A Novel Framework for Sanskrit-Gujarati Symbolic Machine Translation System

Jaideepsinh K. Raulji[1]
Navrachana University
Vadodara, India

Kaushika Pal[3]
Sarvajanik College of Engineering and Technology
Surat, India

Jatinderkumar R. Saini[2]*
Symbiosis Institute of Computer Studies and Research,
Symbiosis International (Deemed University), Pune, India

Ketan Kotecha[4]
Symbiosis Centre for Applied Artificial Intelligence,
Symbiosis International (Deemed University), Pune, India

Abstract—Sanskrit falls under the Indo-European language family category. Gujarati, which has descended from the Sanskrit language, is a widely spoken language particularly in the Indian state of Gujarat. The proposed and realized Machine Translation framework uses a grammatical transfer approach to translate the written Sanskrit language to Gujarati. Because both languages are morphologically rich, studying the morphology of each item is difficult but necessary to incorporate into implementation. To improve the implementation accuracy and translation clarity, an in-depth research of the creation of Nouns, Verbs, Pronouns, and Indeclinables, as well as their mappings, has been carried out. Tokenization, lemmatization, morphological analysis, Sanskrit-Gujarati bilingual synonym-based dictionary, language synthesis, and transliteration are the proposed framework's primary components. The implementation outcome was tested on 1,000 phrases, using the automated Bilingual Evaluation Understudy (BLEU) scale which yielded a value of 58.04 It was also tested on the ALPAC scale, yielding the Intelligibility score of 69.16 and the Fidelity score of 68.11. The results are encouraging and prove that the proposed system is promising and robust for the implementation in the real world applications.

Keywords—*Bilingual synonym dictionary; Gujarati; lemmatization; machine translation system (MTS); morphological analyzer; Sanskrit; synthesizer; transliteration*

I. INTRODUCTION

Aside from computers' incredible processing capacity, researchers have traditionally found it difficult to create and execute Machine Translation Systems (MTS) with great precision. The complexity of natural languages is due to lexical, semantic and contextual aspects, sophisticated morphological nature, and most importantly the pragmatics and discourse, which refers to the speaker's intent. The designing and the implementation of a Machine Translation (MT) system can be done in a variety of ways.

In this paper, a technique for constructing a symbolic MT implementation from Sanskrit to Gujarati is offered due to rare availability of bilingual parallel corpora which form the basis for machine learning techniques. A pure dictionary- based translation system uses no intermediate representation to convert from source to target language.

The Machine Translation (MT) approaches could be classified broadly into four categories, as is depicted diagrammatically in Fig. 1. Notably, two of these four broad categories can be further divided into two sub-categories for each broad category. Historically speaking, the correlation of the categorization of the machine translation approaches existing in the pertinent scientific literature could also be done for the rationalistic, empirical and the hybrid approaches.

For the present research work, a dictionary has been used to accomplish the task, as it will offer a word to word transformation through sub-tasks like morphological analysis supplemented with lemmatizer, grammatical transfer, synthesis. It will later rearrange the words in the sentences of the target language. The method is simple to use, but it is not versatile enough to be applied several other pairs.
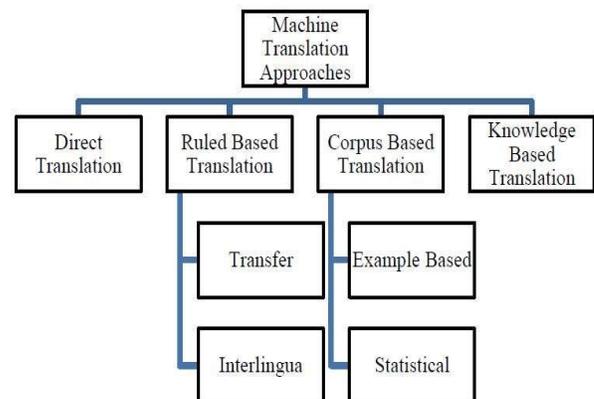

Fig. 1. MT Approaches [2].

The transfer approach is more complicated than the preceding one since it examines properties as lexical, syntactic & semantics and morphological aspects of language. Because it is built to accommodate various languages, the Interlingua approach is still more versatile than transfer. Interlingua is used to construct an intermediate representation of natural language also known

---

*Corresponding Author

as pivot language which is then transformed to target [1]. The relativeness of Direct, transfer, and interlingua methods are strategically connected, as shown in Fig. 1. If a significant number of labelled, aligned, or parallel corpora are available, the corpus-based technique tends to be accurate enough. Because the grammatical mechanics of a language have no effect on corpus-based models, a single corpus-based MT model can be used to train a model in any language.

II. LITERATURE REVIEW

The amount of study and money invested on the MT system after World War-II is notable. However, after the Automated Language Processing Committee (ALPAC) issued a report in 1966 CE, the funding for the MT system was substantially decreased. After the 1990s, a ray of optimism emerged, thanks to lower computer hardware costs and increased memory and calculation capacity, which led to new techniques. MT-related work used to be limited to languages such as English, Russian, French, and Spanish, but in today's world, MT systems are being developed for a wide range of languages, including Sanskrit.

As shown in Fig. 2, Cancedda et al. [3] presented a diagrammatic representation of the various methods used for machine translation. Many MT systems use Sanskrit and Gujarati in some form or another. Rathod and Sondur presented English-Sanskrit Translator and Synthesizer (ETSTS) which is a combination of rules and example-based MT implementation which transforms sentences to speech [5]. E-Trans is an English to Sanskrit MT tool based on Synchronous CFG proposed by Bahadur et al. The language representation part is implemented through SCFG [6]. Subramaniam [7] built Sanskrit to English rule-based translator. Sandhi Splitter, Translation Generator with Morphological parser are the two important components of the implementation. English to Sanskrit Example-Based MT system is developed by Mishra and Mishra [8] [9]. The main components of the system are Part-of-Speech (POS) tagger, Gender-Number-Person (GNP) detection, as well as Noun, Root Verb, and Adverb detection. A nice piece of work which translates Sanskrit to Hindi has been developed at Jawaharlal Nehru University (JNU). Word sense disambiguation, anaphora resolution, prose order generation, and other modules were studied by the researchers while it was claimed that Yoga and Ayurveda will be added to the system's capabilities [10]. AnglaBharti MT system translates English to Sanskrit. It is based on Paninian Grammar rules also known as PLIL code [11]. Raulji and Saini [4] presented a comparison of the various machine translation systems involving Sanskrit and Gujarati as the language pair.

Sreedeepa and Idicula [12] developed Sanskrit-English MT implementation based on Interlingua. In analysis of language, LFG is used which helps in finding semantic relation between words in a sentence. The semantic analysis was done through Karaka analyzer through Paninian grammar framework. Using interlingua approach, Sanskrit to English MT is developed by Sreedeepa and Idicula [12].

It used Lexical Function Grammar (LFG) build using Paninian Karaka Analysis. The karaka analysis is used to analyse syntactico- semantic relations between words in a sentence. Gupta et al. developed Sanskrit to English MT system. The system is based on grammatical aspect of the language pair [13]. Singh et al. [24] deployed the hybrid usage of Neuro Machine Translation (NMT) and Rule Based Machine Translation (RBMT) to design the MTS for the Sanskrit-Hindi language pair. Akhand et al. [25] while reviewing the MT systems for the Bangla language, found that no MTS exists that involves Bangla-Sanskrit language pair. In addition to the above mentioned MT systems, the researchers have also attempted to evaluate the accuracy of MTS. For instance, Sabtan [26] used the data of social media itself as a language for translation. Ehab et al. [27] investigated the MT using the example based approach for the language pair comprising of Arabic and English languages. Pudaruth et al. [28], similarly, discussed the Rule Based Machine Translation (RBMT) system for the language pair comprising of English and Creole.

Given the richness of the Sanskrit language, there have been several attempts by the researchers involving the analysis of the language. Derivative nouns [29], word segmentation and morphological parsing [30], noun declension and verb conjugation [31], dependency parsing [32], lemmatization [33], and constituency mapper [34] are a few such instances. Similarly, for the Gujarati language, the researchers have explored chunking [35], stemming [36], inflections [37], lexicon-based analysis [38], speech recognition [39], character recognition [40], and spell checking [41]. Based on the detailed literature review till date, we have observed that there is a definite dearth of research on MTS for the Sanskrit-Gujarati language pair. It has also been observed that no formal research works are dedicated to the morphological analysis, comparison and linking of both languages together. The present research work bridges all these gaps and presents not just the theoretical framework but also the working model of the MTS involving these two Indian languages. The results have been found to be encouraging and motivating. Rest of the paper is organized as follows: Section III presents the characteristics of Sanskrit and Gujarati languages while Section IV presents a detailed discussion on the research methodology. This is followed by a section each on results, and conclusions and future work.
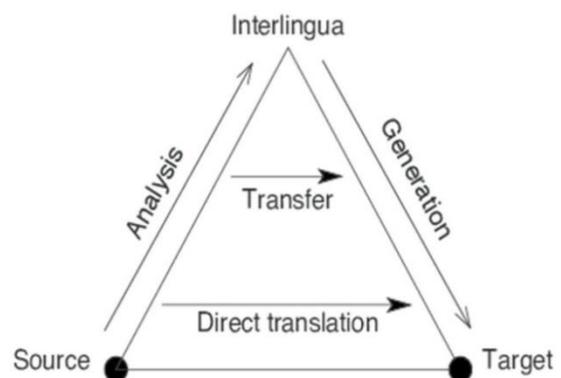

Fig. 2. The Translation Methods [3].

III. Characteristics of Sanskrit and Gujarati Languages

Sanskrit and Gujarati are included in the Indian Constitution as scheduled languages historically belong to Indo-Aryan family of languages. Gujarati is less ordered and regular than Sanskrit. Sanskrit is rich and morphologically structured hence tends to be focused internationally for research in computational linguistics domain. Gujarati is official language of state of Gujarat. Apart from state of Gujarat, it is also spoken in adjoining parts of Rajasthan, Madhya-Pradesh and Maharashtra states of India.

Many Gujarati community are also found in countries viz. UK, USA, Canada, Australia, New Zealand, and few African continent's countries. Sanskrit is an ancient spoken language with tradition dating back to the Vedic period since 2000 BCE. Gujarati is a contemporary language compared to Sanskrit, with a spoken heritage dating back to roughly 1100 CE. [14] [15] [16]. Sanskrit is written in a variety of scripts, the most common of which being Devanagari [17], whereas Gujarati is written in Abugida script, which is a variant of Devanagari. Table I lists a few characteristics of these language pairs [18].

TABLE I. Characteristics of Sanskrit and Gujarati Languages

LanguageElements	Sanskrit	Gujarati
Consonants	33	33
Vowels	12	12
Gender (3 genders in each)	Masculine	Masculine
	Feminine	Feminine
	Neuter	Neuter
Number (3 numbers in Sanskrit and 2 in Gujarati)	Singular	Singular
	Dual	Plural
	Plural	Plural
Case Markers (8 Cases in each)	Nominative	Nominative
	Accusative	Accusative
	Instrumental	Instrumental
	Dative	Dative
	Ablative	Ablative
	Genitive	Genitive
	Locative	Locative
	Vocative	Vocative
Persons (3 persons in each)	First	First
	Second	Second
	Third	Third
Tense (6 tenses in Sanskrit and 5 in Gujarati)	Present	Present
	Aorist	Past (Simple)
	Past (Imperfect)	Past (Imperfect)
	Past (Perfect)	Past (Perfect)
	Future (First)	Future
	Future (Second)	Future
Moods (4 in Sanskrit and 3 in Gujarati)	Imperative	Imperative
	Potential	Potential
	Conditional	Conditional
	Benedictive	No equivalent

IV. Methodology

The strength of the language analysis performed on the source and target languages determines the success of a rule-based system. Better findings come from a thorough examination of source and target language divergence and similarity mappings. The rule-based paradigm is given here, with an emphasis on grammatical similarities and divergence between Sanskrit and Gujarati, as well as extensive dictionary support. Due of its complexity, the main MT work entails a large number of subs and ancillary tasks. The following sub-sections present the various Natural Language Processing (NLLP) and Computational Linguistic (CL) tasks to finally yield complete MTS. The diagrammatic flow of the working of the proposed system is depicted in Fig. 3. The input text provided in Sanskrit language gets translated to the Gujarati language after passing through stages like tokenization, morphological analysis, lemmatization, translation, synthesis and transliteration.
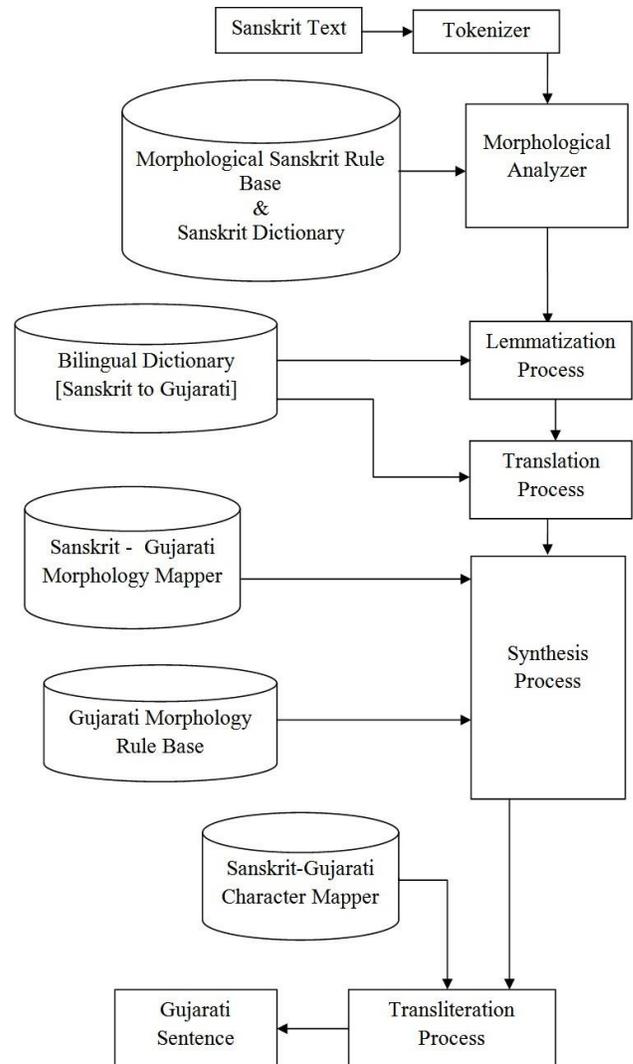

Fig. 3. Framework of Sanskrit-Gujarati MT Implementation.

1) Tokenization phase: Tokenization is the process of breaking down paragraphs into sentences, with each sentence serving as a token. If the sentence is broken down into multiple words, each word serves as a token. Because Sanskrit has a lot of word morphology, the text has to be tokenized into words before it can be properly analyzed. In the language, space separates each word. Fig. 4 depicts the procedure. The single vertical line depicts end of sentence ('|') with 2404 as its Unicode and double vertical lines ("||") depicts end of poetic stanza with 2405 as its Unicode. These two symbols are used to Sanskrit sentence tokenizers. Although the use of '.' (full stop) in modern Sanskrit literature is incorrect, it is nonetheless included in the method for Sentence Boundary Detection (SBD). The space delimiter is used to tokenize Sanskrit words.

2) Morphological-analysis phase: Except for indeclinables, every Sanskrit word can reflect its unique grammatical qualities by adding inflection to the root word. Indeclinables are words that do not possesses any kind of inflectional variants and hence added to dictionary/wordnet. Sanskrit pronouns also have irregular declension patterns; hence they were entered straight into the datastore. The inflectional affixes of the remaining nouns are examined using a grammar rule base and dictionary. The surface grammatical information for the word is provided by the Sanskrit dictionary, such as pronoun, noun, verb, and so on. The G (Gender)-N (Number)-C (Case) labels for noun constituent and adjective constituents are used to tag a word using deep structure research employing Sanskrit grammatical rules [19]. For verbs, there are Tense-Aspect- Modality (TAM), Person, Number, 'Parasmaipada', and 'Aatmanepada' labeling modes [19]. Finally, morphological analyzer produces words that have been tagged with grammatical information. To quickly develop the prototype, high-frequency words from corpora of about 75000 words were used to find 75 stop-words, which were then put to the dictionary. This reduces translation time-complexity [20]. The author in [42] presents Sanskrit stop-word analysis while comparison of such analyzers is presented in [43]. The algorithm is shown in Fig. 5 as a logic flow diagram.
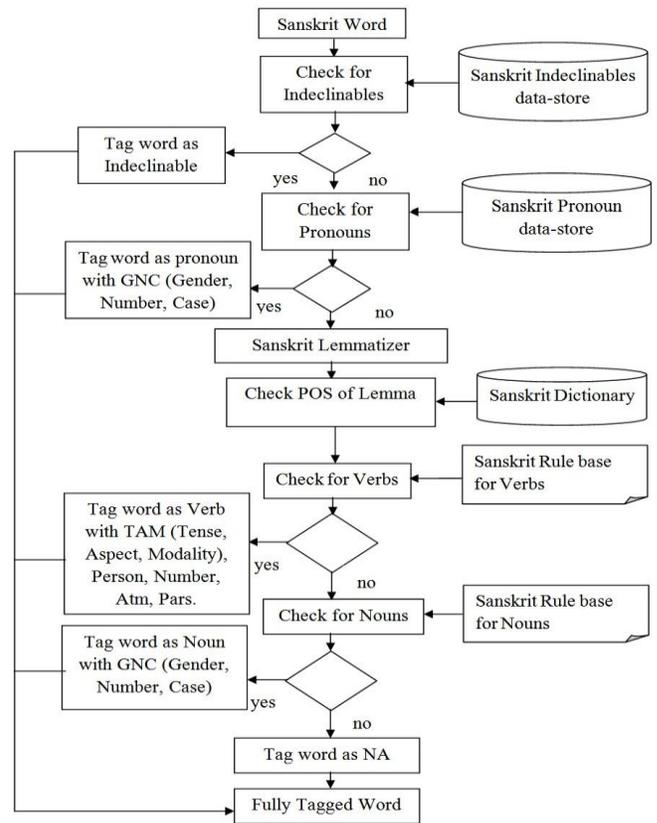

Fig. 4.   Tokenizing Sanskrit Text.

Fig. 5.   Morphological Analyzer.

3) Lemmatization phase: A lemma (root word or dictionary form) is derived from an inflected word using this method. Nominal and verbal inflections abound in Sanskrit. If Aatmanepada and Parasmaipada are included, a single Sanskrit noun has 24 variants and 18 verb variants in its inflected forms. As a result, storing all Sanskrit words with such inflection forms necessitates a large number of dictionary entries, and computational retrieval becomes time-consuming. As a result, the dictionary will only contain Sanskrit terms in their basic form. After applying suffix stripping rules, the lemmatizer examines the token and searches the dictionary for the word. Fig. 6 depicts the process diagram.

4) Translation phase: For the translation procedure, the lemma obtained from the Lemmatizer phase is used as the input. The obtained lemma is compared with a bilingual Sanskrit- Gujarati dictionary. It is notable that the output of the lemmatization phase is the root form of the word. It is also noteworthy that we have directly implemented the lemmatizer instead of a stemmer which does not necessarily give the root form. The Sanskrit root word is matched within a bilingual Sanskrit-Gujarati dictionary to get the Gujarati equivalent as mentioned in Fig. 7. To get the Gujarati equivalent, the Sanskrit root word (Sanskrit lemma) is matched in order. The order of matching is as follows: Indeclinables, Pronouns, Verbs, and the remaining Nominals.
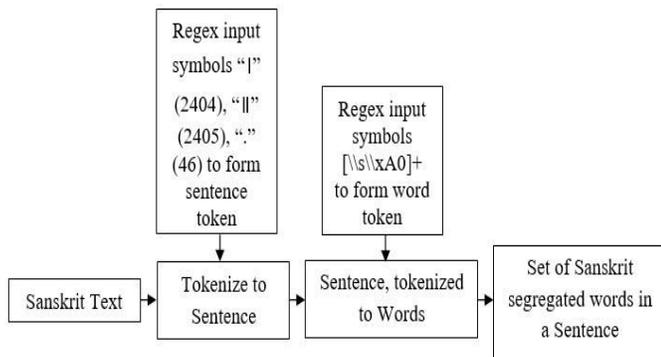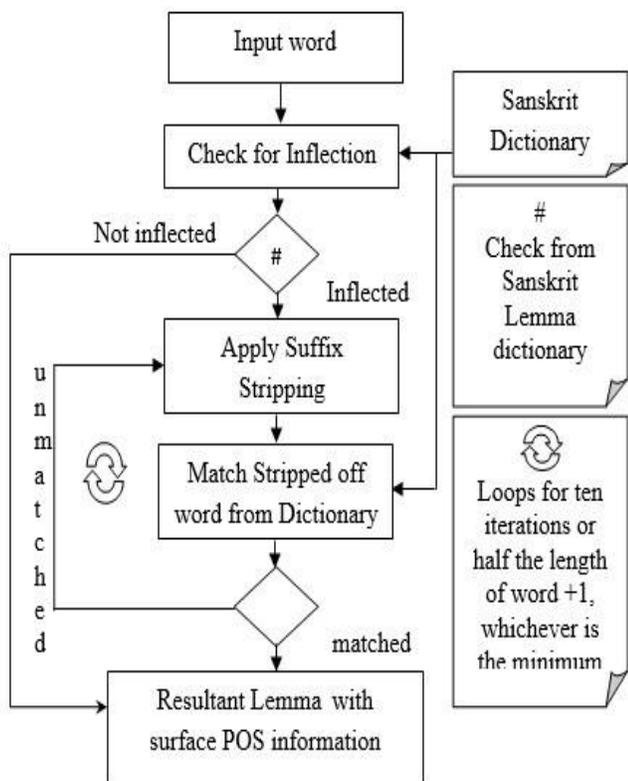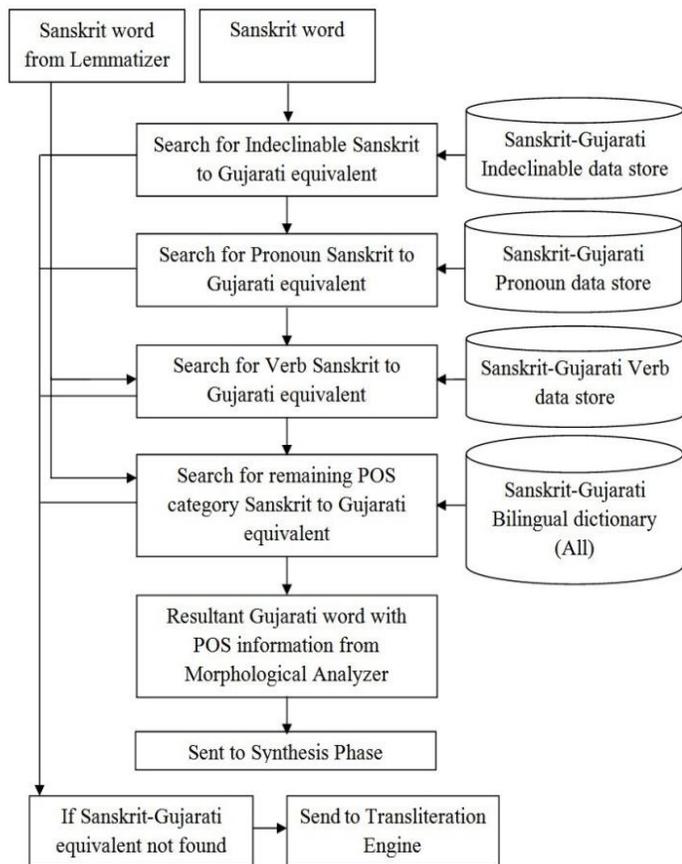
Fig. 6.   Sanskrit Lemmatizer.

Fig. 7.   Translation Phase.

5) Synthesis phase: This phase has mapping repository of morphology of Sanskrit to Gujarati for various Parts of Speech (POS) including nouns, adjectives and verb constituents. Based on the morphological rules derived from the grammar of the source language, it maps to rules of target language and is finally applied on Gujarati lemma to form a meaningful word. Fig. 8 depicts this process diagrammatically.
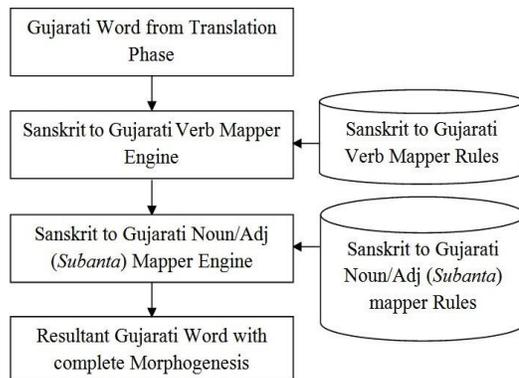

Fig. 8.   Synthesis Phase.

6) Transliteration phase: The process of converting language script X to language script Y without harming pronunciation is called as Transliteration. Here the unmatched words from the translation phase are supplied to the transliteration phase, which finally changes Sanskrit (Devanagari) script into Gujarati (Gujarati-Devanagari) equivalents script letters while maintaining their pronunciation. Unmatched terms are mostly seen in the Named Entity class. A Unicode UTF-8 Devanagari scripted font is used to identify the single characters of a Sanskrit word. To generate UTF-8 Gujarati script characters, add 384 to the word, as illustrated in Fig. 9. Because Sanskrit and Gujarati are both free-word order languages, rearranging words in a phrase has little impact on the meaning of the sentence.
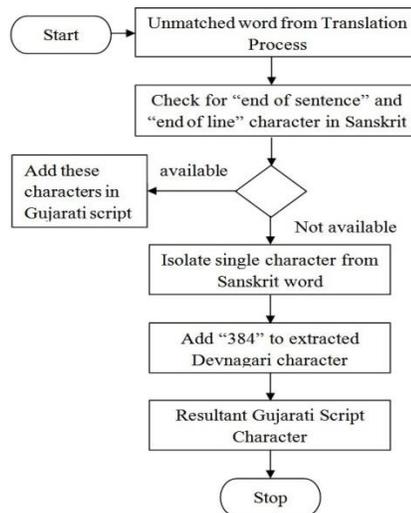

Fig. 9.   Transliteration Phase.

V. Results and Findings

Automatic evaluations are significantly more objective because they cover a limited element of the attributes to be examined, whereas human evaluations are too subjective. As a result, it's impossible to compare machine and human results. For morphologically complex language pairs, evaluation by human is considered appropriate, albeit arduous, resource-intensive, and time-intensive task. Despite the fact that BLEU is inappropriate for language with rich morphological characteristics and does not even handle word synonym factor and inflections. The suggested implementational framework is evaluated using the Bi-Lingual Evaluation Understudy (BLEU). However, the general acceptance of BLEU in the MT community is the rationale for its use in evaluation. BLEU was curated and designed by Papinene et al. at IBM [22]. Pn is a modified n-gram precision used by BLEU. Because the BLEU approach is based solely on Precision, it does not use Recall. However, it compensates for recollection by including a Brevity- Penalty feature for short sentences that are translated. The formula can be found below.

$$BLEU = BP * \exp[\sum_{n=1}^{N}\{(\frac{1}{n}) * \log(Pn)\}]$$

Here,

N = Maximum n-gram (n=1 to N) BP = Brevity Penalty

Here 1000 sentences from varied grammatical categories were chosen to test the system [21], the implemented algorithm received a BLEU score of 58.04. The ALPAC scale was used to assess the same set of sentences manually. The Automated Language Processing Committee's (ALPAC) Fidelity and Intelligibility measure is a two-scale metric [23]. The identical set of sentences used in BLEU were manually assessed using ALPAC's Intelligibility and Fidelity scale by ten language specialists. The Intelligibility score was 69.16, while the Fidelity score was 68.11.

VI. Conclusion and Future Work

To implement a rule-based system is always a challenge due to complexity of rules as well as the number of rules. This is particularly true for morphologically rich languages like Gujarati and Sanskrit. The challenge is to cover each and every grammatical element. Through the proposed implementation, the robust results are obtained due to inclusion of each grammatical feature in details. The unique MT framework on Sanskrit to Gujarati received satisfactory results utilizing the ALPAC's manual scale on intelligibility and fidelity parameter. It also received good results on the BLEU scale. As the architectural implementation may improve the results by covering a larger range of dictionary words and accounting for any grammatical language exceptions, in future we will consider these additional elements as well. Also on availability of huge bilingual corpus in future, machine and deep learning frameworks can be implemented to make the system more accurate and generic.

References

[1] Poibeau T. (2017), "Machine Translation", Book – The MIT Press, Massachusetts Institute of Technology, Cambridge, ISBN: 978-0-262-53421-5, Vol-1, PP 296.

[2] Siddiqui T., and Tiwary U.S. (2008), "Natural Language Processing and Information Retrieval", Oxford University press, ISBN- 0195692322.

[3] Cancedda N., Dymetman M., Foster G., and Goutte C. (2008), "A Statistical Machine Translation Primer". doi: 10.7551/mitpress/9780262072977.003.0001.

[4] Raulji J., and Saini J.R. (2016), "Sanskrit Machine Translation Systems: A Comparative Analysis", International Journal of Computer Applications, 136(1):1-4. doi: 10.5120/ijca2016908290.

[5] Rathod S., and Sondur S. (2012), "English to Sanskrit Translator and Synthesizer (ETSTS)", International Journal of Emerging Technology and Advanced Engineering, 2(12):379-383. Available online: https://ijetae.com/files/Volume2Issue12/IJETAE_1212_68.pdf.

[6] Bahadur P., Jain A.K., and Chauhan D.S. (2012), "Etrans – A complete framework for English to Sanskrit Machine Translation", International Journal of Advanced Computer Science and Applications, Special Issue from ICWETT-2012, 4:52-59. doi: 10.14569/SpecialIssue.2012.020107.

[7] Subramaniam A. (2005), "Sanskrit to English Translator", Language in India, 5(1). Available online: http://www.languageinindia.com/jan2005/aparnasans kritdissertation1.html.

[8] Mishra V., and Mishra R.B. (2009), "Divergence patterns between English and Sanskrit Machine Translation", INFOCOMP Journal of Computer Science, 8(3):112-138.

[9] Mishra V., and Mishra R.B. (2010), "ANN and Rule Based Model for English to Sanskrit Machine Translation", INFOCOMP Journal of Computer Science, 9(1):80-89.

[10] "JNU Sanskrit Hindi MT" (2022), Available online: http://sanskrit.jnu.ac.in/shmt/index.jsp.

[11] Goyal P., and Sinha R.M.K. (2009), "A Study towards Design of an English to Sanskrit Machine Translation System", proc. of Sanskrit Computational Linguistics: First and Second International Symposia, ACM, pp.287-305. doi: https://doi.org/10.1007/978-3-642- 00155-0_13.

[12] Sreedeepa H.S., and Idicula S.M. (2017), "Interlingua based Sanskrit-English Machine Translation", proc. of International Conference on Circuits Power and Computing Technologies (ICCPCT-2017), IEEE. doi: 10.1109/ICCPCT.2017.8074251, Vol-16, Issue-9.

[13] Gupta V.K., Tapaswi N., and Jain S. (2013), "Knowledge representation of grammatical constructs of Sanskrit language using rule based Sanskrit Language to English Language machine translation", proc. of International Conference on Advances in Technology and Engineering (ICATE-2013), IEEE. doi: 10.1109/ICADTE.2013.6524744.

[14] Kale M.R. (2016), "A Higher Sanskrit Grammar", Motilal Banarsidass Publishers Pvt. Ltd. 11th Reprint Edition.

[15] Patel P. (2013), "Sanskrit Vyakaran Prabodh", AB Shah Publications, Saraswati Pustak Bhandar, Ahmedabad, Edition.

[16] Phadke G. (1852), "The Principles of Gujarati Grammar", digitized by Google. Available online: https://books.google.co.in/books?id=E0IAAAAQAAJ.

[17] Edalji S. (1867), "A Grammar of Gujarati Language", digitized by Google.

[18] Patel D. (2015), "Gujarati Vyakaran Ane Lekhan", Gujarati Sahitya Prakashan, Ahmedabad, First Edition.

[19] Raulji J., and Saini J.R. (2019), "Morphological Analyzer for Sanskrit Language", proc. of International Conference on The Journey of Indian Languages: Perspectives on Culture and Society, 3:198-209.

[20] Raulji J., and Saini J.R. (2017), "Generating Stopword List for Sanskrit Language", proc. of 7th IEEE International Advance Computing Conference (IACC- 2017), pp. 799-802. doi:

10.1109/IACC.2017.0164.

[21] Automatic BLEU MT Evaluation, Available online: https://www.letsmt.eu/Bleu.aspx, Accessed since 2017.

[22] Papineni K., Roukos S., Ward T., and Zhu W. (2002), "BLEU: a Method for Automatic Evaluation of Machine Translation", proc. of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia, pp. 311-318. Available online: https://aclanthology.org/P02- 1040.pdf.

[23] Automatic Language Processing Advisory Committee (ALPAC) (1966), "Language and Machines – Computer in Translation and Linguistics", report by ALPAC, National Academy of Sciences National Research Council, Washington D.C.

[24] Singh M., Kumar R., and Chana I. (2020), "Hybrid machine translation system using deep learning", ASM Science Journal, 13(2):31–45. Available online: https://www.akademisains.gov.my/asmsj/?mdocs- file=4679.

[25] Akhand M.A.H., Roy A., Dhar A.C., and Kamal M.A.S. (2021), "Recent Progress, Emerging Techniques, and Future Research Prospects of Bangla Machine Translation: A Systematic Review", International Journal of Advanced Computer Science and Applications, 12(9):291-307. doi: 10.14569/IJACSA.2021.0120933.

[26] Sabtan Y.M.N.S., Hussein M.S.M., Ethelb H., and Omar A. (2021), "An Evaluation of the Accuracy of the Machine Translation Systems of Social Media Language", International Journal of Advanced Computer Science and Applications, 12(7):406-415. doi: 10.14569/IJACSA.2021.0120746.

[27] Ehab R., Gadallah M., and Amer E. (2019), "English- Arabic Hybrid Machine Translation System using EBMT and Translation Memory", International Journal of Advanced Computer Science and Applications,10(1):195-203. doi: 10.14569/IJACSA.2019.0100126.

[28] Pudaruth S., Sookun L., and Ruchpaul A.K. (2013), "English to Creole and Creole to English Rule Based Machine Translation System", International Journal of Advanced Computer Science and Applications, 10(1):25-29. doi: 10.14569/IJACSA.2013.040805.

[29] Singh A.K., Dave S., Mehta S., and Prathosh A.P. (2022), "A data driven approach for Sanskrit Derivative Nouns analysis", proc. of ACM CODS- COMAD-2022 5th Joint International Conference on Data Science & Management of Data, pp. 288–289. doi: 10.1145/3493700.3493745.

[30] Li J., and Girrbach L. (2022), "Word Segmentation and Morphological Parsing for Sanskrit", Computation and Language arxiv, Cornell University, doi: 10.48550/arXiv.2201.12833.

[31] Biltoo A.K. (2022), "First Steps Towards Sanskrit: Language, Linguistics and Culture", Routledge, Taylor and Francis, First Edition. doi: 10.4324/9780429325434.

[32] Sandhan J., Behera L., and Goyal P. (2022), "Systematic Investigation of Strategies Tailored for Low-Resource Settings for Sanskrit Dependency Parsing", Computation and Language arxiv, Cornell University, doi: 10.48550/arXiv.2201.11374.

[33] Raulji J.K., and Saini J.R. (2019), "Sanskrit Lemmatizer for Improvisation of Morphological Analyzer", Journal of Statistics & Management Systems,22(4):613-625. doi: 10.1080/09720510.2019.1609186.

[34] Raulji J.K., and Saini J.R. (2020), "Sanskrit-Gujarati Constituency Mapper for Machine Translation System", proc. of IEEE Bombay Section Signature Conference (IBSSC-2019), pp.1-8. doi: 10.1109/IBSSC47189.2019.8972989.

[35] Tailor C., and Patel B. (2020), "Chunker for Gujarati Language Using Hybrid Approach", proc. of Rising Threats in Expert Applications and Solutions, Advances in Intelligent Systems and Computing, 1187:77-84. doi: 10.1007/978-981-15-6014-9_10.

[36] Patel C.D., and Patel J.M. (2021), "Influence of GUJarati STEmmeR in Supervised Learning of Web Page Categorization", International Journal of Intelligent Systems and Applications, 13(3):23-34. doi: 10.5815/ijisa.2021.03.03.

[37] Chauhan A., and Shah A. (2021), "Improving Semantic Coherence of Gujarati Text Topic Model Using Inflectional Forms Reduction and Single-letter Words Removal", ACM Transactions on Asian and Low-Resource Language Information Processing, 20(1):1-18. doi: 10.1145/3447760.

[38] Shah P., and Swaminarayan P. (2021), "Lexicon-based sentiment analysis on movie review in the Gujarati language", International Journal of Information Technology, Communications and Convergence, 4(1):63-72. doi: 10.1504/IJITCC.2021.119111.

[39] Raval D., Pathak V., Patel M., and Bhatt B. (2022), "Improving Deep Learning based Automatic Speech Recognition for Gujarati", ACM Transactions on Asian and Low-Resource Language Information Processing, 21(3):1–18. doi: 10.1145/3483446.

[40] Doshi P.D., and Vanjara P.A. (2021), "A Comprehensive Survey on Handwritten Gujarati Character and Its Modifier Recognition Methods", proc. of Information and Communication Technology for Competitive Strategies (ICTCS-2020), 191:841- 850. doi: 10.1007/978-981-16-0739-4_79.

[41] Patel H., Patel B., and Lad K. (2021), "Jodani: A spell checking and suggesting tool for Gujarati language", proc. of 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence- 2021).pp.94-99. doi: 10.1109/Confluence51648.2021.9377072.

[42] Raulji J.K., and Saini J.R. (2020), "Sanskrit Stopword Analysis through Morphological Analyzer and its Gujarati Equivalent for MT System", proc. of the International Conference on ICT for Sustainable Development (ICT4SD-2019), Panaji, India. Vol. 93, pp. 427-433. doi: 10.1007/978-981-15-0630-7_42.

[43] Saini J.R., and Raulji J.K. (2020), "Peer Analysis of "Sanguj" with Other Sanskrit Morphological Analyzers", proc. of 2nd International Conference on Computing Analytics and Networking (ICCAN-2019), Bhubaneshwar, India. Vol. 1119, pp. 65-73. doi: 10.1007/978-981-15-2414-1_7.