi sentences in Devanagari script. Example: अमेरिकी लोग अब पहले जितनी गैस नहीं खरीदते।
- **Hicm**: Hindi sentences with selected words substituted in English to create code-mixed text. Example: American people अब पहले जितनी gas नहीं खरीदते।
- **Hicmrom**: Romanized version of Hicm where Hindi words are transliterated into the Roman script. Example: American people ab pahle jitni gas nahin kharidte.
- **Hicmdvg**: Code-mixed sentences where English words are transliterated into Devanagari. Example: अमेरिकन पेओपल अब पहले जितनी गैस नहीं खरीदते।

[9] https://huggingface.co/datasets/kartikagg98/HINMIX_hi-en

- **NoisyHicmrom**: Synthetic noise is injected into Hicmrom sentences to improve robustness against spelling variations and informal typing patterns. Example: Aerican people ab phle jtni gas nain khridte.

Example 1.9: HINMIX

english	The next day I met a cuddly baby girl playing on a swing.
devnagari hindi (Hi)	दूसरे दिन मैंने झूले पर खेलती हुई एक छोटी सी प्यारीप्यारी लड़की देखी।
hinglish (Hicm)	दूसरे दिन मैंने झूले पर खेलती हुई a छोटी सी प्यारीप्यारी girl देखी।
hinglish devnagari (Hicmdvg)	दूसरे दिन मैंने झूले पर खेलती हुई आ छोटी सी प्यारीप्यारी गर्ल देखी।
romanized hindi (Hicmrom)	dusre din maine jhule par khelti hui a chhoti si pyaripyari girl dekhi.
noisy romanized hindi (NoisyHicmrom)	druse din mniae jluhe par khteli hui a cohhti si payiapryri girl dikeh.

1.2.5 LinCE

The **LinCE** (Linguistic Code-switching Evaluation) benchmark [1] provides a centralized leaderboard for various code-mixed tasks. The Hindi-English component comprises roughly 8,000 sentences and is used to evaluate standard NLP tasks such as Language Identification, Named Entity Recognition (NER), and Part-of-Speech (POS) tagging in code-mixed contexts.[10]

Example 1.10: LinCE

english	Not very popular it seems.
hinglish	lagta hai bahut popular nahi hai.

1.2.6 PHINC

PHINC (Parallel Hinglish Social Media Code-Mixed Corpus for Machine Translation) [9]: The dataset tackles challenges in translating noisy, informal, code-mixed social media text, offering 13,738 Hinglish-English sentence pairs manually annotated by 54 annotators for low-resource machine translation task.[11] The dataset contains the following fields:

- Hinglish Code-Mixed Sentence: The original sentence in Romanized Hindi-English (Hinglish).
- Human Translated English Sentence: The corresponding English translation provided by human annotators.

[10] `https://huggingface.co/Huggmachas`
[11] `https://aclanthology.org/2020.wnut-1.7/`

Example 1.11: PHINC: contains no english word in hinglish

hinglish	mujhe lagta hai ye baatein dil ki, hoti lafzon ki hai dhokbaazi.
english	i think these talking are heartbreaking, the words are a shock.
hinglish_dev	मुझे लगता है ये बातें दिल की, होती लफ्जों की है धोकबाज़ी।

Example 1.12: PHINC: contains english words like internet and ban in hinglish

hinglish	ghar pe internet khelne pe ban hai kya ? @CriminalSingh
english	is there a ban on playing internet at home ? @CriminalSingh
hinglish_dev	घर पे इंटरनेट खेलने पे बन है क्या ? क्रिमिनलसिंह

2 Experiment

This report evaluates the efficacy of Low-Rank Adaptation (LoRA), a parameter-efficient fine-tuning (PEFT) technique, to adapt IndicTrans2 to these two domains:

1. **Spoken Domain (TED Talks):** This setting evaluates the translation of TED Talk transcripts, which contain disfluencies, colloquialisms, rhetorical questions, and first-person narratives distinct from formal written text. The evaluation is conducted on a test set of 9,527 examples.

2. **Code-Mixed Domain (PHINC):** This setting targets the translation of informal, Romanized Hindi mixed with English grammar (Hinglish), a dominant form of social media communication in South Asia. As IndicTrans2 is trained on Devanagari Hindi, a transliteration pipeline is employed to bridge the script gap. The evaluation uses the PHINC dataset with a test set of 2,748 examples.

Performance is measured on the test split of the respective datasets using a holistic suite of metrics: lexical overlap is assessed via **BLEU** and **CHRF**, while semantic fidelity and learned quality estimation are evaluated using **BERTScore**, **COMET**, and **BLEURT**.

2.1 Evaluation Metrics

To comprehensively evaluate translation quality, we employ four complementary automatic metrics that capture lexical, character-level, semantic, and learned quality signals.

- **BLEU (Bilingual Evaluation Understudy):** BLEU measures modified n-gram precision between a candidate translation c and a reference translation r, combined with a brevity penalty to discourage overly short outputs.

  For n-grams up to order N (typically $N = 4$), the modified precision p_n is defined as:

$$p_n = \frac{\sum_{g \in c} \min\left(\text{Count}_c(g), \text{Count}_r(g)\right)}{\sum_{g \in c} \text{Count}_c(g)}$$

The brevity penalty (BP) is:

$$\text{BP} = \begin{cases} 1 & \text{if } |c| > |r| \\ \exp\left(1 - \frac{|r|}{|c|}\right) & \text{if } |c| \leq |r| \end{cases}$$

The final BLEU score is computed as:

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

where w_n are typically uniform weights such that $\sum_{n=1}^{N} w_n = 1$.

- **CHRF (Character n-gram F-score):** CHRF operates at the character level and computes an F-score over character n-grams, making it more robust for morphologically rich languages.

  Let P_n and R_n denote character n-gram precision and recall:

$$P_n = \frac{\sum_{g \in c} \min(\text{Count}_c(g), \text{Count}_r(g))}{\sum_{g \in c} \text{Count}_c(g)}$$

$$R_n = \frac{\sum_{g \in r} \min(\text{Count}_c(g), \text{Count}_r(g))}{\sum_{g \in r} \text{Count}_r(g)}$$

  The F-score for order n is:

$$F_{\beta,n} = \frac{(1+\beta^2) P_n R_n}{\beta^2 P_n + R_n}$$

  The overall CHRF score is computed by averaging over character n-gram orders:

$$\text{CHRF}_\beta = \frac{1}{N} \sum_{n=1}^{N} F_{\beta,n}$$

  where β controls the relative importance of recall and precision, typically $\beta = 2$.

- **BERTScore (F1):** BERTScore evaluates semantic similarity by computing cosine similarity between contextual embeddings of tokens from a pretrained language model.

  Let $\mathbf{h}_i^c$ denote the embedding of token i in the candidate sentence and $\mathbf{h}_j^r$ the embedding of token j in the reference sentence. The pairwise similarity is:

$$s_{ij} = \cos(\mathbf{h}_i^c, \mathbf{h}_j^r)$$

  Precision and recall are computed via maximum matching:

$$P = \frac{1}{|c|} \sum_{i \in c} \max_{j \in r} s_{ij}$$

$$R = \frac{1}{|r|} \sum_{j \in r} \max_{i \in c} s_{ij}$$

The final BERTScore (F1) is:

$$F_1 = \frac{2PR}{P + R}$$

- **COMET:** COMET is a learned evaluation metric based on neural regression trained to predict human judgment scores. It takes as input the source sentence x, candidate translation c, and reference translation r.

  Let $\mathbf{z}_x$, $\mathbf{z}_c$, and $\mathbf{z}_r$ denote contextual embeddings extracted from a pretrained encoder. These representations are combined and passed through a feed-forward regression model f_θ:

  $$\text{COMET}(x, c, r) = f_\theta\left(\mathbf{z}_x, \mathbf{z}_c, \mathbf{z}_r\right)$$

  The model is trained to minimize mean squared error with respect to human scores y:

  $$\mathcal{L}(\theta) = \frac{1}{M} \sum_{i=1}^{M} \left(f_\theta(x_i, c_i, r_i) - y_i\right)^2$$

  Unlike overlap-based metrics, COMET directly models human preferences and captures adequacy, fluency, and semantic consistency.

- **BLEURT (Bilingual Evaluation Understudy with Representations from Transformers):**

  BLEURT is a learned evaluation metric based on a pretrained Transformer encoder (typically BERT) fine-tuned to predict human judgment scores. Unlike surface-based metrics such as BLEU, BLEURT directly models semantic adequacy and fluency through regression.

  Given a candidate translation c and a reference translation r, BLEURT first constructs a joint input sequence:

  $$\mathbf{x} = [\text{CLS}, c, \text{SEP}, r, \text{SEP}]$$

  This sequence is encoded using a pretrained Transformer encoder to obtain contextual representations:

  $$\mathbf{H} = \text{Encoder}(\mathbf{x})$$

  Let $\mathbf{h}_{\text{CLS}}$ denote the final hidden representation of the CLS token. A regression head g_ϕ maps this representation to a scalar quality score:

  $$\text{BLEURT}(c, r) = g_\phi(\mathbf{h}_{\text{CLS}})$$

  The regression head typically consists of a linear projection:

$$g_\phi(\mathbf{h}) = \mathbf{w}^\top \mathbf{h} + b$$

BLEURT is trained to minimize the mean squared error between predicted scores and human evaluation scores y:

$$\mathcal{L}(\phi) = \frac{1}{M} \sum_{i=1}^{M} \Big(g_\phi(\mathbf{h}_{\text{CLS}}^{(i)}) - y_i \Big)^2$$

To improve robustness, BLEURT is often pre-trained on large amounts of synthetically perturbed sentence pairs before fine-tuning on human-rated datasets. This allows it to generalize better to noisy and diverse translation outputs. ‘

2.2 Experiments: Spoken Domain

2.2.1 Experimental Setup

Fine-tuning utilizes the LoRA technique, which injects trainable rank decomposition matrices into the Transformer layers while freezing the pre-trained weights. The complete set of optimization, regularization, and LoRA-specific hyperparameters is provided in Table 3.

Table 3: Hyperparameter Configuration for Fine-Tuning

Category	**Configuration Details**
Optimization	Adam optimizer with early stopping based on BLEU improvement
Training Config	Batch size: 32, Gradient accumulation steps: 2, Epochs: 3
Learning Dynamics	Learning rate $\eta = 5 \times 10^{-5}$, Weight decay: 0.01, Warmup ratio: 0.1
Regularization	Max gradient norm: 1.0, Label smoothing: 0.05
LoRA Configuration	Rank $r = 8$, Scaling factor $\alpha = 32$, Dropout 0.2
Target Modules	Attention mechanism query, key, value, and output projections (q_proj, k_proj, v_proj, out_proj)

2.2.2 Results and Discussion

The results for the spoken domain indicate that IndicTrans2 is already capable of handling conversational English and Hindi. Figure 3 illustrates the validation BLEU and evaluation loss trends across epochs for both translation directions, while Table 4 summarizes the final evaluation metrics. In the **English** $\rightarrow$ **Hindi** direction, evaluation BLEU initially decreases slightly before stabilizing, whereas the validation loss consistently decreases from approximately 2.78 to 2.71. The steady reduction in loss indicates improved optimization and better likelihood estimation over epochs. However, the marginal fluctuation in BLEU suggests that lower loss does not necessarily translate into proportional improvements in surface-form overlap with reference translations. As shown in Table 4, fine-tuning yields a modest BLEU improvement (0.2152 $\rightarrow$ 0.2220). However, CHRF

slightly decreases (47.83 → 46.94), and COMET drops from 0.7729 to 0.7611, while BERTScore F1 remains nearly unchanged. BLEURT also shows a small decline (0.6884 → 0.6772), further suggesting that the observed BLEU gain may not correspond to a genuine improvement in semantic fidelity. Since BLEURT is designed to approximate human judgment by capturing contextual and adequacy-level nuances beyond surface n-gram overlap, its reduction indicates that the fine-tuned model may introduce subtle distortions or less natural phrasing despite improved lexical alignment. This pattern reinforces the observation that optimization improvements reflected in loss reduction do not necessarily guarantee consistent gains across learned semantic evaluation metrics.

This behavior is also reflected qualitatively in Example 2.1. Although the fine-tuned model produces a slightly more literal rendering **छोटी हो गई है** compared to the base model's progressive construction **छोटी होती जा रही है**, the semantic content remains nearly identical. This subtle lexical adjustment may contribute to minor BLEU variation without indicating a substantial semantic improvement, consistent with the observed decline in BLEURT and COMET scores.

In the **Hindi → English** direction, evaluation BLEU exhibits a gradual upward trend across epochs, while validation loss decreases steadily from approximately 1.93 to 1.86, suggesting stable convergence without evidence of overfitting. Nevertheless, Table 4 shows that the overall improvements remain marginal across all metrics. BLEU increases only slightly (0.3596 → 0.3610), CHRF remains almost constant, BERTScore F1 improves minimally, and COMET shows a negligible decline. BLEURT, in contrast to the English → Hindi direction, improves from 0.7232 to 0.7260. Although the increase is modest, this upward shift in a learned quality metric indicates a slight enhancement in semantic adequacy and fluency from Hindi to English. The discrepancy between COMET and BLEURT trends highlights that different neural evaluation models may emphasize distinct aspects of translation quality, such as adequacy, fluency, or contextual consistency.

Table 4: Performance on Spoken Domain (TED Talks)

Direction	Model	BLEU	CHRF	BERTScore F1	COMET	BLEURT
English → Hindi	Base	21.52	**47.83**	0.9468	**0.7729**	**0.6884**
	Fine-tuned	**22.20**	46.94	**0.9471**	0.7611	0.6772
Hindi → English	Base	35.96	**57.94**	0.9361	**0.8632**	0.7232
	Fine-tuned	**36.10**	57.85	**0.9374**	0.8626	**0.7260**

The limited gains in both directions may indicate that IndicTrans2's pre-training already sufficiently captures spoken-style Hindi–English translation patterns. Alternatively, it is plausible that the TED2020 corpus, despite being conversational in format, does not comprehensively reflect spontaneous and diverse real-world spoken discourse. Consequently, the dataset may not introduce a strong enough domain shift to yield substantial adaptation benefits. Furthermore, the mixed behavior of BLEURT across directions suggests that domain adaptation effects may be asymmetric, potentially influenced by differences in linguistic structure, tokenization granularity, or the relative difficulty of generating morphologically rich Hindi versus comparatively analytic English.

Overall, the loss curves demonstrate stable optimization behaviour, while the evaluation metrics confirm that domain-specific fine-tuning provides only modest improvements in the spoken setting. The inclusion of BLEURT further supports this conclusion by revealing that semantic-level gains are limited and, in some cases, direction-dependent, thereby underscoring the importance of employing

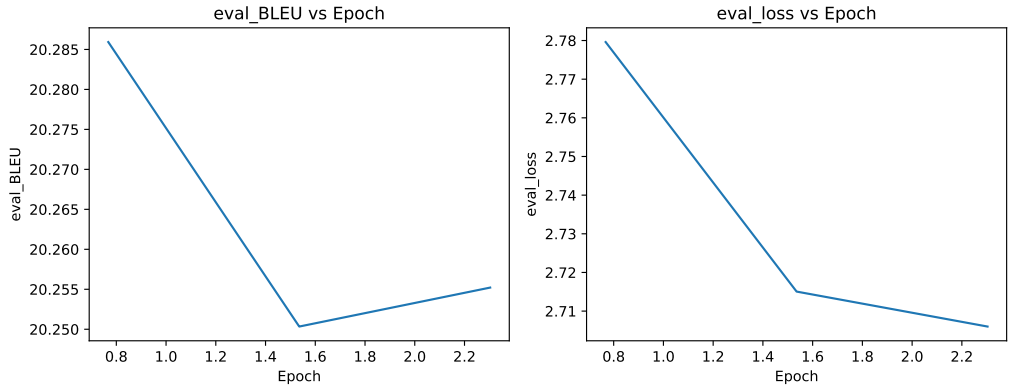
multiple complementary evaluation metrics when assessing domain adaptation performance. A qualitative illustration is provided in Example 2.2. Here, the fine-tuned model aligns more closely with the conversational phrasing of the reference ("figure this out") compared to the base model's slightly more neutral alternative ("find out"). While the semantic difference is minimal, the refined lexical choice reflects improved stylistic adequacy in spoken discourse, which may explain the slight increase observed in BLEURT.

Example 2.1: English → Hindi

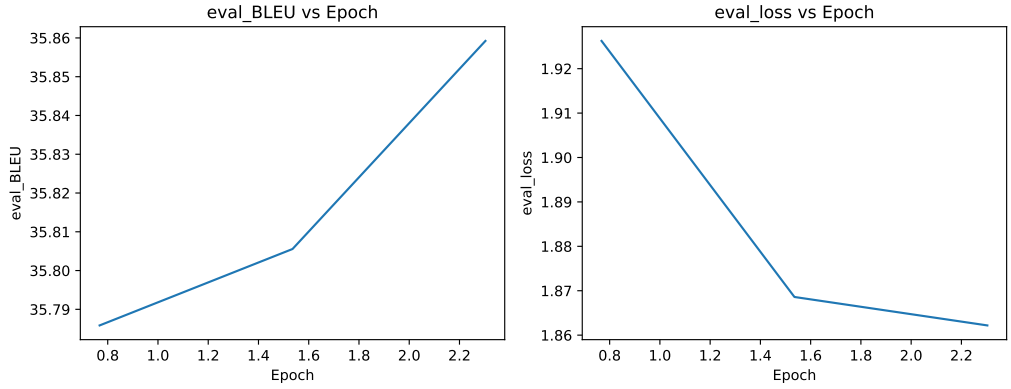
english	(Laughter) Ladies and gentlemen, the world has gotten smaller.
hindi_dev	(ठहाका) देवियों और सज्जनों, दुनिया आज छोटी हो गयी है.
hindi	(ठहाका) देवियों और सज्जनों, दुनिया आज छोटी हो गयी है.
hindi_pred_base	(हँसी) देवियों और सज्जनों, दुनिया छोटी होती जा रही है।
hindi_pred_finetuned	(हँसी) देवियों और सज्जनों, दुनिया छोटी हो गई है।

Example 2.2: Hindi → English

hindi	तो उन्होंने ये कैसे पता लगाया होगा?
english_dev	So how'd they figure this out?
english	So how'd they figure this out?
english_pred_base	So how did they find out?
english_pred_finetuned	So how did they figure this out?



(a) English → Hindi: Evaluation BLEU and Loss vs Epoch



(b) Hindi → English: Evaluation BLEU and Loss vs Epoch

Figure 3: Validation performance across epochs for the spoken domain. BLEU exhibits a modest upward trend while evaluation loss consistently decreases.

2.3 Experiments: Code-Mixed Domain

The code-mixed domain represents a much more severe distribution shift involving Romanized Hindi scripts and grammatical mixing (Hinglish). Fine-tuning utilizes the LoRA technique, which injects trainable rank decomposition matrices into the Transformer layers while freezing the pre-trained weights. The complete set of optimization, regularization, and LoRA-specific hyperparameters is provided in Table 5.

2.3.1 Pipeline and Setup

Since IndicTrans2 does not natively support Romanized Hinglish, its predefined translation directions such as indic to English, English to indic, and indic to indic do not directly satisfy the

Table 5: Hyperparameter Configuration for Fine-Tuning

Category	**Configuration Details**
Optimization	Adam optimizer with early stopping based on BLEU improvement
Training Config	Batch size: 32, Gradient accumulation steps: 2, Epochs: 10 (Hinglish → English), 20 (English → Hinglish)
Learning Dynamics	Learning rate $\eta = 5 \times 10^{-5}$, Weight decay: 0.01, Warmup ratio: 0.1
Regularization	Max gradient norm: 1.0, Label smoothing: 0.05
LoRA Configuration	Rank $r = 8$, Scaling factor $\alpha = 32$, Dropout 0.2
Target Modules	Attention mechanism query, key, value, and output projections (q_proj, k_proj, v_proj, out_proj)

requirement of translating Romanized Hinglish to English and vice versa. To address this limitation, we introduce a preprocessing step that transliterates Romanized Hinglish into Devanagari Hinglish. The resulting Devanagari text is then used to fine tune the model under the indic to English and English to indic settings, enabling bidirectional translation between Romanized Hinglish and English.

However, this transliteration based preprocessing can introduce performance degradation. Errors or ambiguities in the Roman to Devanagari conversion may propagate to the translation stage, and since the underlying model parameters remain frozen, the system has limited capacity to adapt to such noise. As a result, the overall translation quality may dip compared to a model trained natively on Romanized Hinglish data. For transliteration, we use `XlitEngine` from `ai4bharat.transliteration`. The transliteration module is used as is, with frozen parameters, and no additional fine tuning or modifications are applied.:

- **Hinglish → English:**
  Hinglish Romanized (Input) $\xrightarrow{\text{Transliterate}}$ Hinglish Devanagari $\xrightarrow{\text{Translate}}$ English (Output).
- **English → Hinglish:**
  English (Input) $\xrightarrow{\text{Translate}}$ Hinglish Devanagari $\xrightarrow{\text{Transliterate}}$ Hinglish Romanized (Output).

Table 6: Performance on Code-Mixed Domain (PHINC)

Direction	**Model**	**BLEU**	**CHRF**	**BERT F1**	**COMET**	**BLEURT**
Hinglish → English	Base	14.49	35.16	0.8303	0.6554	0.5546
	Fine-tuned	**24.58**	**43.55**	**0.8429**	**0.6948**	**0.5944**
English → Hinglish	Base	**9.25**	34.82	0.8573	**0.6612**	0.4059
	Fine-tuned	6.44	**36.06**	**0.8608**	0.6451	**0.4305**

2.3.2 Results and Discussion

Fine-tuning is performed for a longer duration (20 epochs) compared to the Hinglish → English, utilizing the same hyperparameters ($r = 8, \alpha = 32$) to allow the model to adapt to predict english words in hindi.

Hinglish → English: This direction shows substantial improvement not only in automatic metrics but also in optimization behavior across epochs. As illustrated in Figure 4a, evaluation loss decreases consistently with training, indicating stable convergence and improved generalization. At the same time, evaluation BLEU exhibits a steady upward trajectory, rising from approximately 6.0 in early epochs to above 9.0 toward the end of training. This monotonic loss reduction coupled with BLEU improvement suggests that fine-tuning effectively adapts the model to the code-mixed distribution rather than overfitting.

Quantitatively, BLEU improves by nearly 70% from 14.49 to 24.58. CHRF increases significantly from 35.16 to 43.55, reflecting better character-level alignment, which is especially important for transliteration-heavy inputs. BERTScore F1 improves from 0.8303 to 0.8429, indicating stronger semantic similarity, while COMET rises from 0.6554 to 0.6948, confirming meaningful gains in translation quality from a learned evaluation perspective. BLEURT also improves from 0.5546 to 0.5944, further confirming that fine-tuning enhances overall translation quality in a way that better aligns with human judgment.

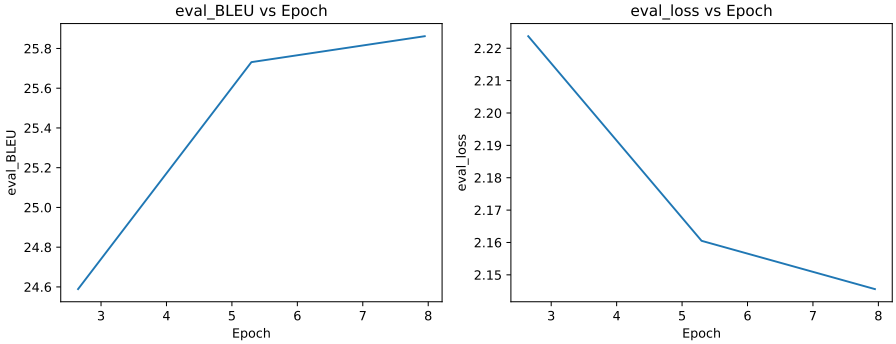
Overall, the consistent decline in evaluation loss, the upward BLEU trend across epochs, and the substantial improvements across semantic and lexical metrics collectively demonstrate that fine-tuning successfully enables the model to act as a denoising and normalization mechanism for noisy code-mixed input, resulting in significantly more fluent and semantically accurate English translations.

Example 2.3: Bekar Translation Error

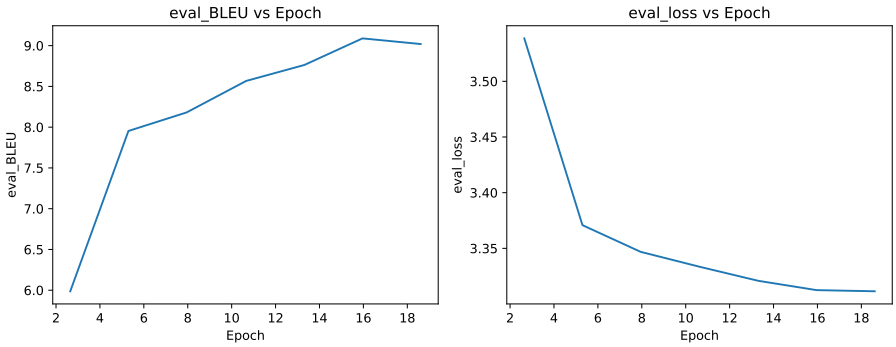
hinglish	@RoflGandhi_ unka intezar karna bhi bekar hai bhai!
hinglish_dev	@रोफलगांधी_ उनका इंतजार करना भी बेकर है भाई!
english	@RoflGandhi_ waitiing for them is also waste!
english_pred_base	@रोफलगांधी_ Waiting for them is also a baker brother!
english_pred_finetuned	@रोफलगांधी_ Waiting for them is also useless!

The Example 2.3 further illustrates the qualitative difference between the base and fine-tuned models. In particular, the Hinglish word *bekar* is incorrectly translated as *baker* by the base model. This error arises from surface-level phonetic similarity, where the model maps the transliterated token to a high-frequency English lexical neighbor rather than its contextual meaning. Since *bekar* is a transliteration of the Hindi word meaning "useless" or "waste," the base model fails to capture the semantic equivalence and instead performs a literal orthographic approximation.

In contrast, the fine-tuned model correctly translates *bekar* as *useless*, demonstrating improved semantic disambiguation and contextual grounding. This indicates that fine-tuning on code-mixed data enables the model to learn systematic correspondences between transliterated Hindi tokens and their true English equivalents. Rather than relying on superficial string similarity, the fine-tuned model leverages contextual cues and distributional adaptation to produce a semantically appropriate translation. This behavior supports the quantitative improvements reported earlier, showing that the gains are not only metric-level but also linguistically meaningful at the lexical level.



(a) Hinglish → English: Evaluation BLEU and Loss vs Epoch



(b) English → Hinglish: Evaluation BLEU and Loss vs Epoch

Figure 4: Validation performance across epochs for the spoken domain. BLEU exhibits a modest upward trend while evaluation loss consistently decreases.

English → Hinglish: In contrast to the previous direction, this setting presents a mismatch between optimization behavior and automatic evaluation. As illustrated in Figure 4b, evaluation loss decreases monotonically from approximately 3.53 in early epochs to nearly 3.31 toward the end of training, indicating stable convergence and effective fitting of the training distribution. Simultaneously, evaluation BLEU increases steadily from around 6.0 to slightly above 9.0, with only a minor fluctuation in the final epoch. This upward trajectory suggests that the model does improve in terms of surface-level overlap during training.

However, despite this positive epoch-level trend, the final fine-tuned model underperforms the base model on the held-out test set. Quantitatively, BLEU drops from 9.25 (base) to 6.44 (fine-tuned), representing a relative decline of approximately 30%. CHRF increases modestly from 34.82 to 36.06, indicating slight improvements at the character level. BERTScore F1 improves marginally from 0.8573 to 0.8608, while COMET decreases from 0.6612 to 0.6451. BLEURT increases from 0.4059 to 0.4305, suggesting a slight improvement in quality when evaluated with a metric designed to better correlate with human judgment, despite the drop observed in BLEU on the test set.

This discrepancy arises primarily from a mismatch in evaluation protocols between validation and

test time. During validation, evaluation is performed between Hinglish written in Devanagari script and standard Hindi, allowing closer lexical alignment at the script level. In contrast, final test-time evaluation is conducted between Roman-script Hinglish and its transliterated Roman counterpart. Because transliteration introduces additional surface variation, even semantically correct outputs may differ orthographically from the reference.

As a result, minor differences in transliteration conventions, spelling choices, or lexical selection can significantly reduce n-gram overlap in Roman script. Metrics such as BLEU are particularly sensitive to these surface mismatches, leading to lower test-set scores despite stable optimization behavior and validation-time improvements. Therefore, the observed performance gap reflects evaluation-level orthographic variability rather than a fundamental collapse in semantic generation quality, highlighting the challenges of measuring code-mixed generation under strict string-matching metrics.

Furthermore, the two-stage pipeline (translation followed by transliteration) amplifies small lexical deviations. If the intermediate Hindi output differs from the reference in word choice, transliteration produces completely different surface forms, leading to minimal overlap and disproportionately large BLEU reductions.

The qualitative examples illustrate this phenomenon. In the Example 2.4 Model performed better case, the fine-tuned model generates "guddu impression banana janata hai," which preserves the intended meaning and adopts a natural mixed structure. Although stylistic markers from the reference are absent, semantic fidelity is maintained. In contrast, the base model produces "prabhaav dalna," a fully Hindi lexical choice that increases divergence from the code-mixed reference and further reduces overlap. Moreover, the fine-tuned model retains the capacity to generate English lexical items when contextually appropriate, preserving the intended code-mixed character of the output. In contrast, the base model consistently defaults to fully Hindi lexical realizations and fails to introduce English words even when the reference exhibits mixed usage. This difference highlights that fine-tuning improves the model's ability to model code-mixing behavior, even when overlap-based metrics such as BLEU do not fully reflect that gain.

The Model performed poor example Example 2.5 reveals instability in certain instances. The fine-tuned and development predictions truncate the sentence after the hashtags, producing repeated punctuation and omitting the explanatory clause entirely. This results in extremely low lexical overlap and severe metric penalties. While the base model attempts a translation, stylistic and lexical mismatches still constrain BLEU.

Overall, even though training curves show consistent loss reduction and validation BLEU improvement, final evaluation indicates that generating code-mixed text remains substantially more challenging than normalizing it. The gap between optimization signals and test-set metrics underscores the difficulty of modeling highly variable code-mixed targets under strict overlap-based evaluation.

Example 2.4: Model performed better

english	guddu knows how to make an impression.
hinglish	guddu ko style marna b aata hai :p
hinglish_dev	गुड्डू को स्टाइल मारना बी आता है :प
pred_hinglish_dev	गुड्डू इम्प्रेशन बनाना जानता है।
pred_hinglish_base	guddu janata hai kii kaise prabhaav dalna hai.
pred_hinglish_finetuned	guddu impression banana janata hai.

Example 2.5: Model performed poor

english	#respect #aus #playlikeawinner . now the respect about india has vanished.
hinglish	#respect #aus #playlikeawinner . . . baki ap to india ka liye respect na ke barabar hi reh gyi hai
hinglish_dev	#रिस्पेक्ट #ओएस #प्लेलाइकअविनर . . . बाकी एपी तो इंडिया का लिए रिस्पेक्ट ना के बराबर ही रेह गयी है
pred_hinglish_dev	#respect #aus #playlikeawinner.....
pred_hinglish_base	#respect #aus #playlikeawinner. ab bharat key baare main sammaan gayab how gaya hai.
pred_hinglish_finetuned	#respect #aus #playlikeawinner.....

3 Conclusion

This study evaluated the effectiveness of Low-Rank Adaptation, a parameter-efficient fine-tuning method, for adapting IndicTrans2 to two challenging settings: spoken-domain translation and Hindi-English code-mixed translation. Performance was assessed using complementary lexical, character-level, and learned semantic metrics including BLEU, CHRF, BERTScore, COMET, and BLEURT. In the spoken-domain setting based on TED Talks, fine-tuning resulted in only marginal improvements over the base model. Although evaluation loss consistently decreased across epochs, gains in BLEU were small and learned metrics showed mixed or negligible changes. In some cases, slight increases in lexical overlap did not correspond to improvements in semantic quality. This indicates that the pretrained model already captures much of the conversational and semi-formal structure present in TED-style speech. As a result, lightweight domain adaptation through LoRA provides limited additional benefit in this relatively mild distribution shift. I want to add that Spoken-domain fine-tuning was performed for only 3 epochs . Although training was stable, additional epochs may allow further domain adaptation and potentially yield improved semantic-level gains.

In contrast, the code-mixed setting revealed a strong asymmetry between normalization and generation. For Hinglish to English translation, fine-tuning produced substantial improvements across all metrics. Both lexical and semantic measures improved significantly, and qualitative analysis showed better contextual disambiguation of transliterated Hindi words. The fine-tuned model learned to resolve phonetic ambiguities and produce semantically appropriate translations rather than relying on surface-level similarity. This demonstrates that LoRA adaptation is highly effective when the target domain represents a substantial deviation from the model's pretraining distribution.
However, the reverse direction, English to Hinglish generation, remained considerably more challenging. Despite stable optimization and improving validation trends during training, final test performance showed inconsistent gains, particularly under overlap-based metrics. Unlike standard translation, there is no single canonical Hinglish target. Multiple semantically valid outputs may differ significantly in lexical choice, script usage, and code-switching patterns. Since BLEU relies on exact n-gram overlap, even valid paraphrastic or stylistic variations are heavily penalized. Additionally, the two-stage translation and transliteration pipeline amplifies small lexical differences, further reducing surface-level metric scores.
Overall, the experiments highlight three central findings. First, parameter-efficient fine-tuning with LoRA can yield substantial gains when the distribution shift is large, as in code-mixed normal-

ization. Second, when the base model already aligns well with the target domain, improvements remain modest despite stable optimization. Third, evaluating code-mixed generation requires more than surface-based metrics, as semantic adequacy and stylistic flexibility are not fully captured by n-gram overlap.

In summary, LoRA-based adaptation is highly effective for handling noisy, transliterated code-mixed input and improving semantic normalization, moderately useful for already well-represented spoken domains, and still limited in fully addressing the open-ended variability of code-mixed generation. These findings underscore both the strengths of parameter-efficient fine-tuning and the need for improved evaluation frameworks for multilingual and code-mixed text generation.

References

[1] Gustavo Aguilar, Sudipta Kar, and Thamar Solorio. LinCE: A centralized benchmark for linguistic code-switching evaluation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1803–1813. European Language Resources Association, 2020.

[2] Arpit Gupta, Rushin Shah, Mrinal Mohit, Anuj Kumar, and Mike Lewis. Semantic parsing for task oriented dialog using hierarchical representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2018.

[3] K. Kartik et al. Synthetic data generation and joint learning for robust code-mixed translation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 2024.

[4] Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. The IIT Bombay English-Hindi parallel corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, 2018. European Language Resources Association (ELRA).

[5] Pierre Lison and Jörg Tiedemann. Opensubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, 2016. European Language Resources Association (ELRA).

[6] Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh M. Khapra. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages. *Transactions of the Association for Computational Linguistics*, 10:145–162, 2022.

[7] Nils Reimers and Iryna Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2020.

[8] Rilka Sheth, Hiren Beniwal, and Mayank Singh. COMI-LINGUA: Expert annotated large-scale dataset for multitask NLP in Hindi-English code-mixing. In *Findings of the Association for Computational Linguistics: EMNLP 2025*. Association for Computational Linguistics, 2025.

[9] Vivek Srivastava and Mayank Singh. 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 48–58. Association for Computational Linguistics, 2020.

[10] Vivek Srivastava and Mayank Singh. HinGE: A dataset for generation and evaluation of code-mixed Hinglish text. In *Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems (Eval4NLP)*. Association for Computational Linguistics, 2021.

[11] Jörg Tiedemann. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012)*, Istanbul, Turkey, 2012. European Language Resources Association (ELRA).