Knowledge-Based Systems 260 (2023) 110182

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Multitasking of sentiment detection and emotion recognition in code-mixed Hinglish data

Soumitra Ghosh [a], Amit Priyankar [a], Asif Ekbal [a,*], Pushpak Bhattacharyya [b]

[a] Department of Computer Science and Engineering, IIT Patna, India
[b] Department of Computer Science and Engineering, IIT Bombay, India

ARTICLE INFO

Article history:
Received 17 May 2021
Received in revised form 11 September 2022
Accepted 4 December 2022
Available online 7 December 2022

Keywords:
Emotion analysis
Sentiment analysis
Multitasking
Deep learning
Transfer learning
Code-mixed data

ABSTRACT

As the number of non-native English speakers on social media has skyrocketed in recent years, sentiment and emotion analysis on regional languages and code-mixed data has gained traction. Despite extensive research on English, the area of Hindi–English code-mixed texts is still relatively new and understudied. We create an emotion annotated Hindi–English (Hinglish) code-mixed dataset by performing emotion annotation on the benchmark SentiMix dataset to solve this problem and enable future researchers to contribute to this domain. We propose an end-to-end transformer-based multitask framework for sentiment detection and emotion recognition from the SentiMix code-mixed dataset. We fine-tune the pre-trained cross-lingual embedding model, XLMR, using task-specific data to further exploit the efficacy of transfer learning to improve the overall efficiency of our methods. Our proposed multi-task solution outperforms the state-of-the-art single-task and multitask baselines by a considerable margin, implying that the auxiliary task (i.e. emotion recognition) increases the efficiency of the primary task (i.e. sentiment detection) in a multi-task environment. It should be noted that the reported findings were obtained without the use of any ensemble techniques, thereby adhering to a model of effective and production-ready NLP.

1. Introduction

There has been significant growth in the number of users using social media platforms. Twitter has around 187 million daily active users worldwide.[1] In multilingual societies like India, code-mixing is one of the most popular forms of communication. Most of the Hindi speaking users frequently use Hindi vocabulary along with English in their writings. This notion of mixing two different languages to emphasize the expression is known as code-mixing [1]. Sentiment classification of code-mixed text is useful in contexts such as socially or politically motivated debates, the spread of false news, and so on. Below is a code-mixed tweet where Hindi words are written in their transliterated form:

> "*@narendramodi It feels awesome to see u as a PM once again ... desh apke sath hai. desh ki ummide aapke sath hai Jai Hind*"
>
> English Translation: @narendramodi It feels awesome to see u as a PM once again ... The country's hopes are with you Jai Hind.

In the above sentence, the sentiment is *positive* and the emotion which is coming out of it is *joy*. We can observe that emotion is a more evolved form of sentiment. A positive sentiment could be inferred from *happy* or *surprised* emotion.

While sentiment analysis and opinion mining are often used interchangeably, opinion mining is an application that uses sentiment analysis and contextualizes polarity scores in subjects, facets, and goals. Sentiment detection is the task where a sentence or text is classified into negative, positive and neutral (sometimes) classes based on the polarity of sentences. Emotion detection is the process of determining the emotion which is being generated from the texts. Unlike the sentiment detection task, identifying emotion is a more complex challenge because the distinctions between certain emotion types are more subtle than the differences between positive and negative emotions. Emotions generally occur to the response of some event, news or remembering past. Ekman came up with the six basic emotions, *viz.* happiness, sadness, disgust, surprise, fear and anger [2], and these are extensively used for several emotion recognition tasks [3,4].

Conversations in the style of code-mixing through different channels, such as social media, online games, online product reviews, and so on, makes it difficult to decipher the text's sentiment. Adding to that, there are various other challenges when working with code-mixed texts such as ambiguity in language

* Corresponding author.
E-mail addresses: ghosh.soumitra2@gmail.com (S. Ghosh), amitpriyankar22@gmail.com (A. Priyankar), asif@iitp.ac.in (A. Ekbal), pb@cse.iitb.ac.in (P. Bhattacharyya).

[1] https://www.statista.com/statistics/242606/number-of-active-twitter-users-in-selected-countries/.

https://doi.org/10.1016/j.knosys.2022.110182
0950-7051/

identification, noisy text, etc. Recent advances in natural language processing tools such as pre-trained transformer-based language models which uses sentencepiece and wordpiece tokenization methods performs significantly well on reducing some of the above mentioned challenges. The welcoming nature of such social-media forums encourages people from all walks of life to express their thoughts on a subject. As a consequence, code-mixed data processing will likely reveal the true feelings about a subject. Despite this widespread enthusiasm for customer input, code-mixed content in under-resourced languages receives little exposure.

To the best of our knowledge, this is the first study to discuss the sentiment detection challenge on code-mixed Hinglish data in a multitasking environment, with emotion recognition as a supporting task. Our proposed method is based on state-of-the-art cross-lingual contextual embeddings, XLM-RoBERTa (XLMR) [5] and transfer learning approach. To tackle the problem of scarce availability of relevant code-mixed data, we create gold labels for emotion by manually annotating 20,000 Hinglish instances of the benchmark SentiMix dataset. We further employ transfer learning to improve our model's performance on the sentiment detection task. We compare the per-task efficiency of our top-performing multitask method to that of the single-task models. Our method is compared to some state-of-the-art architectures developed exclusively for emotion and sentiment analysis in code-mixed texts. On the sentiment detection task, our best model achieves an F1 score of 71.61%, which is 3.30% higher than the single-task model and 1.93% higher than the state-of-the-art multitask method on comparable tasks. The proposed method attains noteworthy improvements of 7.14% and 1.18% on the emotion task over the single-task and best performing multitask baselines, respectively.

The following are the key contributions of our proposed work:

- This is the first study to look into the role of emotion in improving sentiment detection efficiency in code-mixed Hinglish results. To detect sentiment and recognize emotion in code-mixed Hinglish messages, we suggest an end-to-end cross-lingual word embeddings-based multitask system.
- We produce a good quality emotion annotated code-mixed Hindi–English (Hinglish) dataset by labelling 20,000 code-mixed instances of the benchmark SentiMix Hinglish dataset with emotion labels from Paul Ekman's basic emotions.
- We employ an effective transfer learning strategy to take advantage of abundantly available monolingual Hindi and English sentiment annotated datasets to improve our proposed method's overall performance on scarcely available code-mixed Hinglish data on the Sentiment task.
- We provide important resources to the community: the codes, the emotion annotated code-mixed Hinglish dataset, and the trained multitask models will be freely available to everyone[2] interested in working on sentiment and emotion analysis in code-mixed data using a similar approach.

The remainder of the paper is organized in the following manner. Some of the previous works on this domain are described in Section 2. In Section 3, we go through the dataset construction and annotations in great details. We explain our proposed approach for multitasking studies in Section 4. In Section 5, we discuss the experiments we did and the results we got. Finally, in Section 6, we wrap up our work and describe the direction of future work.

[2] Download link: http://www.iitp.ac.in/~ai-nlp-ml/resources.html#EmoSen.

2. Related work

With the rise in popularity of code-mixing on social media platforms, there has been an increase in interest in researching the various nuances of code-mixing. Sentiment study of code-mixed languages is a hot topic in academia. Initially, traditional machine learning techniques were employed for the sentiment detection task of monolingual text. The authors in [6] experimented with a machine learning-based approach, Support Vector Machine (SVM) for sentiment analysis. They also used graph-cut techniques on WordNet synonym graph to improve the classification accuracy.

Authors in [7] introduced Recursive Neural Tensor Networks for classifying a single sentence into positive/negative labels. For Hindi–English sentiment analysis, many systems used traditional machine learning models like SVM, Naive Bayes and Random Forests for classification in the shared task, SAIL-2015 [8]. The best method used n-grams at the word and character level as features and an SVM to classify the sentiment. Sentiment analysis of social media posts was experimented with by [9]. After pre-processing the data and removing noises, the authors employed a multi-layer perceptron to perform the polarity detection task. The problem of hate speech detection was analysed by [10] in a Hindi–English code-mixed dataset comprising of tweets. For classifying tweets into hate-speech and natural speech, they used supervised machine learning systems such as SVMs and Random Forests.

[11] have studied the sentiment detection of English–Hindi code-mixed dataset and introduced the sub-word level representations in Long Short Term Memory (LSTM) to learn the details of sentiment value. They have also investigated the challenges of misspellings in transliterated Hindi. [12] proposed a joint multitask learning framework that handles the multiple complex natural language processing (NLP) tasks by successively growing its depth. A real-time system for Twitter sentiment analysis was proposed in [13]. Their system architecture consisted of two major steps: pre-processing of the data and sentiment prediction. Pre-processing involved tokenization which correctly handled URLs, common emoticons, phone numbers, HTML tags, Twitter mentions and hashtags, numbers with fractions and decimals, repetition of symbols and Unicode characters. For the sentiment detection part, they used the statistical methods. A deep learning model based on Bi-LSTM was experimented with by [14] for word-level language identification in transliterated Hindi and English data. They showed that the word embedding model performs better as compared to the character embedding model. In the work, [15], a multi-input multi-channel transfer learning-based model was presented to detect the offensive tweets in Hinglish. The proposed model used multiple embeddings and parallel CNN-LSTM architecture which out-performed the naive transfer learning models.

Sentiments and emotions are closely related. Most emotional states have a clear distinction of being a positive or negative situation. Emotional states e.g. 'anger', 'fear', 'disgust', 'sad' etc. belong to negative situations, whereas 'happy' and 'surprise' reflect positive situations. Multi-task learning (MTL) framework targets to achieve generalization by leveraging the inter-relatedness of multiple problems/tasks [16]. In a typical MTL scenario, the same input representation is used for several supervised learning tasks or outputs. MTL aims to leverage the interdependence among multiple correlated tasks to increase the confidence of individual tasks in prediction. The authors [17] observed empirically that performance of the sentiment detection task can be improved by learning it jointly with the emotion detection task. Similar mutually benefiting performances have been observed in findings of several studies [18,19] on joint learning of sentiment and

emotion detection tasks. Ghosh et al. [20] developed a multi-task framework for emotion detection from suicide notes as the primary task and depression detection and sentiment detection as auxiliary tasks. Overall results for the primary task of emotion recognition indicated that depression and sentiment information helped in improving the predictive performance of the emotion recognition task in a multitasking scenario. MTL also provides the benefit to develop only one unified model, in contrast, to separate models for each task in a single-task setting, resulting in reduced complexity in terms of learnable model parameters. Motivated by the association of sentiment & emotion and the advantages of the multi-task learning paradigm, we presented a multi-task framework that jointly learns and classifies the sentiments and emotions from code-mixed texts.

Multitask systems that leverage emotion labels of texts have also been attempted for sentiment detection. [21] proposed a two-layered multitask attention-based on Bidirectional Long Short-Term Memory neural networks that perform sentiment detection through emotion analysis. For stance detection in Hindi–English code-mixed language, [22] introduced a multitask Learning-based Deep Learning architecture. [23] integrated multitasking paradigm to a BERT-based model which classifies texts from 3 different languages: Hindi, Bengali and English into different aggression classes. Their proposed model uses an attention mechanism on the top of the BERT followed by fully connected layers to do the classification. [24] investigated transfer learning-based approach using cross-lingual embeddings for sentiment classification of Hindi–English code-mixed tweets. Their results show that incorporating cross-lingual embeddings improves the accuracy against a monolingual approach. For emotion classification of code-mixed Hindi–English tweets, [25] proposed a candidate sentence generation and selection-based method on top of the Bi-LSTM.

Ensemble approach is based on the hybridization of various classical machine learning algorithms like Naive Bayes, SVM, Linear Regression, and SGD classifiers were experimented in [26]. Their method provided decent results for code-mixed Hindi–English text. [27] experimented with various deep learning-based approaches to detect the emotion of code-mixed (Hinglish) tweets. They also tested transformer-based architectures such as BERT, RoBERTa, and ALBERT, and discovered that the BERT model outperforms all the others. [28] analysed the emotion detection of code-mixed Hindi–English tweets. The authors presented a code-mixed Hindi–English corpus of tweets and experimented with supervised machine learning algorithms to detect the emotions. [29] presented the process of creating a code-mixed Hindi–English dataset from social media for sentiment analysis. Authors in the paper included inconsistent spellings and ambiguous meaning words in their study. [30] introduced a gold standard corpus for sentiment analysis of code-mixed Malayalam–English.

Transformer models have been used successfully for a variety of NLP tasks. Since the majority of the pre-trained transformer models were trained on English data, the majority of the tasks were based on the English language. Even though there were some multilingual models, such as Multilingual BERT (mBERT) [31], there was much debate regarding its ability to represent all languages [32], and although mBERT displayed some cross-lingual characteristics, it was not trained on cross-lingual data [33]. The recently published cross-lingual transformer models — XLMR [5], which has been trained on 104 languages, provided the impetus for this methodology. XLMR has the unique property of being very compatible in monolingual benchmarks while still delivering the highest performance in cross-lingual benchmarks [5].

Table 1
Class-wise distribution of sentiment labels for train, test and validation splits.

Split	Total	Positive	Neutral	Negative
Train	14000	4634	5264	4102
Test	3000	1000	1100	900
Validation	3000	982	1188	890

3. Corpus details

The dataset used in this paper was introduced in Task 9 of the SemEval 2020 shared task [34]. The authors have released a code-mixed Hindi–English (Hinglish) corpora annotated with sentence level sentiment labels. The classified labels are *negative*, *neutral* and *positive*. Tweets expressing joy, praising or applauding a person, group or country were considered as *positive* whereas attack on a group, person or product, criticism, abuse were considered as *negative*. Tweets containing facts, news or advertisements were labelled as *neutral*. We performed emotion annotations on the entire dataset to enhance the utility of this dataset and meet the scarcity of available code-mixed Hinglish emotion annotated data.

3.1. Corpus statistics

The dataset shared in the shared task is divided into three distinct splits: train, test and validation containing 14,000, 3000 and 3000 instances, respectively. Table 1 shows the distribution of train, test and validation split for each class. For each split, the neutral class is in a slight majority in the dataset, but still, the dataset is nearly balanced. Each tweet contains the Twitter handle of the user(s), URL link to the tweet and sometimes the "RT" keyword which denotes retweet. The average sentence length is 134.9 characters. In the training set, the average sentence length is 136.9 characters and the vocabulary size is 60,115. With an average sentence length of 127.7 characters, the validation set has a vocabulary size of 19,499. The test set has an average sentence length of 129.9 characters and a vocabulary size of 19,331 words.

3.2. Annotation setup for emotion labels

Three annotators were asked to review each instance of the dataset, with each sentence bearing no more than one emotion from Ekman's basic emotions [35], *viz. anger, disgust, fear, happiness, sadness, surprise*. The necessity to add another class arose when the annotators encountered instances that have no emotion or some non-neutral emotion that do not fall into the scope of Ekman's basic emotions. We name this class as *others*. Initially, while starting the annotation process, we labelled the class for the non-emotive instances as *Neutral*. However, in due time we observed certain instances, even though very few, contains some emotion[3] which do not fall under the scope of the Ekman's six-basic emotions. So to avoid any confusion and/or mis-interpretation of the annotation scheme, we renamed the *Neutral* class of *Others*. Also, adding an extra class by considering separate classes for *Others* and *Neutral* classes would increase the data sparsity problem that is already existent among the various emotion classes. Table 2 shows the distribution of instances over the various emotion classes.

Inter-annotator agreement: Finding agreement among the various annotators is an important task in any annotation task involving multiple raters to produce a reliable annotated dataset.

[3] Such as anticipation, optimism, sarcasm, etc.

Table 2
Class-wise distribution of emotion labels for train, test and validation splits.

Split	Train	Test	Validation
Anger	2095 (14.96%)	680 (22.67%)	415 (13.83%)
Disgust	1048 (7.49%)	105 (3.5%)	148 (4.93%)
Fear	56 (0.4%)	13 (0.43%)	4 (0.13%)
Joy	3893 (27.81%)	1008 (33.6%)	973 (32.43%)
Sadness	856 (6.11%)	122 (4.07%)	307 (10.23%)
Surprise	51 (0.36%)	7 (0.23%)	6 (0.2%)
Others	6001 (42.86%)	1065 (35.5%)	1048 (34.93%)
Total	14000	3000	3000

Table 3
Average per-class agreement among the annotators.

Class	Anger	Disgust	Fear	Joy	Sadness	Surprise	Others
Score	0.71	0.65	0.74	0.90	0.82	0.58	0.86

Table 4
Sample instances from the training set. ET: English Translation.

Tweet	Sentiment	Emotion
@theskindoctor13 Happy Birthday Doctor sahab bhagwaan aapko khush or swasth rkhe or aap hme creativity se hansaate rhe aise hi apni ET: *Happy Birthday Doctor Sir. Wish God keeps you happy and healthy and you continue to make us laugh by your creativity.*	Positive	Joy
EVRYONE SHUT THE FUCK UP https//t.co/RINOmqkPsZ	Negative	Anger
@RailwaySeva @dr _abhi_voice @drm_kir I proud where I'm from love you modi ji finally you make digital india	Neutral	Joy
Seen karke log mujhe ignore karty msgs https//t.co/Yfop5lks45 ET: *People ignore me after seeing msgs*	Negative	Sadness
@sunoo_chanda make a gentle and nice relation with someone. life me kisi k sahare ki zarort parti ha ET: *@sunoo_chanda make a gentle and nice relation with someone. Need someone's support in life*	Neutral	others

Cohen's Kappa coefficient [36] is one such metric that is considered a reliable measure for evaluating inter-annotator agreement. It is defined as:

$$K = \frac{Pr(a) - Pr(e)}{1 - Pr(e)}$$

where $Pr(a)$&$Pr(e)$ are the observed and by chance agreement among raters. Our dataset has an average agreement of 0.75, indicating that the annotated dataset is of substantial quality. Table 3 shows the average agreement among the annotators for each emotion class. We observe that classes like *fear*, *joy* and *others* have better agreement than the classes like *anger*, *disgust* and *surprise*. The low scores of the classes *anger* and *disgust* can be attributed to the fact that they are very closely related emotions, which is also realized in Plutchik's wheel of emotions [37]. Class *surprise* may consist of the instances from both *positive* and *negative* polarities. Also, implicit *surprise* instances were often marked as *others* by some annotators. The gold-label annotations are determined by a majority vote on the three annotators' annotations. In case of instances where all three annotators have marked with a different emotion label, a fourth annotator was employed to resolve the conflict. Table 4 shows some sample instances from the emotion and sentiment annotated SentiMix dataset.

3.3. Difficult examples

- **Sarcastic Instances**
  The annotators faced certain level of difficulty in tagging instances which were sarcastic in nature or were implicit in nature. Consider the following instances:

  1. @rishav_sharma1 Nahi vaha log itne educated hai ki direct ISIS me hi placement lete hai.
     English translation: *@Rishav_Shame1 No those people are so educated that they take placement in direct Isis only.*
  2. RT @awarastic Pakistan ko kisi ne btaya nahi t20 world cup nahi h ye .. jo 20 over me hi ludak gaye ..
     English translation: *RT @awarastic Did no one tell Pakistan that this is not a t20 World Cup that they wrapped up within 20 overs ..*

- **Context Dependent Tweets**

  1. @Shankar27273 Enko Italywali MAA ne Bataya! Bade Dynasty se hota hai Karma Se Nahi.
     English translation: *He was told by his Italian Mother! You become big by dynasty, not by karma.*
  2. @not_dat_guy Ek article aya tha jisme likha tha Mission Mangal mega budget film hai lmao
     English translation: *One article came that read Mission Mangal is a big budget film lmao*

  In the 1st tweet, to label the emotions it is required to know the background of the person who is being referred here. Similarly, in the 2nd tweet, we can see that some extra information is needed to detect the emotion.

4. Methodology

The key concept behind our methodology is to fine-tune a pre-trained cross-lingual transformer model, XLMR, on a task-specific dataset, then pass the saved weights of the encoder module to the proposed multitask classification model, which is trained on the SentiMix dataset.

Problem definition: Given a code-mixed Hinglish tweet (x), classify the Emotion (e) of the tweet as ϵ {*anger, disgust, fear,*

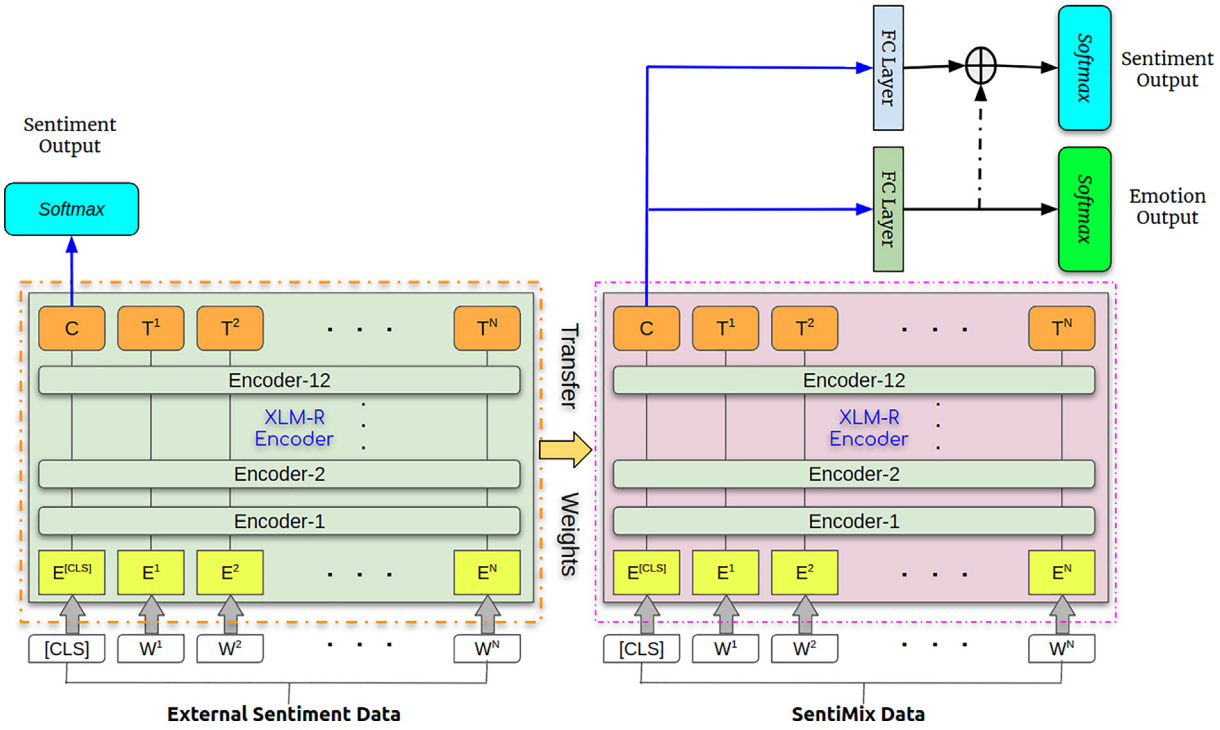


Fig. 1. Architecture of the Transfer Learning-based XLMR Multitask Learning Framework.

joy, sadness, surprised and others} and Sentiment as $s \in$ {*positive, negative, neutral*}. Given a training set $\{(x_1, e_1, s_1), \ldots, (x_n, e_n, s_n)\}$ of n samples, the network is trained to minimize the following multi-task loss function:

$$E(\theta) = \lambda_{emo} E_{emo}(\theta) + \lambda_{sent} E_{sent}(\theta), \quad (1)$$

$E_{emo}(\theta)$ and $E_{sent}(\theta)$ in the above equation represents the emotion-class loss and sentiment-class loss, respectively. The terms λ_{emo} and λ_{sent} represent the weighting coefficients for the emotion and sentiment losses, respectively. θ denotes the network's trainable parameters.

In Section 4.1, we describe the general classification architecture using XLMR. We define a transfer learning-based multitask method for detecting sentiment and emotion simultaneously in Section 4.2. Fig. 1 depicts the overall architecture of our proposed solution.

4.1. XLMR for text classification

The XLMR transformer architecture, like other transformer architectures, can be used to perform sequence classification tasks [5]. We use the XLMR-base model which is trained in 100 languages. It has 12 encoder layers, 768 hidden states, 12 attention heads, feed-forward layer dimension of 3072 and 270M parameters. Unlike XLM which is trained on Wikipedia dumps, the XLMR is trained on filtered CommonCrawl data.

It takes a sequence of up to 512 tokens as input and returns the representation of the sequence. The corresponding token sequence of N words for an input tweet is shown as below:

$$[CLS], W^1, W^2, W^3, \ldots, W^N, [SEP],$$

where the [*CLS*] token is inserted in the starting position of the sequence which is used as an indicator of the start of the sentence. The [*SEP*] is a symbol that distinguishes one series from the next and indicates the end of a statement. The token embeddings are denoted by E^i, the hidden vector for each ith term in the input utterance is denoted by T^i, and the final hidden vector of the special [*CLS*] symbol is denoted by C. The contextualized sentence representation (h) is formed by the representation of [C]. The bidirectional nature of XLMR guarantees mutual conditioning on both the left and right contexts of a token. In an ideal single-task classification task, to predict the likelihood of sentiment label s we apply a simple softmax classifier on top of XLMR, as seen in Eq. (2), where θ is the task-specific parameter matrix.

$$p(s|h) = softmax(\theta h) \quad (2)$$

The XLMR output is passed through two task-specific dense layers (one for each sentiment and emotion task), followed by their respective output layers with softmax activation, to fit our model to a multitask environment. Also, we linearly concatenate ($\bigoplus$) the task-specific sentiment features with the task-specific emotion features to form sentiment-aware emotion features. The rationale behind performing this step is to exploit the underlying co-relatedness among the two tasks of sentiment and emotion.[4] This is realized as follows:

$$p(s|h_1) = softmax(\theta_1 h_1) \quad (3)$$

$$p(e|h_2) = softmax(\theta_2 h_2) \quad (4)$$

h_1 and h_2 represent the outputs from the task-specific dense layers for sentiment and emotion, respectively. We fine-tune all the parameters from XLMR (shared parameters) as well as θ_1 and θ_2 (task-specific) jointly by maximizing the log-probability of the correct sentiment and emotion label.

4.2. Transfer Learning-based XLMR Multitask Learning (TL − $XLMR_{LS}^{MTL}$)

Although XLMR is quite popular for its superior performance with a variety of cross-lingual tasks, we perform task-specific

[4] Also, we empirically observed that this setup produces better emotion output score than without the sentiment knowledge.

fine-tuning on a sentiment dataset which we prepare by combining multiple other publicly available datasets.

We prepare our external sentiment dataset for fine-tuning by consolidating data from the following four sources:

1. Domain-specific Twitter data: This is an in-house domain-specific Twitter corpus developed as a part of a project which consists of 9088 tweets in Devanagari Hindi. This dataset is annotated with Ekman's basic emotions plus a non-emotive class *Others*. We associate each emotion to their corresponding sentiment label and obtain the following distribution: Positive: 2698; Negative: 3513; Neutral: 2877.
2. Event extraction dataset from disaster domain: This dataset [38] contains emotion-annotated 3847 instances collected from various news website articles on the disaster domain. The instances were annotated as per Plutchik's Basic Emotion which we map to respective polarity label achieving the following distribution: Positive: 782; Negative: 2628; Neutral: 437.
3. Sentiment analysis dataset: We collected 9080 sentiment annotated instances (Devanagari Hindi) from Github (User: sid573).[5] The distribution over the sentiment classes are as follows: Positive: 3255; Negative: 3175; Neutral: 2650.
4. Movie reviews dataset: Fetched from Github (User: subham721),[6] this dataset consists of 1018 Hindi (Devanagari) movie reviews (Positive: 516; Negative: 502) as well as 10662 English instances (Positive: 5,331; Negative: 5,331), all annotated with sentiment labels.

First, we train the XLMR classification model on the consolidated data and we save the weights of the entire model. Next, to train a classifier on the shared task data, we tweak the above architecture by removing the single output layer and add two task-specific dense layers after the XLMR encoder. We pass the output from the XLMR encoder to the task-specific dense layers which finally passes through two task-specific output layers. We transfer the previously saved weights of only the XLMR-module from the fine-tuned model to initialize the weights of the XLMR-module in the tweaked architecture.

5. Experiments, results and discussion

In this section, we go through the specifics of our experiments as well as our thoughts on the outcomes.

5.1. Model parameters

We use the Huggingface[7] Transformers package to import the pre-trained XLM-R model and also used Keras[8] and Scikit-learn[9] libraries at different stages of our implementation. All the experiments have been performed on a GeForce GTX 1080 Ti GPU. In the multitasking experiments, we weigh the losses from the Sentiment and Emotion tasks, using the *loss_weights* parameter of Keras's[10] *compile* function. We implemented a simple Grid Search algorithm and iterated over the following set of values for each task : [0.1, 0.3, 0.5, 0.7, 0.9, 1]. We obtained best results when the loss weights are set as 0.3 for both the sentiment and emotion tasks. We set the input sentence length to 128 for all the developed models. The validation set's best model was preserved for testing. Table 5 shows the details of various hyper-parameters related to our experiments.

5 https://github.com/sid573/Hindi_Sentiment_Analysis.

6 https://github.com/shubham721/Sentiment-Analysis-On-Hindi-Reviews.

7 https://huggingface.co/transformers/.

8 https://keras.io/api/.

9 https://scikit-learn.org/.

10 https://keras.io/api/models/model_training_apis/.

5.2. Results

As demonstrated in Table 6, it is worth noting that language model fine-tuning and label smoothing provided noticeable improvements over direct classifier fine-tuning. The precision and weighted F1-score for our different models are shown in Table 6. Our best model attains an F1 score of 71.61% on the emotion detection task, which is 3.30% higher than the single-task model and 1.93% higher than the state-of-the-art multitask approach on comparable tasks. On the emotion task, the suggested approach outperformed the single-task and highest performing multitask baselines by 7.14 and 1.18 points, respectively. To examine the importance of the sentence encoder (in our case XLMR), we conduct an ablation experiment by replacing the XLMR module with Multilingual BERT (mBERT). We report the results in Table 6. We observe considerable fall in scores for both the tasks, depicting superior capability of the cross-lingual XLMR pre-trained model to encode code-mixed data than the multilingual mBERT model. Our observations are also consistent with the past findings [41]. Fig. 2 shows the confusion matrices for the sentiment (Fig. 2(a)) and emotion recognition tasks (Fig. 2(b)). We observe that the correct classification counts for each emotion class resonates the representation of samples of that class in the overall dataset (as shown in Table 2). Also, majority of the instances of the disgust class are mis-classified as anger, which is a very closely related class as depicted in Plutchik's Wheel of Emotions [37] (adjacent petals). Our proposed system's results are statistically significant[11] with the next best performing baseline when tested against the null hypothesis with $p < 0.05$. Also, the improvement in results of the developed $XLMR_{LS}^{MTL}$ model than the task-specific single-task systems are found to be statistically significant with $p < 0.05$.

5.2.1. Comparison with state-of-the-arts

Our proposed multi-task system outperforms the various state-of-the-art systems on the sentiment detection task by significant margins. Our model is shown to outperform the system introduced in [39] by 2.61 F1-points. This baseline employs both fine-tuning and label smoothing as similar to our approach. In addition, they considered the rule-based features to boost their system performance, however, our multitask transfer learning approach proved to be better in attaining the overall objective. The single-task mBERT-based system experimented in [40] could attain a 69.15% F1-score, which is 2.46% higher than our system, which shows the efficacy of cross-lingual transformer-based model such as XLMR over the traditional BERT-based model. The authors in [40] proposed their final model which is a multitask system developed on top of XLMR and achieved significant performance improvement from the mBERT model. Here also, our developed system outperforms the aforementioned model with an improvement of 1.93 points.

5.3. Qualitative analysis

Experimental results have shown that the multi-task system with transfer learning has significantly improved the result of the sentiment detection task. This improvement can be justified by the fact that emotions such as *sadness, anger, and disgust* are closely associated with *negative* emotion. Similarly, the emotions like *joy* and *surprise* (sometimes) are associated with positive emotion. In Table 7, we can see that the TL + MTO system can classify the sentiment correctly, whereas STO and MTO could not. The prediction of Table 6 Sentence 1 as positive by STO is a little surprising. We passed various segments of the Tweet separately

11 We performed Student's t-test for the test of significance.

Table 5
Details of various hyper-parameters related to our experiments.

Parameters	Details
Task-specific layers	All developed multitask models have 1 fully-connected layer (768 neurons) each for Sentiment and Emotion tasks
Output layer(s)	*Single-task models:* 1 output layer each (3 and 7 neurons for Sentiment and Emotion tasks respectively) *Multitask models:* 2 output layers each representing Sentiment (3 neurons) and Emotion task (7 neurons)
Hidden activation	*ReLU* (Dense layers)
Output activation	*Softmax* (for both Sentiment and Emotion)
Batch size	16
Epochs	10 epochs for all models except proposed. For proposed, 30 epochs during transfer learning; 10 epochs during training on task data
Dropout	0.5
Loss	*Categorical CrossEntropy* (for both Sentiment and Emotion)
Loss weights	[0.3, 0.3] for $TL - XLMR_{LS}^{MTL}$ model
Optimizer	*Adam*

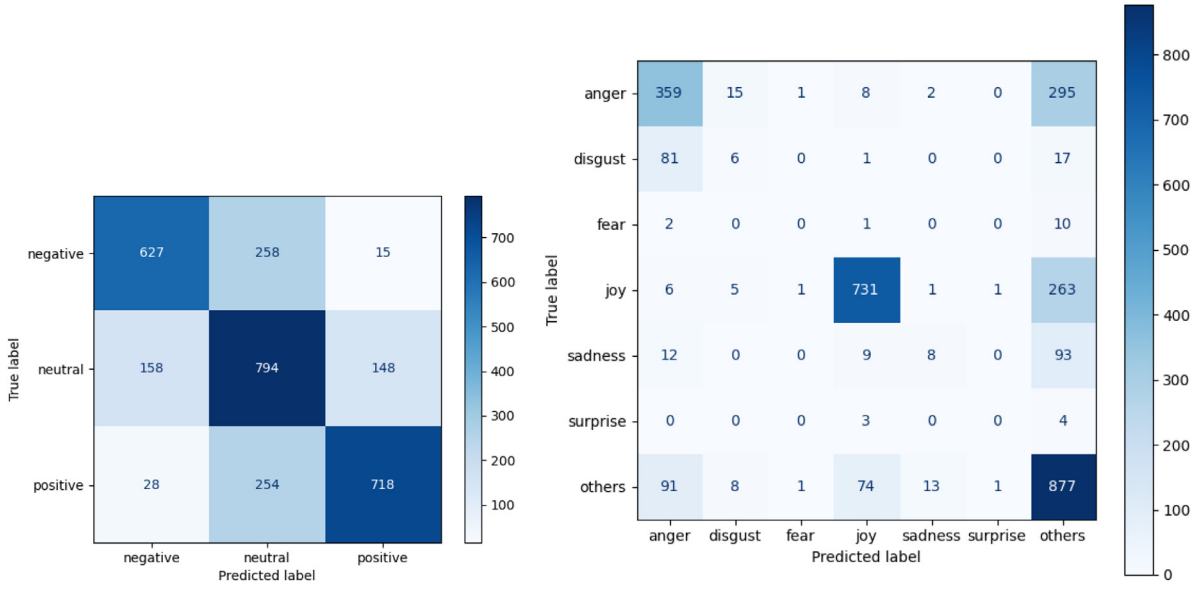
Table 6
Accuracy and weighted F1-score of different systems are shown. Values in bold shows the highest attained score for a particular metric. STL: Single-task learning; MTL: Multitask learning; LS: Label Smoothing; FT: Fine-tuning; RF: Rule-based Features; Emo: Emotion; Sent: Sentiment.

Tasks	Sentiment		Emotion	
Models	Accuracy (%)	F1 (%)	Accuracy (%)	F1 (%)
Developed baselines				
$XLMR^{STL-Sent}$	68.37	68.31	–	–
$XLMR_{LS}^{STL-Sent}$	69.16	69.2	–	–
$XLMR^{STL-Emo}$	–	–	60.26	56.33
$XLMR_{LS}^{STL-Emo}$	–	–	62.57	58.17
$XLMR_{LS}^{MTL}$	70.47	70.48	65.2	62.29
State-of-the-Art Baselines				
$XLMR_{FT+LS+RF}^{STL}$ [39]	–	69	–	–
$mBERT^{STL}$ [40]	68.66	69.15	–	–
$XLMR^{MTL}$ [40]	69.05	69.68	–	–
Proposed system				
$TL - XLMR_{LS}^{MTL}$	**71.3**	**71.61**	**66.03**	**64.47**
Ablation experiments				
$TL - mBERT_{LS}^{MTL}$	69.17	69.55	63.33	61.43

Table 7
Sample predictions where multitask system outperformed the Single-Task Systems. Here, STO: Single-task Output; MTO: Multitask Output; TL: Transfer Learning. ET: English Translation.

Tweet	Actual	STO	MTO	TL + MTO
@Saurabh_MLAgk Modi is so bad. Making ppl work n sweat. Aur @TajinderBagga tumhe AAP samaj kabhi maaf nahi karega	Negative Anger	Positive –	Neutral Anger	Negative Anger
ET: *Modi is so bad. Making ppl work n sweat. And @TajinderBagga the AAP society will never forgive you.*				
@Sicilian__Mafia Bhaai Engineers sirf love hi krte hain. Wo love marriage mai convert naaai hoti	Neutral Sadness	Negative –	Negative Sadness	Neutral Sadness
ET: *Engineers do only love. That love does not convert into marriage.*				
@yupptv Inke hulchul se hota hai screen par hungama aur performance par inki bolta hain Oh My God zamaana! As #Bollywood's iconic	Positive Joy	Negative –	Neutral Neutral	Positive Joy
ET: *@yuppatav Their hustle and bustle causes ruckus on the screen and on their performances they say Oh My God generation! As #Bollywood's iconic*				
Debates me aakar bhi kya ukhad lete hai fake news ki stories pe agenda chalate hai r tameez to hai Hi nahi kisi	Negative Anger	Neutral –	Negative Anger	Negative Anger
ET: *Even after coming in debates, what do you uproot? Agenda runs on the stories of fake news and there is no good*				

through the single-task sentiment model to better understand the model's behaviour. The predicted sentiment is negative when we pass '*Modi is so bad.*'. When we pass '*Making ppl work n sweat.*', the predicted sentiment is Positive. The predicted sentiment is Neutral when we pass '*Aur @TajinderBagga tumhe AAP samaj kabhi maaf nahi karega*'. This shows the single-task model performs



(a) Sentiment task (b) Emotion task

Fig. 2. Confusion matrix of the two tasks.

Table 8
Sample predictions where multitask system is not able to predict the sentiment correctly. ET: English Translation.

Tweet	Actual	STO	MTO	TL + MTO
@akki_dhoni @_iRajeev You are stalking me bro. Just shut up and nikal. Le. Faltu logo ki mention mein jagah nahi haii ET: *You are stalking me bro. Just shut up and get out. Take. There is no place in the mention for useless people.*	Negative Disgust	Negative –	Neutral Sadness	Neutral Anger
I found this awesome recording of Aawaz deke hamein tum bulao on #Smule ET: *I found this awesome recording of Lend me your voice and call out to me #Smule*	Neutral Joy	Neutral –	Positive Joy	Positive Joy
@nirahua1 @msunilbishnoi Wah bhai Shandaar jabab ET: *Nice brother great answer*	Positive Joy	Positive –	Neutral Neutral	Neutral Neutral

comparatively better when segments of the Tweet are presented to it individually and not as a whole. Studying the predicted sentiment labels closely on this Tweet, we can notice that the model finds it difficult to comprehend the Tweet's code-mixed segment (last segment). Also, in the middle segment, the model possibly misinterprets the words *work* and *sweat* in the given context to be Positive, which ideally should have been Negative or Neutral. We observe from the performance of the MTO and our proposed model how the predicted sentiment labels improve gradually on the same tweet, indicating the importance of multi-tasking and transfer learning. Furthermore, the multi-task systems MTO and TL + MTO can also predict the emotion classes correctly, along with the sentiment (except sentence 3, where TL + MTO outperforms MTO). In sentence 2, we can see that the TL + MTO system is correctly detecting the *neutral* sentiment. Keeping Bert's pre-trained parameters trainable during the process outperformed freezing BERT parameters during fine-tuning. This is attributed to data disparity in data distributions of BERT pre-training and Sentimix tasks.

5.3.1. Error analysis

It has been observed that our models are facing difficulty in classifying neutral and non-neutral classes. This is the reason why the accuracy of the *neutral* class is very low compared to the accuracies of *positive* and *negative* class. Table 8 shows some misclassified instances by the multitask systems for the sentiment task. For comprehensive understanding of the multitask systems performance, we also present the actual and predicted emotion classes. For sentence 1, the multitask models predicted neutral sentiment but the actual sentiment is negative since the user who wrote the tweet was feeling bothered or slightly anger. Also, the predicted emotion by both the multitask systems is neutral. In sentence 2, the multitask systems predicted the positive sentiment which is correct since the user is feeling good about something and the emotion predicted is joy which is also correct. Here, the gold annotation for sentiment is wrong, however the emotion label annotated by us is correct. In sentence 3, the multitask systems failed to distinguish between neutral and positive sentiment. The predicted emotion by the multitask systems was neutral, which can be due to the lack of the sufficient contextual information.

The two main obstacles in defining emotions in Hindi–English code-mixed data are dealing with the linguistic problems that come with code-mixed data and the lack of clean data. As a result, even more, class-specific cleaner data would be needed to

reduce the effects of noise created by spelling mistakes, stemming sentences, and the use of multiple contexts.

6. Conclusion

In this paper, we present a high-quality emotion-annotated Hinglish corpus of tweets annotated with *happiness, sadness, anger, surprise, fear, disgust, others* classes. As far as we know, this work is the first to investigate the role of the emotion recognition task that aids in improving the performance of sentiment detection task on code-mixed data in a multitask learning setting. The proposed multi-task system is built on top of a cross-lingual embedding-based transformer model, XLMR, which yields better results than the existing XLMR-based state-of-the-art models on a similar task. We further show that transfer learning from a couple of high-resource mono-lingual languages to their low-resource code-mixed variant improves the overall performance of the system on the Sentiment task. On the *sentiment* task, the best accuracy attained by our multi-task model is 71.61% outperforming the state-of-the-art multitask system by 1.93% and the best performing multitask baseline by 1.31%. On the emotion task, our model attains 63.47%, outperforming the best performing baseline system by 1.18 points. While multitask learning enhanced generalization by using domain-specific information found in the training signals of the correlated tasks of sentiment and emotion, transfer learning improves the semantic coverage and association between the word vectors of Hindi and English information used for code blended Hinglish information, thereby acting as prior evidence for Hinglish embeddings information.

The emotion annotated dataset introduced in this work has the limitation of some severely under-represented classes which we would like to address in future work by adding samples to these classes and present a balanced distribution. Furthermore, the annotations and experiments outlined in this paper can be applied to code-mixed texts from multilingual communities that include more than two languages in the future.

CRediT authorship contribution statement

Soumitra Ghosh: Methodology, Implementation, Resource creation, Experiments, Writing – original draft. **Amit Priyankar:** Resource creation, Experiments, Writing – original draft. **Asif Ekbal:** Conceptualization, Methodology, Writing – review & editing, Funding acquisition, Supervision. **Pushpak Bhattacharyya:** Conceptualization, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors gratefully acknowledge the partial support from the project titled 'Development of CDAC Digital Forensic Centre with AI based Knowledge Support Tools', supported by Ministry of Electronics and Information Technology (MeitY), Government of India and Government of Bihar (project number P-264). Asif Ekbal acknowledges the Young Faculty Research Fellowship (YFRF), supported by Visvesvaraya PhD scheme for Electronics and IT, Ministry of Electronics and Information Technology (MeitY), Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia). All authors approved the version of the manuscript to be published.

References

[1] C. Hoffman, An Introduction to Bilingualism 4th impression, Longman Ltd, UK, 1996.
[2] P. Ekman, E.R. Sorenson, W.V. Friesen, Pan-cultural elements in facial displays of emotion, Science 164 (3875) (1969) 86–88.
[3] O. Bălan, G. Moise, L. Petrescu, A. Moldoveanu, M. Leordeanu, F. Moldoveanu, Emotion classification based on biophysical signals and machine learning techniques, Symmetry 12 (1) (2020) 21.
[4] S.-Y. Chen, C.-C. Hsu, C.-C. Kuo, L.-W. Ku, et al., Emotionlines: An emotion corpus of multi-party conversations, 2018, arXiv preprint arXiv:1802.08379.
[5] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020, Association for Computational Linguistics, 2020, pp. 8440–8451, http://dx.doi.org/10.18653/v1/2020.acl-main.747.
[6] A. Agarwal, P. Bhattacharyya, Sentiment analysis: A new approach for effective use of linguistic knowledge and exploiting similarities in a set of documents to be classified, in: Proceedings of the International Conference on Natural Language Processing, Vol. 22, ICON, 2005.
[7] R. Socher, A. Perelygin, J. Wu, J. Chuang, C.D. Manning, A.Y. Ng, C. Potts, Recursive deep models for semantic compositionality over a sentiment treebank, in: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2013, pp. 1631–1642.
[8] B.G. Patra, D. Das, A. Das, Sentiment analysis of code-mixed Indian languages: An overview of SAIL_Code-Mixed Shared Task@ ICON-2017, 2018, arXiv preprint arXiv:1803.06745.
[9] S. Ghosh, S. Ghosh, D. Das, Sentiment identification in code-mixed social media text, 2017, CoRR abs/1707.01184, URL http://arxiv.org/abs/1707.01184.
[10] A. Bohra, D. Vijay, V. Singh, S.S. Akhtar, M. Shrivastava, 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.
[11] A. Joshi, A. Prabhu, M. Shrivastava, V. Varma, Towards sub-word level compositions for sentiment analysis of Hindi-English code mixed text, in: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, 2016, pp. 2482–2491.
[12] K. Hashimoto, C. Xiong, Y. Tsuruoka, R. Socher, A joint many-task model: Growing a neural network for multiple nlp tasks, 2016, arXiv preprint arXiv:1611.01587.
[13] H. Wang, D. Can, A. Kazemzadeh, F. Bar, S. Narayanan, A system for real-time Twitter sentiment analysis of 2012 U.S. presidential election cycle, in: Proceedings of the ACL 2012 System Demonstrations, Association for Computational Linguistics, Jeju Island, Korea, 2012, pp. 115–120, URL https://www.aclweb.org/anthology/P12-3020.
[14] S. Shekhar, D. Sharma, M. Beg, An effective bi-LSTM word embedding system for analysis and identification of language in code-mixed social media text in English and Roman Hindi, Comput. Sist. 24 (2020) http://dx.doi.org/10.13053/cys-24-4-3151.
[15] P. Mathur, R. Sawhney, M. Ayyar, R. Shah, Did you offend me? Classification of offensive tweets in Hinglish language, in: Proceedings of the 2nd Workshop on Abusive Language Online, ALW2, 2018, pp. 138–148.
[16] R. Caruana, Multitask learning, Mach. Learn. 28 (1) (1997) 41–75.
[17] A. Kumar, A. Ekbal, D. Kawahara, S. Kurohashi, Emotion helps sentiment: A multi-task model for sentiment and emotion analysis, in: International Joint Conference on Neural Networks, IJCNN 2019 Budapest, Hungary, July 14-19, 2019, IEEE, 2019, pp. 1–8, http://dx.doi.org/10.1109/IJCNN.2019.8852352.
[18] M.S. Akhtar, D.S. Chauhan, D. Ghosal, S. Poria, A. Ekbal, P. Bhattacharyya, Multi-task learning for multi-modal emotion recognition and sentiment analysis, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), Association for Computational Linguistics, 2019, pp. 370–379, http://dx.doi.org/10.18653/v1/n19-1034.
[19] M.S. Akhtar, D. Ghosal, A. Ekbal, P. Bhattacharyya, S. Kurohashi, All-in-one: Emotion, sentiment and intensity prediction using a multi-task ensemble framework, IEEE Trans. Affect. Comput. 13 (1) (2022) 285–297, http://dx.doi.org/10.1109/TAFFC.2019.2926724.
[20] S. Ghosh, A. Ekbal, P. Bhattacharyya, A multitask framework to detect depression, sentiment and multi-label emotion from suicide notes, Cogn. Comput. (2021) 1–20.

[21] A. Kumar, A. Ekbal, D. Kawahara, S. Kurohashi, Emotion helps sentiment: A multi-task model for sentiment and emotion analysis, 2019, CoRR abs/1911.12569, URL http://arxiv.org/abs/1911.12569.

[22] S.R. Sane, S. Tripathi, K.R. Sane, R. Mamidi, Stance detection in code-mixed Hindi-English social media data using multi-task learning, in: Proceedings of the Tenth Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, 2019, pp. 1–5.

[23] N. Safi Samghabadi, P. Patwa, S. PYKL, P. Mukherjee, A. Das, T. Solorio, Aggression and misogyny detection using BERT: A multi-task approach, in: Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying, European Language Resources Association (ELRA), Marseille, France, 2020, pp. 126–131, URL https://www.aclweb.org/anthology/2020.trac-1.20.

[24] P. Singh, E. Lefever, Sentiment analysis for Hinglish code-mixed tweets by means of cross-lingual word embeddings, in: Proceedings of the the 4th Workshop on Computational Approaches to Code Switching, European Language Resources Association, Marseille, France, 2020, pp. 45–51, URL https://www.aclweb.org/anthology/2020.calcs-1.6.

[25] V. Srivastava, M. Singh, IIT gandhinagar at SemEval-2020 task 9: Code-mixed sentiment classification using candidate sentence generation and selection, in: Proceedings of the Fourteenth Workshop on Semantic Evaluation, International Committee for Computational Linguistics, Barcelona (online), 2020, pp. 1259–1264, URL https://www.aclweb.org/anthology/2020.semeval-1.168.

[26] K. Yadav, A. Lamba, D. Gupta, A. Gupta, P. Karmakar, S. Saini, Bi-LSTM and ensemble based bilingual sentiment analysis for a code-mixed Hindi-English social media text, in: 2020 IEEE 17th India Council International Conference, INDICON, IEEE, 2020, pp. 1–6.

[27] A. Wadhawan, A. Aggarwal, Towards emotion recognition in Hindi-English code-mixed data: A transformer based approach, 2021, arXiv:2102.09943.

[28] D. Vijay, A. Bohra, V. Singh, S.S. Akhtar, M. Shrivastava, Corpus creation and emotion prediction for Hindi-English code-mixed social media text, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop, Association for Computational Linguistics, New Orleans, Louisiana, USA, 2018, pp. 128–135, URL https://www.aclweb.org/anthology/N18-4018.

[29] N. Garg, K. Sharma, Annotated corpus creation for sentiment analysis in code-mixed Hindi-English (Hinglish) social network data, Indian J. Sci. Technol. 13 (40) (2020) 4216–4224.

[30] B.R. Chakravarthi, N. Jose, S. Suryawanshi, E. Sherly, J.P. McCrae, A sentiment analysis dataset for code-mixed Malayalam-English, in: Proceedings of the 1st Joint Workshop on Spoken Language Technologies for under-Resourced Languages (SLTU) and Collaboration and Computing for under-Resourced Languages, CCURL, European Language Resources association, Marseille, France, 2020, pp. 177–184, URL https://www.aclweb.org/anthology/2020.sltu-1.25.

[31] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), Association for Computational Linguistics, 2019, pp. 4171–4186, http://dx.doi.org/10.18653/v1/n19-1423.

[32] T. Pires, E. Schlinger, D. Garrette, How multilingual is multilingual BERT? in: Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers, Association for Computational Linguistics, 2019, pp. 4996–5001, http://dx.doi.org/10.18653/v1/p19-1493.

[33] K. Karthikeyan, Z. Wang, S. Mayhew, D. Roth, Cross-lingual ability of multilingual BERT: An empirical study, in: International Conference on Learning Representations, 2019.

[34] P. Patwa, G. Aguilar, S. Kar, S. Pandey, P. Srinivas, B. Gambäck, T. Chakraborty, T. Solorio, A. Das, SemEval-2020 task 9: Overview of sentiment analysis of code-mixed tweets, in: Proceedings of the Fourteenth Workshop on Semantic Evaluation, SemEval@COLING 2020, Barcelona (Online), December 12-13, 2020, International Committee for Computational Linguistics, 2020, pp. 774–790, URL https://www.aclweb.org/anthology/2020.semeval-1.100/.

[35] P. Ekman, An argument for basic emotions, Cogn. Emot. 6 (3–4) (1992) 169–200.

[36] J. Cohen, A coefficient of agreement for nominal scales, Educ. Psychol. Meas. 20 (1) (1960) 37–46.

[37] R. Plutchik, The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice, Am. Sci. 89 (4) (2001) 344–350.

[38] Z. Ahmad, R. Jindal, A. Ekbal, P. Bhattachharyya, Borrow from rich cousin: Transfer learning for emotion detection using cross lingual embedding, Expert Syst. Appl. 139 (2020) 112851.

[39] A. Malte, P. Bhavsar, S. Rathi, Team_Swift at SemEval-2020 task 9: Tiny data specialists through domain-specific pre-training on code-mixed data, in: Proceedings of the Fourteenth Workshop on Semantic Evaluation, 2020, pp. 1310–1315.

[40] G.-E. Zaharia, G.-A. Vlad, D.-C. Cercel, T. Rebedea, C.-G. Chiru, UPB at SemEval-2020 task 9: Identifying sentiment in code-mixed social media texts using transformers and multi-task learning, 2020, arXiv preprint arXiv:2009.02780.

[41] B. Braaksma, R. Scholtens, S. van Suijlekom, R. Wang, A. Üstün, FiSSA at SemEval-2020 task 9: Fine-tuned for feelings, in: Proceedings of the Fourteenth Workshop on Semantic Evaluation, International Committee for Computational Linguistics, Barcelona (online), 2020, pp. 1239–1246, URL https://aclanthology.org/2020.semeval-1.165.