PRALEKHA: Cross-Lingual Document Alignment for Indic Languages

Sanjay Suryanarayanan[1] **Haiyue Song**[3]
Mohammed Safi Ur Rahman Khan[1,4] **Anoop Kunchukuttan**[1,2] **Raj Dabre**[1,3,4,5*]
[1]Nilekani Centre at AI4Bharat [2]Microsoft
[3]National Institute of Information and Communications Technology, Japan
[4]Indian Institute of Technology, Madras [5]Indian Institute of Technology, Bombay
huggingface.co/Pralekha github.com/Pralekha

Abstract

Mining parallel document pairs for document-level machine translation (MT) remains challenging due to the limitations of existing Cross-Lingual Document Alignment (CLDA) techniques. Existing methods often rely on metadata such as URLs, which are scarce, or on pooled document representations that fail to capture fine-grained alignment cues. Moreover, the limited context window of sentence embedding models hinders their ability to represent document-level context, while sentence-based alignment introduces a combinatorially large search space, leading to high computational cost. To address these challenges for Indic languages, we introduce PRALEKHA[1], a benchmark containing over 3 million aligned document pairs across 11 Indic languages and English, which includes 1.5 million English–Indic pairs. Furthermore, we propose Document Alignment Coefficient (DAC), a novel metric for fine-grained document alignment. Unlike pooling-based methods, DAC aligns documents by matching smaller chunks and computes similarity as the ratio of aligned chunks to the average number of chunks in a pair. Intrinsic evaluation shows that our *chunk-based method is 2–3× faster while maintaining competitive performance*, and that *DAC achieves substantial gains over pooling-based baselines*. Extrinsic evaluation further demonstrates that document-level MT models trained on DAC-aligned pairs consistently outperform those using baseline alignment methods. *These results highlight DAC's effectiveness for parallel document mining.* The dataset and evaluation framework are publicly available to support further research.

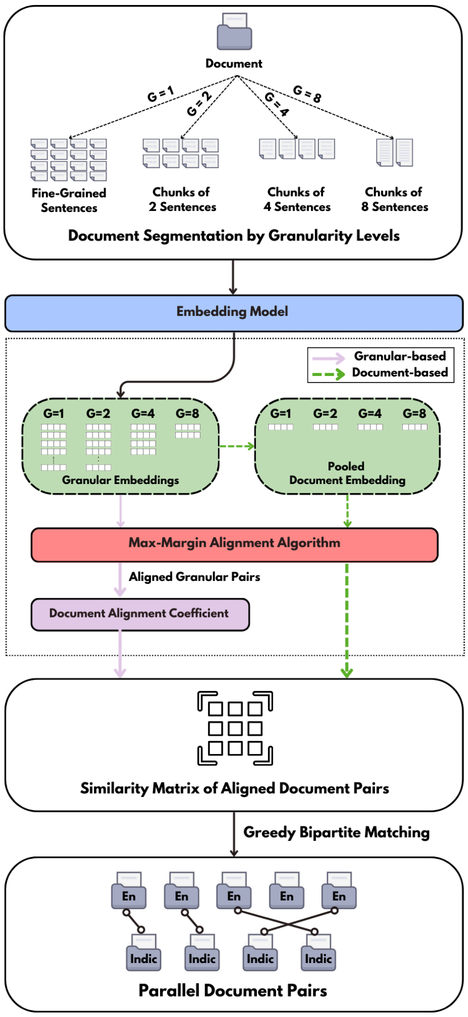

Figure 1: An Overview of the Evaluation Framework for Cross-Lingual Document Alignment. The **pink path** illustrates the proposed approach leveraging DOCUMENT ALIGNMENT COEFFICIENT (DAC), while the **green path** represents pooling-based baseline methods.

1 Introduction

The advent of Large Language Models (LLMs) has significantly improved the ability to process long-context information, enabling advancements in document-level tasks such as machine translation (MT). Document-level MT extends beyond sentence-level translation by capturing discourse-level dependencies, coherence, and contextual information across entire documents, leading to more

*Corresponding author: raj.dabre@cse.iitm.ac.in

[1]PRALEKHA means *document* in Sanskrit, an ancient Indo-Aryan language.

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 662–676

accurate and fluent translations. While there are efforts (Ramesh et al., 2022; Siripragada et al., 2020) that provide sentence-level parallel data, large-scale document-level parallel corpora remain scarce (Sannigrahi et al., 2023; Pal et al., 2024).

To address this gap, parallel document mining is essential for building high-quality training corpora for document-level MT. However, this progress is hindered by two key challenges: *(a) the lack of large-scale document alignment evaluation benchmarks and (b) the limitations of current Cross-Lingual Document Alignment (CLDA) techniques.* Reliable evaluation benchmarks are crucial because the effectiveness of an alignment method cannot be improved without accurate performance measurements. As for the limitations of CLDA, most existing embedding models were developed for sentence-level parallel corpus mining (Schwenk et al., 2020, 2021; Bañón et al., 2020). As a result, they have limited context windows and can represent only parts of a document (Zhu et al., 2024). Document-level embeddings are therefore often obtained by truncating documents or pooling embeddings of independently encoded chunks, both of which can be lossy (Sannigrahi et al., 2023). These approaches overlook chunk-level cues, which reduces their ability to capture complete document semantics. Even without pooling, the reliance of existing alignment techniques on sentence-level operations leads to combinatorial growth in the search space, making large-scale document alignment computationally expensive.

In this paper, we focus on Indic languages, which have abundant sentence-level parallel corpora and benchmarks (Ramesh et al., 2022; Siripragada et al., 2020; Gala et al., 2023) but remain low-resource for CLDA and Document-level MT. We first introduce PRALEKHA, a large-scale parallel document dataset for evaluating CLDA techniques across 11 Indic languages and English. PRALEKHA comprises over 3 million document pairs between 11 Indic languages and English, of which 1.5 million are English–Indic pairs. These are high-quality, human-verified aligned documents sourced from reliable domains such as the Indian Press Information Bureau (PIB) and the Mann Ki Baat (MKB) radio program. It serves both as a benchmark for assessing CLDA and as a domain-specific parallel corpus for training document-level MT models for Indic languages. Alongside PRALEKHA, we propose DOCUMENT ALIGNMENT COEFFICIENT (DAC), a novel metric for CLDA at the fine-granular level. Unlike existing approaches that rely on pooled document-level embeddings, we focus on smaller granular units, called chunks. We first embed and align these chunks, then compute the DAC of a document pair as the ratio of aligned chunks to the average number of chunks in the pair.

In our experiments using PRALEKHA under realistic noisy document mining settings, we evaluate the proposed DAC approach against existing pooling-based baselines. *DAC achieves average precision, recall, and F1 scores of 0.8932, 0.6312, and 0.7372, respectively, showing improvements of 15–20% in precision and 5–10% in F1 score over the baselines.* We further benchmark DAC on the CCAligned (El-Kishky et al., 2020) dataset, where it attains average precision, recall, and F1 scores of 0.8110, 0.6511, and 0.7203, respectively, outperforming the baselines across all metrics. Additionally, *our chunk-based method demonstrates a 2–3× speedup over sentence-based alignment techniques while maintaining competitive performance*. Having established DAC's intrinsic effectiveness, we conduct an extrinsic evaluation by training document-level MT models on mined documents using open-source LLMs. The results show that *MT models trained on DAC-aligned pairs consistently outperform those trained with baseline alignment methods*. The PRALEKHA dataset and CLDA evaluation framework is publicly available to support further research in this area.

2 Related Work

Cross-Lingual Sentence Alignment. Early approaches to sentence-level alignment, such as Europarl (Koehn, 2005), relied on manually curated metadata (Abdul-Rauf and Schwenk, 2009; Do et al., 2009). Recent advancements in multilingual embeddings have significantly improved large-scale mining of parallel data from unstructured sources. Methods such as LaBSE (Feng et al., 2022), LASER (Schwenk and Douze, 2017), SONAR (Duquenne et al., 2023), and hierarchical bilingual retrieval (Guo et al., 2019) leverage multilingual sentence representations to enhance cross-lingual alignment accuracy. These techniques have enabled large-scale efforts such as CCMatrix (Schwenk et al., 2020), WikiMatrix (Schwenk et al., 2021), ParaCrawl (Bañón et al., 2020), NLLB (Team et al., 2022), and BPCC (Gala et al., 2023), which extract and align sentence pairs from massive web crawls.

Cross-Lingual Document Alignment. For document-level alignment, early approaches relied on metadata rather than content, using features such as URL and page structure similarity (Resnik and Smith, 2003). While multilingual encoders have improved sentence-level parallel data extraction, document-level alignment remains challenging due to the limited context windows of existing sentence embedding models (Zhu et al., 2024). In this work, we explore a faster chunk-based method that effectively leverages fine-granular embeddings for document-level alignment.

Indic Language Corpora. Despite India's large population (1.43 billion), its languages remain low-resource, particularly for high-quality document-level data. Significant efforts have been made to build monolingual corpora, such as IndicNLP Suite (Kakwani et al., 2020) and Sangraha (Khan et al., 2024), as well as parallel corpora and benchmarks like IN22 and BPCC (Gala et al., 2023). However, no publicly available cross-lingual parallel document datasets exist for Indic languages (Sannigrahi et al., 2023). The only work referencing parallel documents uses them solely for mining sentence pairs, without releasing the document-level data (Siripragada et al., 2020). This underscores the urgent need to construct parallel document corpora for cross-lingual document alignment and document-level machine translation, a gap that our work aims to fill.

3 PRALEKHA

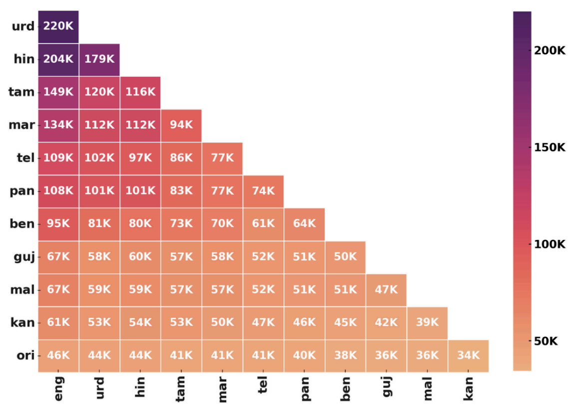

Figure 2: Heat-map of *alignable* document pairs for each language pair in PRALEKHA. Darker cells indicate higher alignment counts.

We present PRALEKHA a large-scale parallel document dataset for evaluating Cross-Lingual Document Alignment (CLDA) techniques. PRALEKHA comprises over 3 million document pairs between 12 languages - Bengali (ben), Gujarati (guj), Hindi (hin), Kannada (kan), Malayalam (mal), Marathi (mar), Odia (ori), Punjabi (pan), Tamil (tam), Telugu (tel), Urdu (urd) and English (eng) containing a mixture of high, medium and low resource languages. Figure 2 shows a heatmap of the distribution of *alignable* document pairs across language combinations. Among these, 1.5 million document pairs are English–Indic.

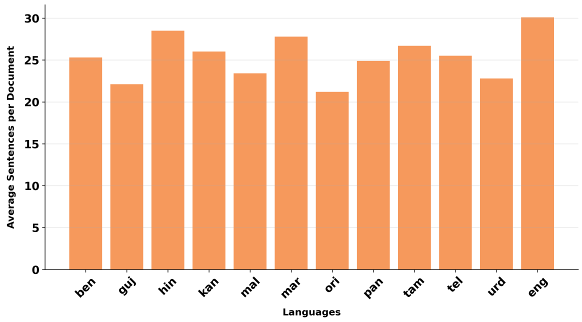

Figure 3: Average document length in PRALEKHA for each language, measured in terms of number of sentences per document.

Figure 3 presents statistics on the average number of sentences per document for each language in PRALEKHA. The average sentence count ranges from 20 to 30, indicating that the documents are relatively long and making the benchmark more challenging.

PRALEKHA covers two broad domains - News Bulletins and Podcast Scripts, therefore containing a mixture of written as well as spoken form of data. Following the methodology of Sangraha (Khan et al., 2024), we custom-scraped data from the Indian Press Information Bureau (PIB)[2] website, aligning documents by matching bulletin IDs interlinking bulletins across languages. These documents are manually written and hence are of high quality. For podcast scripts, we employed the approach used by Khan et al. (2024) and Siripragada et al. (2020) to collect transcripts from Mann Ki Baat,[3] a radio program hosted by the Indian Prime Minister. This program is typically spoken in Hindi, then manually transcribed and translated into various other Indian languages. Sometimes, the website's meta-data incorrectly marks the document's languages, which we fix with a simple combination of IndicLID (Madhani et al., 2023) and script-specific Unicode ranges.

[2] https://pib.gov.in
[3] https://www.pmindia.gov.in/en/mann-ki-baat

All the data is human-written/verified. *This dataset serves both as a benchmark for assessing document alignment techniques and as a domain-specific parallel corpus for training document-level MT models in Indic Languages.*

4 Cross-Lingual Document Alignment

Given two document collections, D_1 and D_2, where D_1 consists of documents written in `lang1` and D_2 consists of documents in `lang2`, the objective of Cross-Lingual Document Alignment (CLDA) is to determine a set of paired documents, $P = \{(d_1, d_2) \mid d_1 \in D_1, d_2 \in D_2\}$, such that each pair (d_1, d_2) exhibits a high degree of semantic similarity, with d_1 and d_2 being direct translations of each other.

We next describe the evaluation framework used to benchmark various CLDA techniques. As illustrated in Figure 1, the framework includes our proposed approach, which leverages the DOCUMENT ALIGNMENT COEFFICIENT (DAC), along with the pooling-based baseline methods. The following subsections discuss each component of the CLDA evaluation framework in detail.

4.1 Granularity

The granularity level G defines the unit of text at which alignment is performed and is a key parameter in our study. We evaluate document alignment at three levels of granularity: $G = 1$ (sentence-level, as in Schwenk et al. (2020)), $G = 2, 4, 8$ (chunk-level, where each chunk consists of 2, 4, or 8 sentences, respectively), and $G = |D|$ (document-level, as in El-Kishky et al. (2020); El-Kishky and Guzmán (2020)).

At one extreme, $G = 1$ produces too many embeddings to align, resulting in a combinatorially large search space that leads to high computational cost, while offering limited context for semantic comparison. At the other extreme, $G = |D|$ leads to overly coarse representations and information loss. It also creates challenges in computing a single document-level embedding because embedding models have a limited context window (Zhu et al., 2024). Both extremes have limitations; therefore, we consider intermediate values of G that strike a balance between efficiency and information retention. A larger granularity helps preserve local coherence and capture broader document-level information, addressing the shortcomings of traditional sentence-based approaches (Xie et al., 2024).

Pooling Strategies for Baselines. We benchmark several pooling strategies used in prior work (El-Kishky et al., 2020; El-Kishky and Guzmán, 2020) to construct document embeddings ($G = |D|$) by aggregating finer-grained unit embeddings.

1. MEAN POOLING (MP): Computes a single embedding by averaging all unit embeddings.
2. LENGTH POOLING (LP): Adjusts unit embeddings based on length, assigning greater importance to longer, more informative segments.
3. INVERSE DOCUMENT FREQUENCY (IDF): It weighs terms according to their rarity, prioritizing informative words while down-weighting common ones to improve alignment precision.
4. LENGTH-INVERSE DOCUMENT FREQUENCY (LIDF): Combines length-based weighting and IDF, emphasizing longer, content-rich segments while reducing the influence of repetitive or generic terms, providing a balanced approach to alignment.

4.2 Embedding Model

To compute multilingual embeddings we consider LABSE (Feng et al., 2022) and SONAR (Duquenne et al., 2023). LABSE is designed for cross-lingual sentence retrieval, leveraging both parallel and monolingual data to generate language-agnostic embeddings. SONAR extends this approach by training on a more diverse set of languages, leading to improved representation quality for both high-resource and low-resource languages.

Computing Embeddings. After segmenting each document into units based on the chosen granularity G, we obtain representations for each segment by encoding the entire unit with the embedding model. Because the values of G considered in our study are relatively small, each segment fits within the context window of the embedding models used.

4.3 Alignment Algorithm

After computing unit embeddings at various text granularities from the monolingual corpora in the source and target languages, we apply the Alignment Algorithm 1 to align these embeddings. The algorithm performs margin-based bitext mining in a shared multilingual space to extract parallel units (Schwenk et al., 2020). It adopts the max-margin criterion, which has been shown to improve alignment quality compared to an absolute similarity threshold (Artetxe and Schwenk, 2019). To build efficient similarity indices, we use the FAISS library[4]. For each unit embedding, we retrieve the top-k nearest neighbors and compute margin scores $M(x, y)$ to capture similarity in a relative context. Candidate pairs are then ranked by their margin scores, and a greedy bipartite matching strategy is applied to select the highest-scoring pairs, ensuring that each embedding is aligned only once.

Algorithm 1 Alignment Algorithm

Input: Embeddings $\mathbf{X} = \{x_i\}$ and $\mathbf{Y} = \{y_j\}$, k nearest neighbors

Output: $\mathcal{U} = \arg\max_{\mathcal{U}'} \sum_{(x,y)\in\mathcal{U}'} M(x, y)$

FAISS Indexing: Construct FAISS index for efficient similarity search

$$\texttt{index\_X} \leftarrow \texttt{IndexFlatIP}(\mathbf{X})$$
$$\texttt{index\_Y} \leftarrow \texttt{IndexFlatIP}(\mathbf{Y})$$

Query top-k nearest neighbors:

$$\mathsf{NN}_k(x) = \{y_1, y_2, \ldots, y_k\} \subset \mathbf{Y} \text{ for each } x \in \mathbf{X}$$
$$\mathsf{NN}_k(y) = \{x_1, x_2, \ldots, x_k\} \subset \mathbf{X} \text{ for each } y \in \mathbf{Y}$$

Compute Margin Score:

for each (x, y) in $\mathbf{X} \times \mathbf{Y}$ where $y \in \mathsf{NN}_k(x)$ or $x \in \mathsf{NN}_k(y)$ **do**

$$A_x = \frac{1}{k} \sum_{z\in\mathsf{NN}_k(x)} \cos(x, z)$$

$$A_y = \frac{1}{k} \sum_{z\in\mathsf{NN}_k(y)} \cos(y, z)$$

$$M(x, y) = \frac{\cos(x, y)}{0.5 \times (A_x + A_y)}$$

end for

Max Strategy: Select pairs to maximize overall margin score

$$\mathcal{U} = \{(x, y) \mid y \in \mathsf{NN}_k(x)\} \cup \{(x, y) \mid x \in \mathsf{NN}_k(y)\}$$

$$\mathcal{U}_{\text{sorted}} = \text{Sort}(\mathcal{U}) \text{ by } M(x, y) \text{ in descending order}$$

for each (x, y) in $\mathcal{U}_{\text{sorted}}$ **do**
  if both x and y are not in aligned pairs **then**
    Add (x, y) to aligned pairs
  end if
end for

Return: $\mathcal{U}$, aligned pairs that maximize the overall margin score

For pooled document embeddings at $G = |D|$, we directly apply the alignment algorithm on these embeddings to obtain aligned document pairs. We extend this idea by introducing a novel DOCUMENT ALIGNMENT COEFFICIENT (DAC) metric for mining parallel documents, which leverages finer-grained units for document alignment. The motivation behind DAC is to avoid relying solely on document-level embeddings, which are typically formed by pooling unit embeddings and may fail to capture detailed semantic information. Instead, we treat each document as a sequence of chunks, where each chunk is a contiguous set of sentences and serves as the basic unit of alignment. Our hypothesis is that embeddings at finer granularity capture semantic nuances more effectively. After performing alignment at the chunk level, the resulting information is aggregated to identify aligned document pairs. The following subsection provides a detailed description of the DAC approach.

[4]https://github.com/facebookresearch/faiss

4.4 DOCUMENT ALIGNMENT COEFFICIENT

Unlike baseline approaches that operate directly on pooled document embeddings, DAC performs chunk-level alignment first and then derives document pairs using a computed score based on these chunk alignments, thereby providing finer-grained semantic signals that improve alignment precision.

Computing DAC for a Document Pair. For a given source–target document pair, the DAC is computed as follows:

$$\text{DAC} = \frac{2 \times N_{\text{aligned}}}{N_{\text{src}} + N_{\text{tgt}}} \tag{1}$$

Here, N_{src} and N_{tgt} represent the total number of chunks in the source and target documents, respectively, and N_{aligned} denotes the number of aligned chunks between the two documents.

The algorithm first aligns the constituent chunks within each document pair and then uses these fine-grained alignments to compute the overall degree of alignment at the document level.

DAC Threshold. The DAC produces a normalized alignment score between 0 and 1, where higher values indicate stronger alignment, and a threshold on this score can be set to control the final alignment quality. A lower DAC threshold allows a balance between quality and quantity, accepting some misalignments while increasing coverage. On the other hand, a higher threshold prioritizes precision, ensuring fewer incorrect alignments at the cost of reduced yield. The threshold can be adjusted based on the specific requirements of the aligned dataset and its intended downstream task. A detailed analysis of various DAC Thresholds is provided in Figure 7 and Table 4 in Appendix A.

Language	ben	guj	hin	kan	mal	mar	ori	pan	tam	tel	urd	eng
Alignable	95K	67K	204K	61K	67K	134K	46K	108K	149K	109K	220K	298K
Unalignable	47K	34K	102K	31K	34K	67K	23K	54K	75K	55K	110K	149K
Total	142K	101K	306K	92K	101K	201K	69K	162K	224K	164K	330K	447K

Table 1: Per language statistics of unique documents divided into two groups: *alignable* and *unalignable*. *Alignable* documents come from the English-Indic part of PRALEKHA and the *Unalignable* documents come from SANGRAHA UNVERIFIED, representing noise for a realistic evaluation setting.

5 Experiments

In this section, we present the experimental setup and describe the two types of evaluations conducted on PRALEKHA: intrinsic evaluation, which measures the effectiveness of document alignment techniques, and extrinsic evaluation, which assesses their impact on downstream tasks. *Unless stated otherwise, all references to* PRALEKHA *hereafter refer to its English–Indic subset.*

5.1 Intrinsic Evaluation

For intrinsic evaluation, we assess how well DAC performs compared to existing pooling-based baselines for document alignment.

Data Setting : PRALEKHA contains only parallel (*alignable*) documents, so aligning only these would not reflect a realistic and challenging scenario. To simulate real-world multilingual corpora setting, we sample random (*unalignable*) documents from SANGRAHA UNVERIFIED (Khan et al., 2024) that cannot be aligned with those in PRALEKHA and inject them as noise into the dataset. We select a number of unalignable documents equal to 50% of the documents in PRALEKHA, resulting in a 1:2 ratio of *unalignable* to *alignable* documents during alignment. Table 1 reports the counts of unique *alignable* and *unalignable* documents for each language used in our experiments. Additionally, we compare the best-performing baseline with our proposed DAC approach on the CCAligned dataset (El-Kishky et al., 2020), using a similar data setting.

Alignment Settings : We evaluate alignment at four granularities: sentence-level ($G = 1$), chunk-level ($G = 2, 4, 8$), and document-level ($G = |D|$). For $G = |D|$, we benchmark all pooling-based baselines described in Section 4.1. Unit embeddings are obtained using LABSE (Feng et al., 2022) and SONAR (Duquenne et al., 2023). We build FAISS `IndexFlatIP`[5] indices and retrieve the top-k nearest neighbors for each embedding with $k = 16$. A DAC threshold of 0.1 is used to balance precision and recall, and a detailed threshold analysis is provided in Figure 7 and Table 4 in the Appendix.

Evaluation Metrics : To assess the intrinsic performance of various document alignment methods, we evaluate three key metrics: precision, recall, and F1 score. While a high F1 score is generally desirable as it balances precision and recall, the ideal metric depends on the specific downstream task. We further measure computational time to analyze efficiency trade-offs across different granularities.

5.2 Extrinsic Evaluation

To assess the extrinsic performance of document alignment techniques, we evaluate their effectiveness in document-level MT tasks.

Data Setting : We focus on translation between English and eight Indic languages: Bengali, Gujarati, Hindi, Kannada, Malayalam, Marathi, Odia, and Tamil. From PRALEKHA, we randomly sample 1,000 document pairs each for the development and test sets. After removing these pairs, we perform document alignment on the remaining data as described in Section 5.1. We obtain two sets of aligned documents: one using the best-performing baseline and the other using our proposed DAC approach. From each set, we sample 10,000 aligned pairs to train two separate document-level MT models using open-source LLMs. Extrinsic evaluation is then conducted on the sampled test set.

Implementation and Training : We fine-tune two open-source LLMs: LLAMA-3.2-1B[6] and SARVAM-1[7]. LLAMA-3.2-1B is a 1 billion parameter multilingual model released by Meta AI,

[5]https://github.com/facebookresearch/faiss/wiki/The-index-factory
[6]huggingface.co/meta-llama/Llama-3.2-1B
[7]huggingface.co/sarvamai/sarvam-1

Metrics	Methods	LaBSE				SONAR			
		G = 1	G = 2	G = 4	G = 8	G = 1	G = 2	G = 4	G = 8
Precision	MP	0.7585	0.7432	0.7232	0.7018	0.7803	0.7643	0.7541	0.7632
	LP	0.7820	0.7709	0.7614	0.7464	0.7943	0.7843	0.7891	0.8111
	IDF	0.7610	0.7463	0.7290	0.7052	0.7821	0.7669	0.7596	0.7661
	LIDF	0.7831	0.7716	0.7619	0.7466	0.7950	0.7849	0.7894	0.8112
	DAC	**0.9152**	**0.9007**	**0.8936**	**0.8912**	**0.9106**	**0.8870**	**0.8742**	**0.8732**
Recall	MP	0.6310	0.6057	0.5653	0.5599	0.6613	0.6349	0.5696	0.5414
	LP	0.6824	0.6807	0.7019	0.6959	0.6924	0.6834	0.6688	0.6630
	IDF	0.6437	0.6192	0.5962	0.5738	0.6702	0.6471	0.5991	0.5544
	LIDF	**0.6908**	**0.6844**	**0.7030**	**0.6971**	**0.6953**	**0.6864**	**0.6700**	**0.6632**
	DAC	0.6588	0.6411	0.6471	0.6407	0.6484	0.6108	0.6005	0.6020
F1 Score	MP	0.6856	0.6639	0.6291	0.6157	0.7133	0.6904	0.6410	0.6237
	LP	0.7252	0.7199	0.7273	0.7170	0.7372	0.7276	0.7213	0.7266
	IDF	0.6940	0.6733	0.6519	0.6266	0.7193	0.6989	0.6647	0.6353
	LIDF	0.7303	0.7223	0.7281	0.7178	0.7392	**0.7297**	**0.7221**	**0.7268**
	DAC	**0.7635**	**0.7463**	**0.7480**	**0.7428**	**0.7550**	0.7210	0.7098	0.7109

Table 2: Average Precision, Recall, and F1 scores from the Intrinsic Evaluation of CLDA techniques on PRALEKHA, computed over 11 Indic languages. Results are shown for LABSE and SONAR embeddings at granularities $G = 1, 2, 4, 8$. We compare pooling-based baselines (MP, LP, IDF, LIDF) with our proposed DAC approach. **Bold** values indicate the best score for each granularity. Appendix A provides further analysis on the intrinsic trends.

while SARVAM-1 is a 2 billion parameter model developed by Sarvam AI, optimized for Indic languages. We used *open-instruct*[8] for *full fine-tuning* using the prompt-completion template. We used default hyperparameters recommended in *open-instruct*, most notably: learning rates of 5×10^{-5}, maximum context (prompt+completion) lengths of 4096, Adam optimizer, and no weight decay. Training was performed on NVIDIA H100 GPUs with mixed-precision (bfloat16) support. We trained only bilingual models, a single model for English $\leftrightarrow$ X translation, for a total of 16 models (8 languages, 2 alignment approaches). Each model was trained for 1 epoch. We performed greedy decoding with a maximum of 4,096 generated tokens to maintain consistency across evaluations.

Evaluation Metrics : To evaluate document-level MT performance, we use DOCCOMET[9], an extension of COMET that incorporates document-level context for improved quality assessment. Specifically, we employ the `Unbabel/wmt22-comet-da` model from the WMT 2022 Metrics Shared Task, a reference-based COMET model designed for document-level MT evaluation. Alongside COMET, we also report ChrF[10], a character-level F-score metric based on n-gram overlaps.

[8] `https://github.com/allenai/open-instruct`

[9] `https://github.com/Unbabel/COMET`

[10] `https://github.com/mjpost/sacreBLEU#chrf--chrf`

6 Results and Discussion

We present results demonstrating the effectiveness of DAC through intrinsic and extrinsic evaluations.

6.1 Intrinsic Performance of DAC

Table 2 reports average precision, recall, and F1 scores across granularities and embedding models.

DAC vs. Baselines: DAC consistently achieves the highest precision across all settings, with gains of 15–20% over pooling-based baselines. While its recall is slightly lower, DAC attains competitive or superior F1 scores, indicating that it produces more precise alignments, a desirable property for high-quality MT training data, where quality is more important than quantity (Ranathunga et al., 2024; Zhou et al., 2023). Among the baselines, LIDF is the strongest competitor, performing closest to DAC due to its effective combination of length normalization and term weighting. LP performs slightly worse, whereas MP and IDF-based pooling lag considerably behind. *Additionally, in experiments conducted on the CCAligned dataset (El-Kishky et al., 2020), DAC outperforms LIDF across all metrics; see Appendix B for more details.*

Effect of Granularity. As shown in Table 3, alignment performance is highest at the finest granularity ($G = 1$) for both LaBSE and SONAR, with only a modest decline as chunk size increases. Be-

Granularity	LaBSE				SONAR			
	Time (min)	Precision	Recall	F1	Time (min)	Precision	Recall	F1
G = 1	47	0.9152	0.6588	0.7635	115	0.9106	0.6484	0.7550
G = 2	36	0.9007	0.6411	0.7463	96	0.8870	0.6108	0.7210
G = 4	25	0.8936	0.6471	0.7480	59	0.8742	0.6005	0.7098
G = 8	18	0.8912	0.6407	0.7428	35	0.8732	0.6020	0.7109

Table 3: Computational time and alignment performance across document granularities (G) for 150k document pairs on an NVIDIA H100 node. Results for LaBSE and SONAR are reported in wall-clock time (minutes), precision, recall, and F1 score. Color gradients illustrate relative trends: **red**→**green** indicates slower→faster configurations for time, while performance metrics follow a unified **green** scale. The drop in alignment quality is far smaller than the reduction in computation time, highlighting a strong computational advantage at higher granularities.

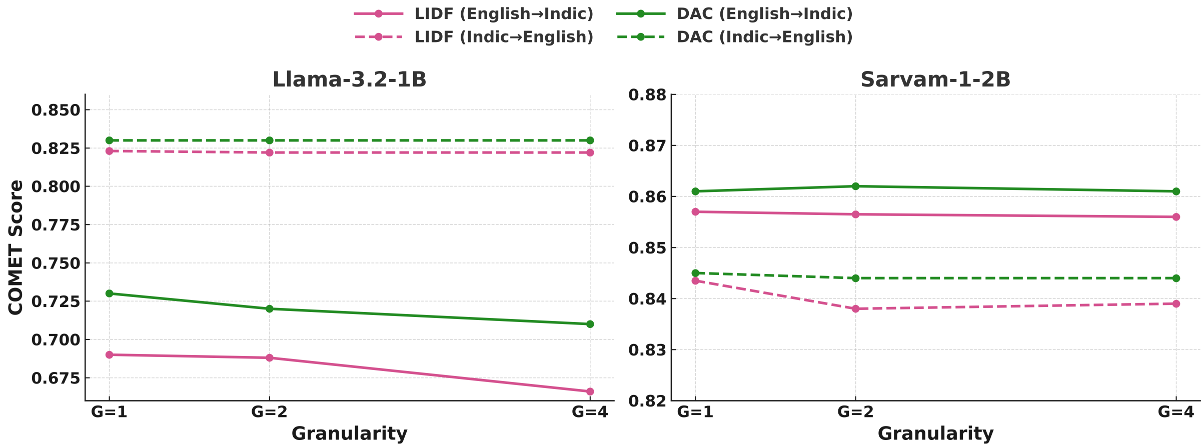

Figure 4: Extrinsic evaluation of DAC with LaBSE embeddings (**green**) and LIDF with SONAR embeddings (**pink**) on PRALEKHA. COMET scores averaged across 8 Indic Languages are reported for granularities $G = 1, 2, 4$ on English→Indic (solid lines) and Indic→English (dashed lines) translation tasks using the LLAMA-3.2-1B (left) and SARVAM-1-2B (right) models. Appendix C provides further analysis of extrinsic trends, and detailed per-language COMET scores are presented in Table 6.

yond $G = 2$, variations across performance metrics remain minimal, indicating that coarser segmentations preserve much of the alignment quality. In contrast, computational time decreases sharply from 47 to 18 minutes for LaBSE and from 115 to 35 minutes for SONAR as G increases from 1 to 8, resulting in roughly a 2–3× reduction in runtime. This efficiency gain stems from a combinatorial reduction in the candidate search space: at higher granularities, each document is represented by fewer, longer chunks, leading to substantially fewer embedding computations and pairwise similarity operations during retrieval.

This trade-off underscores an important consideration for large-scale document alignment. While fine-grained ($G = 1$) configurations yield slightly better alignment quality, their computational cost becomes impractical at scale. Coarser granularities thus offer a balanced solution, maintaining competitive alignment performance while enabling significantly faster processing, offering a favorable efficiency–performance trade-off .

Effect of Embedding Model. Across embedding models, SONAR performs better with pooling-based methods, whereas LABSE achieves higher performance with DAC. This contrast reflects their underlying training objectives: SONAR is trained with mean-pooled sentence representations, making it naturally compatible with aggregation-based alignment, while LABSE relies on the [CLS]-token embedding, which encodes context from the entire sequence while remaining sensitive to token-level structure. Consequently, LABSE benefits from methods that preserve token-level information, whereas SONAR aligns more effectively with pooling strategies that emphasize global semantics. See Appendix A for a detailed analysis of intrinsic performance trends.

6.2 Extrinsic Performance of DAC

Based on the intrinsic evaluations, the optimal configurations for document alignment on our PRALEKHA dataset are LABSE with DAC at granularities $G = 1$, $G = 2$, and $G = 4$, followed closely by SONAR with LIDF at the same granularities. These configurations were chosen solely on the basis of alignment performance metrics, without accounting for computational cost. In this section, we evaluate their extrinsic performance using the experimental setup described in Section 5.2, to examine whether the intrinsic alignment trends translate into measurable downstream gains.

Table 6 reports COMET scores for extrinsic evaluation, comparing DAC with LIDF (the strongest baseline). *DAC consistently outperforms LIDF across all granularities, translation directions, and fine-tuned models*, demonstrating that higher-precision alignments yield cleaner parallel data and result in measurable downstream improvements in translation quality. Detailed per-language COMET scores are presented in Appendix C.

Differences in MT performance across granularities are relatively small and align closely with intrinsic performance trends. The strong results for $G = 2$ and $G = 4$ suggest that moderate chunking can substantially reduce computational cost while maintaining translation quality. This suggests that coarse-grained segmentation balances alignment efficiency and performance in large-scale settings.

Across models, Sarvam-based systems outperform their LLaMA-based counterparts, particularly for English→Indic translation. This is likely due to Sarvam's tokenizer being optimized for Indic languages, reducing token fertility and generation burden. Conversely, Indic→English translations achieve consistently higher scores, reflecting the greater morphological and syntactic complexity of Indic languages, which makes it more challenging.

Overall, performance differences remain modest but consistent across settings. Extrinsic results confirm that DAC-aligned corpora at $G = 2$ & 4 achieve strong translation quality. However, it is worth noting that intrinsic performance does not always translate to downstream performance. Appendix C provides further discussion of these trends, and Table 7 reports corresponding ChrF scores, which follow patterns similar to the COMET results.

7 Conclusion

Mining parallel document pairs for document-level MT is challenging because existing CLDA methods often rely on metadata or pooled sentence embeddings due to the limited context windows of embedding models. Moreover, sentence-level alignment introduces a combinatorially large search space, leading to high computational costs that become impractical at scale. To address these challenges for Indic languages, we introduced PRALEKHA, a large-scale benchmark comprising over 3 million aligned document pairs across 11 Indic languages and English. Furthermore, we proposed the Document Alignment Coefficient (DAC), a novel metric for fine-grained document alignment. Unlike pooling-based methods, DAC aligns documents by matching smaller chunks and computes similarity as the ratio of aligned chunks to the average number of chunks in a pair. Intrinsic evaluation shows that our chunk-based approach is 2–3× faster while maintaining competitive performance, and that DAC achieves substantial gains over pooling-based baselines. Extrinsic evaluation further demonstrates that document-level MT models trained on DAC-aligned pairs consistently outperform those trained using baseline alignment methods. These results highlight DAC's effectiveness for parallel document mining, balancing performance with computational efficiency. By releasing the PRALEKHA dataset and CLDA evaluation framework, we aim to facilitate future research on scalable and reliable document-level alignment and to advance the development of high-quality document-level MT systems for Indic languages.

Limitations

While PRALEKHA and the proposed chunk-based alignment approach advance research in cross-lingual document alignment, some limitations remain. Our dataset covers 11 Indic languages and English, so the findings may not generalize to other language families with different linguistic properties. The documents in PRALEKHA are primarily drawn from structured, high-quality sources, making the benchmark less representative of noisy web data that includes informal language, OCR errors, or code-switching. In addition, the benchmark mostly contains documents of moderate length, leaving performance on very short or long documents, common in other domains, largely untested. Because the dataset is domain-specific, MT models trained solely on it may overfit and exhibit reduced generalization, or even hallucinate when evaluated on out-of-domain data. DAC may also be less robust when translations are semantically equivalent but highly paraphrased. Finally, our extrinsic evaluation does not fully establish how improvements in alignment quality affect document-level MT performance, as current MT metrics capture discourse-level phenomena only to a limited extent. Future work should expand CLDA evaluation to more diverse language families, noisier domains, and a wider range of document lengths. It should also explore alignment strategies that are more robust to paraphrasing and develop document-level MT metrics that better correlate with alignment quality.

Ethics

All datasets used in this paper are available under permissible licenses, and we adhere strictly to their intended usage, maintaining full compliance with licensing requirements. Additionally, the code used for our evaluation framework will be made publicly available under the MIT License.[11] Generative AI systems were used solely to assist with the language of this paper, specifically for paraphrasing, spell-checking, polishing the authors' original text, and generating boilerplate code.

Acknowledgements

We would like to thank EkStep Foundation and Nilekani Philanthropies for their generous grant towards building datasets, models, tools, and other resources for Indian languages. This work was partly supported by JSPS KAKENHI Grant-in-Aid for Early-Career Scientists 25K21290.

References

Sadaf Abdul-Rauf and Holger Schwenk. 2009. On the Use of Comparable Corpora to Improve SMT Performance. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 16–23. Association for Computational Linguistics.

Mikel Artetxe and Holger Schwenk. 2019. Margin-based Parallel Corpus Mining with Multilingual Sentence Embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3197–3203, Florence, Italy. Association for Computational Linguistics.

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

Thi-Ngoc-Diep Do, Viet-Bac Le, Brigitte Bigi, Laurent Besacier, and Eric Castelli. 2009. Mining a Comparable Text Corpus for a Vietnamese-French Statistical Machine Translation System. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 165–172. Association for Computational Linguistics.

Paul-Ambroise Duquenne, Holger Schwenk, and Benoît Sagot. 2023. SONAR: Sentence-Level Multimodal and Language-Agnostic Representations. *Preprint*, arXiv:2308.11466.

Ahmed El-Kishky, Vishrav Chaudhary, Francisco Guzmán, and Philipp Koehn. 2020. CCAligned: A Massive Collection of Cross-Lingual Web-Document Pairs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 5960–5969. Association for Computational Linguistics.

Ahmed El-Kishky and Francisco Guzmán. 2020. Massively Multilingual Document Alignment with Cross-lingual Sentence-Mover's Distance. *arXiv preprint arXiv:2002.00761*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-Agnostic BERT Sentence Embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 878–894. Association for Computational Linguistics.

Jay Gala, Pranjal A. Chitale, Raghavan AK, Varun Gumma, Sumanth Doddapaneni, 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. *Preprint*, arXiv:2305.16307.

Mandy Guo, Yinfei Yang, Keith Stevens, Daniel Cer, Heming Ge, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2019. Hierarchical Document Encoder for Parallel Corpus Mining. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 64–72. Association for Computational Linguistics.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian

[11] https://opensource.org/licenses/MIT

Languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961. Association for Computational Linguistics.

Mohammed Safi Ur Rahman Khan, Priyam Mehta, Ananth Sankar, Umashankar Kumaravelan, Sumanth Doddapaneni, G Suriyaprasaad, Varun G Balan, Sparsh Jain, Anoop Kunchukuttan, Pratyush Kumar, Raj Dabre, and Mitesh M. Khapra. 2024. IndicLLMSuite: A Blueprint for Creating Pre-training and Fine-Tuning Datasets for Indian Languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15831–15879. Association for Computational Linguistics.

Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.

Yash Madhani, Mitesh M. Khapra, and Anoop Kunchukuttan. 2023. Bhasa-Abhijnaanam: Native-script and romanized Language Identification for 22 Indic languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 816–826, Toronto, Canada. Association for Computational Linguistics.

Proyag Pal, Alexandra Birch, and Kenneth Heafield. 2024. Document-Level Machine Translation with Large-Scale Public Parallel Corpora. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13185–13197, Bangkok, Thailand. 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.

Surangika Ranathunga, Nisansa de Silva, Menan Velayuthan, Aloka Fernando, and Charitha Rathnayake. 2024. Quality Does Matter: A Detailed Look at the Quality and Utility of Web-Mined Parallel Corpora. *Preprint*, arXiv:2402.07446.

Philip Resnik and Noah A. Smith. 2003. The Web as a Parallel Corpus. *American Journal of Computational Linguistics*, 29(3):349–380.

Sonal Sannigrahi, Josef van Genabith, and Cristina España-Bonet. 2023. Are the best multilingual document embeddings simply based on sentence embeddings? In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2306–2316, Dubrovnik, Croatia. Association for Computational Linguistics.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021. WikiMatrix: Mining 135 Million Parallel Sentences from Wikipedia. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1625–1639. Association for Computational Linguistics.

Holger Schwenk and Matthijs Douze. 2017. Learning Joint Multilingual Sentence Representations with Neural Machine Translation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 157–167, Vancouver, Canada. Association for Computational Linguistics.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, and Armand Joulin. 2020. CCMatrix: Mining Billions of High-Quality Parallel Sentences on the WEB. *Preprint*, arXiv:1911.04944.

Shashank Siripragada, Jerin Philip, Vinay P. Namboodiri, and C V Jawahar. 2020. A Multilingual Parallel Corpora Collection Effort for Indian Languages. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3743–3751. European Language Resources Association.

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. *Preprint*, arXiv:2207.04672.

Jiawen Xie, Pengyu Cheng, Xiao Liang, Yong Dai, and Nan Du. 2024. Chunk, Align, Select: A Simple Long-sequence Processing Method for Transformers. *Preprint*, arXiv:2308.13191.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. LIMA: Less Is More for Alignment. *Preprint*, arXiv:2305.11206.

Dawei Zhu, Liang Wang, Nan Yang, Yifan Song, Wenhao Wu, Furu Wei, and Sujian Li. 2024. LongEmbed: Extending Embedding Models for Long Context Retrieval. *Preprint*, arXiv:2404.12096.

Appendix

A Intrinsic Performance Trends in CLDA

This section analyzes the Precision, Recall, and F1 score trends of various Cross-Lingual Document Alignment (CLDA) methods based on their intrinsic performance on PRALEKHA, as shown in Table 2.

A.1 Impact of Granularities

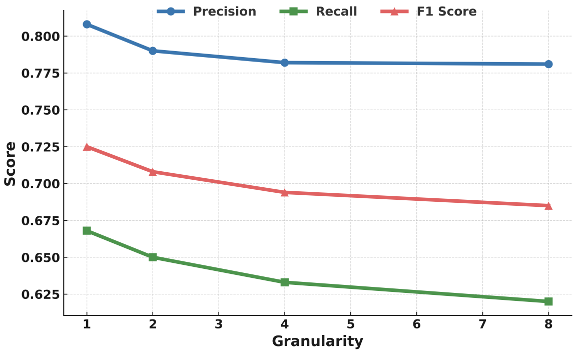

Figure 5: Impact of Granularity on the Intrinsic Performance of CLDA techniques on PRALEKHA.

As shown in Figure 5, increasing granularity is associated with a gradual reduction in precision, recall, and F1 score. The decline is most pronounced for recall, suggesting that fewer relevant matches are retrieved as segmentation becomes finer. Precision, however, remains relatively stable across granularities. These trends indicate that while finer segmentation can provide more detailed units for alignment, it may also reduce the amount of contextual information available to embedding models, which are typically optimized for shorter text spans.

A.2 Impact of Embedding Models

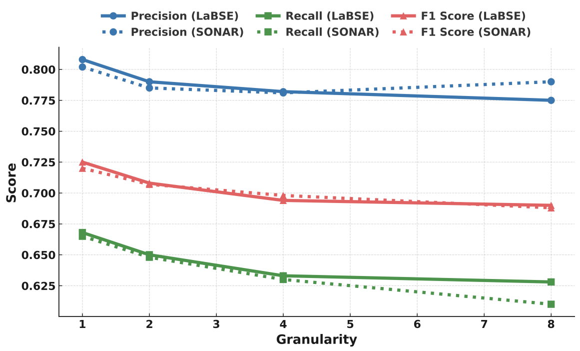

Figure 6: Impact of Embedding Models on the Intrinsic Performance of CLDA techniques on PRALEKHA.

Figure 6 shows how increasing granularity affects the performance of both LaBSE and SONAR across precision, recall, and F1 score. SONAR tends to maintain slightly higher precision, indicating more accurate alignments, whereas LaBSE achieves higher recall, retrieving a larger number of matches. At finer granularities, both models show a gradual decrease in overall performance, with SONAR being relatively more stable in precision. LaBSE achieves a higher F1 score overall, reflecting a more balanced trade-off between precision and recall. These results suggest that SONAR may be better suited for scenarios where precision is prioritized, while LaBSE can be advantageous for broader retrieval needs.

A.3 Impact of DAC Threshold

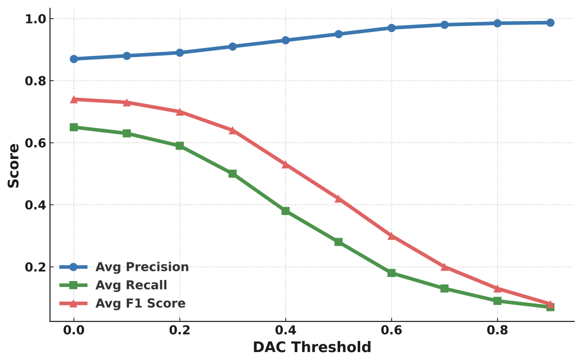

Figure 7: Impact of DAC Threshold on the Intrinsic Performance of CLDA techniques on PRALEKHA.

Figure 7 illustrates the trends of precision, recall, and F1 score across different DAC thresholds, highlighting the trade-offs between precision and recall. Increasing the DAC threshold leads to higher precision but lower recall and F1 scores in all settings. We adopt a threshold of 0.1 for both intrinsic and extrinsic evaluations, as it provides a better balance between quality and yield. Table 4 compares the intrinsic performance of DAC at thresholds 0.1 and 0.5.

A.4 DAC vs. Baseline Methods

We compare our DAC approach with the two strongest pooling-based baselines, Length Pooling (LP) and Length-IDF (LIDF), as shown in Table 2. Figure 8 highlights that DAC consistently delivers the highest precision across all granularities for both LaBSE and SONAR. Although LP and LIDF yield slightly higher recall, DAC achieves a better balance between precision and recall, reflected in its F1 scores. Moreover, DAC shows minimal

Metrics	Methods	LaBSE				SONAR			
		G = 1	G = 2	G = 4	G = 8	G = 1	G = 2	G = 4	G = 8
Precision	DAC (0.1)	0.9152	0.9007	0.8936	0.8912	0.9106	0.8870	0.8742	0.8732
	DAC (0.5)	**0.9555**	**0.9617**	**0.9570**	**0.9404**	**0.9546**	**0.9600**	**0.9480**	**0.9307**
Recall	DAC (0.1)	**0.6588**	**0.6411**	**0.6471**	**0.6407**	**0.6484**	**0.6108**	**0.6005**	**0.6020**
	DAC (0.5)	0.2930	0.2568	0.3104	0.3960	0.2530	0.2005	0.2542	0.3297
F1 Score	DAC (0.1)	**0.7635**	**0.7463**	**0.7480**	**0.7428**	**0.7550**	**0.7210**	**0.7098**	**0.7109**
	DAC (0.5)	0.4368	0.3915	0.4564	0.5486	0.3896	0.3229	0.3931	0.4814

Table 4: Comparison of DAC threshold performance (0.1 and 0.5) in the intrinsic evaluation on PRALEKHA. **Bold** values indicate the best score for each granularity.

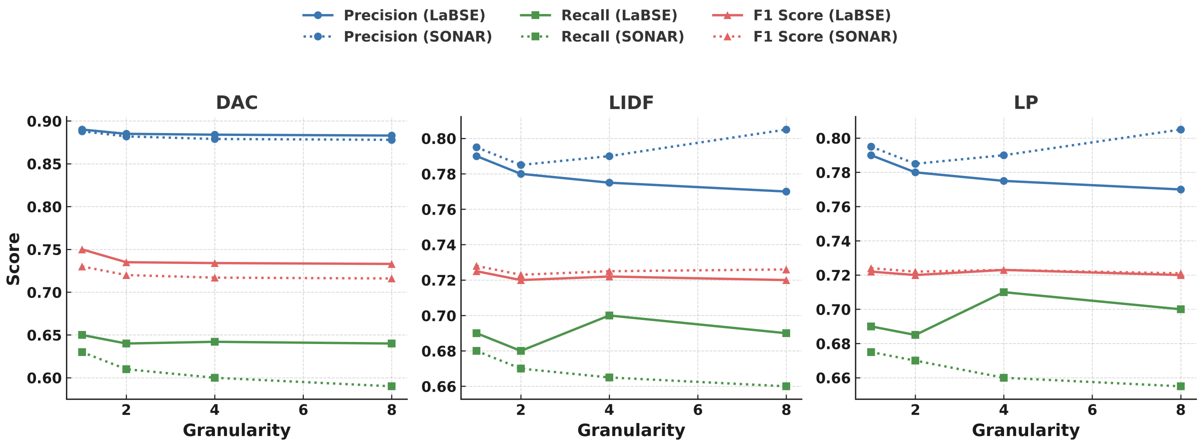

Figure 8: Intrinsic evaluation performance of DAC, LP, and LIDF on PRALEKHA.

degradation in precision as granularity increases, demonstrating its robustness compared to the baselines. These trends confirm that DAC more effectively leverages contextual information, making it the strongest overall alignment method.

B Intrinsic Evaluation on CCALIGNED

While the primary intrinsic evaluation was conducted on the PRALEKHA dataset, we also perform an additional benchmark on CCALIGNED (El-Kishky et al., 2020), a large-scale web-mined parallel corpus. This experiment evaluates whether the performance gains of DAC generalize beyond the structured and high-quality domains of PRALEKHA. We follow the same data settings described in Section 5.1. For comparison, we use the strongest baseline from the main experiments: the LIDF pooling method with SONAR embeddings, evaluated at granularities $G = 1, 2, 4$ and the DAC approach with LaBSE embeddings at the same granularities.

Table 5 presents the results, showing that *DAC consistently outperforms LIDF across all metrics and settings. Its effectiveness is therefore not limited to* PRALEKHA *but also extends to large, noisy web-mined corpora, confirming DAC as a robust alignment method across diverse domains.*

C Extrinsic Performance Trends in CLDA

Alongside COMET, we report ChrF scores. COMET uses neural models to capture contextual information, whereas ChrF measures surface-level lexical similarity, providing a complementary perspective on translation quality.

Figure 9 shows COMET and ChrF scores across granularities ($G = 1, 2, 4$) for the LLAMA 3.2-1B and SARVAM-1 models. SARVAM-1 consistently outperforms LLAMA 3.2-1B across all granularities, and ChrF follows trends similar to COMET as discussed in Section 6.2. *DAC-aligned documents at $G = 2$ and $G = 4$ slightly outperform LIDF in both metrics, confirming DAC's effectiveness for document alignment.* These results also highlight that strong intrinsic performance does not always translate to gains in downstream tasks.

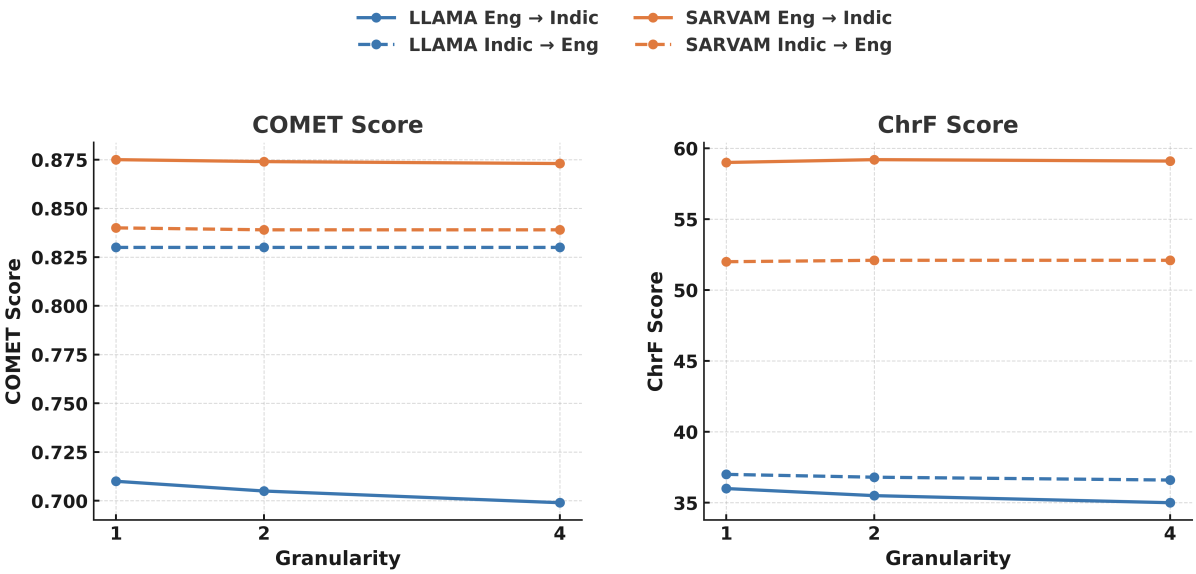

Figure 9: Extrinsic Evaluation of LLAMA-3.2-1B (**blue**) and SARVAM-1-2B (**orange**) on PRALEKHA, reporting COMET (left) and ChrF (right) scores averaged across 8 Indic Languages on PRALEKHA across granularities $G = 1, 2, 4$ on English→Indic (solid lines) and Indic→English (dashed lines) translation tasks.

Language	**Method**	**Precision**			**Recall**			**F1**		
		G = 1	**G = 2**	**G = 4**	**G = 1**	**G = 2**	**G = 4**	**G = 1**	**G = 2**	**G = 4**
Assamese	LIDF	0.4596	0.4842	0.4743	0.3511	**0.3862**	**0.3781**	0.3981	0.4297	0.4208
	DAC	**0.5978**	**0.5792**	**0.5635**	**0.3966**	0.3840	0.3497	**0.4768**	**0.4618**	**0.4316**
Bengali	LIDF	0.7569	0.7870	0.7785	0.6791	**0.7274**	**0.7153**	0.7159	0.7560	0.7456
	DAC	**0.8330**	**0.8317**	**0.8358**	**0.6848**	0.7125	0.6961	**0.7517**	**0.7675**	**0.7596**
Gujarati	LIDF	0.8206	0.8248	0.8138	0.7660	**0.7768**	**0.7646**	0.7924	0.8001	0.7884
	DAC	**0.8967**	**0.8768**	**0.8559**	**0.7833**	0.7671	0.7375	**0.8390**	**0.8183**	**0.7923**
Kannada	LIDF	0.7042	0.7079	0.7001	0.6399	**0.6470**	**0.6415**	0.6705	0.6761	0.6695
	DAC	**0.8240**	**0.8043**	**0.7825**	**0.6625**	0.6410	0.6069	**0.7345**	**0.7134**	**0.6836**
Malayalam	LIDF	0.7196	0.7247	0.7090	0.6350	0.6308	0.6113	0.6747	0.6745	0.6565
	DAC	**0.8417**	**0.8116**	**0.7816**	**0.7032**	**0.6717**	**0.6172**	**0.7662**	**0.7350**	**0.6898**
Marathi	LIDF	0.7741	0.7787	0.7738	0.7121	**0.7194**	**0.7110**	0.7418	0.7478	**0.7411**
	DAC	**0.8591**	**0.8368**	**0.8122**	**0.7359**	0.7109	0.6748	**0.7928**	**0.7687**	0.7372
Odia	LIDF	0.5793	0.5739	0.5566	0.4014	0.4036	0.3886	0.4742	0.4739	0.4576
	DAC	**0.7322**	**0.7326**	**0.7057**	**0.4262**	**0.4270**	**0.3991**	**0.5388**	**0.5395**	**0.5099**
Tamil	LIDF	0.7115	0.7079	0.6976	0.6493	**0.6413**	**0.6332**	0.6789	0.6729	0.6638
	DAC	**0.8265**	**0.8034**	**0.7775**	**0.6628**	0.6404	0.6006	**0.7356**	**0.7127**	**0.6777**
Telugu	LIDF	0.7416	0.7519	0.7517	0.6758	**0.6905**	**0.6877**	0.7072	0.7199	**0.7183**
	DAC	**0.8359**	**0.8145**	**0.7962**	**0.7081**	0.6852	0.6515	**0.7667**	**0.7443**	0.7166

Table 5: Precision, Recall, and F1 scores from Intrinsic Evaluation on CCALIGNED at varying granularities (G). We compare the best-performing pooling-based baseline from Table 2 (LIDF) with our proposed approach (DAC). **Bold** values indicate the best score for each granularity.

Language	Method	English → Indic						Indic → English					
		LLAMA-3.2-1B			SARVAM-1-2B			LLAMA-3.2-1B			SARVAM-1-2B		
		G = 1	G = 2	G = 4	G = 1	G = 2	G = 4	G = 1	G = 2	G = 4	G = 1	G = 2	G = 4
Bengali	LIDF	0.7242	0.7277	0.7112	0.8748	0.8738	0.8711	0.8131	0.8079	0.8109	0.8283	0.8212	0.8235
	DAC	**0.7693**	**0.7694**	**0.7548**	**0.8764**	**0.8778**	**0.8755**	**0.8270**	**0.8168**	**0.8217**	**0.8331**	**0.8307**	**0.8296**
Gujarati	LIDF	0.7315	0.7036	0.6771	0.8907	0.8897	0.8925	0.8467	0.8484	0.8447	0.8622	0.8627	0.8603
	DAC	**0.7326**	**0.7458**	**0.7379**	**0.8944**	**0.8905**	**0.8966**	**0.8485**	**0.8541**	**0.8541**	**0.8640**	**0.8702**	**0.8677**
Hindi	LIDF	0.7874	0.7823	0.7849	0.8334	0.8336	0.8324	0.8525	0.8561	0.8552	0.8541	0.8520	0.8552
	DAC	**0.7894**	**0.7848**	**0.7908**	**0.8357**	**0.8338**	**0.8366**	**0.8584**	**0.8562**	**0.8553**	**0.8558**	**0.8531**	**0.8554**
Kannada	LIDF	0.6340	0.6202	0.6317	0.8707	0.8699	0.8702	0.8257	0.8272	0.8231	0.8387	0.8392	0.8435
	DAC	**0.6643**	**0.6417**	**0.6619**	**0.8713**	**0.8771**	**0.8717**	**0.8334**	**0.8290**	**0.8284**	**0.8437**	**0.8446**	**0.8450**
Malayalam	LIDF	0.6829	0.6793	0.6598	0.8794	0.8815	0.8809	0.8229	0.8218	0.8279	0.8432	0.8419	0.8397
	DAC	**0.7139**	**0.7139**	**0.6958**	**0.8848**	**0.8825**	**0.8831**	**0.8315**	**0.8340**	**0.8342**	**0.8496**	**0.8442**	**0.8453**
Marathi	LIDF	0.6552	0.6553	0.6479	0.7424	0.7403	0.7407	0.8241	0.8163	0.8223	0.8501	0.8395	0.8412
	DAC	**0.6828**	**0.6701**	**0.6655**	**0.7474**	**0.7528**	**0.7442**	**0.8289**	**0.8307**	**0.8308**	**0.8578**	**0.8529**	**0.8547**
Odia	LIDF	0.5142	0.5328	0.5394	0.8761	0.8752	0.8767	0.8241	0.8163	0.8223	0.8501	0.8395	0.8412
	DAC	**0.6391**	**0.6233**	**0.5910**	**0.8778**	**0.8754**	**0.8766**	**0.8362**	**0.8427**	**0.8295**	**0.8509**	**0.8545**	**0.8508**
Tamil	LIDF	0.7112	0.7177	0.7131	0.8927	0.8929	0.8934	0.7846	0.7917	0.7892	0.8115	0.8089	0.8082
	DAC	**0.7559**	**0.7525**	**0.7518**	**0.8990**	**0.9024**	**0.8996**	**0.7979**	**0.8056**	**0.8029**	**0.8180**	**0.8186**	**0.8167**

Table 6: Per-language COMET scores from Extrinsic Evaluation of LIDF and DAC on PRALEKHA for LLAMA-3.2-1B and SARVAM-1-2B on English→Indic and Indic→English translation tasks. **Bold** values indicate the best score for each granularity.

Language	Method	English → Indic						Indic → English					
		LLAMA-3.2-1B			SARVAM-1-2B			LLAMA-3.2-1B			SARVAM-1-2B		
		G = 1	G = 2	G = 4	G = 1	G = 2	G = 4	G = 1	G = 2	G = 4	G = 1	G = 2	G = 4
Bengali	LIDF	34.17	34.22	33.52	48.57	48.79	48.78	57.32	55.40	55.98	68.34	68.75	68.68
	DAC	**39.25**	**38.43**	**37.47**	**50.62**	**50.06**	**50.90**	**58.59**	**55.71**	**57.20**	**69.95**	**69.68**	**69.74**
Gujarati	LIDF	35.11	32.84	31.60	64.06	63.48	63.13	36.12	38.54	36.29	64.31	66.37	65.81
	DAC	**36.12**	**38.54**	**36.29**	**64.31**	**66.37**	**65.81**	**40.75**	**41.80**	**40.95**	**67.80**	**68.21**	**67.92**
Hindi	LIDF	57.32	55.40	55.98	68.34	67.75	67.68	37.20	36.90	36.50	49.80	49.60	49.90
	DAC	**58.59**	**55.71**	**57.20**	**69.95**	**68.68**	**68.74**	**42.00**	**41.60**	**41.20**	**52.40**	**52.10**	**52.50**
Kannada	LIDF	29.71	29.01	29.47	62.91	63.42	62.81	33.20	32.80	32.40	45.80	45.60	46.00
	DAC	**32.70**	**31.23**	**32.15**	**64.84**	**64.66**	**64.82**	**37.80**	**37.40**	**37.00**	**48.60**	**48.30**	**48.70**
Marathi	LIDF	40.70	39.79	40.28	55.86	55.61	54.76	35.50	35.10	34.60	47.50	47.20	47.70
	DAC	**45.53**	**43.55**	**42.94**	**60.02**	**58.30**	**57.17**	**40.00**	**39.50**	**39.00**	**50.40**	**50.10**	**50.60**
Malayalam	LIDF	33.65	33.65	31.94	57.65	58.08	58.60	33.90	33.50	33.10	45.90	45.60	46.10
	DAC	**37.03**	**36.59**	**35.40**	**62.44**	**59.86**	**60.35**	**38.20**	**37.80**	**37.40**	**48.50**	**48.20**	**48.60**
Odia	LIDF	23.00	23.77	24.42	57.65	56.15	56.85	33.90	33.50	33.10	45.90	45.60	46.10
	DAC	**30.12**	**29.33**	**27.43**	**58.06**	**57.26**	**58.58**	**38.20**	**37.80**	**37.40**	**48.50**	**48.20**	**48.60**
Tamil	LIDF	35.01	35.01	35.04	52.87	52.51	54.14	33.20	32.80	32.40	45.80	45.60	46.00
	DAC	**39.28**	**38.95**	**38.47**	**57.96**	**58.86**	**54.45**	**37.80**	**37.40**	**37.00**	**48.60**	**48.30**	**48.70**

Table 7: Per-language ChrF scores from Extrinsic Evaluation of LIDF and DAC on PRALEKHA for LLAMA-3.2-1B and SARVAM-1-2B on English→Indic and Indic→English translation tasks. **Bold** values indicate the best score for each granularity.