Bridging the Digital Language Divide: A Comprehensive Survey of English-to-Indian Low-Resource Machine Translation

Subha Mahajan[1], Baijnath Kaushik[2], and Pooja Sharma[3]
{subhamahajan.03@gmail.com[1], baijnath.kaushik@smvdu.ac.in[2], and pooja.sharma@smvdu.ac.in[3]}

School of Computer Science and Engineering, Shri Mata Vaishno Devi University Katra, India[1,2,3]

Abstract. In a nation marked by unparalleled linguistic diversity, translating between English and India's regional languages poses challenges due to limited parallel corpora, rich morphology, and varied scripts. The paper presents a comprehensive survey of English-to-Indian low-resource machine translation approaches, covering data acquisition from public repositories and in-house sources, corpus preprocessing through tokenization, sub word segmentation, alignment, and synthetic data generation via paraphrasing, back-translation, and noise injection.

We analyze three generations of translation paradigms rule-based, statistical, and neural highlighting architectures from RNN-based encoder–decoder models to transformer-based networks and adapter-augmented large language models with sparse Mixture-of-Experts components. We compare evaluation metrics from BLEU and chrF to learned quality estimators like COMET and BERTScore and discuss human assessment protocols for adequacy and fluency. Finally, we identify future directions in transfer learning, unsupervised and semi-supervised adaptation, in-context learning with large language models, and collaborative corpus development, with the goal of bridging India's digital language divide.

Keywords: Language Translation, Evaluation Metrics, BLEU, chrF, COMET, BERTScore, Large Language Models

1 Introduction

Natural languages are those that have organically developed in human societies over time, shaped by usage, culture, and interactions. These languages serve as powerful mediums for people to share knowledge, express emotions, and communicate their thoughts and feelings with one another. They are the building blocks of human connection and understanding, expressing the variety and richness of human experience. Languages are fundamental to human communication, with thousands of native tongues spoken worldwide. Each language carries its unique alphabet, signs, grammar, and phonetics, shaped by centuries of cultural and geographical evolution. If there are a

IICTDS 2025, December 19-20, Chandigarh, India
Copyright © 2026 EAI
DOI 10.4108/eai.19-12-2025.2366265

nation where old and morphologically rich varieties of regional languages exist that is India. Computers find it easier to process the data represented in English language with standard ASCII codes than other natural languages. Yet, the development of machines that have the capability of understanding other natural languages is challenging and done with different techniques. There are much research works and applications like Multilingual Information Retrieval, Intelligent Virtual Assistant, Speech Recognition, Certified Translation, Social Media Analytics, Language Identification, Sentiment Analysis etc., developed to handle these natural languages for real time needs. This paper reviews the techniques behind the systems described above, with particular attention to how they have been adapted for the regional languages of India, hereafter referred to as IRL [1].

In today's world, the internet is no longer dominated by a single language, the presence of content in various regional languages is expanding rapidly. Google Translate has expanded its support for Indian regional languages significantly. It now offers translation services for 19 Indian languages, including Sanskrit, Assamese, Bhojpuri, Dogri, Konkani, Maithili, Mizo, and Manipuri, alongside the previously supported languages like Hindi, Bengali, Tamil, Telugu, Kannada, Malayalam, Marathi, Punjabi, and Gujarati. Much research is being carried out to facilitate users to work and interact with computers in their own regional natural languages.

In the current scenario, India continues to embrace its linguistic diversity, with significant advancements in technology to support regional languages. The Eighth Schedule of the Indian Constitution currently recognizes 22 languages, and efforts are being made to protect and advance these languages.

Machine translation (MT), sentiment analysis (SA), parts-of-speech (POS) tagging, and named entity recognition (NER) are the main fields of study in Indian regional language (IRL) processing. Text can be translated between languages while maintaining its original meaning with the aid of machine translation [2]. Sentiment analysis is used to determine whether the content is neutral, negative, or positive, as well as to comprehend the opinions or feelings expressed in the text [3]. By classifying each word in a sentence according to grammatical categories like nouns, verbs, or adjectives, POS Tagging promotes a deeper comprehension of language. [4] Important details in the text, such as names of individuals, locations, organizations, or dates, are recognized and categorized using named entity recognition.

Deep learning-based methods are gradually replacing conventional machine learning in modern applications for Indian Regional Languages (IRL). To create robust NLP solutions that can handle tasks end-to-end without requiring a lot of manual feature engineering, today's systems mainly rely on sophisticated architectures, particularly transformer models.

Researchers can concentrate on developing sizable pre-training datasets, implementing transfer learning strategies, and investigating unsupervised learning approaches with the aim to advance these deep learning approaches for IRL. These projects may uncover deeper linguistic patterns and enhance NLP systems' adaptability to India's diverse linguistic landscape.

Large language models (LLMs), such as GPT-4 and similar models, are now widely used in machine translation. LLMs are particularly beneficial for Indian regional languages, many of which have complicated grammar and little training data. They are excellent at handling languages with rich morphology and intricate sentence structures because they can learn and represent complex grammatical and semantic patterns.

2 Related Literature

Shaolin Zhu et.al. presented MoE-LLM, a novel framework developed to enhance multilingual translation in large language models. This method adds sparse Mixture-of-Experts (MoE) modules using a hybrid transfer learning strategy while maintaining the primary LLM parameters fixed, as opposed to full fine-tuning, which occasionally weakens the model's current linguistic abilities.MoE-LLM achieves up to 2.5 BLEU points more than conventional fine-tuning techniques in tests on the WMT benchmark, which covers 10 translation directions across 6 languages. Also, it outperforms other popular adaptation methods like Adapters and LoRA-F and performs well in zero-shot translation [5].

Wenhao Zhu et al. investigates the application of large language models (LLMs) to multilingual machine translation using the in-context learning (ICL) method. Two main questions are the focus of the study. The first is: a) What is the performance of LLMs when translating between multiple languages? The study examines the number of popular LLMs. They look at OPT, LLaMA2, and Falcon for English-centric models and XGLM, BLOOMZ, ChatGPT, and GPT-4 for multilingual models (OpenAI, 2023). In total, the research covers 102 languages across 606 translation directions, which are divided evenly into 202 English-centric, 202 French-centric, and 202 Chinese-centric directions. Their findings indicate that LLMs are steadily improving their multilingual translation abilities, with GPT-4 attaining a new level of performance. In fact, when compared with the popular supervised multilingual machine translation system NLLB (Team, 2022), GPT-4 outperforms it on 40.91% of the English- centric translation directions [6].

(b) What factors influence the translation performance of LLMs? The authors identify several new working patterns. First, LLMs are shown to acquire translation skills in a resource-efficient manner—producing acceptable outputs even for language pairs with virtually no training data (zero-resource languages). Second, LLMs can perform translations even when the provided instructions are suboptimal or "unreasonable," if they are supplemented by appropriate in-context learning exemplars.

Andrabi et al. developed an NMT system that translates text from English to Urdu by leveraging an LSTM encoder-decoder architecture enhanced with an attention mechanism. They compiled a diverse dataset sourced from newspapers, religious texts, and other materials to train the model, which ultimately achieved a BLEU score of 45.0 for English-to-Urdu translations [7].

Mujadia et al. introduced an NMT system that employs a transformer-based sequence-to-sequence model to facilitate bidirectional translation between English and Marathi. They trained their system using a customized dataset along with the Samanantar [8] parallel corpus to achieve strong bilingual alignment. The model performed well, scoring 22.0 BLEU for English-to-Marathi translation and 31.0 BLEU for Marathi-to-English. These results show that the system is effective in both translation directions [9].

Madaan et al. introduced a multilingual NMT system that integrates Bhojpuri, Sindhi, English, and Magahi (a dialect of Maithili). Their approach achieved a peak BLEU score of 36.2 for translating Sindhi into English [10]. To validate its performance, the system was benchmarked against Google's Multilingual NMT tool and both the training and evaluation processes leveraged data from the IIT Bombay English-Hindi corpus and the MT Summit 2019 dataset.

3 Generic Block Diagram

The generic model for language processing in a machine-translation (MT) involves several stages to convert text from a source language to a target language while preserving its meaning as shown below in **Figure 1**:

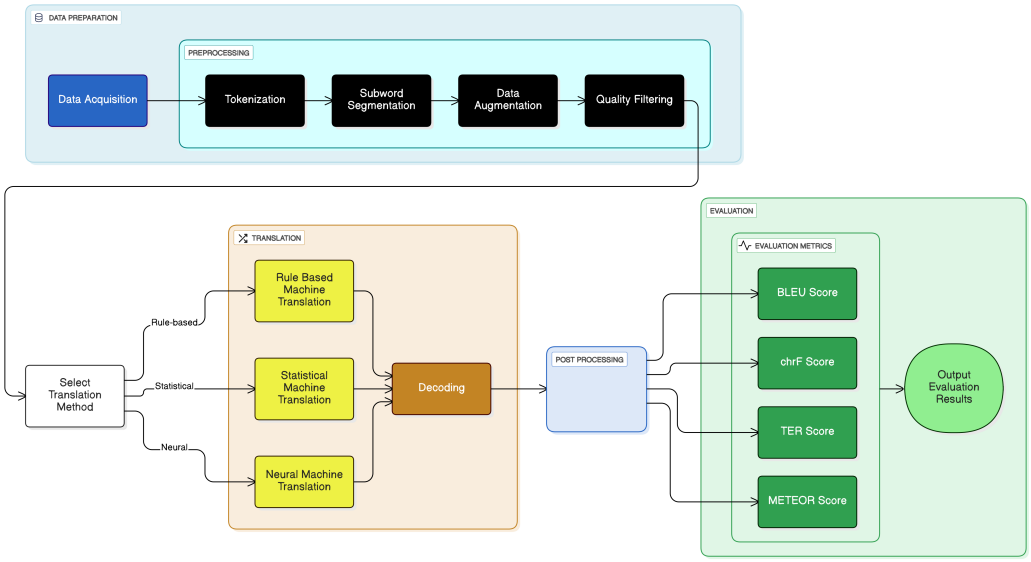


Fig. 1. Generic Model for Language Translation

Each block feeds into the next, turning an unstructured source sentence into a polished target-language output. The roles of the individual stages are:

3.1 Data Acquisition

Data acquisition involves sourcing parallel corpora from public datasets such as EMILLE, the IIT Bombay English–Hindi corpus, and Ai4Bharat, as well as harvesting in-house bilingual text scraped from news portals and government websites. Monolingual target text is collected from diverse sources including news articles, Wikipedia dumps, and social media streams. To further enrich training data for low-resource language pairs, synthetic parallel data can be generated by back-translating monolingual Hindi or other regional texts into English.

3.2 Preprocessing Techniques

Corpus preprocessing lays the groundwork for any neural or statistical machine translation system. When building an English to Indian language MT pipeline, we typically assemble a parallel corpus drawn from diverse domains such as newswire, web articles, religious texts, and everyday conversational sentences, to ensure broad coverage of vocabulary, style, and named entities. Preprocessing steps involve following phases:

Tokenization, Sub word Segmentation, and Filtering. In any machine translation pipeline, tokenization is a crucial preprocessing step that is applied to both the source and target texts. By breaking down sentences into smaller units, such as words, phrases, punctuation, and script-specific elements, it minimizes vocabulary ambiguity and gives the model consistent input. Accurate tokenization not only aids in the proper alignment of source and target tokens but also teaches the model reliable translation patterns, which ultimately enhances the accuracy and fluency of the output.

During preprocessing, the choice of tokenization tools depends on the language and script being used. For English, widely adopted tools such as Moses or spaCy work well. In contrast, for Indian languages including Hindi, Bengali, Kannada, Malayalam, Tamil, Telugu, Punjabi, Gujarati, Odia, and Assamese specialized libraries like the Indic-NLP Library or iNLTK are more appropriate. Typically, sentences are split using whitespace and punctuation. However, this process must be handled carefully to respect each language's unique features such as clitic rules, compound word structures, and script-specific writing conventions so that the text is segmented accurately and meaningfully.

Data Argumentation. The term refers to the procedure of creating new training examples from already existing data by means of transformations that do not affect the original meaning or labels. In machine translation, as well as in other natural language processing tasks, this often means the creation of synthetic sentence pairs by paraphrasing, back-translation, or introducing controlled noise. This method improves the parallel corpus without necessarily gathering more raw data. Data augmentation exposes the model to a greater range of linguistic contexts by altering elements like word choice, sentence structure, or even character patterns; as a result, it improves system robustness and its capacity to handle real-world text.

By expanding the dataset and adding more linguistic variation, such as synonyms, colloquialism, and paraphrasing, this method addresses the lack of training data in contemporary neural models. It further helps to increase the robustness of the system to typos and unusual sentence structures and reduces overfitting because it prevents the model from memorizing a limited set of examples. These advantages have made data augmentation one of the most important methods in machine translation, particularly for low-resource languages. It is part of maintaining the reliability of translation systems when facing noisy text or domain shifts.

Alignment and Quality Filtering. Sentence-level alignment for English Indian language pairs is done by applying methods like length-ratio analysis, lexical overlap metrics, and embedding-based similarity scores to ensure accurate matching. Pairs that are below the stipulated alignment confidence thresholds or those that fail language-identification checks are removed to take out misaligned

or low-quality data. Exact or near-duplicate pairs are removed, and this cleaned data is shuffled to avoid domain-based clustering. Each sentence pair is also further annotated with domain information, such as newsgroups, web content, religious texts, or conversational speech, to allow for balanced sampling during training and to support robust model performance across a wide array of text types.

Wherever possible, we can also enforce one-to-one correspondence between source and target tokens to streamline sentence-level filtering and ensure a consistently aligned, high-quality bilingual corpus. A joint sub word vocabulary spanning both source and target corpora is then constructed to promote the transfer of rare roots and loanwords across languages. Finally, parallel sentence pairs that fall outside predefined length boundaries (for example, fewer than three tokens or more than 250 tokens) or that exhibit monolingual noise single-language segments within a parallel instance are filtered out to preserve corpus consistency and alignment quality.

3.3 Machine Translation

In natural-language processing, machine translation (MT) powers applications such as cross-language information retrieval, multilingual chatbots, document localization, and question answering. MT is the automatic conversion of text or speech from a source language into a target language while preserving meaning, style and fluency. Formally, MT finds

$$y^* = argmax_y P(y|x) \tag{1}$$

the target sentence y that maximizes the conditional probability given the source x. The three classic paradigms of machine translation are:

Rule-based machine translation (RBMT). It is the classical MT paradigm that derives translations by applying hand-crafted linguistic knowledge, bilingual dictionaries with morphological, syntactic and semantic grammars, rather than learning from data. In a rule-based NLP translation framework, every decision is driven by an explicitly encoded set of linguistic patterns and transformation rules. For example, English sentences typically follow a Subject–Verb–Object (SVO) order, whereas Hindi favours Subject–Object–Verb (SOV). Since purely structural mappings often miss deeper semantic or contextual cues, these systems enrich their rule base with external lexical and world-knowledge resources thesauri, ontologies or lexical databases like WordNet to supply the reasoning and background knowledge essential for accurate, fluent translations.

RBMT relies on a handcrafted grammar and a bilingual lexicon to map source-language strings into target-language equivalents. By contrast, Knowledge-Based MT (KBMT) and Principle-Based MT (PBMT) augment this pipeline with deep parsing: they analyze source text at the lexical, phrasal and syntactic levels and then apply those structured representations to generate the target-language output. [11].

Statistical Machine Translation (SMT). It is a data-driven paradigm that generates translations by learning probabilistic models from large bilingual corpora rather than hand-crafting linguistic rules.

In the statistical paradigm, machine translation selects the most likely target rendition by scoring candidates with probability distributions learned from bilingual or multilingual corpora [12]. By contrast, Example-Based MT (EBMT) relies on a database of previously translated sentence pairs: it retrieves examples like the input, aligns matching segments, and recombines them analogically to produce the final translation [13].

Beyond Statistical machine translation, the field has largely shifted to Neural machine translation which subsumes many Statistical machine translation components into end-to-end encoder–decoder architectures.

Neural Machine Translation (NMT). Beyond rule-based and statistical methods, researchers have embraced neural-based architectures, deep-learning models such as encoder–decoder networks to boost performance across language-processing tasks. It is an end-to-end, data-driven approach that trains a single neural network to map entire source sentences directly to target sentences. It models the conditional probability of a target sequence given a source sequence, and today stands as the dominant paradigm in MT, often rivalling human performance on high-resource language pairs.

Machine translation's goal is to pick, for a given source sentence x, the target sentence y^* that maximizes the conditional probability $P(y \mid x)$. Neural MT realizes this by training a single, end-to-end encoder–decoder network often augmented with attention that compresses x into a continuous representation and decodes it into the most likely y, thereby learning $P(y \mid x)$ directly. In neural MT, the encoder compresses a source sentence of arbitrary length into a fixed-size context vector that captures its meaning; the decoder then unfolds that vector into the target-language output. The whole encoder–decoder network is trained end-to-end to maximize the conditional probability $P(y \mid x)$, driving it to learn the most likely translation for any given source sentence [14, 15].

3.4 Decoding

We can begin the decoding phase by mapping the model's probability distributions into a coherent target-language string. In practice, this involves running inference with strategies such as beam search, where we can retain the top N hypotheses at each time step and apply length or coverage penalties to discourage truncated or repetitive outputs or sampling methods like top-k and nucleus (top-p) sampling when diversity of renderings is desired. Because English Indian language systems typically operate on sub word units (e.g., BPE or Sentence Piece), the decoder emits a sequence of sub word tokens that must later be merged. To obtain a raw token stream that accurately reflects the model's best approximation of the source sentence, trade-offs between translation adequacy, fluency, and inference speed can be achieved by carefully adjusting the beam width, temperature, and penalty parameters.

3.5 Post-Processing

Post-processing pipeline is employed to further transform the tokenized text into a fluid, natural-sounding sentence in the target Indian language. Subword units are first detokenized into complete words, and punctuation marks like quotation marks, commas, and full stops are corrected. Next,

either using monolingual language models or predefined rule sets, true casing is applied to correct initial sentence capitalization and named entity capitalization.

To make the result culturally authentic, the next step is script-specific orthographic normalization, which involves normalizing the conventions for placing diacritical marks in Tamil and applying schwa-deletion rules for Hindi. To preserve proper names and technical terms, named-entity transliteration and domain-specific glossary substitutions come next. Dates, numbers, and URLs are substituted with their original forms in placeholders. Together, these processes improve the text so that the final translation adheres to the target script's conventions and reads naturally.

3.6 Evaluation Metrics

We can complete the workflow by evaluating the decoded and post-processed translations using both automated metrics and human assessments, as shown in **Figure 2**.

Human Evaluation. By using native speakers or bilingual annotators to evaluate translation quality, these metrics can identify subtleties that are often overlooked by automated evaluation techniques.

Adequacy and Fluency.

- **Adequacy:** The degree to which a translation allows the original meaning to come across is what this criterion represents. It usually receives scores on a five-point scale, with higher ratings corresponding to closer rendering of the intended message.
- **Fluency:** This is the rating for grammatical correctness and naturalness of the translation. It is similarly evaluated on a five-point scale, with higher scores indicating greater linguistic naturalness and grammatical correctness.

Machine Translation systems depend on robust evaluation metrics, whether rule-based, statistical, or neural, to estimate translation quality correctly. Typically, such metrics fall into categories such as n-gram overlap, edit distance, and semantic similarity, with each capturing distinct facets of translation accuracy, fluency, and overall adequacy.

Automatic Metrics.
n-gram Overlap Metrics.

- **BLEU (Bilingual Evaluation Understudy):**BLEU measures the degree of n-gram matches between a candidate translation and one or more reference translations. It computes a modified precision for each n-gram order, applies a brevity penalty to discourage overly short outputs, and aggregates via a geometric mean:

$$BLUE = BP.exp(\sum_{n=1}^{N} w_n ln(p_n)) \tag{2}$$

$$\sum_{n=1}^{\infty} (a_n cos\frac{n\pi x}{L} + b_n sin\frac{n\pi x}{L}) \tag{3}$$

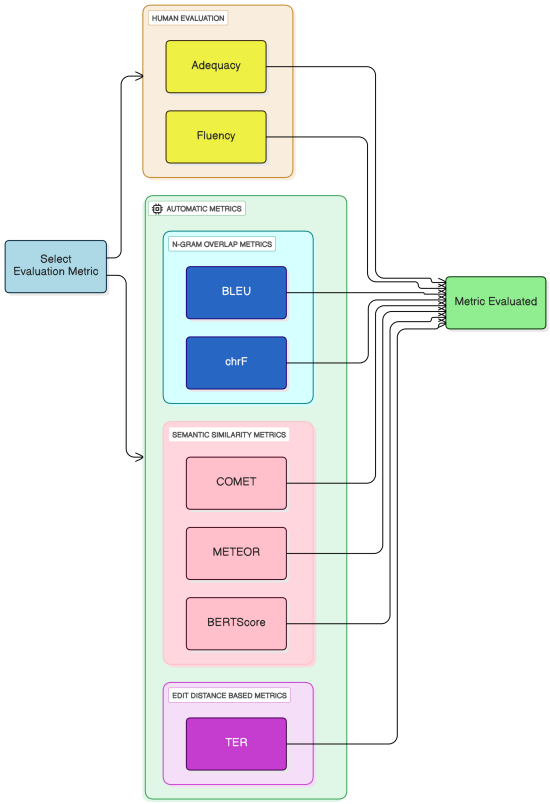


Fig. 2. Workflow of Evaluation Metrics for Machine Translation

where

- p_n is the modified precision for n-grams of size n
- w_n are usually uniform weights (e.g; $w_n = \frac{1}{N}$ for $N = 4$

where info weight$(ngram) = log_2 \frac{(count(ngram\ in\ corpus))}{(count\ (prefix\ in\ corpus))}$

- **chrF:** chrF operates on character n-grams instead of word tokens, improving robustness on morphologically rich languages. It computes an F-score over character n-gram precision and recall:

$$(1-\beta^2)\frac{P_c R_c}{\beta^2 P_c + R_c} \tag{4}$$

Where $P_c R_c$ are precision and recall of character n-grams, and β balances recall vs precision.

Edit-Distance-Based Metrics

- **TER(Translation Edit Rate):** TER counts the minimum number of edits insertions, deletions, substitutions, and shifts needed to change the candidate into the reference, normalized by reference length:

$$TER = \frac{\#edits}{avg\ ref\ length} \tag{5}$$

A lower TER indicates fewer required corrections. Eg: For "the cat is on mat" vs "the cat is on the mat"

$$TER = \frac{1}{6} \approx 0.17(one\ insertion) \tag{6}$$

Sematic Similarity Metrics

- **METEOR:** METEOR aligns unigrams via exact, stem, synonym, and paraphrase matches, computes precision P and recall R, then applies a fragmentation penalty Pen:

$$F_\alpha = \frac{PR}{\alpha P + (1-\alpha)R'} \quad Score = (1-Pen)F_\alpha \tag{7}$$

where α typically $= 0.9$, and $Pen = \gamma(\frac{\#chunks}{\#matches})\beta$ with tuning parameters γ, β

- **BERT Score:** BERT Score uses contextual embeddings from BERT to match each candidate token to a reference token by cosine similarity, then computes precision, recall, and F1 over those similarities:

$$BERTScore_F = \frac{2\sum_i^{max\ cos(E_i^c, E_j^r)}}{\sum_i^{mac\ cos(E_i^c, E_j^r)} + \sum_j^{mac\ cos(E_i^c, E_j^r)}} \tag{8}$$

where E_i^c, E_j^c are the hidden states of candidate and reference tokens.

- **COMET:** COMET trains a regression model on human judgments, using encoder representations of candidate, reference, and source sentences. It outputs a scalar quality score calibrated to human scales. Architecturally, it fine-tunes a multilingual encoder with a pooling and regression head to minimize mean squared error against human scores.

4 Conclusion

This paper discusses state of English-to-Indian low-resource machine translation from end to end covering data acquisition, corpus preprocessing, and three generations of MT paradigms (rule-based, statistical, and neural models). We highlighted how India's morphological richness, script diversity, and limited parallel corpora pose unique challenges and how targeted techniques such as data augmentation, back-translation, and noise injection can amplify scarce resources. Ultimately, integrating transfer learning and in-context learning with dedicated pre-training corpora emerges as a promising path toward narrowing India's digital language divide.

We can further elevate translation quality by integrating transfer learning and unsupervised methods, including sparse Mixture-of-Experts adapters and in-context learning with large language models. By developing dedicated pre-training corpora for regional languages, refining domain adaptation pipelines, and systematically varying input representations, we prepare models to capture complex grammatical patterns and remain robust to real-world noise.

References

[1] Harish BS, Rangan RK. A comprehensive survey on Indian regional language processing. SN Applied Sciences 2020 2:7. 2020 6;2(7):1-16. Available from: https://link.springer.com/article/10.1007/s42452-020-2983-x.

[2] Reddy MV, Hanumanthappa M. Indic Language Machine Translation Tool: English to Kannada/Telugu. Lecture Notes in Electrical Engineering. 2013;213 LNEE:35-49. Available from: https://link.springer.com/chapter/10.1007/978-81-322-1143-3$_4$.

[3] Kumar A, Kohail S, Ekbal A, Biemann C. IIT-TUDA: System for Sentiment Analysis in Indian Languages Using Lexical Acquisition. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2015;9468:684-93. Available from: https://link.springer.com/chapter/10.1007/978-3-319-26832-3$_6$5.

[4] BR S, Kumar P R. Kannada Part-Of-Speech Tagging with Probabilistic Classifiers. International Journal of Computer Applications. 2012 6;48(17):26-30.

[5] Zhu S, Pan L, Jian D, Xiong D. Overcoming language barriers via machine translation with sparse Mixture-of-Experts fusion of large language models. Information Processing & Management. 2025 5;62(3):104078. Available from: https://www.sciencedirect.com/science/article/pii/S0306457325000202?via

[6] Zhu W, Liu H, Dong Q, Xu J, Huang S, Kong L, et al. Multilingual Machine Translation with Large Language Models: Empirical Results and Analysis. In: Findings of the Association for Computational Linguistics: NAACL 2024. Stroudsburg, PA, USA: Association for Computational Linguistics; 2024. p. 2765-81.

[7] Andrabi SAB, Wahid A. Machine Translation System Using Deep Learning for English to Urdu. Computational Intelligence and Neuroscience. 2022 1;2022:1-11.

[8] Ramesh G, Doddapaneni S, Bheemaraj A, Jobanputra M, Raghavan AK, Sharma A, et al. Samanantar: The Largest Publicly Available Parallel Corpora Collection for 11 Indic Lan-

guages. Transactions of the Association for Computational Linguistics. 2022 2;10:145-62. Available from: https://aclanthology.org/2022.tacl-1.9/.

[9] Mujadia V, Sharma DM. English-Marathi Neural Machine Translation for LoResMT 2021; 2021. Available from: https://aclanthology.org/2021.mtsummit-loresmt.16/.

[10] Madaan P, Sadat F. Multilingual Neural Machine Translation involving Indian Languages; 2020. Available from: https://aclanthology.org/2020.wildre-1.6/.

[11] Goyal V, Lehal GS. HINDI TO PUNJABI MACHINE TRANSLATION SYSTEM. 2011:1-6. Available from: https://dl.acm.org/doi/pdf/10.5555/2002440.2002441.

[12] Godase A, Govilkar S. MACHINE TRANSLATION DEVELOPMENT FOR INDIAN LANGUAGES AND ITS APPROACHES. International Journal on Natural Language Computing (IJNLC). 2015;4(2).

[13] Somers H. Review Article: Example-based Machine Translation. Machine Translation. 1999;14(2):113-57. Available from: https://link.springer.com/article/10.1023/A:1008109312730.

[14] Tab-delimited Bilingual Sentence Pairs from the Tatoeba Project (Good for Anki and Similar Flashcard Applications);. Available from: http://www.manythings.org/anki/.

[15] Part-Of-Speech Tagging using Neural Network;. Available from: https://www.researchgate.net/publication/228594059.