



Review Paper


A comprehensive survey on Indian regional language processing

B. S. Harish¹  · R. Kasturi Rangan¹ 

Received: 24 December 2019 / Accepted: 29 May 2020 / Published online: 12 June 2020

© Springer Nature Switzerland AG 2020

Abstract

In recent information explosion, contents in internet are multilingual and majority will be in the form of natural languages. Processing of these natural languages for various language processing tasks is challenging. The Indian regional languages are considered to be low resourced when compared to other languages. In this survey, the various approaches and techniques contributed by the researchers for Indian regional language processing are reviewed. The tasks like machine translation, Named Entity Recognition, Sentiment Analysis and Parts-Of-Speech tagging are reviewed with respect to Rule, Statistical and Neural based approaches. The challenges which motivate to solve language processing problems are presented. The sources of dataset for the Indian regional languages are described. The future scope and essential requirements to enhance the processing of Indian regional languages for various language processing tasks are discussed. 

Keywords Language processing · Machine translation · Named entity recognition · POS tagging

1 Introduction

Any language that has evolved naturally in humans through its usage over the time is called natural language. People exchange their knowledge, emotions and feelings with others through the means of natural language. There are different native languages existing in various parts of the world, each with its own alphabet, signs and grammar. If there is a nation where old and morphologically rich varieties of regional languages exist that is India [57]. It is comparatively easy for computers to process the data represented in English language through standard ASCII codes than other natural languages. However, building the machines capability of understanding other natural languages is arduous and is carried out using various techniques. There are many research works and applications like (1) Chatbot (2) Text-to-speech conversion (3) Language Identification (4) Hands-free computing (5) Spell-check (6) Summarizing-electronic medical records (7) Sentiment Analysis and so on, developed to handle these natural languages for real time needs. In this paper,

various methods used to develop the aforementioned applications; especially on Indian Regional Languages (IRL) are presented.

Nowadays, the internet is no more monolingual; contents of the other regional languages are growing rapidly. According to the 2001 census, there are approximately 1000 documented languages and dialects in India. Much research is being carried out to facilitate users to work and interact with computers in their own regional natural languages [3]. Google offers searching in 13 languages and provides transliteration in Indian Regional languages (IRL) like Kannada, Hindi, Bengali, Tamil, Telugu, Malayalam, Marathi, Punjabi, and Gujarati [51]. The major concentrated tasks on IRL are Machine Translation (MT), Sentiment Analysis (SA), Parts-Of-Speech (POS) Tagging and Named Entity Recognition (NER). Machine translation is inter-lingual communication where machines translate source language to the target language by preserving its meaning [75]. Sentiment analysis is identification of opinions expressed and orientation of thoughts in a piece of text [47]. POS Tagging is a process in which each word in

✉ B. S. Harish, bsharish@jssstuniv.in; R. Kasturi Rangan, rkrangan3@gmail.com | ¹Department of Information Science and Engineering, JSS Science and Technology University, Mysuru, Karnataka State, India.



a sentence is labeled with a tag indicating its appropriate part of speech [15]. Named Entity Recognition identifies the proper names in the structured or unstructured documents and then classifies the names into sets of predefined categories of interest. Majorly, machine learning algorithms and natural language processing techniques are used to develop applications for IRL. Language processing techniques are widely and deeply investigated for English. However, not much work has been reported for IRL due to the richness in morphology and complexity in structure. The generic model for the language processing is as shown in Fig. 1.

1.1 Generic block diagram

The generic model for language processing consists of various stages viz., machine transliteration, preprocessing, lexical and morphological analysis, POS tagging, feature extraction and evaluation. The raw text block in the diagram represents the natural language which is in unstructured form. The contributions of aforementioned techniques for success of the language processing tasks are as follows:

1.1.1 Tokenization

In natural language processing applications, the raw text initially undergoes a process called tokenization. In this

process, the given text is tokenized into the lexical units, which are the most basic units. After tokenization, each lexical unit is termed as token. Tokenization can be at sentence level or word level, depending on the category of the problem [91]. Hence, there are 3 kinds of tokenization - a) sentence level tokenization b) word level tokenization and c) n-gram tokenization. Sentence level tokenization deals with the challenges like sentence ending detection and sentence boundary ambiguity. In word level tokenization, words are the lexical units, hence the whole document is tokenized to the set of words. The word level of tokenization is used in various language processing and text processing applications. The n-gram tokenization is a token of n-words where 'n' indicates the number of words taken together for a lexical unit. If 'n=1' then lexical unit is called as unigram, similarly if 'n=2' lexical unit is bigram and trigram if 'n' value is '3'. During n-gram tokenization (where $n \geq 2$), to satisfy the n-words in the tokens there will be overlapping of terms in the tokens. Figure 2 presents all the 3 ways of tokenization for some set of sentences in Kannada which is one of the Indian Regional Languages.

1.1.2 Machine transliteration

In natural language processing, machine transliteration plays a vital role in applications like cross-language machine translation, named entity recognition,

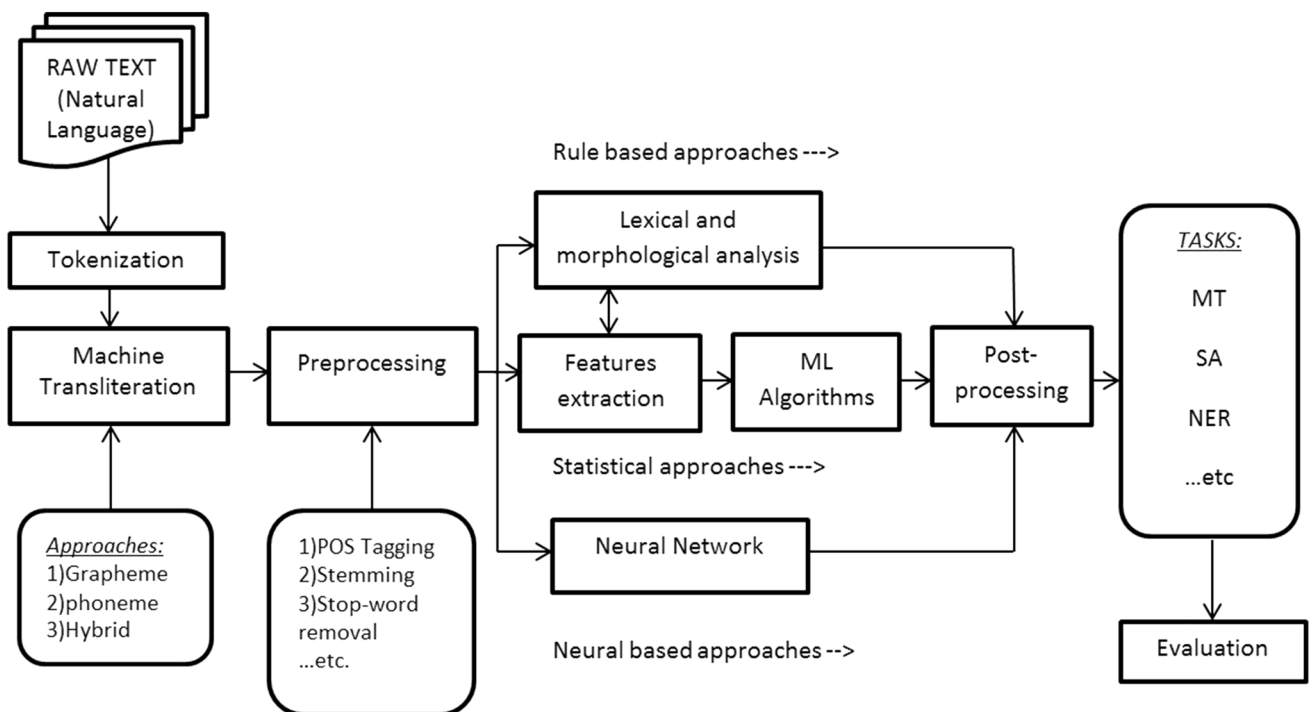
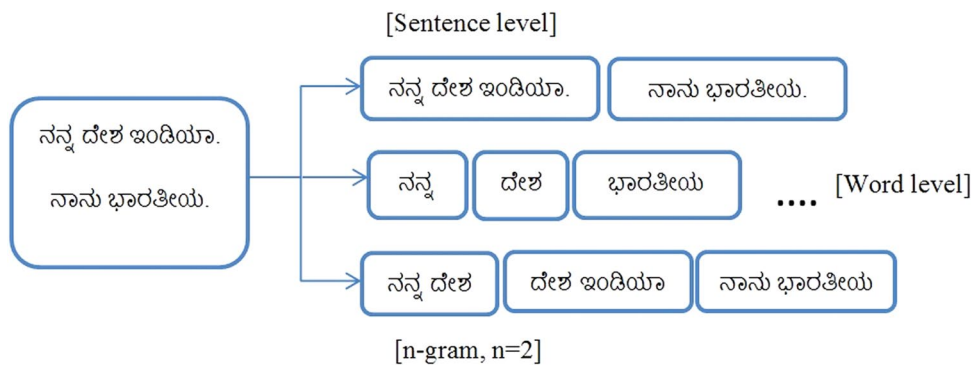


Fig. 1 Generic model for language processing

Fig. 2 An example of Tokenization in Kannada Language



Language	Word	Pronunciation
English	anybody	ə n i : b ɒ d i :
Bengali	ভালবাসা	Bhālabāsā
Kannada	ಭಾರತ	Bhārata
Telugu	ప్రేమ	Prēma

Fig. 3 Examples of pronunciation dictionary of few languages

information retrieval, etc. Transliteration is a process of converting a word or character from the source languages alphabetical system to the target languages alphabetical system, without losing the phonetics of the source languages word or character. Before transliteration, words are divided into syllabic units using Unicode and character encoding standards. Then each of the syllabic units of a word gets converted to target language [50]. For example:

Hindi	English
/अ/	/a/
/आ/	/'ā' or 'A'/

There are 3 main types of transliteration; grapheme, phoneme and hybrid [8]. The grapheme based transliteration model directly transliterates the source language word to the target language grapheme without phonetic knowledge. The phoneme based transliteration model uses the phonetics of the source language word to transliterate to target language grapheme. This model conserves the tone of the word or character and brings out proper transliteration of target language graphemes. Examples of phonetics dictionary for some words of different languages are presented in Fig. 3 using International Phonetic Alphabet (IPA) [26]. The hybrid transliteration model uses

both the source language grapheme and phoneme to produce the target language grapheme.

1.1.3 Preprocessing techniques

Once the raw text of natural language is tokenized and transliterated, some of the preprocessing techniques are used in enhancing the efficiency of the applications, as per the requirement. Some of the major techniques are as follows:

- (a) **Stemming** Stemming normalizes the given word into its root/stem by cleaving the suffixes and prefixes of the word. Root word is modified to express different grammatical categories (tense, voice, person, gender, etc.) in a sentence, which is called as inflection of the language. However, the obtained root word may not be a valid word in the language. There are some stemming techniques developed for IRL based on longest-matched method, n-gram method, brute-force method, etc. [14].

Example: "Plays", "Playing", "Played" leads to root word "Play"
 "Troubling", "Troubles", "Troubled" leads to root word "Trouble"
 "ಮರಗಲು", "ಮರಗಲಿದೆ", "ಮರಗಲಿಗಿ" leads to root word "ಮರ"

- (b) **Stop-word removal** There are some words which frequently occur in documents yet convey no additional meaning. Hence, removal of these non-informative words eases language processing tasks. There are several techniques to remove stop-words like dictionary based stop-words removal, DFA [45] (Deterministic Finite Automata) based stop-word removal, etc.
- (c) **Lemmatization** It is similar to the stemming technique but the word is reduced to an acceptable form in the language after the removal of suffixes and prefixes. The reduced word is called "Lemma"; it is valid

and accepted by the language. For example, “runs”, “running”, “ran” are all the different forms of the root word called “run” in English language; thus “run” is the lemma for all those formerly mentioned forms of it. Researchers are also working on lemmatization techniques for IRL, which can help in building various applications on multilingual platforms [67].

- (d) *POS Tagging* As the name indicates, it refers to the process of tagging words in a sentence with parts of speech like Noun, Pronoun, Verb, Adjective, etc. This process will be different for different languages owing to differences in grammatical structure. POS tagging helps in the natural language processing applications like cross-lingual machine translation, Documents-Tagging using named entity recognition, sentiment analysis, etc. However, understanding the grammatical structure of a sentence for automatic POS tagging is a challenging task. In India, many researchers are working towards proposing POS tagging for various regional languages [15]. It helps in development of various applications for native languages.
- (e) *Unicode Normalization* Unicode is the Universal character encoding standard, which represents the text characters by a unique hexadecimal value. This representation is used for information processing. Some sequence of Unicode characters are equivalent to single abstract Unicode character, this multiple representations for an abstract character leads to complication. To eliminate these non-essential differences Unicode normalization is performed during preprocessing. Unicode normalization transforms equivalent sequence of characters into the same representation. For example: The string “fi” can be represented either by the characters “f” and “i” ($U + 0066, U + 0069$) or by the ligature “fi” ($U + FB01$). Even in Indian regional languages especially in Hindi language, Nuktha based characters forms multiple representations as shown in below example.

फ़ (U+0958) = फ (U+0915) + ी (U+093C)

1.1.4 Statistical based approaches

After preprocessing, the model is trained for language processing using either the machine learning/statistical approaches or rule based language processing approaches. In machine learning approach, feature extraction method is used to extract features from the preprocessed data. Later, these features are used for the purpose

of training learning algorithms. These machine learning algorithms constitute statistical based models.

For example, statistical approach uses probability distribution function to choose the best translation in machine translation task of language processing. During this translation, Multi/bi-lingual corpus is used [40]. In another approach called Example Based Machine Translation (EBMT), corpus of the translated examples is used to train the model. The test input is matched with the corpus example and matched words of test input sentence are recombined later in an analogical manner for proper translation [85]. Similarly, there are different machine learning based solutions for other applications which are mentioned earlier.

1.1.5 Rule based approaches

This approach existed before the statistical based models were created for language processing. The lexical and morphological analyses using techniques like regular expressions, Suffix stripping and so on are applied after preprocessing. In rule based natural language processing approach, the set of rules and patterns guide the machine to translate the language. E.g.: English has the language structure of SVO (Subject, Verb, Object) while Hindi has SOV (Subject, Object, Verb). Researchers believe natural language translation is incomplete without the support of some external knowledge like reasoning and basic knowledge of the language. Hence, rule based approach uses thesaurus and data sources like Wordnet.

For example, Rule Based Machine Translation (RBMT) technique translates source language to target language using various set of rules and bilingual dictionary in machine translation task of language processing. Similarly, other techniques like Knowledge Based Machine Translation (KBMT), Principle Based Machine Translation (PBMT) make use of parsers for the lexical, phrasal and grammatical information of the language [42].

1.1.6 Neural based approaches

Other than the rule and statistical based approach, researchers also worked using neural based approaches for language processing tasks to find better results. In this approach, the input data is processed through the artificial neurons in the architecture. The circuit of neurons forms a neural network. For language processing tasks, neural based approaches provide better results for various complex situations such as training huge datasets, better and fast learning, owing to the presence of features like uniformity, computation power, learning ability and generalization ability. For instance, machine translation task aims to find the best similar target language sentence for

a source in language processing task. From the probabilistic point of view, it is the maximum of $P(y/x)$ where y is target and x is source language. But in neural based approach, it tries to build an end to end large neural network model in order to complete the same task. The core purpose is to encode a variable length sequence of words into a fixed length vector, which is the summary of the whole source sentence. It is further translated to the target language using decoder. This encoder-decoder model is trained to attain best conditional probability $P(y/x)$ in neural based translation [24, 60].

1.1.7 Post-processing

In this phase, the results generated by the techniques of language processing models are made much more refined or efficient. The results from the models are checked for spelling corrections, sentence arrangements, grammatical errors, missed translations, etc. [42].

1.1.8 Evaluation

The results of the applications are measured and evaluated to know the efficiency of the models using statistical measurements like Accuracy, Precision, Recall and F-Score (Harmonic mean of Precision and Recall). BLEU (BiLingual Evaluation Understudy) scores are calculated for machine translation tasks to check the quality of translations [59]. Other measures are also used. UNK (Unknown Word) count is used for measuring the Out-Of-Vocabulary (OOV) words in translation task, while WER (Word Error Rate) metric is used to analyze human translated output and machine translated output.

2 Challenges

There are several challenges faced in all the stages of the language processing tasks because of differences in grammar and phonetics. The challenges faced in Indian regional language processing are as follows:

- (1) Tokenization of the text. Some of the regional languages don't have common delimiters like white space or punctuations.
E.g.: Urdu language.

I Love You	ہوں کرتا پیار سے تم میں
------------	-------------------------

- (2) Language structure i.e. order of the words in the sentences will differ from one language to another [16].

E.g.: Subject Verb Object (SVO) (English), Subject Object Verb (SOV) (Kannada).



- (3) Ambiguity in translation or transliteration of regional language words.

E.g.: In English-Hindi translation, the word mount translates but Everest remains same. In English-Kannada both the words are just transliterated.

Mount Everest	एवरेस्ट पर्वत (Hindi)
	ಮೌಂಟ್ ಎವರೆಸ್ಟ್ (Kannada)

- (4) Some languages support Homograph words whose meaning changes with context [56].

E.g.: In Heart Attack and Dog attacks cat, attack is the homograph word.

- (5) Some languages have multiple scripts.
E.g.: Punjabi (Gurmukhi, Shahmukhi).
- (6) Grammatical variations between languages lead to ambiguity.
- (7) Judging of speakers intention is difficult. Meanings of sentences or words vary with the speakers intention (like sarcasm, sentiment, metaphor, etc.).
- (8) Code-Mixed language processing is challenging as user uses multiple languages in a sentence or an utterance.

E.g.: User tweet : "listening to Bombae Haelutaitae from Rajakumara"

3 Motivation

India is a multilingual country. Indian constitution lists 22 languages, referred to as scheduled languages. These languages are given status, recognition and official encouragement. Of the entire population, barely 10% Indians use English to transact and most prefer regional languages, which have evolved over centuries. As there is diversity in languages, language processing applications are a boon to the people for their day-to-day transactions. However, understanding and generation of these natural languages i.e. processing of these natural languages by machine is complex. Therefore, we review the work carried out by researchers on various techniques developed for processing Indian Regional Languages.

4 Review in detail

George University and IBM jointly developed the first machine translation application in 1954 for translating more than sixty Russian sentences into English. This was the first milestone achieved in the field of natural language processing. Real progress was much slower in NLP. Until 1980, NLP techniques were complex and based on hand written rules. Post the introduction of Moores law by Gordon Moore, the former CEO of Intel, the computational power of the system increased and paved way for the development of statistical models based machine learning algorithms, which led to a revolution in NLP.

4.1 Machine transliteration

Early in 1994, Arbabi worked on Arabic-English language transliteration using phoneme based model [10]. Later in 2008-2010, researchers developed statistical transliteration techniques which are language independent. Many works have been proposed with regard to Indian regional languages too. In [9], Antony et al., addressed the problem of transliterating English to Kannada language using SVM kernel model, which trained over 40k names of Indian towns. It is based on sequence labeling method. The transliteration module uses an intermediate code, which is designed for preserving the phonetic properties. Authors also compared their results with the Google Indic transliteration system and found better results. The process of converting the words to pronunciation is called as grapheme-to-phoneme (g2p). The statistical grapheme-to-phoneme (g2p) transliteration learning models are trained on language specific pronunciation dictionaries which are expensive, time consuming and require the intervention of language experts. To address these issues, [26] worked on grapheme to phoneme (g2p) transliteration model for low resource languages using Phoible [53] phonological inventory data (having 37 phonological features such as nasal, consonantal, sonorant, etc.). Low resource language words are converted to their pronunciation using phonological information of high resource language words, which are similar in linguistic and phonological information. In [29], Dhore et al., focused on direct phonetic based transliteration approach for Hindi and Marathi to English, without training bilingual database. They used hybrid stress analysis approach for deletion of schwa, which refers to the vowel sounds presented in many unaccented syllables of words and are removed after transliteration. Ekbal et al., [36] made substantial contribution to develop transliteration systems for Indian languages to English and especially for Bengali-English transliteration. They proposed modified joint source-channel model, which is based on regular

and non-probabilistic expression. It uses linguistic knowledge to transliterate person names from Bengali-English. In [50], Lakshmi et al., worked on Back-Transliteration of Kannada language. The Romanized Kannada words are transliterated back to Kannada script. Bilingual corpus (around 1 lakh words) and Bidirectional Long Short-Term Memory (BLSTM) are used in this Back-Transliteration, which obtained good results.

4.2 Preprocessing techniques

4.2.1 Stemming

It is a process of reducing morphologically variant terms into a single term, without performing complete morphological analysis. Ramanathan et al., [71] presented their light weight stemmer on Indian regional language Hindi. This work is based on stripping of word endings by longest matching suffix of words, using manually created suffix list consisting of 65 suffixes. Pandey et al., [58] proposed an improvised unsupervised stemmer for Hindi, which is a probabilistic approach to achieve better stemming. They used EMILLE corpus and WordNet of Hindi for training and testing, respectively. This approach showed better results than light weight stemmers. Ramachandran et al., [70] applied longest match suffix removal technique for the Tamil language stemmer. Saharia et al., [78] worked on Assamese language stemmer based on suffix removal technique. For Gujarati, [61] presented a light weight stemmer which is based on hybrid technique of both unsupervised morphological parsing method [41] and rule based method (manual listing of handcrafted suffixes). Similarly [5, 86] worked on Gujarati language stemmer based on hybrid approach. In [20, 52] researchers presented Bengali stemmers based on longest suffix matching technique, distance based statistical technique and unsupervised morphological analysis technique.

4.2.2 Lemmatization

The aim of lemmatization is to obtain meaningful root word by removing unnecessary morphemes. English and other European languages are not highly inflected when compared to Indian languages, which have more stemmers and lemmatizers [63]. Compared with other Indian regional languages, Hindi words have finite set of inflections morphologically [6]. Hence, [63] worked on optimization of lemmatization technique for Hindi words using rule based and knowledge based approach. Here, knowledge refers to the storage of grammatical features and in lemmatization, it refers to the storage of root words. [67] worked on one of the south Indian regional languages Kannada which consists of more inflectional words than

Hindi. They used Kannada language dictionary for the lemmatization of words under rule based approach.

4.2.3 Parts of speech (POS) tagging

In language understanding, POS tagging plays a vital role. It helps in achieving language processing tasks more efficiently. POS tagging is a disambiguation task and the goal of tagging is to find the exact role of a word in the sentence.

E.g.:

In English sentence "I am Xyz", 'I'-Pronoun (PN), 'am'-Verb (V), 'Xyz'-Noun (N).

In Kannada sentence "ನಾನು ರಾಮ್.", 'ನಾನು'-Pronoun (PN), 'ರಾಮ್'-Noun (N).

where PN, V, N are called tagsets, representing the grammatical identities of words.

In Hindi, Singh et al., [84] presented POS tagger with detailed morphosyntactic analysis, skillful handling of suffixes and decision tree based learning algorithm. Dalal et al., [18] used maximum entropy markov model, which is statistical based and considers multiple features simultaneously such as context based features, word features, dictionary features and corpus-based features to predict the tag for a word. Avinesh et al., [68] used conditional random field and transformation based learning statistical methods for POS tagging of Hindi, Telugu and Bengali. [83] presented POS tagger for Hindi by using Hidden Markov Model (HMM). Working on local word grouping for Hindi, Ray et al., [73] presented POS tagging algorithm based on morphological analysis and lexical rules. In Bengali, [19] built POS tagger based on HMM and Maximum Entropy (ME) methods. They also found that accuracy increased with addition of Morphological Analysis (MA). Ekbal et al., [37] worked on Bengali POS taggers based on Conditional Random Field (CRF) and Support Vector Machine (SVM). They found the performance of SVM to be better [32]. For the south Indian language Tamil, [28] proposed statistical based Support Vector Machine method for POS tagging. Similarly [81] presented a POS tagger for Tamil which is built on combination of both rule based morphological analysis and statistical based methods. Kannada is also a south Indian regional language where Antony et al., [7] worked on POS tagger, based on lexicon dictionary and support vector machine method. Later, Shambavi et al., [15] presented POS tagger built on Hidden Markov Model and Conditional Random Fields methods. In social media, users with multilingual knowledge interact using words from multiple languages in a sentence or an utterance; this is called Code-mixing. [44] worked on English-Hindi social media code-mixed text and experimented POS tagging of these corpora using four machine learning algorithms

(Conditional Random Fields, Sequential Minimal Optimization, Nave Bayes, and Random Forests).

As POS tagging is needed for many of the language processing tasks, researchers used multilayer perceptron neural network for more efficiency. [60] presented neural based POS tagger for Hindi language and claimed it to be the first work on neural based Hindi POS tagger. Comparatively, neural method works better than CRF and HMM statistical methods. Todi et al., [87] worked on Unknown or Out-of-vocabulary words, which is the major challenge in POS tagging task. This challenge is addressed by character embedding and word embedding solutions with simple RNN, LSTM and biLSTM methods. Narayan et al., [54] presented neural based solution for the disambiguation of corpus problem in Hindi language. All these POS taggers are presented in Table 1.

4.3 Approaches for language processing tasks

4.3.1 Rule based approaches [25, 65]

If the language processing tasks are achieved based on lexical rules, morphological analysis and linguistic knowledge after preprocessing, then this approach is termed as rule based solution/approach. The language processing tasks are handled by the decisions taken by the lexical rules, which should be specific and clear. Each language has its own own linguistic rules and all these are to be taken into consideration for achieving the language processing tasks efficiently. Machine Translation (MT) is one of the most difficult and major tasks in language processing. Rule based MT are of three types; the first being Dictionary based or Direct based where multilingual dictionaries are used for translation and which is easy to implement. [42] presented Hindi to Punjabi MT based direct rule based method. Dictionary based English to Kannada/Telugu translation tool is proposed by [75]. Next is Transfer based translation, which concentrates on the grammatical structure of source and target languages. Lastly Interlingual translation, in which source language is translated to intermediate representation called Interlingua (E.g.: Universal Networking Language (UNL)) [46], from which target language is generated. This representation is independent of languages. Dave et al. [25] worked on English to Hindi MT using Interlingua. Rule based sentence simplification technique is proposed by [65] for English to Tamil translation task.

In language processing, Named Entity Recognition (NER) refers to the process of identifying the proper nouns in the text and classifying them into named entity classes like person, location, date, organization, numbers etc. and is a major task. The linguistic handcrafted rules are used in rule based NER. As NER is a classification task of given

Table 1 List of POS taggers on Indian Regional languages

Citations	Rule based	Statistical based	Neural based	Language	Methods	Results
Singh et al. [84]	✓	–	–	Hindi	Decision Tree (CN2)	93.45%
Dalal et al. [18]	–	✓	–	Hindi	Maximum Entropy Markov Model	89.35%
Avinesh et al. [68]	–	✓	–	Hindi, Telugu, & Bengali	Conditional Random Field, Transformation based learning	77.37% (Telugu) 78.66% (Hindi) 76.08% (Bengali)
Shrivastava et al. [83]	–	✓	–	Hindi	Hidden Markov Models	93.12%
Dandapat et al. [19]	✓	✓	–	Bengali	Hidden Markov Model, Maximum Entropy	87.95% (HMM) 88.41% (ME)
Ekbal et al. [37]	–	✓	–	Bengali	Conditional Random Field	90.30%
Ekbal et al. [32]	–	✓	–	Bengali	Support Vector Machine	86.84%
Selvam et al. [81]	✓	✓	–	Tamil	Morphological Analysis, Statistical Projection and Injection Technique	83.00%
Dhanalakshmi et al. [28]	–	✓	–	Tamil	Support Vector Machine	95.63%
Antony et al. [7]	–	✓	–	Kannada	Support Vector Machine	86.00%
Shambavi et al. [15]	–	✓	–	Kannada	Hidden Markov Models Conditional Random Fields	79.90% (HMM) 84.58% (CRF)
Jamatia et al. [44]	–	✓	–	Hindi, English (Code-mixed)	Conditional Random Fields, Sequential Minimal Optimization, Nave Bayes and Random Forests	64.91%
Parikh [60]	–	–	✓	Hindi (ILMT corpus)	Multi-Neuro Tagger (Neural Network)	92.19%
Todi et al. [87]	–	–	✓	Kannada	RNN, LSTM, biLSTM	92.00% (F-Score)
Narayan et al. [54]	–	–	✓	Hindi	Artificial Neural Network	91.30%

language entities into any one of the named entity classes, machine learning methods perform better than rule based methods. Hence, machine learning methods are used widely by the researchers [55]. [43] presented conditional based or rule based NER system for Punjabi language. They developed and used gazetteer lists like prefix list, suffix list, last name list and so on for proper name identification.

Most of the business decisions are based on choices of customers; thus gauging sentiment and sarcasm is crucial for proper decision making. In language processing, Sentiment Analysis (SA) is also a major task [48]. In rule based, SA dictionaries of words annotated with the word's semantic orientation or polarity are used. In Indian regional language, Balamurali et al. [12] worked on Cross-Lingual Sentiment Analysis (CLSA) using WordNets of Hindi and Marathi. CLSA is a task of analyzing sentiment where

languages are different for testing and training processes. WordNets avoid translation between test language texts while training language texts. [47] present SA system for Bengali and Hindi languages using lexicons, distributional thesaurus (DTs) and sentence level co-occurrences.

4.3.2 Performance and limitations

Though rule based approach considers the morphological analysis and linguistic knowledge, it falls short while making complex rules and processing resource deficient languages. It is also a tedious approach because it demands high linguistic acquaintance of languages and updation of rules with evolution of language. During machine translation task, especially in dictionary based, there is no consideration of structure of source text sentence beyond

morphological analysis of words (idioms and phrases, slogans). In transfer based translation, there must be compatibility between the languages. Interlingua is time consuming as it does double translations; however it is supportive of multi languages [46, 51, 72, 79]. NER for Indian regional languages is difficult as it lacks capitalization and has complex phonetics [62]. The language processing task, namely Sentiment Analysis (SA) for Indian regional languages, is difficult due to language constructs, morphological variations and grammatical differences [47]. Further, the lack of WordNets for regional languages renders an uphill task for SA. The rule based approach for SA task is applied whenever the goal is to analyze the sentiment at the document or sentence level, because this approach helps in contextual based analysis of sentiment using various rules. But for cases where contextual factors are not essential or contribute less, feature based statistical methods are preferable.

4.3.3 Statistical based approaches [4, 33, 35, 38, 39, 72, 89, 90]

If the preprocessed data are analyzed with statistical metrics to achieve the desired result in language processing tasks, it is called statistical based approach. This approach looks for statistical relations in preprocessed data (such as distance metric, probability metric, etc.). Here the features of the data guide the statistical models towards efficient results. In translation task, the document is translated on the basis of probability distribution function indicated by $P(k/e)$. The $P(k/e)$ represents the probability of translating a sentence "e" in the source language 'E' (E.g.: English) to a sentence "k" in the target language 'K' (E.g.: Kannada). The parallel corpora of languages play a vital role in statistical based language processing tasks. Unnikrishnan P et al. [89] proposed Statistical Machine Translation (SMT) system for English to Kannada and Malayalam languages, where they concentrated on aspects like reordering the sentences of source language as structure of target language sentence, root-suffix separation for both source and target words and efficient morphological information usage. [72] worked on SMT for English-Hindi translation task. The incorporation of sentence structure reordering method (as per target language) and better suffix-root separation method (of words), enhanced the efficiency of their SMT system.

Named Entity Recognition (NER) task performs better in statistical approach. [4, 33, 39, 90] worked on developing NER for regional languages like Hindi, Bengali, Kannada and Tamil using Conditional Random Field (CRF) method. The SVM statistical method is used by [35] and [31] on Hindi and Bengali languages. The Hidden Markov Model (HMM) method is used for Kannada and Bengali NER task

by [30, 38]. [34] presented NER system by hybrid of these methods in Bengali language and found better results.

Statistical based Sentiment Analysis (SA) task uses machine learning algorithms and is trained by known datasets. Rohini et al. [77] worked on SA for movie reviews in Kannada regional language using Decision tree classifier. The same reviews are translated to English and polarity is analyzed with the classifier mentioned formerly. Location based SA was carried out to identify trends during the Indian election campaign in 2014, using twitter dataset [2]. They used Nave based classifier to classify tweets into positive or negative. [80] worked on classifying Tamil, Hindi and Bengali tweets into positive, negative or neutral using sentiWordNet for features extraction and Nave Bayes classifier.

4.3.4 Performance and limitations

The major significance of the statistical based approach is that it doesn't require more linguistic acquaintances. This is a boon for languages with less resources and leads to efficient processing of language tasks. Among languages, we can find similarly structured languages and non-similarly structured languages i.e. whether the order of Subject-Verb-Object remains the same. In Machine Translation (MT) task, statistical based translation is more efficient for languages with different structures, rather than similarly structured languages. For similarly structured languages, rule based method is efficient and performs better. For the statistical based MT, good parallel-corpora of languages are required. However, dictionaries are more widely available when compared with parallel-corpora and bilingual dictionaries [72]. The main features deciding efficiency in translation are quality and coverage of the corpora and dictionaries, be it rule based or statistical. Proper probability estimation is also a difficult task as it requires sufficient training [46]. The NER task produces more efficient results with statistical methods than with rule based methods. As formerly mentioned, coverage of annotated corpora is the key factor for efficiency of statistical based methods [55]. Even though statistical method for Sentiment Analysis (SA) takes the upper hand when compared to lexicon based, it trails when it comes to the highly inflected regional language. Hence, researchers use WordNets for feature extraction and machine learning classifiers for later classification into positive, negative or neutral classes. Dependency on WordNets is also a drawback because there is lack of WordNets for regional languages [77]. [49] compared the semantic approaches (E.g.: Baseline algorithm) and machine learning approaches on web based Kannada documents for sentiment analysis and found that machine learning approach (using Weka software suite) performs better.

4.3.5 Neural based approaches

Recently, many researchers have worked on neural based solutions for language processing tasks. It is quite successful in giving better results for some language processing tasks but also gives below par results at times due to lack of resources for some regional language processing tasks. There are many artificial neural network architectures like Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), used for building learning models. The task of neural based machine translation is called Neural Machine Translation (NMT). Most of the proposed NMT belong to the encoder-decoder approach. An encoder transfers the variable source sentence into a fixed length vector, while the decoder later translates into target language sentence [11]. Revanuru et al. [76] worked on NMT for 6 Indian language pairs like Telugu-Hindi, Konkani-Hindi, Gujarati-Hindi, Punjabi-Hindi, Tamil-Hindi and Urdu-Hindi. They claim to be the first to apply NMT on Indian regional languages. The neural architecture consists of bi-directional LSTM and BLEU metric for evaluation. In comparison with Google translate, their model outperformed by a BLEU score of 29 for Punjabi-Hindi translation, 17 for Urdu-Hindi translation and 30 for Gujarati-Hindi translation for the dataset given from Indian Language Technology Proliferation and Deployment Center (TDIL-DC), C-DAC. Sentiment Analysis (SA) is a language processing task wherein emotions are studied computationally. Ravi et al. [74] worked on Hinglish (Code-mixed) text that is Romanized Hindi for sentiment classification. They claim that the combination of gain ratio based feature selection and Radial Basis Function Neural Network performs well for their dataset in sentiment classification. Similarly, [66] worked on Code-mixed Hindi-English texts at sub-word level compositions using LSTM for sentiment analysis. Akhtar et al. [1] proposed hybrid deep learning approach where features are extracted from Convolution Neural Network (CNN) and sentiments are classified later by SVM classifier. To prove the independence of method over language, Choudhary et al. [17] proposed Siamese Network Architecture, which is composed of twin Bi-directional LSTM Recurrent Neural Networks (Bi-LSTM RNN) for sentiment analysis task, which they tested on both Hindi and benchmark English datasets. They considered both resource rich languages (English and Spanish) and resource-poor languages (Hindi and Telugu) for training to overcome problems like out-of-vocabulary and spelling errors. This approach takes aid of resource rich language for sentiment analysis of resource poor languages. Bhargava et al. [13] worked on monolingual tweets of Hindi, Bengali and Tamil languages. The experimentation is on binary classification (positive/negative) of tweets using the combination of RNN, CNN and LSTM neural networks. [82] worked on code-mixed data

of Bengali and English languages for sentiment analysis. Convolutional Neural Network (CNN) is applied on code-mixed data by them. Similarly, they extended experiments on monolingual language (Telugu) using CNN network architecture.

Further, the advancement of deep learning leads to usage of various neural networks in many language processing tasks. Neural models alleviate the feature engineering problem faced in non-neural methods which depends on handcrafted features. But Neural models require large parameters for best generalization else model will be overfit. Hence neural models are not significant for low resources. Recently, in language processing due to the availability of large corpus, researchers have developed pretrained neural models which are trained on these large benchmark corpus/dataset. And it could be tested on different datasets for the similar tasks. Basically all these models are pretrained word vectors built on using large corpus. These benchmark pretrained models reduces the time from building the model from scratch. The various pretrained models used for language processing tasks are CoVe(Context Vectors), GLUE(General Language Understanding Evaluation), ELMo(Embedding from Language Models) [64], BERT(Bidirectional Encoder Representations from Transformers), etc. BERT is the recent efficient pretrained model developed by Google [27]. Few months' back BERT had been adopted by Google search and it is trained over 70 languages. Among these languages there are few major Indian regional languages too. Better pretrained models and its research experiments are yet to be done for Indian regional languages by using large language resources.

4.3.6 Performance and limitations

Neural approach evolved for providing efficient solutions to various tasks. The concept of neurons in neural approach duplicates the functions of biological neurons which have features like self-learning, fault tolerance and noise immunity. Many architectures such as LSTM, RNN, CNN have evolved in the recent past and achieved commendable efficiency in various tasks, especially in language processing. However, they fail in situations like less resource/dataset and overfitting. They also especially require sufficient hardware support for faster execution. The Neural Machine Translation (NMT) task performs better than other state-of-the-art methods but care needs to be taken while using unknown and rare words. Multitask learning and multilingual models are suggested for translations of low resource languages [11]. Sentiment analysis task for Indian languages gives better results using neural approach, yet understanding of a low resource language's words is challenging, because words are agglutinative and

often differ in meaning with usage. During preprocessing, emoticons and punctuations are usually removed but these matter a lot in analyzing the sentiment or sarcastic nature of a word/sentence in any given language (E.g.: "What!" and "What?" - Even though the word 'What' is common, meanings differ owing to different punctuation) [13]. This affects the NMT too, because punctuation changes the meaning of sentences significantly (E.g.: Hang him, not leave him (&) Hang him not, leave him). Table 2 gives insights into the discussed researches on Indian regional languages for different language processing tasks (Transliteration, Lemmatization, Machine Translation, Sentiment Analysis, Named Entity Recognition) using rule, statistical and neural based models.

Other than some focused language processing tasks, researchers worked on other tasks of Indian regional languages too. [88] explored Question classification task for Hindi and Telugu languages using neural networks, where they considered both character and word level embedding for their task, with good results. Rajan et al. [69] worked on classification of Tamil text documents using vector space model and neural networks. Among these models, their experimentation results show that both methods are competent while neural network performs slightly better. They used Tamil corpus/Dataset taken from CIIL-Mysore-India.

5 Datasets

EMILLE (Enabling Minority Language Engineering) [22] corpus has been created as a collaboration between Central Institute of Indian Languages (CIIL), Mysuru, India and EMILLE project, Lancaster University, UK. This EMILLE/CIIL corpus is available free for non-profit research works and constitutes monolingual, parallel and annotated corpora. Monolingual corpora have been constructed for 14 south Asian languages namely Assamese, Bengali, Gujarati, Hindi, Kannada, Kashmiri, Malayalam, Marathi, Oriya, Punjabi, Sinhala, Tamil, Telugu and Urdu. It includes both written and spoken data (some among former mentioned languages). Parallel corpora consists of 2,00,000 words in English and respective translated words in languages like Hindi, Punjabi, Bengali, Gujarati and Urdu. Annotated corpora are available for Hindi and Urdu languages, especially for parts-of-speech tagging. These corpora are encoded using Unicode.

IJCNLP-2008 [23] data set for Named Entity Recognition (NER) task was created during the workshop on NER for South and South East Asian languages organized by IIT, Hyderabad and contains datasets of Hindi, Bengali, Oriya, Telugu, and Urdu. Scarcity of resources in regional languages for various computational tasks was the

motivation behind the creation of these datasets, especially for NER task.

Tab-delimited Bilingual Sentence Pairs [24] datasets have been developed by Tatoeba, a non-profitable organization, by collecting sentences from various languages. They specially focused on the development of large number of linguistic datasets of various low resource language sentences and its translations. The dataset can be utilized for translating any low resource language to English. Tab key acts as delimitation between source and translated sentences. There are a minimum of 100 or more sentences and their translations in each dataset. Figure 4 gives an example of the data in the dataset.

Center for Development of Advanced Computing C-DAC [21] is an R&D organization which comes under the Ministry of Electronics and Information Technology (MeitY) of Indian government. As India is a multilingual nation, this organization developed many multilingual tools and solutions to reduce the barriers between Indian languages. All these tools and solutions are available to users for research work. It also provides Indian languages Corpora, and Dictionaries.

These are some sources of dataset aids for the exploration of new avenues in language processing tasks. As there is a scarcity of resources for many regional languages, researchers contributed their own datasets and also conducted their desired language processing tasks on these.

6 Discussion and future directions

According to research firm Common Sense Advisory, 72.1% of online customers spend their time on sites in their own language while 72.4% customers prefer to buy a product with information in their own language. People understand precisely if anybody communicates to them in their mother tongue. These are some of the reasons that it's essential to make machines understand and communicate with the user in their own language. This paper has explored various methods for language processing tasks are explored for Indian regional languages. Since Indian regional languages are morphologically rich, agglutinative and have sentences with difficult to analyze structures, less research work has been attempted in these, compared with English. There is still need for good quality dictionaries such as WordNets, Corpora for the less resourced Indian languages. Even though some good language processing systems have been developed for some Indian languages with large number of speakers, there are still many areas which are untouched such as Code-Mixed language processing, Opinion extraction and so on for many Indian regional languages. Neural based approaches are yet to be experimented in many language processing tasks.

Table 2 List of various research works on IRL

Citations	Languages	Dataset	Results In (Accuracy)	Task	Remarks on methods
Antony et al. [9]	English-Kannada	Own (40,000 Indian place names)	87.28%	Machine Transliteration	SVM based classification.
Ekbal et al. [36]	Bengali-English	Own (6000 Indian person names)	89.80% (TUAR)		Modified joint-source channel model is used.(using Transliteration units)
Paul et al. [63]	Hindi	Own (2500 words)	89.08%	Lemmatization	Rule and Knowledge base model is used.
Prathibha et al. [67]	Kannada	Own (2720 words)	85.00%		Linguistic Rule based method is used.
Dave et al. [25]	English-Hindi	Own, United Nations Charter (180 sentences)	95.00%	Machine Translation	Interlingua, UNL based method is used.
Goyal et al. [42]	Hindi-Punjabi	Own, from various sources (221 documents)	95.40%		Direct translation approach.
Poornima et al. [65]	English-Tamil	Own (200 sentences)	57.50%		Rule based using sentence simplification method.
Gupta et al. [43]	Punjabi	Own (50 Punjabi news documents)	86.25% (F-Score)	Named Entity Recognition	Rule (condition) based approach.
Balamurali et al. [12]	Hindi-Marathi	Own (user destination travel reviews 200 Hindi, 150 Marathi)	72.00% (Hindi) 84.00% (Marathi)	Sentiment Analysis	Linked WordNets of 2 languages, train data and test data are in different languages.
Kumar et al. [47]	Bengali, Hindi	Indian Tweets as part of SAIL-2015	46.25% (Hindi) 42.00% (Bengali)		Distributional thesaurus and sentence level co-occurrences are utilized.
Unnikrishnan et al. [89]	English-Kannada & Malayalam	Own (1100 sentences for both English-Kannada/Malayalam)	24.9(Malayalam)24.5(Kannada) (BLEU-Score)	Machine Translation	Statistical based translation with consideration of reordering, suffix separation are used.
Ramanathan et al. [72]	English-Hindi	Own (5000 training 400 testing sentences)	15.88(BLEU-Score)		Statistical based translation, reordering & suffix separation methods are used.
Ekbal et al. [33]	Bengali, Hindi	IJCNLP-08 (1,22,467 tokens-Bengali & 5,02,974 tokens-Hindi)	78.29%(Hindi) 81.15% (Bengali) (F-Score)	Named Entity Recognition	Conditional Random field method is used.
Ekbal et al. [39]	Bengali	Own (150k words)	90.70% (F-Score)		Conditional Random Field method is used.
Vijayakrishna et al. [90]	Tamil	Own (94k words from tourism domain)	80.44% (F-Score)		Conditional Random Field method is used.
Ekbal et al. [31]	Bengali	Own (150k words from Bengali newspaper)	91.80% (F-Score)		SVM model is used.
Ekbal et al. [38]	Bengali	Own (34 million word from newspaper)	83.79%(F-Score)		Hidden Markov Model is used.
Amarappa et al. [4]	Kannada	Own (100k words, Mixture of EMILLE, Web, Books)	85.40%(F-Score)		Conditional Random Field method is used.

Table 2 (continued)

Citations	Languages	Dataset	Results In (Accuracy)	Task	Remarks on methods
Rohini et al. [77]	Kannada	Own (100 movie reviews from kannada website)	0.79 (Kannada) 0.67 (English) (Recall)	Sentiment Analysis	Comparison between approaches like regional language SA analysis to Machine translated SA analysis
Se et al. [80]	Tamil,Hindi, Bengali	SAIL-2015	39.28% (Tamil) 55.67% (Hindi) 33.60% (Bengali)		Nave Bayes classifier based on SentiWordNet features.
Kumar et al. [49]	Kannada	Own (287 reviews from Kannada web documents)	81.20% (Nave Bayes) 89.70% (Baseline)(Precision)		Comparison between semantic and Machine Learning approaches
Revanuru et al. [76]	Telugu-Hindi, Konkani-Hindi, Gujarati-Hindi, Punjabi-Hindi, Tamil-Hindi, Urdu-Hindi	Indian Language Technology Proliferation And Deployment Center (TDIL-DC), C-DAC	14.16 24.35 35.26 45.97 7.56 22.47 (BLEU scores) (respectively)	Machine Translation	RNN, (Bi-LSTM) methods are used.
Ravi et al. [74]	Hindi-English (Cross-Lingual)	Own (300 news article, 276 Facebook comments)	86.01% (news) 84.88% (comments)	Sentiment Analysis	Radial Basis Function Network, gain ratio are used.
Prabhu et al. [66]	Hindi-English (Cross-Lingual)	Own (3879 Facebook comments, 7549 words)	69.70%		Subword-LSTM method is used.
Akhtar et al. [1]	Hindi	Own (Online product reviews, Online movie reviews) Hindi (SAIL-Twitter dataset)	62.52% (twitter) 65.96% (review) 44.88% (movie reviews)		CNN-SVM is used.
Choudhary et al. [17]	Hindi, Telugu English, Spanish	English (Movie reviews), Spanish (Twitter), Hindi (Product review), Telugu (NewsDataset)	80.50% (Hindi- English) 80.30% (Telugu- English) 81.5% (English- Spanish)		Bi-LSTM RNN method is used.
Bhargava et al. [13]	Hindi, Tamil, Bengali	SAIL-2015	77.63% (Hindi) 57.37% (Bengali) 71.56% (Tamil)		RNN, CNN, LSTM methods are used.
Shalini et al. [82]	Bengali-English (Code-Mixed) Telugu	SAIL-2017 (Code-mixed) 8500 Movie reviews sentences (Telugu)	73.20% (Bengali- English) 51.30% (Telugu)		CNN is used.

She is very pretty.	वह बहुत सुंदर है।
Happy New Year!	नव वर्ष की शुभकामनाएं!
Happy New Year!	नए साल की बधाईयाँ।

Fig. 4 Examples from Hindi-English translation dataset

Neural Unsupervised machine translations for various low resource Indian languages are not yet experimented. Another area of potential interest is Transfer Learning, where knowledge for less resourced task is obtained by gaining knowledge from resource rich domain/tasks. This reduces the problem of overfitting in neural networks. This is being used in image processing tasks but yet to be experimented in NLP applications/tasks. Visual Question Answering [92] is also another language processing task, where language processing of questions has not been experimented in Indian languages.

7 Conclusion

In this paper, various state-of-the-art techniques and approaches used for language processing tasks are reviewed in detail. Comprehensive reviews on language processing, especially on the Indian regional languages are presented. Various methods like tokenization, machine transliteration, lemmatization, stemming, POS tagging and so on, which are the building blocks for many natural language processing tasks, are reviewed. Major approaches like lexicon/rule based, statistical based and neural networks for various tasks like Machine Translation, Sentiment Analysis and Named Entity Recognition are discussed. In this article, detailed description of various research works for tackling the problems on low resource languages (especially Indian languages) is presented. The challenges faced in making machine understand natural languages and enabling machines for natural language generation are described. The dataset sources which are available for some Indian language processing tasks are also presented. Further to the descriptive review, promising future avenues like enabling machines to understand low resource Indian languages by generating corpora, multilingual models, Transfer learning and other natural language generation tasks for Indian languages are listed. With these particular points on future work and exploration of ongoing methods, we believe that the research on Indian regional language processing will be aided.

Acknowledgements This work is supported by Vision Group on Science and Technology (VGST), Department of IT,BT and Science and Technology, Government of Karnataka, India. [File No.: VGST/2019-20/GRD No.:850/397]

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Akhtar MS, Kumar A, Ekbal A, Bhattacharyya P (2016) A hybrid deep learning architecture for sentiment analysis. In: Proceedings of COLING 2016, the 26th international conference on computational linguistics: technical papers, pp 482–493
2. Almatrafi O, Parack S, Chavan B (2015) Application of location-based sentiment analysis using twitter for identifying trends towards indian general elections 2014. In: Proceedings of the 9th international conference on ubiquitous information management and communication, ACM, p 41
3. Amarappa S, Sathyanarayana S (2013) Named entity recognition and classification in kannada language. *Int J Electron Comput Sci Eng* 2(1):281–289
4. Amarappa S, Sathyanarayana S (2015) Kannada named entity recognition and classification using conditional random fields. In: 2015 International conference on emerging research in electronics. Computer Science and Technology (ICERECT), IEEE, pp 186–191
5. Ameta J, Joshi N, Mathur I (2012) A lightweight stemmer for gujarati. arXiv preprint [arXiv:12105486](https://arxiv.org/abs/12105486)
6. Anand Kumar M, Dhanalakshmi V, Soman K, Rajendran S (2010) A sequence labeling approach to morphological analyzer for tamil language. *Int J Comput Sci Eng* 2(06):2201–2208
7. Antony P, Soman K (2010) Kernel based part of speech tagger for kannada. In: 2010 international conference on machine learning and cybernetics, IEEE, vol 4, pp 2139–2144
8. Antony P, Soman K (2011) Machine transliteration for indian languages: a literature survey. *Int J Sci Eng Res IJSER* 2:1–8
9. Antony P, Ajith V, Soman K (2010) Kernel method for english to kannada transliteration. In: 2010 international conference on recent trends in information. Telecommunication and Computing, IEEE, pp 336–338
10. Arbabi M, Fischthal SM, Cheng VC, Bart E (1994) Algorithms for arabic name transliteration. *IBM J Res Dev* 38(2):183–194
11. Bahdanau D, Cho K, Bengio Y (2014) Neural machine translation by jointly learning to align and translate. arXiv preprint [arXiv:14090473](https://arxiv.org/abs/14090473)
12. Balamurali A, Joshi A, Bhattacharyya P (2012) Cross-lingual sentiment analysis for indian languages using linked wordnets. In: Proceedings of COLING 2012: Posters, pp 73–82
13. Bhargava R, Arora S, Sharma Y (2019) Neural network-based architecture for sentiment analysis in indian languages. *J Intell Syst* 28(3):361–375
14. Bijal D, Sanket S (2014) Overview of stemming algorithms for indian and non-indian languages. arXiv preprint [arXiv:14042878](https://arxiv.org/abs/14042878)
15. Br S, Kumar R (2012) Kannada part-of-speech tagging with probabilistic classifiers. *Int J Comput Appl* 975:888
16. Broadwell GA, Butt M, King TH (2005) It aint necessarily s (v) o: Two kinds of vso languages. In: Proceedings of the LFG 05 conference. <http://csli-publications.stanford.edu/LFG/10/lfg05.html>. Stanford, CSLI Publications

17. Choudhary N, Singh R, Bindlish I, Shrivastava M (2018) Emotions are universal: Learning sentiment based representations of resource-poor languages using siamese networks. arXiv preprint [arXiv:18040805](https://arxiv.org/abs/1804.0805)
18. Dalal A, Nagaraj K, Sawant U, Shelke S (2006) Hindi part-of-speech tagging and chunking: a maximum entropy approach. In: Proceedings of the NLPAL machine learning contest, vol 6
19. Dandapat S, Sarkar S, Basu A (2007) Automatic part-of-speech tagging for bengali: an approach for morphologically rich languages in a poor resource scenario. In: Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions, Association for Computational Linguistics, pp 221–224
20. Dasgupta S, Ng V (2006) Unsupervised morphological parsing of bengali. *Lang Resour Eval* 40(3–4):311–330
21. DatasetSource-CDAC (2003) https://www.cdac.in/index.aspx?id=products_services. Accessed on 12 Dec 2019
22. DatasetSource-EMILLE (2003) <https://www.lancaster.ac.uk/fass/projects/corpus/emille/>. Accessed on 12 Dec 2019
23. DatasetSource-IJCNLP (2008) <http://ltrc.iit.ac.in/ner-ssea-08/>. Accessed on 12 Dec 2019
24. DatasetSource-Manythingsorg (2015) <http://www.manythings.org/anki/>. Accessed on 12 Dec 2019
25. Dave S, Parikh J, Bhattacharyya P (2001) Interlingua-based english-hindi machine translation and language divergence. *Mach Transl* 16(4):251–304
26. Deri A, Knight K (2016) Grapheme-to-phoneme models for (almost) any language. In: Proceedings of the 54th annual meeting of the association for computational linguistics (Volume 1: Long Papers), pp 399–408
27. Devlin J, Chang MW, Lee K, Toutanova K (2018) Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint [arXiv:181004805](https://arxiv.org/abs/1810.04805)
28. Dhanalakshmi V, Shivapratap G, Soman Kp RS (2009) Tamil pos tagging using linear programming
29. Dhore M, Dixit S, Dhore R (2012) Hindi and marathi to english ne transliteration tool using phonology and stress analysis. In: Proceedings of COLING 2012: Demonstration Papers, pp 111–118
30. Ekbal A, Bandyopadhyay S (2007) A hidden markov model based named entity recognition system: Bengali and hindi as case studies. In: International conference on pattern recognition and machine intelligence, Springer, pp 545–552
31. Ekbal A, Bandyopadhyay S (2008a) Bengali named entity recognition using support vector machine. In: Proceedings of the IJCNLP-08 workshop on named entity recognition for South and South East Asian languages
32. Ekbal A, Bandyopadhyay S (2008b) Part of speech tagging in bengali using support vector machine. In: 2008 international conference on information technology, IEEE, pp 106–111
33. Ekbal A, Bandyopadhyay S (2009) A conditional random field approach for named entity recognition in bengali and hindi. *Linguist Issues Lang Technol* 2(1):1–44
34. Ekbal A, Bandyopadhyay S (2010a) Named entity recognition using appropriate unlabeled data, post-processing and voting. *Informatica* 34(1):459
35. Ekbal A, Bandyopadhyay S (2010b) Named entity recognition using support vector machine: a language independent approach. *Int J Electr Comput Syst Eng* 4(2):155–170
36. Ekbal A, Naskar SK, Bandyopadhyay S (2006) A modified joint source-channel model for transliteration. In: Proceedings of the COLING/ACL on main conference poster sessions, Association for Computational Linguistics, pp 191–198
37. Ekbal A, Haque R, Bandyopadhyay S (2007a) Bengali part of speech tagging using conditional random field. In: Proceedings of seventh international symposium on natural language processing (SNLP2007), pp 131–136
38. Ekbal A, Naskar SK, Bandyopadhyay S (2007b) Named entity recognition and transliteration in bengali. *Lingvisticae Invest* 30(1):95–114
39. Ekbal A, Haque R, Bandyopadhyay S (2008) Named entity recognition in bengali: A conditional random field approach. In: Proceedings of the third international joint conference on natural language processing, Vol II
40. Godase A, Govilkar S (2015) Machine translation development for Indian languages and its approaches. *Int J Nat Lang Comput* 4:55–74
41. Goldsmith J (2001) Unsupervised learning of the morphology of a natural language. *Comput Ling* 27(2):153–198
42. Goyal V, Lehal GS (2011) Hindi to punjabi machine translation system. In: Proceedings of the 49th annual meeting of the association for computational linguistics: systems demonstrations, Association for Computational Linguistics, pp 1–6
43. Gupta V, Lehal GS (2011) Named entity recognition for punjabi language text summarization. *Int J Comput Appl* 33(3):28–32
44. Jamatia A, Gambäck B, Das A (2015) Part-of-speech tagging for code-mixed english-hindi twitter and facebook chat messages. In: Proceedings of the international conference recent advances in natural language processing, pp 239–248
45. Jha V, Manjunath N, Shenoy PD, Venugopal K (2016) Hsra: Hindi stopword removal algorithm. In: 2016 international conference on microelectronics. Computing and Communications (Micro-Com), IEEE, pp 1–5
46. Kaur B, Veer D (2016) Translation challenges and universal networking language. *Int J Comput Appl* 133(15):36–40
47. Kumar A, Kohail S, Ekbal A, Biemann C (2015a) lit-tuda: System for sentiment analysis in indian languages using lexical acquisition. In: International conference on mining intelligence and knowledge exploration, Springer, pp 684–693
48. Kumar H, Harish B, Kumar S, Aradhya V (2018) Classification of sentiments in short-text: an approach using msmtf measure. In: Proceedings of the 2nd international conference on machine learning and soft computing, ACM, pp 145–150
49. Kumar KA, Rajasimha N, Reddy M, Rajanarayana A, Nadgir K (2015b) Analysis of users sentiments from kannada web documents. *Procedia Comput Sci* 54:247–256
50. Lakshmi BS, Shambhavi B (2019) Bidirectional long short-term memory for automatic english to kannada back-transliteration. *Emerging Research in Computing*. In: Information, communication and applications, Springer, pp 277–287
51. Madankar M, Chandak M, Chavhan N (2016) Information retrieval system and machine translation: a review. *Procedia Comput Sci* 78:845–850
52. Majumder P, Mitra M, Parui SK, Kole G, Mitra P, Datta K (2007) Yass: Yet another suffix stripper. *ACM Trans Inf Syst* 25(4):18
53. Moran S, McCloy D (eds) (2019) PHOIBLE 2.0. Max Planck Institute for the Science of Human History, Jena, <https://phoible.org/>
54. Narayan R, Chakraverty S, Singh V (2014) Neural network based parts of speech tagger for hindi. *IFAC Proc Vol* 47(1):519–524
55. Nayan A, Rao BRK, Singh P, Sanyal S, Sanyal R (2008) Named entity recognition for indian languages. In: Proceedings of the IJCNLP-08 workshop on named entity recognition for South and South East Asian Languages
56. Olinsky C, Black AW (2000) Non-standard word and homograph resolution for asian language text analysis. In: Sixth international conference on spoken language processing
57. Pal U, Chaudhuri B (2004) Indian script character recognition: a survey. *Pattern Recogn* 37(9):1887–1899
58. Pandey AK, Siddiqui TJ (2008) An unsupervised hindi stemmer with heuristic improvements. In: Proceedings of the second workshop on Analytics for noisy unstructured text data, ACM, pp 99–105

59. Papineni K, Roukos S, Ward T, Zhu WJ (2002) Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting on association for computational linguistics, Association for Computational Linguistics, pp 311–318
60. Parikh A (2009) Part-of-speech tagging using neural network. In: Proceedings of ICON
61. Patel P, Popat K, Bhattacharyya P (2010) Hybrid stemmer for gujarati. In: Proceedings of the 1st workshop on South and Southeast Asian Natural Language Processing, pp 51–55
62. Patil N, Patil AS, Pawar B (2016) Survey of named entity recognition systems with respect to indian and foreign languages. *Int J Comput Appl* 134(16):88
63. Paul S, Tandon M, Joshi N, Mathur I (2013) Design of a rule based hindi lemmatizer. In: Proceedings of Third international workshop on artificial intelligence, soft computing and applications, Chennai, India, Citeseer, pp 67–74
64. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zetlemoyer L (2018) Deep contextualized word representations. *arXiv preprint [arXiv:1802.05365](https://arxiv.org/abs/1802.05365)*
65. Poornima C, Dhanalakshmi V, Anand K, Soman K (2011) Rule based sentence simplification for english to tamil machine translation system. *Int J Comput Appl* 25(8):38–42
66. Prabhu A, Joshi A, Shrivastava M, Varma V (2016) Towards subword level compositions for sentiment analysis of hindi-english code mixed text. *arXiv preprint [arXiv:1611.00472](https://arxiv.org/abs/1611.00472)*
67. Prathibha R, Padma M (2015) Design of rule based lemmatizer for kannada inflectional words. In: 2015 international conference on emerging research in electronics. Computer Science and Technology (ICERECT), IEEE, pp 264–269
68. PVS A, Karthik G (2007) Part-of-speech tagging and chunking using conditional random fields and transformation based learning. *Shallow Parsing for South Asian Languages* 21
69. Rajan K, Ramalingam V, Ganesan M, Palanivel S, Palaniappan B (2009) Automatic classification of tamil documents using vector space model and artificial neural network. *Expert Syst Appl* 36(8):10914–10918
70. Ramachandran VA, Krishnamurthi I (2012) An iterative stemmer for tamil language. In: Asian conference on intelligent information and database systems, Springer, pp 197–205
71. Ramanathan A, Rao DD (2003) A lightweight stemmer for hindi. In: The proceedings of EACL
72. Ramanathan A, Hegde J, Shah RM, Bhattacharyya P, Sasikumar M (2008) Simple syntactic and morphological processing can help english-hindi statistical machine translation. In: Proceedings of the third international joint conference on natural language processing, Vol I
73. Ranjan P, Basu HVSSA (2003) Part of speech tagging and local word grouping techniques for natural language parsing in hindi. In: Proceedings of the 1st international conference on natural language processing (ICON 2003), Citeseer
74. Ravi K, Ravi V (2016) Sentiment classification of hinglish text. In: 2016 3rd international conference on recent advances in information technology (RAIT), IEEE, pp 641–645
75. Reddy MV, Hanumanthappa M (2013) Indic language machine translation tool: english to kannada/telugu. In: Multimedia processing. Springer, Communication and Computing Applications, pp 35–49
76. Revanuru K, Turlapaty K, Rao S (2017) Neural machine translation of indian languages. In: Proceedings of the 10th annual ACM India compute conference, ACM, pp 11–20
77. Rohini V, Thomas M, Latha C (2016) Domain based sentiment analysis in regional language-kannada using machine learning algorithm. In: 2016 IEEE international conference on recent trends in electronics, information and communication technology (RTEICT), IEEE, pp 503–507
78. Saharia N, Sharma U, Kalita J (2012) Analysis and evaluation of stemming algorithms: a case study with assamese. In: Proceedings of the international conference on advances in computing, communications and informatics, ACM, pp 842–846
79. Saini S, Sahula V (2015) A survey of machine translation techniques and systems for indian languages. In: 2015 IEEE international conference on computational intelligence and communication technology, IEEE, pp 676–681
80. Se S, Vinayakumar R, Kumar MA, Soman K (2015) Amrita-cen@sail2015: sentiment analysis in indian languages. In: International conference on mining intelligence and knowledge exploration, Springer, pp 703–710
81. Selvam M, Natarajan A (2009) Improvement of rule based morphological analysis and pos tagging in tamil language via projection and induction techniques. *Int J Comput* 3(4):357–367
82. Shalini K, Ravikurnar A, Vineetha R, Aravinda RD, Anand KM, Soman K (2018) Sentiment analysis of indian languages using convolutional neural networks. In: 2018 international conference on computer communication and informatics (ICCCI), IEEE, pp 1–4
83. Shrivastava M, Bhattacharyya P (2008) Hindi pos tagger using naive stemming: harnessing morphological information without extensive linguistic knowledge. In: International conference on NLP (ICON08), Pune, India
84. Singh S, Gupta K, Shrivastava M, Bhattacharyya P (2006) Morphological richness offsets resource demand-experiences in constructing a pos tagger for hindi. In: Proceedings of the COLING/ACL on main conference poster sessions, Association for Computational Linguistics, pp 779–786
85. Somers H (1999) Example-based machine translation. *Mach Transl* 14(2):113–157
86. Suba K, Jiandani D, Bhattacharyya P (2011) Hybrid inflectional stemmer and rule-based derivational stemmer for gujarati. In: Proceedings of the 2nd workshop on South Southeast Asian natural language processing (WSSANLP), pp 1–8
87. Todi KK, Mishra P, Sharma DM (2018) Building a kannada pos tagger using machine learning and neural network models. *arXiv preprint [arXiv:1808.03175](https://arxiv.org/abs/1808.03175)*
88. Tummalapalli M, Mamidi R (2018) Syllables for sentence classification in morphologically rich languages. In: Proceedings of the 32nd Pacific Asia conference on language, information and computation
89. Unnikrishnan P, Antony P, Soman K (2010) A novel approach for english to south dravidian language statistical machine translation system. *Int J Comput Sci Eng* 2(08):2749–2759
90. Vijayakrishna R, Sobha L (2008) Domain focused named entity recognizer for tamil using conditional random fields. In: Proceedings of the IJCNLP-08 workshop on named entity recognition for South and South East Asian Languages
91. Webster JJ, Kit C (1992) Tokenization as the initial phase in nlp. In: COLING 1992 Volume 4: The 15th international conference on computational linguistics
92. Wu Q, Teney D, Wang P, Shen C, Dick A, van den Hengel A (2017) Visual question answering: A survey of methods and datasets. *Comput Vis Image Underst* 163:21–40

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.