Complex & Intelligent Systems
https://doi.org/10.1007/s40747-021-00487-7

ORIGINAL ARTICLE

Hatred and trolling detection transliteration framework using hierarchical LSTM in code-mixed social media text

Shashi Shekhar[1] · Hitendra Garg[1] · Rohit Agrawal[1] · Shivendra Shivani[2] · Bhisham Sharma[3]

Received: 8 May 2021 / Accepted: 24 July 2021

Abstract

The paper describes the usage of self-learning Hierarchical LSTM technique for classifying hatred and trolling contents in social media code-mixed data. The Hierarchical LSTM-based learning is a novel learning architecture inspired from the neural learning models. The proposed HLSTM model is trained to identify the hatred and trolling words available in social media contents. The proposed HLSTM systems model is equipped with self-learning and predicting mechanism for annotating hatred words in transliteration domain. The Hindi–English data are ordered into Hindi, English, and hatred labels for classification. The mechanism of word embedding and character-embedding features are used here for word representation in the sentence to detect hatred words. The method developed based on HLSTM model helps in recognizing the hatred word context by mining the intention of the user for using that word in the sentence. Wide experiments suggests that the HLSTM-based classification model gives the accuracy of 97.49% when evaluated against the standard parameters like BLSTM, CRF, LR, SVM, Random Forest and Decision Tree models especially when there are some hatred and trolling words in the social media data.

Keywords Hatred detection · Trolling · Social media · HLSTM · Embedding · Transliteration

Introduction

Doubtful online users are hiding their real identities and are using their social media accounts for deceptive online messages and incitement comments and tweets for spreading hatred in the society. Hate content detection is the automated assignment for extracting hatred-based words available in contents on the social media. Hateful contents can further be used to misinform people in the society and thus can result in violent incidents. With online hateful contents culminating in gruesome scenarios like the Rohingya issue in India, Article 307 issue, anti-social elements mob violence, and the Terrorist shooting issues, etc., there is a dire need to understand the dynamics of user interaction that facilitate the spread of such hateful content. It is also observed that the content generated by the hateful users tend to spread faster as compared to the content generated by the normal users [1]. An intelligent tool for detecting hate speech or hatred contents present on social media in Indian context is the need for the present time. The internet medium is the widely used source for communication and information exchange in current scenario. In present times language used for communication over internet is not limited to one rather people are using mixed language for communication. The following sentence illustrates an example showing the mixed language format used for communication exhibiting hatred and trolling terms which can provoke violence in the society. **Sentence 1:** @–inke last 5 years ke work ko dekh lijiy ***nafrat*** ho

✉ Bhisham Sharma
Bhisham.sharma@chitkarauniversity.edu.in;
Bhisham.pec@gmail.com

Shashi Shekhar
shashi.shekhar@gla.ac.in

Hitendra Garg
hitendra.garg@gla.ac.in

Rohit Agrawal
rohit.agrawal@gla.ac.in

Shivendra Shivani
shivendra.shivani@thapar.edu

[1] Department of Computer Engineering and Applications, GLA University, Mathura 281406, India

[2] Department of Computer Science & Engineering, Thapar University, Patiala 147004, India

[3] Chitkara University School of Engineering and Technology, Chitkara University, Chandigarh, Himachal Pradesh, India

Published online: 17 August 2021

jayegi aapko inse. Desh mein julus nilkegi aur public khule aam ***patthro se maaregi*** inko. Phir se ***dhamkana*** hoga. Aise logo se ***Nafrat*** karo. The sentence 1 uses the code-mixed content posted on social media related to introducing hatred which is translated as '@.. look at the past 5 years' work of these people; one will start hating them. A rally will be organized and public will throw stones at them. Again, there is a need to threaten them. Hate these people. Kill them. In context to above sentence 1, the mechanism of annotation is used for better understanding of scenario. Hindi words are tagged as H, English words are tagged as E, and the words belonging to Hindi or English denoting the hatred are tagged as E/HT and H/HT, respectively. The word "*last*", "*work*", "*public*" is tagged as E. The word "*Hate*" *and* "*Kill*" will be annotated as E/HT. The word "*Desh*", "*inke*", "*dekh*", "*julus*" will be tagged as H and the words like "*nafrat*", "*Patthar mareenge*" *and* "*dhamaka*" will be annotated as H/HT. The extensive use of social platforms leads to the availability of these contents in multilingual form and it is a challenging task for processing.

HLSTM (Hierarchical Long Short Term Memory) is a novel classification technique that can be applied to extract language information from code-mixed data. The HLSTM-based learning model is the extension of the existing LSTM model. Neural learning approach is the primary option for processing any user-generated text which further helps in extracting language information of the text. The mixing of languages for expressing any opinion is very common in context to Indian scenario and is widely applied on social media. The main challenge in processing and analyzing these data is the availability of multiple variants of writing a word using transliteration mechanism. The work presented in the paper points out the confusion that rose among the languages which are semantically and linguistically related. As we human beings are more inclined towards the use of our native language for communication [2] and the same native language is used on the internet social media domains for expressing opinions. These native languages provide the freedom to express the things in a very easy manner for those users who are not proficient in using English as a language.

The work presented in the paper points out a novel framework, describing the information available for the word level context identification for predicting the hatred content. Here in regard to the dataset used for evaluation the hatred words are tagged as E/HT. There are many applications where these types of scenarios are needed to be normalized and processed for better understanding. The applications of question answering system and sentiment analysis along with text classification can be potential use-cases for context analysis based on discussed scenarios.

The ambiguity extraction for words belonging to English language is a challenge; however, the use of HLSTM learning helps in analyzing the context of the word. The model discussed in the paper is based on recent research undertaken in the area of computer vision and NLP tasks [1, 3]. The HLSTM technique of learning is applied and tested on various parameters.

The text analysis and embedding technique is applied [4] for processing. The character embedding along with word embedding models has been involved for ambiguity extraction and thus the model has been trained to exhibit the learning outcomes in the term of hatred and trolling word detection [5].

The contribution and deliverables of this work are as follows:

(1) Introduction of HLSTM classification and learning model for code-mixed data. The developed HLSTM model is a computational linguistic architecture based on LSTM learning for hatred detection in social media contents.
(2) Implementation and proper alignment of information on the basis of phonetics context is used for training.
(3) The evaluation uses both statistical measures and deep learning-based parametric measures for retrieving hatred information.
(4) Novel voting technique is further applied to cross validate the obtained results. The F1-measure of this technique gives better accuracy in case the hatred ambiguity is not resolved automatically.
(5) Limited featural aspects have been used for processing hatred contents in multi-lingual text.

The remaining portion of the paper is represented under the following headings: Literature review is presented in Sect. "Literature review". Proposed methodology is available in Section "Proposed work". Dataset description and definition is in Section "Dataset". The Section "Evaluations" provides experimental discussion and evaluation measures. Last the overall inference extracted is available in Section "Conclusions".

Literature review

This section points out the related work surveyed in context to the language transliteration, code-mixing and ambiguity detection in code mixed data. Section "Language transliteration" describes the state-of-art in the field of transliteration considering the use of machine learning techniques for language transliteration because transliterated text is usually used on internet social media sites for expressing opinions. Section "Code-mixing" points out the research development in the field of code-mixing. The code-mixing is frequent pattern used by the users for writing posts using two languages where one language is English and other is the transliterated form of native language. Section "Hatred and trolling detection" provides the summarization of work undertaken in identifying the hatred and trolling words used in the written sentences. The proposal presented in the paper points out the potential research gap in terms of identifying hatred and trolling terms used in social media posts in Indian language context based on language English and transliterated versions of Hindi.

Language transliteration

The transliteration domain is the current research area for text analysis. The practice of language mining is the foremost task in textual content processing [6]. The paper [7] points the application and use of CRF (Conditional Random Field) method in bigram processing. The paper [8] focuses on applications and use of LR (Logistic Regression) with probability distribution function in code-mixed domain. The paper [9] points out the applicability of dictionary in normalization of transliterated terms.

The paper [10] presents the creation of linguistic resource for the language English and Gujarati. The approach of translation into native language using transliteration is the approach for identifying language. The work presented in the paper mainly concerns with the transliteration of Guajarati for identifying the language used in code-mixed language.

A mixed script based language identification task was conducted for Indian Languages [11]. Here the use of machine learning techniques using SVM (Support Vector Machines) classifier [12] was proposed. The technique of classification and its related machine learning techniques for English-Hindi [13–15] languages were taken care. This task gives the opportunity for the emerging researched to enhance their learning and understanding the domain area covered under transliteration field [16]. Various emotion identification models have been described based on learning approach [17] for language mining [18–20].

Code-mixing

The paper [12, 21] describes the code mixing patterns in text contents. The work on entity mining in code-mixed [22] data is discussed with the use of embedding techniques [23]. The paper [24] focused on communication medium where short forms are used and its meaning extraction work is presented in the research. Use of regional dialects has been pointed out in communication and identification of its context meaning is handled in the paper. The paper [25, 26] presents the state-of-art in language identification. The paper [27] points the use of MNN (Multi-Layer Neural Network) along with LSTM (Long Short Term Memory) for ambiguity minimization in mixed script textual data.

The paper [28] provides the detailing of challenges added to the code-mixed data for analyzing the sentiments. The author presents the use of BLSTM (Bi-directional Long Short Term Memory) sentence generation and selection using neural classification model for classifying the code-mixed text into predefined sentiment classes. This classification approach based on BLSTM model overcomes the nuances of detecting sentiments in code-mixed data.

Hatred and trolling detection

The work presented in [29] contains the approach for hatred ambiguity removal with the aid of learning models mostly in intrinsic language domain for finding effective context meaning. This section tries to model the ambiguity problem available in code mixed data using embedding technique [30]. The embedding model is widely used in finding ambiguous words which are commonly used in both the languages, as it is the most common research issue [31] in multilingual dataset [32] used in case of NER (Named Entity Recognition) [32] extraction in transliterated domain.

The semantic similarity identification is handled in the paper [33] for analyzing two concepts in the domain of NLP. A method based on WordNet 3.1 is used for determining the similarity. The work presented by the author overcomes the ambiguities found in social media text using the feature selection technique for improving the semantic similarity. The findings suggest that similarity or ambiguity identification in terms of concepts or words depends on selected balanced features rather based on unbalanced features.

The paper [34] provides the detailing regarding evaluation measures, for semantic representation based on the parameters of including shortest path for context measurement. The paper addresses the ambiguity removal mechanism by using the synonymy concept through imparting knowledge based lexicons. The knowledge-based approach and statistical approach have been used for representing the semantic representation. Knowledge-based approach uses dictionary and thesaurus along with ontologies, whereas the statistical approach is based on finding the word frequencies for identifying semantic relations among the words.

Hatred- and trolling-based ambiguous word detection is a challenging issue. Thus, the use of LSTM and BLSTM [30, 31] are nowa days incorporated for effective results. Code mixing gives the flexibility to the users to mix more than one language for expressing the thoughts. So, to process such code-mixed text, identification of language used in each word is important for language processing. The main research issue is to propose a technique for identifying the language of the hatred words in Hindi–English code-mixed data. The focus needs to be done on retrieving the word level language identification for hatred words based on the user's intention to use that word in code mixed environment. Thus, the word level intent identification based on the availability of hatred words is an open issue in transliterated domain. The study also reflects that the use of current hierarchical approach of learning can lead towards better learning in predicting hatred, trolling and ambiguous words available in code mixed environment [35–37]. As the HLSTM model has a similar architecture to the LSTM where the use of a 0/1 boundary detection is avoided to detect ambiguity or availability of hatred terms instead hierarchical policy gradient method is used which gradually learns a policy to select better actions from inside the phrase for each word in code-mixed environment. The proposal available in next section uses the HLSTM learning model to detect hatred and trolling terms used in social media domain. The next section describes the proposed model for hatred and trolling word identification in code-mixed text.

Proposed work

The work discussed in this section enhances the work projected in the paper [38] regarding scarce language lexicons. The area of research in transliteration is explored with the use of embedding framework. Language extraction is potential area to explore in regard to transliterated environment. The reason behind selecting HLSTM model for the proposed work is that the HLSTM model has a similar architecture to the LSTM but instead of a 0/1 boundary detection, the HLSTM uses a policy gradient method that gradually learns a policy to select better actions from inside the phrase for each word. The HLSTM determines a structured representation for a sentence for effective identification of hatred words as compared to LSTM, as LSTM is suited well to classify process and predict time series-based ambiguity for known duration. In the next section, the architecture of artificial LSTM network model is explained.

Design principles

Generating information from the language is a challenging task; thus we propose a certain design principles for evaluating the proposed algorithm. The design guidelines are provided as follows:

a. The document must contain words from two different languages.
b. A single script nomenclature must be followed for writing the contents. Here the single script selected for writing is Roman script.
c. Scenarios are based considering the fact that in India code mixing is done between two language and out of these two one language is English.
d. The comments length must be between 2 and 15 words.
e. The sematic and syntactic rules are applied for validating the mixing points of the languages.

The model proposed is depicted in Fig. 1. The proposed model follows the training sequence of HLSTM [39] at word level. The applied embedding technique helps in retrieving

Table 1 Phonemic representation for ambiguous roman words

Ambiguous Roman words	Hindi-phonemic	English-phonemic
Dust	दुस्ट	डस्ट
Hat	हट	हैट
Fat	फट	फैट
Mat	मत	मैट
Log	लोग	लॉग
OR	ओर → और	और
MAIN	मैं	मेन
TO	तो	टू
Use	उसे	यूज़
IS	इस	इज

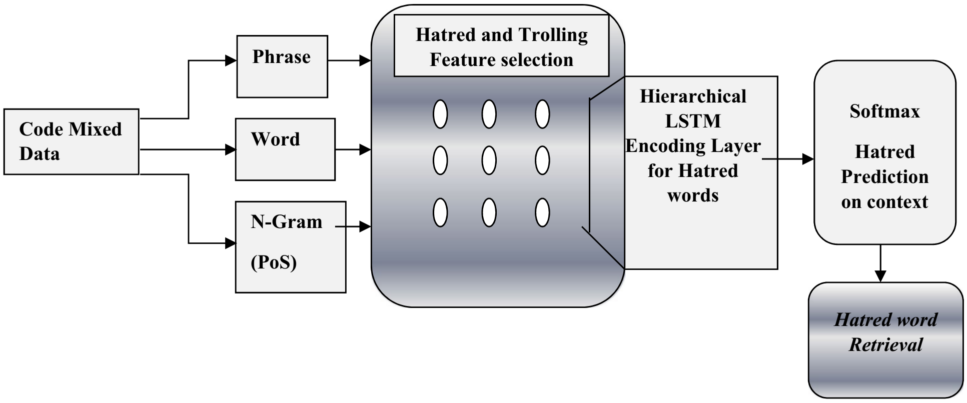

Code Mixed Data
Phrase
Word
N-Gram (PoS)
Hatred and Trolling Feature selection
Hierarchical LSTM Encoding Layer for Hatred words
Softmax Hatred Prediction on context
Hatred word Retrieval

Fig. 1 HLSTM model for hatred and trolling detection

Table 2 Phonemic representation for hatred roman words

Hatred Roman Hindi words
गुस्सा → (*ġuSSaa*)
नाराज़ → (*Naaraaz*)
नाराज़गी → (*Naaraazagii*)
गुस्सा होना → (*ġuSSaa hoNaa*)
नाराज़ होना → (*Naaraaz hoNaa*)
गुस्सैल → (*ġuSSaiL*)
नफरत →(Nafrat)

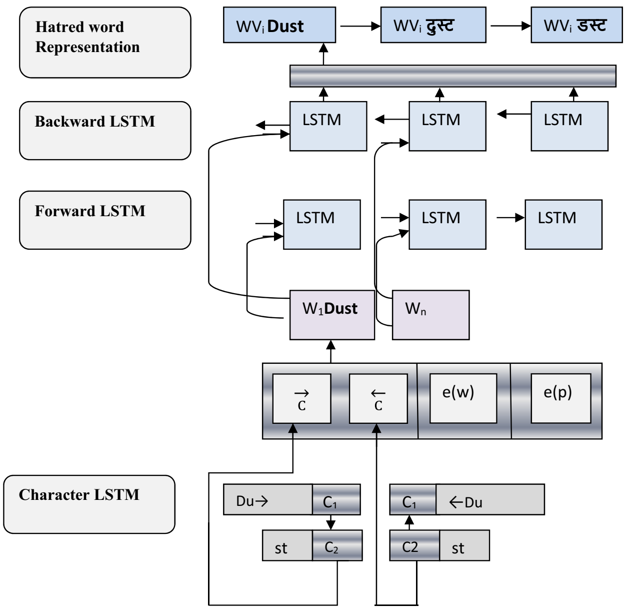


Fig. 2 HLSTM learning model

the hatred words. Tables 1, 2 depict the parameters of LSTM model and Fig. 2 describes the learning model.

Figure 2 depicts the architecture of proposed HLSTM model for hatred terms detection. The proposed model takes the input in terms of code-mixed sentence containing sequence of words in form of ($w_{1\_}w_n$). The mixed sentence is converted into tokens for further processing. As there exists the possibility of English word to be used in Hindi context also like the word dust which can be used as दुस्ट or as डस्ट, this ambiguity needs to be resolved prior to identifying hatred terms from the code-mixed sentence. This is being done using the BLSTM learning model where embedding has been applied to extract the correct context meaning of the ambiguous words. The HLSTM model depicted in Fig. 2 is based on the concept of Vector representation. This vector representation helps in capturing context the context meaning of that word in regard to previous words and next available words in the sentence sequence. The following are the generated vectors: (i) Vector → Forward character ($\overrightarrow{C}$), (ii) Vector ← Backward character ($\overleftarrow{C}$), (iii) Vector: Embedding word ($e(w_i)$) and i(v) PoS vector Embedding ($e(p_i)$). The transformation w$_e$ is applied to *Softmax* function g. The computation is depicted using Eq. 1. Embedding's are necessary for computations as the words needs to be converted into numbers. The vector representation of the words is necessary for computations. Similar words have similar kind of vectors while the dissimilar words have differences in their vectors. Vector representation helps in capturing context meaning of the words. Consider this example: '*gussa*' and '*gussail*' can be described in similar context when compared with another word '*gussa*' and '*dust*'. This technique of representation is the baseline of the algorithm used for training and testing. The classifier feature training parameters are depicted in Table 3.

$$S = g\left(W_{\mathrm{e}}\left[\overrightarrow{\boldsymbol{C}};\overleftarrow{C};e\left(W_i\right);e\left(P_i\right)\right]\right. \quad (1)$$

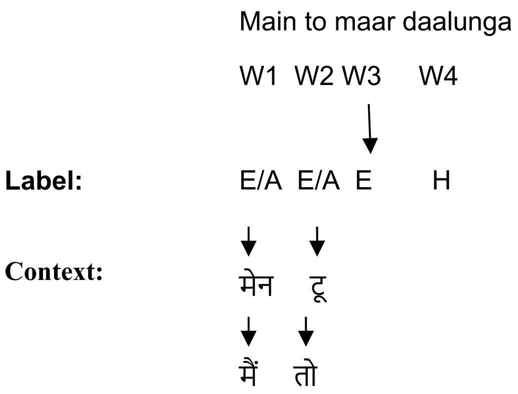
The model is trained to predict the belongingness of the word to the languages used. The tagging technique applied are used to distinct the context [40, 41]. The trained model is an example of learning applied for text analysis. The activation function softmax helps in classification based on embedding parameters *e*(*p*). The classification is done on the basis of pre-trained tagging constraints. Tables 1, 2 depict the sample hatred words used for training and classification.

The embedding technique uses the input for extracting features associated with the words. The use of directional mapping enhances the embedding features. Thus, the use of softmax in vector concatenation as an activation function helps in retrieving the ambiguity.

The consideration of the terms that exist prior to pivot term and next to pivot word forming the things as ($i + 1$) term and ($j + 1$) term are used as word features. Considers the example sentence (Main *to maar daalunga*), here the word of the language is extracted and are tagged accordingly. The learning feature of the model will classify the words which are hatred as E/A, E/HT, H/HT. The tagging scenario is depicted as follows:

Table 3 Parameters for feature extraction

S. no	Feature parameters
1	Context words
2	First & last words
3	Action verb and Adjectives
4	POS
5	English word and Hindi words
6	Hatred word
7	Digit/Special symbols



Embedding: word

The embedding mechanism supports in extracting feature vectors. The purpose behind embedding is for generating computational vectors [42, 43]. Consider the below document which illustrates the embedding technique.

> ***Doc. A:*** "agar tum ***is actor*** ke dushman ho to ***is actor*** ko maara karo".
> ***Doc. B:*** "agar tum ***is actor*** ke dushman ***nahi*** ho to ***is actor*** ko maara ***naa*** karo".

11 features are there in Doc A and Doc B contains 13 features. The feature is computed on the basis of [44] and can be expressed as follows:

> { is, agar, tum, actor, dushman, ke, ho,, ko, to, maara, karo}

The methodology of skip- gram helps in embedding [45]. The Skip-gram description is pointed in Fig. 3. The input word is T_0 this input is provided to the classifier for finding the next sequence word based on probability of log [46] as depicted in Eq. 1. This log probability facilitates the computation in terms of distributed dimensions.

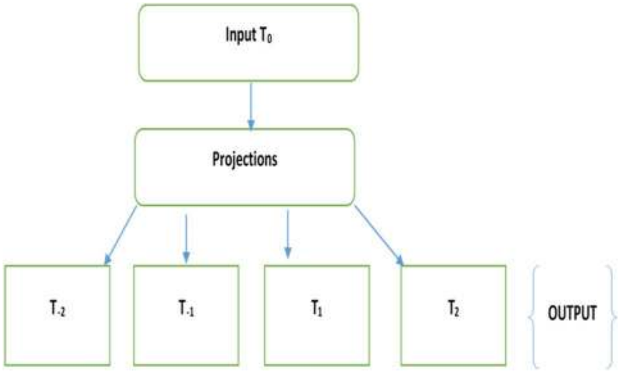


Fig. 3 Skip-gram model

$$L = \frac{1}{N}\sum_{n=1}^{N}\sum_{-x\le i\le x}\log P(Tn + i|Tn) \tag{2}$$

$$P\left(T_j|T_k\right) = \frac{\exp\left(V'T_j\left(VT_k\right)\right)}{\sum_{w=1}^{w}\exp\left(V'T_j\left(VT_k\right)\right)} \tag{3}$$

$$Y = x + S_h\left(W, C_{t-k}, \dots C_{t+k}, C\right). \tag{4}$$

Here in terms of above equation, the $P(T_j\,|T_k)$ is the word probability, Output vector is represented as V'. The Eq. 4 denotes the significance of character embedding especially used for English words which have been tagged as E/A. This Eq. 4 is used to create feature vectors based on character representation for the words which are tagged as ambiguous English. Consider the word "*main*" tagged as E/A This word can have phonetics as मैं in Hindi and मेन in English. The log scaling computation for English words are obtained to further optimize the context probability [47].

Algorithm for hatred detection

```
Input: Mixed format data
Output: Context based word language

    Step1- Calculating English word score
    M1=max(log (frequency (word))
    Word ∈Dictionary (English)
    For any English word
    Word (Score) =log(frequency(w))/M 1
    Word (Score English) = 1
    If w is in Dictionary (English)
    Else
    Word (Score English) = 0

    Step2- Calculating Hindi word score
    M1=max (log (frequency (word))
    Word ∈ Dictionary (Hindi)
    For any Hindi word
    Word (Score) =log(frequency(Hindi word))/M1

    Step 3- Compute TF-IDF score V1
    V1 (t,w)=n(t,w)*log(Hindi Term(HT))/English Term(ET)

    // n (t,w)→ Term (t) count in word (w)
    // HT and ET→ occurrence of Hindi and English term in dictionary
    Step4- calculate maxLikelihood and Classify the Word as E, H, E/HT,H/HT
    Sentence S =W1, W2, W3…Wn
     Interpreted as Sx=W1H W2E W3E W4H …WnH
    Where w1 is H, W2 is E, w3 is E, W4 is H and Wnis H and so on
    // Applying Conditional probability to +1 and -1 word
    P(S=SX) =P (W1H)*P (W2E)*P (W3E)*P (W4H)*P (WnH)
    ScoreE/A=P(S=Sx)
    V1←ProbE*trans_Prob [PwiE][Pwi-1E][Pwi+1E]
     V1←ProbH*trans_Prob [PwiH][Pwi-1H][Pwi+1H]
    Score←max (V1, V2)
Classify the Word as E/HT, H/HT, E/A
```

The use of probability distribution is applied for word identification based on pre-defined classification. Consider an example the word "to" represented as Word_E, and its phonetic similarity in Hindi Word_H is represented as तो. This

Table 4 Dataset

	Total sentences		Total words	
	Training	Testing	Training	Testing
WhatsApp	883	376	3929	903
Twitter	1387	273	25,749	4027
Instagram	782	343	18,742	3879
Facebook	1372	489	24,632	3423

scenario illustrates that one will be having 2^N possibilities for any sentence containing N number of words. The technique of dynamic programming is suitable for computing the possibilities of the words. The concept of context capturing is applied to retrieve all hatred terms used in the sentence. The training for word representation is imparted in the learning phase to forecast the probable words. The learning model applied uses the representation of the words in the sentence as sequence terms ($W_{1,}$ W_2, $W_3 \ldots W_T$) [48–50]. This approach seems to be a computational approach for learning with minimum memory and highly scalable in nature [51, 52]. Thus, the learning of HLSTM based approach is quite beneficial in terms of finding ambiguity and hatred terms and it can be used as a novel tool for solving problems in the area of NLP.

Dataset

The dataset used in the training and testing comes from [53]. The social media contents provide the base for this dataset. The detailing is presented in the Table 4. The dataset contains English–Roman Hindi code-mixed data. As monolingual English and romanized Hindi and other Indian language text messages are prevalent in social media in Indian context. Here we will be concentrating only on code-mixed English–Hindi to extract hatred and trolling terms used on the social media domains. The corpus used is mostly bi-lingual mix. While two languages are blending, one important aspect is to know about the mixing. The blending of languages states which language is mixed in what manner or in what ratio. This leads towards the notion of MI (Multilingual Index) and CMI (Code Mixing Index).

The dataset depicted in Table 4 is standard dataset referred by many researchers for evaluating their hypothesis. This is important for finding patterns for validating the results. The data and resource of WordNet [54] are further used in case of ambiguity identification and normalization. This WordNet is specifically used for analyzing Indian nased languages. The idea behind this resource is to retrieve most frequently used words in the sentence so that the model can be enabled to understand the frequency of the words used in the sentence.

Table 5 MI and CMI values

Language pair	MI	CMI
EN-HN	0.773	32.346

Evaluations

The measures used for evaluating the process discussed in the paper are presented showing the various evaluation measures applied on the above dataset. Two valuation measures are used to evaluate the results, First the statistical measure is used for assessing similarity measure of words, and second context evaluation measure is used based on proposed HLSTM model which is compared against state-of-art other machine learning models. The evaluation measures based on statistical technique are used to find the similarity measures of words which are represented in code-mixed data having different spelling variations, e.g. the word *khauf* can be represented as *khof, khaof, khaoph* and so on in transliterated manner. The statistical evaluation presented in section "Experimental results" and in section "Context retrieval evaluation" describes the evaluation measures for context identification for finding ambiguous hatred terms in code-mixed data based on left and right context of the word in regard to entire sentence.

Experimental results

The evaluation measures selected for testing the hypothesis are based on the techniques of statistical evaluations and learning techniques imparted to the machine based on the concept of HLSTM. The following sections provide the detailing of the evaluation standards used for evaluating the results:

MI (*Multilingual Index*) [9] The concept of MI quantifies the language distribution based on tagging mechanism. The multilingual Index available in the dataset is measured using Eq. 5.

$$\mathrm{MI} = x = \frac{1 - \sum P^2 J}{(k-1)\sum P^2 J}. \tag{5}$$

Here the symbol K denotes the language specification towards the word P_j.

CMI (*Code Mixing Index*) [9] The concept of CMI quantifies the distribution of language used mostly in the dataset. Its mixing index is computed using Eq. 6.

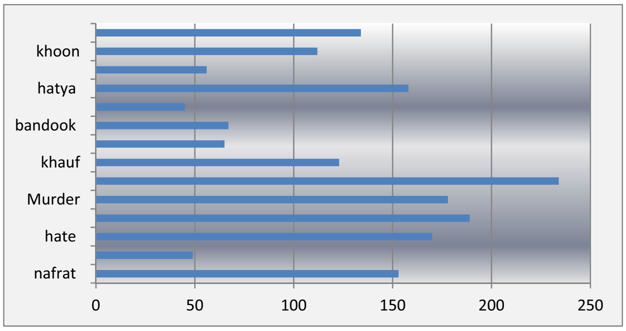


Fig. 4 Hatred word similarity

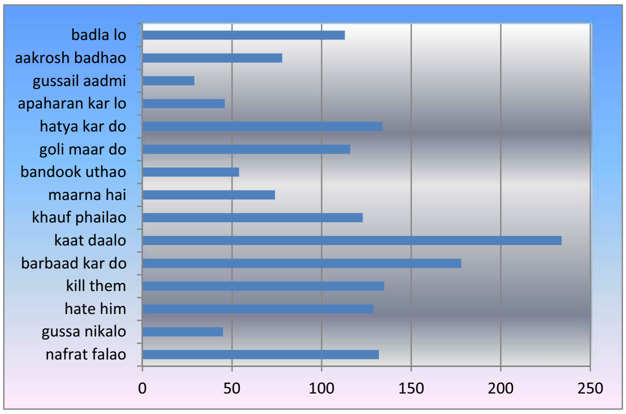


Fig. 5 Hatred word at sentence level

Table 6 Dataset parametric description

Label	Description	Hindi-English %
ENG	ENG Hatred words	24.2
HIN	HIN Hatred words	60.01
PN_{E1}	Proper noun	3.01
Ot	Hatred symbology	0.11
AMG	Ambiguous	11.63
MXD	Mixed format	0.99
UN	Unknown	0.05

$$\mathrm{CMI} = \frac{\sum_{i=1}^{n}\left(w_i\right) - \max\left(w_i\right)}{n-u}, \tag{6}$$

Here in Eq. 6, the $\sum_{i=1}^{n} w_i$ denotes the summation of languages used. The notation max $\{w_i\}$ represents the word available in the dataset in terms of maximum availability. The notations n and u describe the tokens and tagging mechanism. Equation 6 is further normalized on the scaling of 0 and 1 as depicted in Eq. 7.

Table 7 Embedding data

	Words used
ICON [53]	3959
MSIR [58]	3423
MSIR [59]	8734

Table 8 F measure (Twitter)

Embedding	E/HT	H/HT	NE/HT
Character			
1 g	84.95	93.31	78.3
3 g	85.34	93.44	77.1
5 g	**85.38**	**93.49**	**80.2**
Word			
1 g	65.86	82.96	62.2
3 g	**85.71**	**93.97**	**83.9**
5 g	85.42	93.16	78.1

Table 9 F measure (Facebook)

Embedding	E/HT	H/HT	NE/HT
Character			
1 g	85.23	90.91	64.26
3 g	83.54	91.25	63.27
5 g	**88.22**	**94.55**	**67.21**
Word			
1 g	83.92	91.06	61.87
3 g	**87.94**	**94.29**	**68.32**
5 g	86.21	93.74	662.73

$$\mathrm{CMI} = \begin{cases} 100 \times \left[1 - \frac{\max\{wi\}}{n-u}\right] : n > u \\ 0 : n = u \end{cases}. \tag{7}$$

Here in Eq. 7 notation max(w_i) describe the labeled words. The CMI value is being computed using this equation and it provides the mixing patterns in the data which are passed to the machine for further processing (see Table 5).

The evaluation based of statistical measure is computed using Eq. 8. The token similarity is measured on the basis of Conf_Score used in the classifier. The presented Figs. 4 and 5 points out the similarity values obtained on the parameters defined at word and sentence level, respectively.

$$\mathrm{Sim}(X, Y) = \frac{\sum_{i=1}^{n} XiYi}{\sqrt{\sum_{i=1}^{n} Xi^2}\sqrt{\sum_{i=1}^{n} Yi^2}}. \tag{8}$$

The evaluation uses the base of BLSTM learning applied to the hierarchical model. The dataset [53] contains textual

Table 10 F measure (WhatsApp)

Embedding	E/HT	H/HT	NE/HT
Character			
1 g	52.4	80.1	28.5
3 g	**54.9**	**80.2**	**37.7**
5 g	54.3	80.9	31.5
Word			
1 g	50.4	79.6	40.0
3 g	**60.8**	**81.9**	**40.2**
5 g	53.7	80.1	40.1

Table 11 F measure (Instagram)

Embedding	E/HT	H/HT	NE/HT
Character			
1 g	50.4	79.6	40
3 g	**60.8**	**81.9**	**40.2**
5 g	53.7	80.1	40.1
Word			
1 g	50.4	79.6	40
3 g	**60.8**	**81.9**	**40.2**
5 g	53.7	80.1	40.1

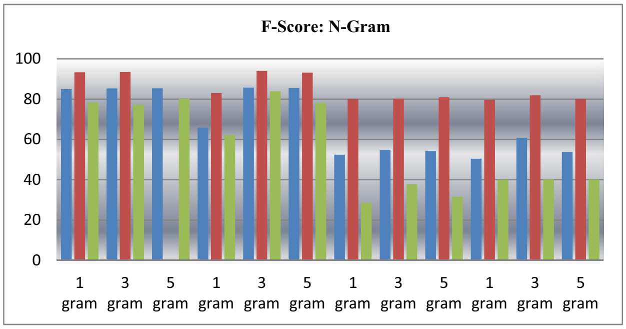


Fig. 6 F-Score for Facebook, WhatsApp, Twitter and Instagram

information posted on social media platforms with defined parameters as illustrated in Table 6.

The training data examples in regard to defined parameters are listed as follows:

1. **EN:** example: Kill, Murder
2. **HN**: example: nafrat, badla, khoon
3. **PNE1:** example: kutta, jaanwar
4. **Ot**: example: C***,F**, @***
5. **AMG:** example: gandelog, victimlog
6. **MXD:** example: Bhai log, Director saheb
7. **UN:** example: F&G, T&S

Fig. 7 HLSTM cloud representation

These defined tagging parameters are used to make the system learn the technique of HLSTM for processing the results. The word available in the data is identified based on these classified parameters for predicting the presence of hatred words. Table 4 provides the detailing of this mechanism which shows data from [55–57]. The embedding features used for further normalizing the process are defined in Table 7.

The table presented below as Tables 8, 9 and 10 provides the detailing testing the proposal using the context finding features.

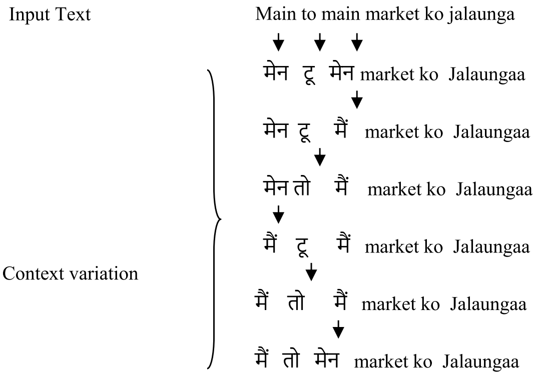
The above table illustrates the embedding parameters in terms of N-gram model. The accuracy in terms of various N-gram parameters is depicted in table in terms of accuracy for Twitter, Facebook, WhatsApp and Instagram (see Table 11) (see Fig. 6).

The representation of cloud depicted in Fig. 7 and 8 provides visual inference of the trained model. The results seem to be clear showing several separations for the various words used in Hindi and English suggesting the correct use of labeling parameters.

Context retrieval evaluation

The process of context evaluation is based on finding the ambiguity in terms of words used in the code-mixed sentence using the left and right contexts in regard to the used pivot word. The evaluation approach is based on the self-learning approach [60]. The basic mechanism for extracting the contextual meaning is based on the condition that the left and right words to the pivot word must belong to two different languages [61–63]. The statistical approach based on set theory intersection concept is applied here in the proposal for annotating the words. The context word is retrieved on the basis of WX notation technique [58]. The tree representation of learning model for finding the context is represented as follows:

Fig. 8 HLSTM code-mixed cloud representation based on hatred words


Input Text
Main to main market ko jalaunga
मेन ट्र मेन market ko Jalaungaa
मेन ट्र मैं market ko Jalaungaa
मेन तो मैं market ko Jalaungaa
मैं ट्र मैं market ko Jalaungaa
Context variation
मैं तो मैं market ko Jalaungaa
मैं तो मेन market ko Jalaungaa

The model uses the Grapheme- GM and Phoneme model-PM for representation. The estimation of ambiguity in regard to context retrieval is jointly modeled using Eq. 9.

$$S = \lambda_1 \times \mathrm{GM}_{\mathrm{score}} + \lambda_2 \times \mathrm{PM}_{\mathrm{score}} . \qquad (9)$$

Here in regard to Eq. 9, symbol S denotes the word score, λ_1. and λ_2 point to the learning weight provided to the model. The scores of *GM* and *PM* are estimated on the basis of probability. Table 12 describes the accuracy. Figure 9 shows the results obtained for developed HLSTM model. The notation TP signifies number of hatred words. FP denotes number of English words detected wrongly as ambiguous. TN signifies number of hatred words in English. FN denotes number ofncorrectly wds detected as English. Figure 10 depicts the matrix parameters for computing various dimensions of the result.

The dataset [60] has been further used as a baseline to measure the accuracy. The feature extraction and classifiers are used to predict the correct tag in regard to sentence context. The HLSTM gives better precision and BLSTM gives better recall. Figure 11 provides the graphical representation of the result.

Table 12 Evaluation parameters for hatred word detection

Parameters	Detection model					
	HLSTM	BLSTM	CRF	LR	SVM	RF
ENG	**7729**					
HIN	2482					
Hatred	1728					
Proper named Entity	764					
O	104					
Mixed	13					
Un	17					
Total Test word	***12,837***					
Accuracy	***97.49%***	***96.67%***	***94.6%***	***93.6***	***96.57***	***94.8***

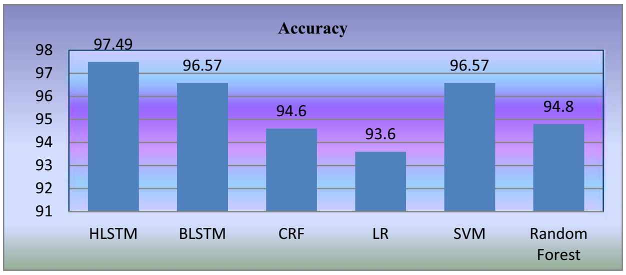


Fig. 9 Accuracy representation

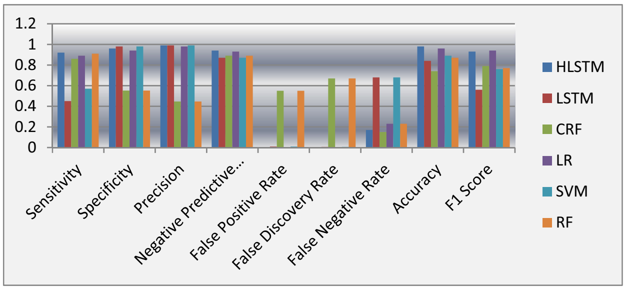


Fig. 10 Parametric evaluation

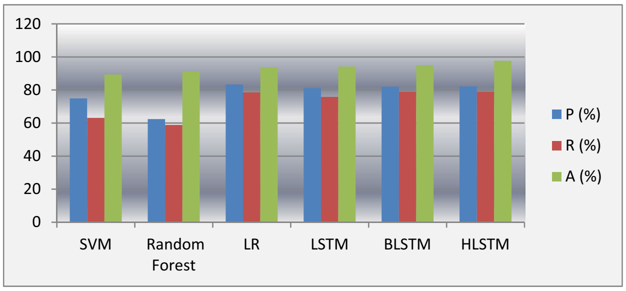


Fig. 11 Precision, recall and accuracy comparison

Conclusions

The objectivity of this work is hereby defined in terms of use of HLSTM based classification. The HLSTM is extension of BLSTM learning model. The use of HLSTM based alignment technique is useful for tagging. The developed HLSTM model is then used for the classification on the basis of voting technique. The main logic behind this is to find multi-lingual features for predicting the language belongingness. The technique context finding using embedding approach suits the model for extracting ambiguity in the sentences. The experiments were organized keeping in mind the lingual features available in the language. The state-of-art techniques in ambiguity detection is studied and compared with the developed approach.

The developed technique is scalable and usable in intent retrieval where intention of the users for using that hatred word in the sentence is not clear. The intent identification helps in understanding the various language models for extracting the context. This intent retrieval in code-mixed sentence helps in solving many linguistic problems related to polysemy. The system is scalable and flexible to carry out other experiments related to other shades of hatred identification available in the form of sarcasm and misinformation.

Declarations

Conflict of Interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Mathew, B, Dutt R, Goyal P, Mukherjee A (2018) Spread of hate speech in online social media. In: Proceedings of the 10th ACM Conference on web science, pp 173–182, 2019
2. Weischedel R, et al (1989) White paper on natural language processing. In: Proceedings of the Workshop on speech and natural language. Association for Computational Linguistics, 1989. pp 481–493
3. Singh VP, Srivastava R, Pathak Y, Tiwari S, Kaur K (2019) Content-based image retrieval-based on supervised learning and statistical-based moments. Mod Phys Lett B 33:1950213
4. Barman U, Das A, Wagner J, Foster J (2014) Code mixing: A challenge for language identification in the language of social media. In: Proceedings of the First Workshop on computational approaches to code switching 2014, pp 13–23
5. Touati R, Messaoudi I, Oueslati AE, Lachiri Z, Kharrat M (2020) New Intraclass Helitrons classification using DNA-image sequences and machine learning approaches. IRBM
6. King B, Abney S (2013. Labeling the languages of words in mixed-language documents using weakly supervised methods. In: Proceedings of the 2013 Conference of the North American chapter of the association for computational linguistics: human language technologies. 2013, pp 1110–1119
7. Nguyen D, Doğruöz AS (2013) Word level language identification in online multilingual communication. In: Proceedings of the Conference on empirical methods in natural language processing 2013, pp 857–862
8. Gella S, Bali K, Choudhury M (2014) "ye word kislang ka hai bhai?" testing the limits of word level language identification. In: Proceedings of the 11th International Conference on natural language processing, 2014, pp 368–377
9. Das A, Gambäck B (2014) Identifying languages at the word level in code-mixed Indian social media text. In: Proceedings of the 11th International Conference on natural language processing 2014, pp 378–387
10. Patel D, Parikh R (2020) Language identification and translation of English and Gujarati code-mixed data. In: 2020 International Conference on emerging trends in information technology and engineering (ic-ETITE), pp 1–4. IEEE, 2020
11. Sequiera R, Choudhury M, Gupta P, Rosso P, Kumar S, Banerjee S, Chakma K (2015) Overview of FIRE-2015 shared task on mixed script information retrieval. In: FIRE Workshops 2015, pp 19–25
12. Vyas Y, Gella S, Sharma J, Bali K, Choudhury M (2014) Pos tagging of English-Hindi code-mixed social media content. In: Proceedings of the 2014 Conference on empirical methods in natural language processing (EMNLP) 2014. pp 974–979
13. Jhamtani H, Bhogi SK, Raychoudhury V (2014) Word-level language identification in bi-lingual code-switched texts. In: Proceedings of the 28th Pacific Asia Conference on language, information and computing 2014, pp 348–357
14. Ethiraj R, Shanmugam S, Srinivasa G, Sinha N (2015) NELIS-named entity and language identification system: shared task system description. In: FIRE Workshops 2015, pp 43–46
15. Qi G, Wang H, Haner M, Weng C, Chen S, Zhu Z (2019) Convolutional neural network-based detection and judgement of environmental obstacle in vehicle operation. CAAI Trans Intell Technol 4(2):80–91. https://doi.org/10.1049/trit.2018.1045
16. Bhargava R, Sharma Y, Sharma S (2016) Sentiment analysis for mixed script indic sentences. In: 2016 International Conference on Advances in computing, communications and informatics (ICACCI) 2016, pp 524–529
17. Sharma M, Singh G, Singh R (2017) Stark assessment of lifestyle-based human disorders using data mining-based learning techniques. IRBM 38(6):305–324
18. Shekhar S, Sharma DK, Sufyan Beg MM (2019) An effective cybernated word embedding system for analysis and language identification in code-mixed social media text. Int J Knowl-Based Intell Eng Syst 23(3):167–179
19. Basavegowda HS, Dagnew G (2020) Deep learning approach for microarray cancer data classification. CAAI Trans Intell Technol 5(1):22–33. https://doi.org/10.1049/trit.2019.0028
20. Tingting Y, Wang Junqian W, Lintai W, Yong X (2019) Three-stage network for age estimation. CAAI Trans Intell Technol 4(2):122–126. https://doi.org/10.1049/trit.2019.0017
21. Bali K, Sharma J, Choudhury M, Vyas Y (2014) I am borrowing ya mixing?" An analysis of English-Hindi code mixing in Facebook. In: Proceedings of the First Workshop on computational approaches to code switching 2014. pp 116–126
22. Shekhar, Shashi, Dilip Kumar Sharma, and MM Sufyan Beg. "Linguistic structural framework for encoding transliteration variants for word origin detection using bilingual lexicon." In 2017 International Conference on Multimedia, Signal Processing and Communication Technologies (IMPACT), pp. 156–160. IEEE, 2017.
23. Remmiya Devi G, Veena PV, Anand Kumar M, Soman KP (2016) (AMRITA-CEN@ FIRE 2016: Code-mix entity extraction for Hindi-English and Tamil-English tweets. In: CEUR Workshop Proceedings 2016, pp 304–308
24. Sapkal K, Shrawankar U (2016) Transliteration of secured SMS to Indian regional language. Proc Comput Sci 78:748–755

25. Zubiaga A, San Vicente I, Gamallo P, Pichel JR, Alegria I, Aranberri N, Fresno V (2016) Tweetlid: a benchmark for tweet language identification. Lang Res Eval 50:729–766
26. Alekseev A, Nikolenko S (2017) Word embeddings for user profiling in online social networks. Comput Sist 21:203–226
27. Zampieri M, Malmasi S, Nakov P, Rosenthal S, Farra N, Kumar R (2019) Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). arXiv preprint arXiv:1903.08983
28. Srivastava V, Singh M (2020) PHINC: a parallel hinglish social media code-mixed corpus for machine translation. arXiv preprint 2004.09447
29. Aina L, Gulordava K, Boleda G (2019) Putting words in context: LSTM language models and lexical ambiguity. arXiv preprint arXiv:1906.05149
30. Bhattacharya P, Goyal P, Sarkar S (2019) Using Communities of words derived from multilingual word vectors for cross-language information retrieval in Indian languages. ACM Trans Asian Low-Resour Lang Inf Process (TALLIP) 18(1):1–27
31. Ajees AP, Mary Idicula S (2019) An improved word representation for deep learning-based NER in Indian languages. Information 10(6):186
32. Mrinalini K, Nagarajan T, Vijayalakshmi P (2018) Pause-based phrase extraction and effective OOV handling for low-resource machine translation systems. ACM Trans Asian Low-Resour Lang Inf Process (TALLIP) 18:1–22
33. Hasan AM, Noor NM, Rassem TH, Noah SAM, Hasan AM (2020) A proposed method using the semantic similarity of WordNet 3.1 to handle the ambiguity to apply in social media text. In: Information science and applications. Springer, Singapore, pp. 471–483
34. Hasan AM, Rassem TH, Noor NM, Hasan AM (2020) A Review of Recent trends: text mining of taxonomy using WordNet 3.1 for the solution and problems of ambiguity in social media. In: Intelligent Computing and innovation on data science. Springer, Singapore, pp 137–152
35. Jadhav SR, Rokade AD, Sable AN, Gade VB (2021) Public hate speech detection using machine learning: a review. Int J 5(12):72–75
36. Shrivastava A, Pupale R, Singh P (2021) Enhancing aggression detection using GPT-2 based data balancing technique. In: 2021 5th International Conference on intelligent computing and control systems (ICICCS), pp. 1345–1350, 2021
37. Shekhar S, Sharma DK, Agarwal DK, Pathak Y (2020) Artificial immune systems-based classification model for code-mixed social media data. IRBM
38. Le NT, Sadat F, Menard L, Dinh D (2019) Low-resource machine transliteration using recurrent neural networks. ACM Trans Asian Low Resour Lang Inf Process 18(2):1–14
39. Pathak Y, Arya KV, Tiwari S (2019) Feature selection for image steganalysis using levy flight-based grey wolf optimization. Multimed Tools Appl 78(2):1473–1494
40. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE Conference on computer vision and pattern recognition 2015, pp 1–9
41. Shekhar S, Sharma DK, Beg MS (2018) Hindi Roman linguistic framework for retrieving transliteration variants using bootstrapping. Proc Comput Sci 125:59–67
42. Sun M, Liu Y, Liu Z, Zhang M (2015) Chinese computational linguistics and natural language processing-based on naturally annotated big data. Springer
43. Shanmugalingam K, Sumathipala S (2019) Language identification at word level in Sinhala-English code-mixed social media text. In: IEEE International Research Conference on smart computing & systems engineering (SCSE) 2019, pp 113–118
44. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems 2013. pp 3111–3119
45. Reddy DA, Kumar MA, Soman KP (2019) LSTM based paraphrase identification using combined word embedding features. In: Soft computing and signal processing. Springer, Singapore, pp 385–394
46. Ramrakhiyani N, Majumder P (2015) Approaches to temporal expression recognition in Hindi. ACM Trans Asian Low-Resour Lang Inf Process (TALLIP) 14:1–22
47. Pathak Y, Sharma K, Singh K, Rana PS (2016) performance study of evolutionary algorithms for structure stability analysis of Al n (n= 2–22). Quantum Matter 5(3):322–329
48. Gupta A, Singh D, Kaur M (2020) An efficient image encryption using non-dominated sorting genetic algorithm-III-based 4-D chaotic maps. J Ambient Intell Humaniz Comput 11(3):1309–1324
49. Kaur M, Kumar V (2018) Adaptive differential evolution-based lorenz chaotic system for image encryption. Arab J Sci Eng 43(12):8127–8144. https://doi.org/10.1007/s13369-018-3355-3
50. Pathak Y, Shukla PK, Tiwari A, Stalin S, Singh S, Shukla PK (2020) Deep transfer learning-based classification model for COVID-19 disease. IRBM. https://doi.org/10.1016/j.irbm.2020.05.003
51. Kaur M, Singh D, Kumar V, Sun K (2020) Color image dehazing using gradient channel prior and guided L0 filter. Inf Sci 521:326–342. https://doi.org/10.1016/j.ins.2020.02.048
52. Singh D, Kumar V, Manjit Kaur V (2020) Classification of COVID-19 patients from chest CT images using multi-objective differential evolution–based convolutional neural networks. Eur J Clin Microbiol Infect Dis 39(7):1379–1389. https://doi.org/10.1007/s10096-020-03901-z
53. http://www.amitavadas.com/ICON2016/ICON_POS.zip. Accessed 14 Apr 2021
54. Narayan D, Chakrabarti D, Pande P, Bhattacharyya P (2002) An experience in building the indo wordnet-a wordnet for Hindi. In: First International Conference on Global WordNet, 2002
55. Shekhar S, Sharma DK, Sufyan Beg MM (2019) Embedding Framework for Identifying Hatred words in Code-Mixed Social Media Text. In: 2019 International Conference on contemporary Computing and Informatics (IC3I), pp. 59–63. IEEE, 2019
56. Sequiera R, Choudhury M, Gupta P, Rosso P, Kumar S, Banerjee S, Chakma K (2015) Overview of FIRE-2015 shared task on mixed script information retrieval. FIRE Workshops 1587:19–25
57. Shanmugalingam K, Sumathipala S, Premachandra C (2018) Word level language identification of code mixing text in social media using NLP. In: 2018 3rd International Conference on Information Technology Research (ICITR) 2018, pp. 1–5
58. Rudra K, Sharma A, Bali K, Choudhury M, Ganguly N (2019) Identifying and analyzing different aspects of English-Hindi code-switching in Twitter. ACM Trans Asian Low-Resour Lang Inf Process (TALLIP) 18:1–28
59. Banerjee S, Chakma K, Naskar SK, Das A, Rosso P, Bandyopadhyay S, Choudhury M (2016) Overview of the mixed script information retrieval (msir) at fire-2016. In: Forum for information retrieval evaluation. Springer, Cham, pp 39–49
60. Bohra A, Vijay D, Singh V, Akhtar SS, Shrivastava M (2018) A dataset of Hindi-English code-mixed social media text for hate speech detection. In: Proceedings of the Second Workshop on Computational modeling of people's opinions, personality, and emotions in social media 2018. pp 36–41
61. Shekhar S, Sharma DK, Sufyan Beg MM (2020) Language identification framework in code-mixed social media text based on quantum LSTM—the word belongs to which language? Mod Phys Lett B 34(6):2050086

62. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780

63. Bernardino HS, Barbosa HJ (2009) Artificial immune systems for optimization. In: Chiong R (ed) Nature-inspired algorithms for optimisation. Springer, Berlin, pp 389–411

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