INMT-*Lite*: Accelerating Low-Resource Language Data Collection via Offline Interactive Neural Machine Translation

Harshita Diddee[*♡2], Anurag Shukla[*1], Tanuja Ganu[1], Vivek Seshadri[1]
Sandipan Dandapat[3], Monojit Choudhury[4♡], Kalika Bali[1]
Carnegie Mellon University[2], Microsoft Research India[1], Microsoft R&D, India[3]
Mohamed Bin Zayed University Of Artificial Intelligence [4]
hdiddee@andrew.cmu.edu, monojit.choudhury@mbzuai.ac.ae
{anuragshukla, visesha,taganu, sadandap,kalikab}@microsoft.com

Abstract

A steady increase in the performance of Massively Multilingual Models (MMLMs) has contributed to their rapidly increasing use in data collection pipelines. Interactive Neural Machine Translation (INMT) systems are one class of tools that can utilize MMLMs to promote such data collection in several under-resourced languages. However, these tools are often not adapted to the deployment constraints that native language speakers operate in, as bloated, online inference-oriented MMLMs trained for data-rich languages, drive them. INMT-Lite addresses these challenges through its support of (1) three different modes of Internet-independent deployment and (2) a suite of four assistive interfaces suitable for (3) data-sparse languages. We perform an extensive user study for INMT-Lite with an under-resourced language community, Gondi, to find that INMT-Lite improves the data generation experience of community members along multiple axes, such as cognitive load, task productivity, and interface interaction time and effort, without compromising on the quality of the generated translations. INMT-Lite's code is open-sourced to further research in this domain.

Keywords: Machine Translation, Corpus Creation and Annotation, Endangered Languages

1. Introduction

The rapidly evolving quality of language technologies driven by the steady improvement of multilingual models (MLMs) does not benefit all languages equally. Several under-resourced language communities are unable to effectively use these advances, as empirical performance gains do not translate effectively to performance standards apt for large-scale community adoption Liebling et al. (2020); Nekoto et al. (2020). Specifically, models in these languages still require a combined effort across axes of more representative metrics, test sets, and domain-diverse training data to scale their adoption proportional to their high-resource language counterparts Team et al. (2022); AI4Bharat et al. (2023). In this work, we focus on tackling the lack of data in these languages. In particular, we argue that the limited availability of machine-readable data for automatic sourcing Rijhwani et al. (2023); Mehta et al. (2022); Jurgens et al. (2017) in these languages calls for an increased effort to assist native language speakers, who can drive data collection in these languages. Accordingly, we leverage the concept of interactive machine translation (INMT) to improve the yield of data collection pipelines while enhancing the overall experience of data providers Lam et al. (2019); Gupta et al. (2021); Santy et al. (2019).

Since under-resourced language communities may not have access to high-capacity, internet-enabled systems, which are dependencies for data generation using INMT tools, we design and demonstrate the efficacy of an INMT service that is adapted to (a) the infrastructural capabilities of the user. Additionally, since low-resource language model development often leverages pre-trained models, deploying the model on an edge device whilst maintaining translation quality is non-trivial and dependent on the user device's edge capacity. Accordingly, we further provide (b) flexible User Interface Choices to account for the quality of the underlying model, which may not be very high performing for extremely low-resource languages.

We describe *INMT-Lite*, an internet-independent INMT service driven by low-latency, compressed MLMs specifically designed to accelerate low-resource data collection. INMT-Lite lends its unique design to 2 primary factors:

1. **Offline, Edge-Capacity Adapted Model**: To achieve this, INMT-Lite provides three different modes of operation: (a) **Native** (Internet-Enabled, Uncompressed model), (b) **Quantized** (Internet-Independent, Compressed model) and (c) **Distilled** Models (Internet-Independent, Compressed model) which language technologists can choose from depending upon what amenities their users want to expend during their participation.

2. **Flexible User Interface Choices**: INMT-

[♡] Work done when the author was at Microsoft Research India.

[*] Equal Contribution

LREC-COLING 2024, pages 9097–9109

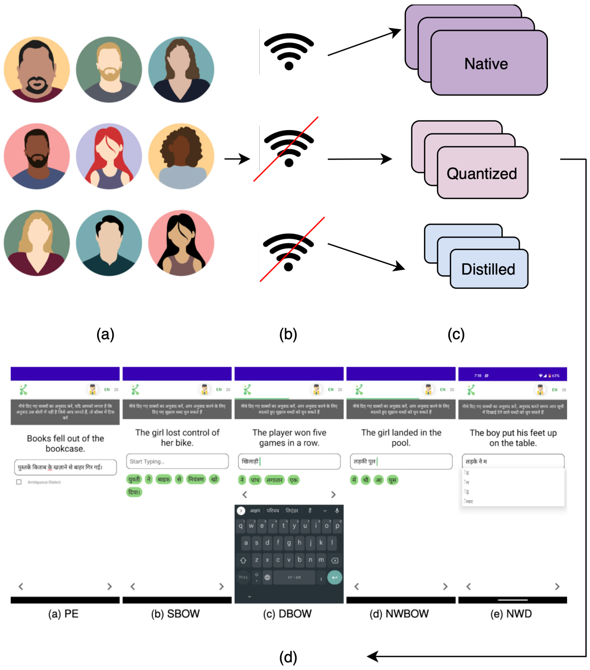

Native
Quantized
Distilled
(a)
(b)
(c)
Books fell out of the bookcase.
The girl lost control of her bike.
The player won five games in a row.
The girl landed in the pool.
The boy put his feet up on the table.
Start Typing...
(a) PE
(b) SBOW
(c) DBOW
(d) NWBOW
(e) NWD
(d)

Figure 1: An Overview of INMT-Lite's Operations: (a) Depending upon the community's constraints (b) an offline (quantized or distilled) or online mode of model deployment is used to drive the (c) suite of compressed models which drive (d) different assistive interfaces which provide data collection support at varying levels of granularity.

> Lite has five interfaces based on naive **Post-Editing, Bag Of Words and Dropdown structures**. These interfaces are meant to give language technologists the freedom to vary the interface according to the target language, more specifically, the quality of the underlying assistive model's performance in that target language.

Through an extensive user study with a severely underresourced language community, Gondi, we substantiate the use of INMT-Lite and the efficacy of the flexibility with different modes and interface choices. We open source the entire pipeline for development of INMT-Lite.

2. Related Work

Existing work supports the utility of interactive neural machine translation for high-resource languages Xiao et al. (2022); Wang et al. (2020); Maheshwari et al. (2023); Lee et al. (2017) by demonstrating a gain in user productivity and translation quality. However, transferring the efficacy of these systems for their operation in languages with lower resources is not trivial. One reason for this impeded transfer is the observation that assistive models in low-resource languages do not necessarily have very high-performance translation models, which interfaces do not take into consideration. For example, Lane and Bird (2022) devise an optimality constraint-based toward interactive word completion in morphologically complex languages. Similarly, additional work such as Santy et al. (2019); Gupta et al. (2021); Lane and Bird (2021) explores different data selection and decoding schemes to adapt these systems to under-resourced setups. Despite these efforts, the adaptation of these systems to the operational constraints of the community is not yet explored. This adaptation is critical as it encourages large-scale adoption in communities that may not have the same set of amenities that other high-resource language communities enjoy Bettinson and Bird (2017); Bird (2018).

INMT-Lite attempts to reconcile these constraints using features across two dimensions; It provides the **flexibility of choosing between multiple interfaces** (varying in degree of instrusion and information density) depending upon the quality of the underlying model. In addition, it provides adaptative modes of operation through which the assistive interface can be **utilized on low-capacity edge devices in areas of limited or no internet connectivity.**

3. Background

INMT-Lite's features have three conceptual dependencies: (a) Interactive Neural Machine Translation (§3.1) (b) Post-Training Quantization (§3.2) and (c) Knowledge Distillation (§3.2). An overview of the general operation pipeline for INMT-Lite is provided in 1.

3.1. Interactive Neural Machine Translation

In a neural machine translation pipeline, a combination of encoder-decoder is used to *encode* the semantic information of an input in one language, which is reconstructed by the *decoder* in the target language. At time step *t*, the probability of generating an output token y_t corresponding to an input x, is conditioned on all previously generated tokens $y_t, y_{t-1} .... y_1$; It is represented as:

$$p(y_t | y_1, ... y_{t-1}, x) = g(y_{t-1}, e_t) \qquad (1)$$

where *g* is a non-linearity that models the encoder hidden state, e_t and the previous token, y_{t-1}. In interactive NMT, instead of conditioning the model prediction y_t on the model's previous inputs, $y_t, y_{t-1}, ... y_1$, we instead model the inputs provided by the user $u_t, u_{t-1}, ..., u_1$ and this modifies equation 1 to:

$$p(y_t | u_1, ... u_{t-1}, x) = g(u_{t-1}, e_t) \qquad (2)$$

All of INMT-Lite's interfaces use equation (2) to incorporate user intent while providing translation for a source sentence. This allows us to dynamically update the suggestions we present to the user through our interfaces, much like other INMT services Santy et al. (2019).

3.2. Compression of MLMs: Post-Training Quantization and Knowledge Distillation

Since INMT-Lite is an offline service, the assistive model has to be ported to the user's device. This introduces the unique challenge of adapting MMLMs to edge resource constraints. To tackle this, the backend model's development pipeline supports the compression of a class of MMLMs by different degrees; depending upon the user's edge device's capacity and the target language. We train our base models on a range of languages, ranging from low-resource languages concretely defined as *LRL*, i.e., languages that have parallel data between 25K-1M data instances and *MRL* i.e., languages having parallel data between 1M-4M data instances. Being a pre-trained model, mT5 is best adapted for *LRL*, whereas the vanilla transformer is a viable architecture for *MRL*.

Post-Training Quantization We use post-training quantization to convert the weights and activations of the model to 8-bit integers (int-8) after training or fine-tuning the model to full precision (fp-32). We observe varying degrees of performance loss due to quantization, with the MRL taking the most significant hit in performance whilst the LRL seeing comparable performance after quantization. Models used in this mode can range from 75MB to 400MB, depending upon the original architecture being quantized.

Hard Distillation We follow Hinton et al. (2015) in distilling a deeper, accurate teacher model by first training it for the target language pair translation. Then, this teacher model generates a larger labeled corpus for monolingual data that it has not seen and this generated paired data can then be used to train a shallower student model.In addition to using language-specific vocabulary embeddings to improve target-language performance, distillation allows us to control finer details of the deployed models, like using a language-specific tokenizer.

Compression through these mechanisms gives rise to three distinct modes of operation in INMT-Lite, described in further detail in §4.2. We use HuggingFace[1] and TFLite to train and generate offline graphs for our backend models. Furthermore, a more detailed language analysis of the performance of both the distilled and quantized models can be found in Table 1 and 2.

3.3. Compressed Models for Languages available for INMT-Lite

In order to establish the feasibility of the compression methods we adopted, we trained and compressed models for 8 languages - for Punjabi, Gujarati, Marathi, Hindi, Bengali, Assamesse, and Odia, we use the Samanantar dataset to train the models Ramesh et al. (2022). Base models are fine-tuned for the configuration specified in Xue et al. (2020). Note that while Gondi is not included in the pretraining languages of mT5, it shares features with several languages in the pretraining corpus of mT5. Hence, we expect that transfer learning, in addition to the language-specific fine-tuning that we perform, generates a reasonable translation model Wang et al. (2022). For Gondi fine-tuning, we utilize a publicly available dataset that is open-source by CGNET Swara. Tables 1 and 2 summarize the performance of the approaches on seven of the languages chosen in addition to Gondi. For our medium resource languages, we quantized the vanilla transformer (without a pre-trained architecture), while for Gondi, we quantized and distilled mt5.

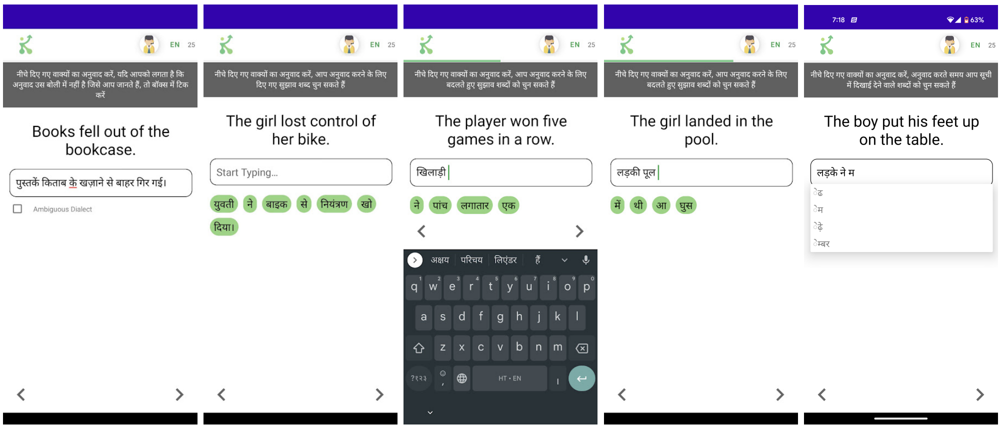
Language	Data	M		Q(M)	
		BLEU	***chrF***	***BLEU***	***chrF***
Punjabi	2.4M	38.4	50.6	27.0	48.0
Gujarati	3.0M	35.9	53.4	28.4	51.4
Marathi	3.3M	27.5	52.7	11.1	40.8
Bengali	8.4M	24.9	46.8	11.4	35.1
Hindi	8.5M	37.7	59.9	27.1	44.9

Table 1: Collection of Languages available with INMT-Lite's Quantized Mode. Here **M** is the best model trained for the language, and **Q(M)** is the quantized variant of the best model, **M**. The source language for all these experiments is English.

Language	Data	M		D(M)	
		BLEU	***chrF***	***BLEU***	***chrF***
Gondi	25K	14.3	32.5	14.2	32.8
Assamesse	140K	10.4	30.4	9.6	27.4
Odia	1M	27.4	47.6	20.2	40.7

Table 2: Collection of Languages available with INMT-Lite's Distilled Mode: Here **M** is the best model trained for the language and **D(M)** is the distilled variant of the best model, **M**. The source language for translation here is Hindi.

[1] HuggingFace MT5 model card



नीचे दिए गए वाक्यों का अनुवाद करें, यदि आपको लगता है कि अनुवाद उस बोली में नहीं है जिसे आप जानते हैं, तो बॉक्स में टिक करें

Books fell out of the bookcase.

पुस्तकें किताब के खज़ाने से बाहर गिर गई।

□ Ambiguous Dialect

नीचे दिए गए वाक्यों का अनुवाद करें, आप अनुवाद करने के लिए दिए गए सुझाव शब्द चुन सकते हैं

The girl lost control of her bike.

Start Typing...

नीचे दिए गए वाक्यों का अनुवाद करें, आप अनुवाद करने के लिए बदलते हुए सुझाव शब्दों को चुन सकते हैं

The player won five games in a row.

खिलाड़ी

नीचे दिए गए वाक्यों का अनुवाद करें, आप अनुवाद करने के लिए बदलते हुए सुझाव शब्दों को चुन सकते हैं

The girl landed in the pool.

लड़की पूल

नीचे दिए गए वाक्यों का अनुवाद करें, अनुवाद करते समय आप सूची में दिखाई देने वाले शब्दों को चुन सकते हैं

The boy put his feet up on the table.

लड़के ने म

(a) PE (b) SBOW (c) DBOW (d) NWBOW (e) NWD

Figure 2: Suite of all the Interfaces available in INMT-Lite. Each interface is intended to provide a different granularity of assistance to accommodate the underlying model's accuracy in the interface's interaction with the user and can be powered by any of the chosen modes (Native, Quantized, or Distilled).

4. System Description: INMT-Lite

In this section, we describe the suite of assistive interfaces we provide for the users (§4.1), the specifications of the pretrained architectures used to generate the backend models (§4.2) and the available modes of operation in INMT-Lite (§4.3).

4.1. Overview of Interface Design

Previous work Li et al. (2022); Krause and Vossen (2020); Santy et al. (2019) has motivated the notion that interactive interfaces may need to adapt the amount and structure of interaction depending upon the quality of the underlying assistance model. With aligned motivation, we provide INMT-Lite s five interfaces, which provide varying degrees of intrusion, latency, and information density. Figure 2 demonstrates all these interfaces, and they are briefly described as follows:

1. **Post-Edit (PE)**: Users provide translations after editing the initial sentence-level recommendations or *gists* that are provided through the back-end model.

2. **Static Bag Of Words (SBOW)**: Users provide translations while being able to see the model's most likely translation as a Bag of Words. Here, the BoW will be provided by the online mode only, and suggestions shall be ported to the device when the sentences are distributed to the users.

3. **Dynamic Bag of Words (DBOW)**: Users provide translations while being able to see sentence-level translations as a BoW. The suggestions provided to the user will continue to change depending on which suggestions the user decides to choose. The intention here is to model the user's input into the suggestion that the model is providing.

4. **Next Word Bag of Words (NWBOW)**: Users provide translations while they see next-word suggestions, i.e. the model's top-k sampled tokens, via a BoW panel. We use k = 5 for all system evaluations.

5. **Next Word Dropdown (NWD)**: Users provide translations while seeing next-word suggestions via a dropdown.

4.2. Backend Models

INMT-Lite's current version supports two primary architectures, which can be trained or fine-tuned for the target language of preference. These include the vanilla seq2seq Vaswani et al. (2017) transformer with 6 Encoder and Decoder Layers and mT5-small Xue et al. (2020) with 8 Encoder and Decoder Layers, vocabulary size - 250,100 and 6 attention heads. To develop our models, we replicate the distillation mechanisms explored systematically in Diddee et al. (2022) because it fits the context of data sparsity in which we compress our models.

4.3. Deployment Modes

The three modes of deployment in INMT-Lite can be chosen considering two factors, architecture of the back-end model, and latency requirements.

1. **Native Mode** An uncompressed model is converted to a static graph and pushed on edge. Since the smallest pre-trained model in our consideration is 1.2GB, pre-trained models

Task Name	Interface	Task Description
Baseline	Default	Users provide translations without any assistance.
Assistive	Default, Bag of Words and Dropdown	Users provide translations by post-editing or using the model's assistance via each of the assistive interfaces.
Scoring	DA Scoring	Users score the translations generated by users. The highest ranked translation here will then further be marked to identify the best mode.

Table 3: Overview of all tasks that are used for the evaluation of the system.

cannot be deployed in this mode. However, vanilla transformer architectures (which range from 180MB to 240MB, depending upon the chosen vocabulary embedding) can be deployed in this mode. Since the model is not compressed at all, the edge-version of the model has no observable loss in accuracy in this mode.

2. **Quantized Mode** The backend model is quantized by 4x depending and then ported to the user's device.

3. **Distilled Mode**: The back-end model is distilled to achieve a size compression between 8x and 12x. Models in this mode occupy approximately 180 - 240MB of on-disk memory once they are ported onto the mobile device.

Decoding Currently, all models use greedy decoding for their inference, except when providing suggestions using the Static Assistance formats (where the recommendations are not computed on edge) and hence, can utilize beam-search (depth = 2) for the suggestion generation.

4.4. Adaptations for Low-Resource Languages

Since INMT-Lite is intended to be predominantly used with low-resource languages - we employed a few features to circumvent typical constraints in under-resourced language data collection. These included (a) **Transliteration Support in Keyboards for supporting Native Script data**: INMT-Lite uses the Google Keyboard to transliterate non-native script inputs into the target language script. (b) **Option for Marking Dialectally Heterogenous Sentences**: Low-resource languages can display a wide range of dialectical heterogeneity, and models trained on such data can generate dialectally ambiguous suggestions. To evaluate the backend model's tendency to do so at run-time, we offer the option to mark sentences (Figure 2) that annotators are not confident of annotating if they feel that the provided suggestions might be a valid output in a dialect that they knew of but did not speak in.

5. User-Evaluation Setup

Since the use of INMT services has not been explored with extremely low-resource languages, we focus on evaluating our system's efficacy with Gondi, a severely under-resourced language spoken in the Indian subcontinent. Through a collaboration with 18 native Gondi speakers who participate as annotators for our study, we organize eight tasks, six of which require the annotators to use all interfaces[2] ***to translate*** the given sentences. The last two tasks require the users ***to score*** the translations generated through all interfaces. We use Direct Assessment (DA) scores (Specia et al., 2020) to understand the quality of the translations generated through each interface. A brief description of these tasks is provided in Table 3 and detailed descriptions of these tasks can be found in Table 5 of the Appendix. We collect *at least* 3 annotations per sample to improve the consistency of our preference and quality assessment. Additional details about the annotation compensations §A.4 and instructions used for our annotations §A.2.2, and the strategy and precautions used while distributing sentences can be found in the Table 6 of the Appendix.

6. Results

We collate our empirical and user-study evaluations to answer the following questions: **RQ1**: Does INMT-Lite lead to a reduction in human effort? (§6.1) **RQ2**: How well do the translations generated by INMT-Lite compare with those generated without assistance? (§6.2) and,**RQ3**: Does INMT-Lite improve the experience of annotators during data generation? (§6.3)

6.1. RQ1: Does INMT-Lite lead to a reduction in Human Effort during the Data Generation Process?

To understand whether INMT-Lite can streamline data generation time and effort, we investigate the

[2] including a setup where no assistance was provided to create as a baseline for quality evaluation

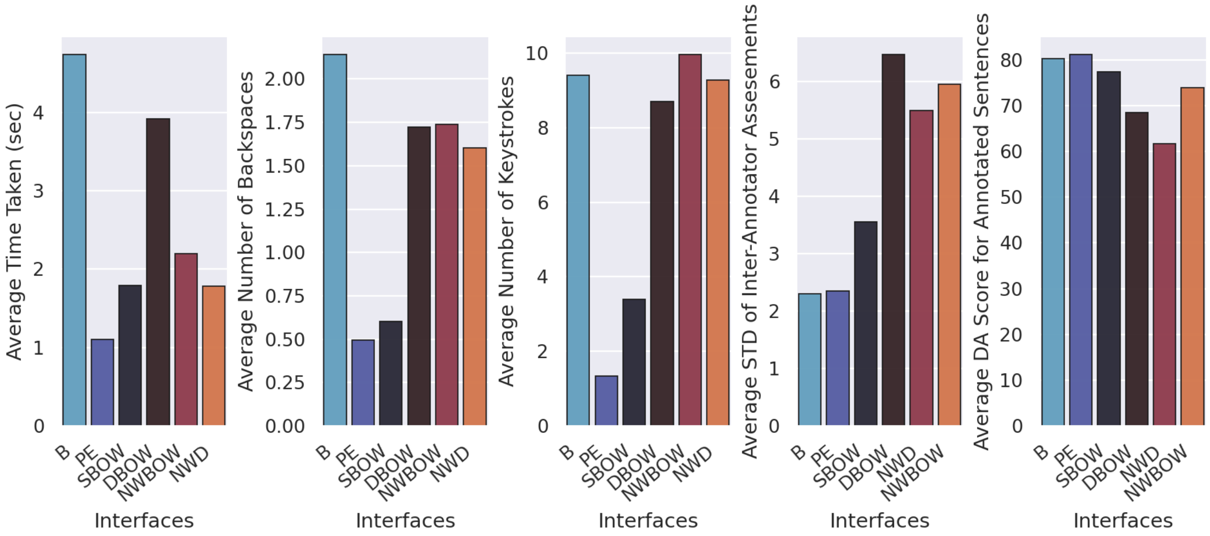


Figure 3: Summary Statistics for INMT-Lite's Evaluation: Comparing the time variation, keystroke load with and without INMT-Lite's assistance; The interannotator evaluation standard deviation and quality of the translations generated with INMT-Lite. All metrics except DA are lower than better. The PE, BOW interfaces for INMT-Lite show consistent gains in time and effort reduction without degrading sentence quality significantly.

(a) keystroke load with and without assistance and the (b) time taken to generate these translations.

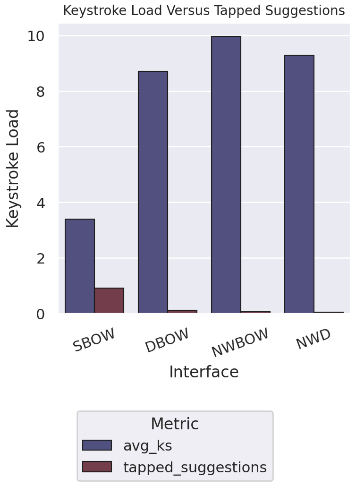
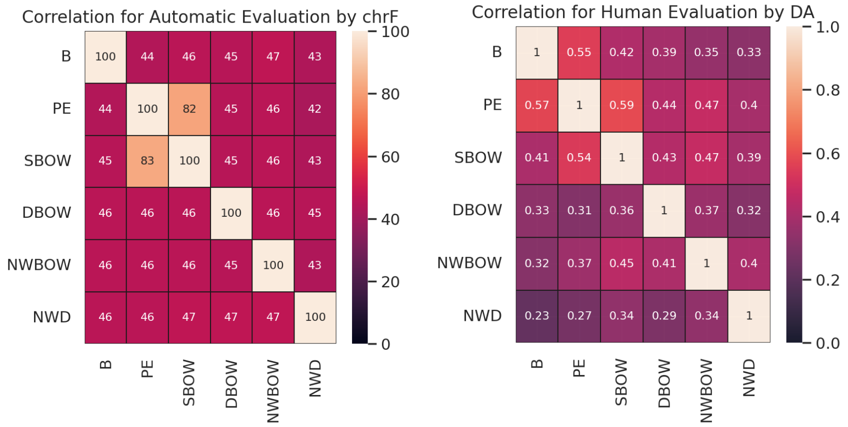


Figure 4: Average Tapped Suggestions vs. Total Keystroke Load for interfaces: SBOW has the highest affinity towards suggestion acceptance: which also reflects in the user's significantly reduced keystroke load. Next-Word interfaces have the least affinity to suggestion acceptance due to their limited breadth-wise coverage.

Keystroke Load We consider the keystroke load to represent the manual effort required to participate in the data generation activity. The keystrokes made to select, edit, or add to the offered suggestion are considered the keystroke load for any interface. Figure 3 (b) and (c) show the average number of keystrokes and backspaces used by users when annotating with each interface. In both cases, the keystroke load is the highest for baseline generation compared to all other interfaces. Among the assistive interfaces, the Next word interfaces have the worst keystroke load. Upon probing this qualitatively, we find that the limiting breadth-wise coverage, i.e., the number of future token suggestions, is counterproductive for users (more details in §6.3). We also monitor the *number of suggestions tapped or suggestion opt-ins* as a proxy of the usefulness of the suggestions. Figure 4 summarizes the average number of accepted suggestions per interface relative to the total keystroke load. We observe that SBOW shows high proclivity to being accepted while the dropdown and next-word interfaces have a much lower rate of acceptance. The annotators mentioned that the dropdown interface's higher degree of intrusion combined with the latency of its suggestion provision, reduced its usability to some extent. This partially describes the low rate of suggestion acceptance for the Dropdown interface as well. We discuss further reasons for the same in §6.3.

Time Taken to Generate Translations Figure 3 shows the difference in the time taken to generate translations with and without assistance. We see a clear reduction in the amount of time taken to generate translation if users are provided INMT-Lite's assistive interfaces. As discussed further in §6.2, we do not see any significant decrease in the



(a) Pair-Wise chrF between Interfaces

(b) F1-score for DA evaluations

Figure 5: F1 correlation and pair-wise chrF evaluation between translations generated through all interfaces. We observe a moderate correlation through both estimations, indicating that the generated translations have observable diversity while maintaining both semantic (evaluated by human evaluations) and structural similarity (evaluated by an automatic metric)

quality of the translation with this reduction in time, thus strongly indicating INMT-Lite's usefulness.

Takeaway: Despite having a relatively low-quality, noisy back-end model, INMT-Lite appears to be more efficient for time and manual effort axes. Seeing the relatively lower gains with Next-Word interfaces, we recommend that language technologists prioritize interfaces with higher breadth-wise coverage: namely PE, SBOW and DBOW when there aren't very strong quality guarantees on the inferences generated by the backend model (as with LRL).

6.2. RQ2: How well do the translations generated with INMT-Lite compare with those generated without assistance ?

An important consideration of using INMT-Lite is to balance the amount of manual effort employed with the effort used to generate the translation with the quality of the provided translation.

Quality of the Generated Translation We compute the average DA scores of the translations generated through each mode and interface to understand the quality of the translations generated through each interface. From Figure 3, we observe that the PE interface gives the most competitive gains in sentence quality, while SBOW gives only an insignificant reduction in sentence quality. Note that the DA scoring system has an average range of 20 points for an interpretation of a score. Bearing this in mind, we can also see that generations from all interfaces lie in this range, indicating that the semantic quality of the generated translation does not take any observable hit due to the use of assistive translations. This, coupled with the reduced time and keystroke load, strongly speaks for the efficacy of the assistive interfaces.

Diversity of the Generated Translation To comment on the diversity of the translations generated through each interface, we compute the pairwise chrF correlation Popović (2015) and the DA Score correlation between the translations generated through each interface. To compute the correlation of the DA score, we have each translation scored by 3 annotators and then compute the pairwise F1 score, Cohen's Kappa (Inter Annotator agreement) for their scores 5 and §7. We find fair to moderate inter-annotator agreement between the annotators. Computing these correlations also gives us preliminary information against the following questions: (Q1) Since the model can only give a limited variety of suggestions, **do the users get biased against the translations they generate**, ultimately generating very similar, low quality translations? Furthermore, (Q2) if users are using the system maliciously, are they simply accepting suggestions irrespective of their correctness? Both these questions are partially answered by checking the correlation between the generated translations; that is, a very high correlation between the translations generated by each of the interfaces would satisfy both of the presented questions. Conversely, a very low correlation between the sentences generated through each interface would call for revisiting the quality of the generated translations, as the same input would not be expected to generate extremely uncorrelated translations on average. Encouragingly, we find a moderate-fair correlation between the translations generated through all the interfaces using both automatic and human-evaluation metrics. This provided evidence of the syntactic (prioritized by the automatic metric) and semantic similarity of the generated translations.

To further investigate the impact of using the assistive interfaces on token diversity, we also investigated the Unique Token Coverage and Token Overlap for the translations generated by all interfaces. For token overlap, we computed the intersection between the translation token set of the target interface with the token set of all other interfaces. A high unique token count and a low overlapping token count would suggest higher diversity (and consequently, low-interface-induced bias). Figure 6 summarizes this comparison: We do not see a large variation in the number of unique tokens for all interfaces. We do see a higher rate of overlap for PE and SBOW; Toward this, annotators reported that they found it more cumbersome to make nuanced edits to translations for PE, where the entire translation would already be visible. Consequently, their tendency to edit was slightly attenuated, which could explain the low unique token count observed with PE.

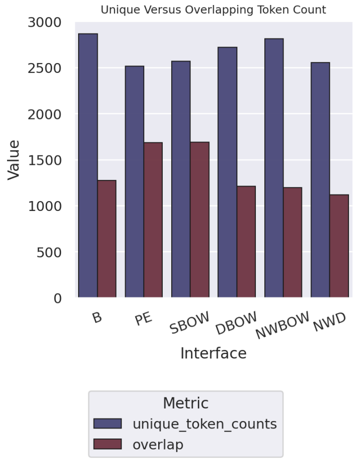


Figure 6: Unique Token Count and Overlapping Token Count for all interfaces: The maintained count of unique tokens and the low token overlap across interfaces denotes that the token diversity of the generated translations is maintained.

Takeaway: INMT-lite's interfaces maintain sentence quality and token diversity of the translations. Additionally, interfaces that call for granular edits to full-length translations over token-level re-edits to partial translations are less conducive to overall diverse generations. Hence, if generation diversity is a priority of the task at hand, we recommend using Next-Word interfaces, which show a higher proclivity to diverse generations.

6.3. Qualitative Observations from the System's Evaluation

During and after our user study, our annotators shared feedback on the efficacy of the interfaces we enlist in this section.

Low Affinity to Tapping Suggestions Annotators mentioned that their incidence of tapping suggestions was relatively low because tapping on the suggestion often disrupted their typing flow. Instead, they preferred to see the suggestion and type the expected token. This, to some extent, explained the poor tapping suggestion ratios that we observed in Figure 4. Annotators also reported that their tapped suggestions were relatively low for assistive interfaces as the provided suggestions guided their initial generation. Specifically, they had a higher chance of spelling the target token rightly if they had a suggestion (even if it was spelled wrongly), jump-starting their translation for the sentence.

Breadth-Wise is Better than Depth-Wise Coverage Annotators reported that breadth-wise coverage interfaces like SBOW, DBOW, and PE were generally more helpful than Next-Word Interfaces. Additionally, they mentioned that Next-Word interfaces took longer to parse since they had to go through all the possible options and there were instances where none of these suggestions would actually be the correct token. Finally, they also mentioned that especially for the Next Word interfaces - they often chose to type out the suggestion and not tap on it - as the overhead of sequentially tapping then making granular edits was less productive *per position*.

7. Conclusion

Acknowledging that representative data collection in under-resourced languages requires tools curated towards the community and languages needs, we present INMT-Lite: An internet-independent, edge-oriented interactive neural machine translation service. Through an extensive user study with native speakers of a severely under-resourced language, Gondi, we show that INMT-Lite boosts translation productivity whilst maintaining sentence quality and diversity. Additionally, using a combination of qualitative analysis of user experience and quantitative investigation of operational feasibility, we offer recommendations for language technologists when developing assistive technologies for data generation under similar constraints.

8. Future Work

INMT-Lite's interface has a vast set of parameters that need deeper investigation: Attributes such as a) the depth of decoding, b) the number of suggestions shown across each structure, and c) the trigger of invocation (when should suggestions be generated per interface if not at a token-level) are expected to affect the efficacy and consequently preference of the users.

9. Acknowledgements

We thank Shubhranshu Choudhury from CGNET Swara for giving us the opportunity and support to collaborate with members of the Gondi community. More crucially, we thank all the members of the Gondi community who gave us their valuable time and feedback in developing INMT-Lite. Finally, we also thank Karya Inc. and Microsoft Research India for providing financial support to further this research.

AI4Bharat, Jay Gala, Pranjal A. Chitale, Raghavan AK, Sumanth Doddapaneni, Varun Gumma, Aswanth Kumar, Janki Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, Mitesh M. Khapra, Raj Dabre, and Anoop Kunchukuttan. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages.

Mat Bettinson and Steven Bird. 2017. Developing a suite of mobile applications for collaborative language documentation. In *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 156–164, Honolulu. Association for Computational Linguistics.

Steven Bird. 2018. 842Designing Mobile Applications for Endangered Languages. In *The Oxford Handbook of Endangered Languages*. Oxford University Press.

Harshita Diddee, Sandipan Dandapat, Monojit Choudhury, Tanuja Ganu, and Kalika Bali. 2022. Too brittle to touch: Comparing the stability of quantization and distillation towards developing low-resource MT models. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 870–885, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Kamal Gupta, Dhanvanth Boppana, Rejwanul Haque, Asif Ekbal, and Pushpak Bhattacharyya. 2021. Investigating active learning in interactive neural machine translation. In *Proceedings of Machine Translation Summit XVIII: Research Track*, pages 10–22, Virtual. Association for Machine Translation in the Americas.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

David Jurgens, Yulia Tsvetkov, and Dan Jurafsky. 2017. Incorporating dialectal variability for socially equitable language identification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 51–57, Vancouver, Canada. Association for Computational Linguistics.

Lea Krause and Piek Vossen. 2020. When to explain: Identifying explanation triggers in human-agent interaction. In *2nd Workshop on Interactive Natural Language Technology for Explainable Artificial Intelligence*, pages 55–60, Dublin, Ireland. Association for Computational Linguistics.

Tsz Kin Lam, Shigehiko Schamoni, and Stefan Riezler. 2019. Interactive-predictive neural machine translation through reinforcement and imitation. In *Proceedings of Machine Translation Summit XVII: Research Track*, pages 96–106, Dublin, Ireland. European Association for Machine Translation.

William Lane and Steven Bird. 2021. Local word discovery for interactive transcription. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2058–2067, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

William Lane and Steven Bird. 2022. A finite state aproach to interactive transcription. In *Proceedings of the first workshop on NLP applications to field linguistics*, pages 1–10, Gyeongju, Republic of Korea. International Conference on Computational Linguistics.

Jaesong Lee, Joong-Hwi Shin, and Jun-Seok Kim. 2017. Interactive visualization and manipulation of attention-based neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 121–126, Copenhagen, Denmark. Association for Computational Linguistics.

Zichao Li, Prakhar Sharma, Xing Han Lu, Jackie Chi Kit Cheung, and Siva Reddy. 2022. Using interactive feedback to improve the accuracy and explainability of question answering systems post-deployment. *ArXiv*, abs/2204.03025.

Daniel J. Liebling, Michal Lahav, Abigail Evans, Aaron Donsbach, Jess Holbrook, Boris Smus, and Lindsey Boran. 2020. Unmet needs and opportunities for mobile translation ai. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–13, New York, NY, USA. Association for Computing Machinery.

Ayush Maheshwari, Ajay Ravindran, Venkatapathy Subramanian, and Ganesh Ramakrishnan. 2023. Udaan - machine learning based post-editing tool for document translation. In *Proceedings of the 6th Joint International Conference on Data Science amp; Management of Data (10th ACM IKDD CODS and 28th COMAD)*, CODS-COMAD '23, page 263–267, New York, NY, USA. Association for Computing Machinery.

Devansh Mehta, Harshita Diddee, Ananya Saxena, Anurag Shukla, Sebastin Santy, Ramaravind Kommiya Mothilal, Brij Mohan Lal Srivastava, Alok Sharma, Vishnu Prasad, U. Venkanna, and Kalika Bali. 2022. Learnings from technological interventions in a low resource language: Enhancing information access in gondi. *ArXiv*, abs/2211.16172.

Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi Fasubaa, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddeen Muhammad, Salomon Kabongo Kabenamualu, Salomey Osei, Freshia Sackey, Rubungo Andre Niyongabo, Ricky Macharm, Perez Ogayo, Orevaoghene Ahia, Musie Meressa Berhe, Mofetoluwa Adeyemi, Masabata Mokgesi-Selinga, Lawrence Okegbemi, Laura Martinus, Kolawole Tajudeen, Kevin Degila, Kelechi Ogueji, Kathleen Siminyu, Julia Kreutzer, Jason Webster, Jamiil Toure Ali, Jade Abbott, Iroro Orife, Ignatius Ezeani, Idris Abdulkadir Dangana, Herman Kamper, Hady Elsahar, Goodness Duru, Ghollah Kioko, Murhabazi Espoir, Elan van Biljon, Daniel Whitenack, Christopher Onyefuluchi, Chris Chinenye Emezue, Bonaventure F. P. Dossou, Blessing Sibanda, Blessing Bassey, Ayodele Olabiyi, Arshath Ramkilowan, Alp Öktem, Adewale Akinfaderin, and Abdallah Bashir. 2020. Participatory research for low-resourced machine translation: A case study in African languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2144–2160, Online. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Mahalakshmi J, Divyanshu Kakwani, Navneet Kumar, Aswin Pradeep, Srihari Nagaraj, Kumar Deepak, Vivek Raghavan, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh Shantadevi Khapra. 2022. Samanantar: The Largest Publicly Available Parallel Corpora Collection for 11 Indic Languages. *Transactions of the Association for Computational Linguistics*, 10:145–162.

Shruti Rijhwani, Daisy Rosenblum, Michayla King, Antonios Anastasopoulos, and Graham Neubig. 2023. User-centric evaluation of ocr systems for kwak'wala.

Sebastin Santy, Sandipan Dandapat, Monojit Choudhury, and Kalika Bali. 2019. INMT: Interactive neural machine translation prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 103–108, Hong Kong, China. Association for Computational Linguistics.

Lucia Specia, Frédéric Blain, Marina Fomicheva, Erick Fonseca, Vishrav Chaudhary, Francisco Guzmán, and André F. T. Martins. 2020. Findings of the WMT 2020 shared task on quality estimation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 743–764, Online. Association for Computational Linguistics.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Ke Wang, Jiayi Wang, Niyu Ge, Yangbin Shi, Yu Zhao, and Kai Fan. 2020. Computer assisted translation with neural quality estimation and automatic post-editing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2175–2186, Online. Association for Computational Linguistics.

Xinyi Wang, Sebastian Ruder, and Graham Neubig. 2022. Expanding pretrained models to thousands more languages via lexicon-based adaptation. *arXiv preprint arXiv:2203.09435*.

Yanling Xiao, Lemao Liu, Guoping Huang, Qu Cui, Shujian Huang, Shuming Shi, and Jiajun Chen. 2022. BiTIIMT: A bilingual text-infilling method for interactive machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1958–1969, Dublin, Ireland. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

A. Appendix

A.1. Compatibility of Interfaces with Different Language Classes

Despite offering various interface and model-operation options, INMT-Lite was found to be unsuitable for certain combinations in terms of ease and efficiency of interactions. The reason for this was primarily the latency of inference associated with all interfaces when used with a backend model in a specific mode. Real-time interaction was not always feasible due to latency. For example, in the Quantized Mode, generating a single suggestion with a 400MB model (used for languages with the lowest resource) resulted in a latency of 0.6 seconds for a single inference. This latency could have a negative impact on the user experience in terms of suggestion generation. Consequently, we provide Table 4 to summarize our recommendations regarding the adoption of such combinations, taking into account the choice of the backend architecture. Note that we define compatibility as a measure of the latency of operating the interface (latency < 500 milliseconds per inference is considered acceptable for interaction).

A.2. User Study Specifications

In this section we present the following details: (a) the task setup, (b) the metrics and instructions employed for our annotations, and (c) the strategy utilized for distributing sentences, along with information about our interannotator evaluation.

A.2.1. Task Setup

The user study conducted for INMT-Lite involved working with native language speakers who participated in a collection of 8 tasks. Detailed descriptions of these tasks can be found in Table 5. The choice of these tasks was motivated by a pilot study presented in (Mehta et al., 2022). The pilot study involved three native members of the Gondi community and aimed to assess the quality of the Hindi-Gondi translation model, the usability of the Bag Of Words (BoW) interface, and the Quantized Mode of INMT's operation. The conclusions drawn from the pilot study were as follows: (a) The model's performance was tolerable, but not excellent, with errors in spellings and difficulties with longer sentences. (b) The efficacy of the BoW interface was evident, indicating its viability for further investigation across all modes. Taking these findings into consideration, the decision was made to test four interfaces in addition to the baseline and post-edited modes of interaction. Furthermore, the use of distilled models was fixed in this system, as distilled models could leverage a language-specific tokenizer.

A.2.2. Metrics and Instructions for Annotation

For the scoring tasks in our study (refer to Table 5), we employ the Direct Assessment method described in (Specia et al., 2020) to evaluate the sentences. We make the evaluation guidelines for this metric easily accessible through our interface. An example of these guidelines can be seen in Figure 7. In the scoring interface, we use a slider-like scale, as shown in Figure 7b.

A.2.3. Distribution Strategy

To prevent cognitive bias and ensure unbiased evaluations, we implemented several safeguards to avoid having the same sentence evaluated by an annotator through different interfaces. This was done to prevent any potential impact on their interaction with other interfaces and their perception of the translations.

Let s_i represent the i^{th} set of sentences, where each set s_i contains 36 sentences. The annotator assigned to annotate a particular set is denoted a_i. The notation (s_k, i_k) represents a pair of sentences s_k annotated through the k^{th} interface. The distribution strategy for all data collection tasks is

Mode* × *Interface Overview	**Interfaces**				
	PE	SBOW	DBOW	NWBOW	NWD
Modes					
Native	↗	↗	↗	↗	↗
Quantized	↗↗	↗↗	↗	↗	↗
Distilled	↗↗	↗↗	↗↗	↗↗	↗↗

Table 4: Overview of Inference Feasibility evaluated as a function of mode and its operation in an interface. A single arrow denotes whether the mode supports offline inference, while a double arrows denotes if the mode supports dynamic and offline inference.

Task Name	**Interface**	**Task Description**
Baseline	Default	Users provide translations without any assistance.
Post-Edited	Default	Users provide translations after editing an initial sentence level recommendation (gist).
Static-BoW	Bag of Words	Users provide translations while they are shown the model's single, top-most sentence-level recommendation as a bag of words.
Dynamic-BoW	Bag of Words	Users provide translations while they are shown sentence-level translations as a BoW. The suggestions provided to the user are updated depending on the user's latest edits and the current state of the translation.
Next-Word-BoW	Dropdown	Users provide translations while they are given next-word suggestions via a BoW panel
Next-Word-Dropdown	Dropdown	Users provide translations while they are given next-word suggestions via a dropdown
Scoring for the Best Interface	DA Scoring	This task will have users score the translations generated by users in Task 3, 4, 5, 6. The highest ranked translation here will be further used in Task 9.
Scoring for the Best Mode	DA Scoring	This task will have users score the translations from Task 1, 2 and 7.

Table 5: Description of all tasks that are used for the evaluation of the system.

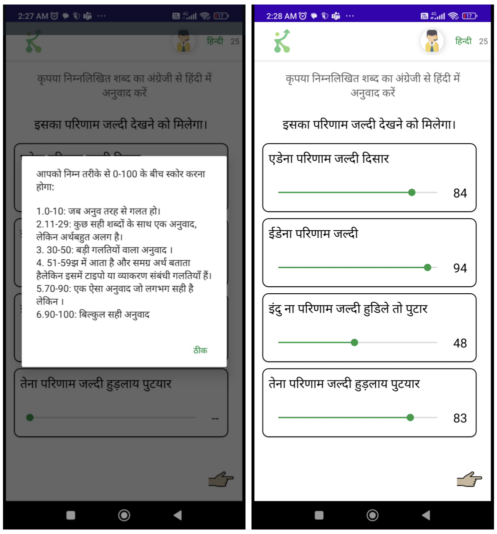
Annotator × Task	a_1	a_2	a_3	a_4	a_5	a_6
t_1	(s_1, i_1)	(s_1, i_2)	(s_1, i_3)	(s_1, i_4)	(s_1, i_5)	(s_1, i_6)
t_2	(s_2, i_2)	(s_2, i_3)	(s_2, i_4)	(s_2, i_5)	(s_2, i_6)	(s_2, i_1)
t_3	(s_3, i_3)	(s_3, i_4)	(s_3, i_5)	(s_3, i_6)	(s_3, i_1)	(s_3, i_2)
t_4	(s_4, i_4)	(s_4, i_5)	(s_4, i_6)	(s_4, i_1)	(s_4, i_2)	(s_4, i_3)
t_5	(s_5, i_5)	(s_5, i_6)	(s_5, i_1)	(s_5, i_2)	(s_5, i_3)	(s_5, i_4)
t_6	(s_6, i_6)	(s_6, i_1)	(s_6, i_2)	(s_6, i_3)	(s_6, i_4)	(s_6, i_5)

Table 6: Distribution Strategy for INMT-Lite's study: We ensure that no user annotator uses two interfaces to annotate the same sentence to avoid any cognitive bias from confounding their interaction with the interface that they use.

described in Table 6. To ensure that the same annotator does not score a sentence they provided, the system flagged each provided sample with the user who provided it. During the redistribution of samples for scoring, any flagged user was excluded from the pool of available annotators.

A.3. Inter Annotator Evaluation

Table 7 presents the results of the pairwise evaluation of the interannotator for our annotations. We consistently observe a fair to moderate correlation in all of our evaluations. However, we do observe a degradation in correlation (random-low correlation) between one pair of annotators for the Baseline and PE interfaces. Annotators have mentioned that such divergences could be attributed to the dialectal heterogeneity among the annotators themselves. This heterogeneity may particularly affect the baseline translation scores, as the initial translations may have been generated by an annotator with a different dialectal preference.



2:27 AM

हिन्दी 25

कृपया निम्नलिखित शब्द का अंग्रेजी से हिंदी में अनुवाद करें

इसका परिणाम जल्दी देखने को मिलेगा।

आपको निम्न तरीके से 0-100 के बीच स्कोर करना होगा:

1.0-10: जब अनुव तरह से गलत हो।
2.11-29: कुछ सही शब्दों के साथ एक अनुवाद, लेकिन अर्थबहुत अलग है।
3. 30-50: बड़ी गलतियों वाला अनुवाद ।
4. 51-59झ में आता है और समग्र अर्थ बताता हैलेकिन इसमें टाइपो या व्याकरण संबंधी गलतियाँ हैं।
5.70-90: एक ऐसा अनुवाद जो लगभग सही है लेकिन ।
6.90-100: बिल्कुल सही अनुवाद

ठीक

तेना परिणाम जल्दी हुड़लाय पुटयार --

2:28 AM

हिन्दी 25

कृपया निम्नलिखित शब्द का अंग्रेजी से हिंदी में अनुवाद करें

इसका परिणाम जल्दी देखने को मिलेगा।

एडेना परिणाम जल्दी दिसार 84

ईडेना परिणाम जल्दी 94

इंदु ना परिणाम जल्दी हुडिले तो पुटार 48

तेना परिणाम जल्दी हुड़लाय पुटयार 83

(a) Annotation Instructions (b) Scoring Interface

Figure 7: Annotation and Scoring interfaces provided while users participated in the scoring activity. The instructions are translated by native speakers and can be accessed at any point during the collection.

Interface	IAA Metric	Pair-Wise Evaluations		
B	κ	0.422	0.324	0.06
	F1	0.72	0.58	0.42
PE	κ	0.421	0.394	0.134
	F1	0.72	0.64	0.49
SBOW	κ	0.431	0.435	0.157
	F1	0.70	0.61	0.49
DBOW	κ	0.325	0.380	0.122
	F1	0.59	0.51	0.40
NWBOW	κ	0.347	0.318	0.074
	F1	0.60	0.47	0.33
NWD	κ	0.421	0.392	0.167
	F1	0.60	0.47	0.32

Table 7: Inter-Annotator Evaluation

A.4. Cost of Annotation

In accordance with the expected median wage for high-skill tasks, we paid our annotators INR 10 for each translation task (all interfaces including the baseline) and INR 15 for each scoring task. We defined our payment scheme keeping in mind the professional rates of compensation provided to skilled translators: Rs 1.25 per word, that is, Rs 15 for a sentence with 9-12 words.