The Journal of Supercomputing (2023) 79:13075–13089
https://doi.org/10.1007/s11227-023-05168-5

Textual emotion detection utilizing a transfer learning approach

Mahsa Hadikhah Mozhdehi[1] · AmirMasoud Eftekhari Moghadam[1]

Accepted: 5 March 2023 / Published online: 22 March 2023

Abstract
Many attempts have been made to overcome the challenges of automating textual emotion detection using different traditional deep learning models such as LSTM, GRU, and BiLSTM. But the problem with these models is that they need large datasets, massive computing resources, and a lot of time to train. Also, they are prone to forgetting and cannot perform well when applied to small datasets. In this paper, we aim to demonstrate the capability of transfer learning techniques to capture the better contextual meaning of the text and as a result better detection of the emotion represented in the text, even without a large amount of data and training time. To do this, we conduct an experiment utilizing a pre-trained model called Emotional-BERT, which is based on bidirectional encoder representations from transformers (BERT), and we compare its performance to RNN-based models on two benchmark datasets, with a focus on the amount of training data and how it affects the models' performance.

Keywords Natural language processing · Emotion classification · Text mining · Emotion detection · Transfer learning · Large language models

1 Introduction

In today's world, the availability of social media platforms is somehow overwhelming, and people spend a noticeable amount of their time communicating with each other through different social networking platforms. This communication could be via text, audio, or video, where people can express their emotions in these ways. There is no way to ignore or set aside emotions in human life because we, humans,

✉ AmirMasoud Eftekhari Moghadam
eftekhari.moghadam@gmail.com

Mahsa Hadikhah Mozhdehi
mahsa.mozhdehi@gmail.com

[1] Faculty of Computer and Information Technology, Islamic Azad University, Qazvin, Iran

use these emotions to communicate with each other, or make decisions [1]. Emotions can be expressed in different ways, for example, facial or body gestures, voice, or text. Detecting a person's emotion, by looking at their facial and body gestures or hearing their voice, would be easier, but emotion detection from text is not that easy, even for humans themselves [2].

However, due to the availability of enormous valuable textual data on social media platforms, such as Instagram, Facebook, and Twitter, and knowing that they contain valuable information about crowd behavior and emotion [3], automating the hard task of detecting emotions has gained popularity in the past few years. For example, during the COVID-19 pandemic [4], people have been sharing their experiences and opinions on the issue. Analyzing these comments can help us understand what they really feel and whether they are dealing with depression or not to take further action. Another example is empathetic chat-bots that need to understand the emotions of their users to respond accordingly [5].

Emotion detection (ED) is now a subfield of natural language processing (NLP), where it tries to detect the emotion lying behind the text, such as joy, love, and sadness. There is now various research employing different models for textual emotion detection such as LSTM [6], BiLSTM, and GRU [7]. Although these models made some promising contributions in this field, they have some limitations, such as being slow, requiring vastly computational resources, and needing a large amount of training data. But this amount of labeled data and computational resources are not always available.

Therefore, our objective is to [8] show the benefit of transfer learning and how this problem can be addressed utilizing pre-trained language models such as EmotionalBERT. In this paper, we used EmotionalBERT, which is based on pre-trained BERT [8]. The knowledge of the BERT model is transferred to train a standard feed-forward neural network with a softmax layer built on top of it, in order to classify tweets based on their emotions. Our results show that not only EmotionalBERT can perform better compared to the RNN-based models considered in this experiment, with only 36% of the dataset, but it significantly improves the accuracy with only a few training epochs. We also test the model on a new small dataset and compare the results.

In the next section, we overview related literature on emotion detection in text. Then, in section 3, we introduce EmotionalBERT, the pre-trained language model used in this experiment. Section 4 details the data preparation, the baseline models, and the experimental results. Section 5 concludes the paper and points out future work.

2 Related work

There have been many attempts at facial emotion recognition [9, 10] or audio-visual emotion recognition [11, 12], but there has been less focus on detecting emotions from textual data, as it is a relatively new area in NLP. Some work has been done using traditional machine learning techniques [1, 9, 13, 14]). A very important aspect of textual data is its sequential pattern. It means that the meaning of a single word obviously depends on the rest of the words presented in the sentence or paragraph. So context can help us to find out what a single word really means and determine the emotion of the text. Unfortunately, these traditional machine learning techniques cannot help with capturing the sequential nature of text [15] and as a result fail to consider the context while classifying text based on their emotion.

This lack of ability of traditional machine learning models made some deep learning models such as recurrent neural networks (RNNs), and their variants long-short term memory (LSTM) [6], and gated recurrent units (GRU) [7], more prominent in emotion detection in text [16–18]. Although recurrent models consider the sequential nature of text [19] and have achieved state-of-the-art results for different NLP tasks, they are slow and need to be trained from scratch, there is a limitation of how much they can capture the long-term dependencies in the text [20], and they need a large amount of labeled data to train. Preparing this large amount of labeled data is a time-consuming and tedious procedure [13], so it is not always available.

This is where transfer learning, transferring knowledge from a general-purpose task into a more specialized target task, comes into play. Using transfer learning, we can achieve better results compared to traditional deep learning models, with much smaller training material. Pre-trained language models, such as bidirectional encoder representations from transformers (BERT) [8], and its variants, Open AI GPT[21], and Transformer-XL [22], have been widely used in various NLP tasks and have shown promising performance. These models have been trained on a huge set of data and gained knowledge from it and now that knowledge can be used for other similar tasks, with no need for the huge amount of data and training time.

Some work has been done using pre-trained language models (LMs) to classify emotions or sentiments in text. [23] utilizes BERT as an embedding layer which then the output passes through a CNN and BiLSTM layer to perform Bangla sentiment analysis. They also compare BERT embedding ability to various word embedding techniques, such as Word2Vec, GloVe, and fastText, and their results show that BERT significantly outperforms all embedding and algorithms. [24] compares BERT, RoBERTa[1], DistilBERT[2], and XLNet [25] pre-trained transformer models' performance in recognizing emotions from texts. The implemented models are fine-tuned on the ISEAR data to classify it into seven emotion classes. Their results show that RoBERTa had the highest accuracy. [26] studies the effectiveness DeepEmotex models, which are fine-tuned USE [27] and BERT pre-trained models to classify text based on their emotions. They also studied the effect of varying the amount of data and found out that using more data for fine-tuning the pre-trained models can improve their performance.

But as we discussed earlier, although more training data can enhance the model's performance, preparing a large amount of labeled data is a time-consuming and tedious task. In this paper, our objective is to demonstrate the benefit of transfer learning and how such pre-trained models maintain their accuracy using a small amount of labeled data compared to traditional deep learning models such as RNNs.

[1] https://arxiv.org/abs/1907.11692

[2] https://arxiv.org/abs/1910.01108

3 EmotionalBERT model

In this paper, we adopted EmotionalBERT, which is based on pre-trained BERT. The knowledge of the BERT model is transferred to train a standard feed-forward neural network with a softmax layer built on top of it, in order to classify tweets based on their emotions. The bidirectional encoder representations from transformers (BERT) [8] are a transformer-based language model that only uses the encoder part of the transformer. It has been trained on a huge amount of data (books, Wikipedia, etc.), and it can be used as a pre-trained model for different NLP tasks such as sentiment analysis (SA), question answering (QA), and text summarization (TS). There are several variants of BERT; here, we use the BERT-based model and fine-tuned it for the target task. The model has 12-layer encoders or as the authors call them, transformer blocks. Each transformer block contains a 768-dimensional hidden layer and a 12-head self-attention layer.

The first input token is supplied with a special token called [CLS] which stands for classification. It is the representation of the whole input sequence and therefore can be used for classification tasks. BERT has been trained based on two different techniques. The first one is masked language modeling in which 15% of the input sequence will be replaced by [MASK] token and the model tries to predict the masked tokens. The second technique is next-sentence prediction. The model gets two sentences as inputs that are separated by the [SEP] token and the model has to find out if the second sentence follows the first one.

We take the final output of the first token [CLS] and feed it to a classifier. The classifier contains a feed-forward neural network layer followed by a softmax function to get the probability of classes. The architecture of the model is demonstrated in Fig. 1.

4 EmotionalBERT evaluation

In this section, we discuss the datasets we used and the way we prepared them for the experiments. We also explain the baseline models that are used to compare our results with and demonstrate our experimental results.

4.1 Data preparation

We conduct our experiment using two different datasets, Wang and MELD, which in the following subsections will be discussed.

4.1.1 Wang dataset

For our first experiment, we adopt the dataset created by Wang et al. [1]. This dataset contains around 2.5 million tweets which over 1.3 million were available to download using their IDs. The tweets are labeled with seven emotion classes, six of which

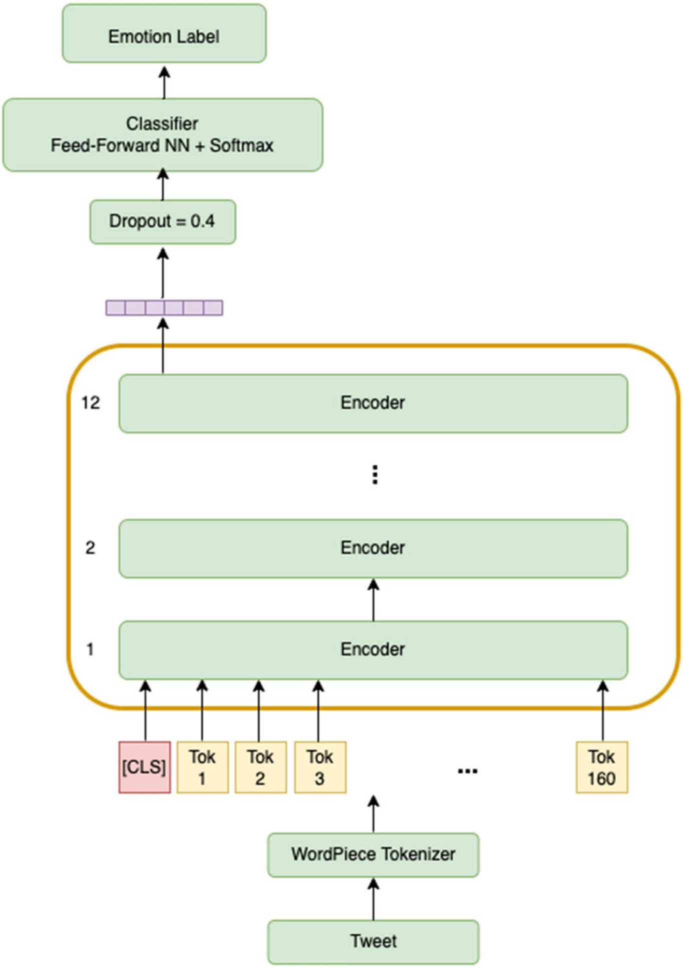

Fig. 1 The EmotionalBERT model architecture

are from [28]: joy, sadness, anger, love, fear, thankfulness, and surprise (Fig. 1), and were labeled using their hashtags. The number of tweets for each emotion class is depicted in Table 1. As can be seen, the dataset is imbalanced, meaning the amount of data for some classes is far fewer than the others. Surprise has the least amount of data, and joy has the most. This is not surprising as emotions like joy or sadness are more common than surprise.

4.1.2 MELD dataset

To prove our point and test the model performance, we ran another experiment on a new publicly available dataset called Multimodal EmotionLines Dataset (MELD) [29], which contains about 16,000 utterances from the TV series Friends.

Table 1 Number of tweets for each emotion class in Wang dataset

Emotion	Number of tweets
Joy	393,631
Sadness	338,015
Anger	298,480
Love	169,267
Fear	73,575
Thankfulness	79,341
Surprise	13,535
Total	**1,387,787**

The total number of tweets for all emotion classes is represented by the bold number

Table 2 Number of utterances for each emotion class in MELD dataset

Emotion	Number of utterances
Neutral	7575
Joy	2807
Anger	2016
Surprise	1842
Sadness	1139
Disgust	463
Fear	398
Total	**16240**

The total number of tweets for all emotion classes is represented by the bold number

This dataset contains video, audio, and text of the utterances. We only used the text data for our experiment. The number of utterances for each emotion class is depicted in Table 2. As can be seen, this dataset is relatively small compared to the first one, and it is also imbalanced. The neutral class has the highest, and fear has the lowest number of utterances. We adopted this dataset to see how the models perform on a much smaller dataset.

To prepare the data, we took some simple preprocessing steps. We removed the stopwords and punctuations, lowercased the letters, and convert the contracted words to their original form. Tokenization has been done using the BERT wordpiece tokenizer. We limited the length of each sentence to 160 tokens. So, if there is any sentence longer than 160 tokens, they will be truncated.

4.2 Baseline models

For comparison, we consider two RNN-based models. The first one is the bidirectional GRU model proposed by Seyeditabari et al.[3]. They used seven identical binary emotion classifiers for each emotion class. For the embedding layer, they tried different models and found out that there was no significant difference in their performance. They published their results based on two embedding models, ConceptNet Numberbatch [30] and fastText [31], both with 300 dimensions. The architecture of their model consists of an embedding layer, a bidirectional GRU layer, max-pooling and average-pooling layers, and a dense neural network layer.

After the embedding layer, there is a bidirectional GRU in order to capture a better understanding of the sequential nature of the tweets. To extract the most important features and an average representation from the output of this GRU layer, a concatenation of max-pooling and average-pooling is used. The output is then fed to a dense classification layer with a dropout rate of 50%. Finally, the sigmoid function produces the probability of each emotion class. For further study, you can refer to the original article.

The second one is an LSTM-based model. The model consists of a unidirectional LSTM layer with 300 hidden units, followed by a fully connected output layer. The pre-trained fastText embedding weights are used to initialize the embedding layer and are fixed during training. The model uses dropout with a rate of 0.5 to prevent overfitting. The AdamW optimizer is used to minimize the cross-entropy loss function. The model is trained on a GPU provided by Google Colaboratory service to speed up the training process. The weights are initialized using Xavier and orthogonal initialization for linear and LSTM layers, respectively.

4.3 First experiment

In the first experiment, we ran the EmotionalBERT and LSTM-based models on the Wang dataset and then compare the results with the bidirectional GRU model. The learning rate and batch size for EmtionalBERT and LSTM models are 2e-5 and 16, respectively. The EmotionalBERT and LSTM model trained for 3 and 5 epochs, respectively.

For EmotionalBERT, we chose three sets of increasing amounts of data to reach optimum F1 levels, and this has been achieved by 500K tweets. We did not feed the whole dataset to the model, in order to prove that despite the fact that feeding more data improves the model's performance, EmotionalBERT can achieve better results than RNN models, even without a huge training material. Also, due to low resources, we were not able to train the LSTM model from scratch on the whole dataset. So, we decided to train it on 500k tweets as well.

[3] https://arxiv.org/pdf/1907.09369.pdf

Table 3 F1-score results for phase 1, with 100,000 training data

Emotion	Precision	Recall	F1-score
Fear	0.77	0.66	0.70
Sadness	0.77	0.65	0.76
Love	0.78	0.64	0.68
Joy	0.81	0.80	0.81
Anger	0.82	0.81	0.81
Thankfulness	0.80	0.76	0.78
Surprise	0.81	0.58	0.63
Average	**0.79**	**0.71**	**0.73**

The bold numbers denote the superior performance or higher f1-score achieved by the model

Table 4 F1-score results for phase 2, with 250,000 training data

Emotion	Precision	Recall	F1-score
Fear	0.79	0.69	0.73
Sadness	0.79	0.76	0.77
Love	0.80	0.79	0.73
Joy	0.80	0.80	0.80
Anger	0.83	0.81	0.82
Thankfulness	0.84	0.75	0.78
Surprise	0.82	0.60	0.65
Average	**0.80**	**0.74**	**0.75**

The bold numbers denote the superior performance or higher f1-score achieved by the model

Table 5 F1-score results for phase 3, with 500,000 training data

Emotion	Precision	Recall	F1-score
Fear	0.84	0.65	0.70
Sadness	1	1	1
Love	1	1	1
Joy	0.82	0.81	0.82
Anger	1	1	1
Thankfulness	1	1	1
Surprise	0.84	0.53	0.55
Average	**0.92**	**0.85**	**0.86**

The bold numbers denote the superior performance or higher f1-score achieved by the model

Table 6 Number of tweets for each emotion class in 36% of the dataset

Emotion	Number of tweets
Joy	144,040
Sadness	123,367
Anger	109,435
Love	62,100
Thankfulness	29152
Fear	26,944
Surprise	4,962
Total	**500,000**

The total number of tweets for all emotion classes is represented by the bold number

4.3.1 Analyzing each phase

The first experiment is done on the Wang dataset in three phases, feeding data to the EmotionalBERT model with 100,000, 250,000, and 500,000 in each phase.

To fine-tune the model, we chose 16 for the batch size, as the amount of input data has a noticeable impact on the model performance. Considering how much the learning rate can affect the learning and convergence of the model, we decided not to choose it so large and chose 2e-5 for the learning rate. We trained the model for three epochs because in fine-tuning, the model has already learned many high-level and low-level features of the text, and there is no need for a large number of epochs. Also, after three epochs the training and validation accuracy was not changing anymore, so there was no point in training the model for more than three epochs.

We used the exact same BERT-based model for each emotion class, which predicts whether it is that specific emotion or the other ones. The results of these three phases are shown in Tables 3, 4, and 5, respectively. The reported numbers are F1-scores. We also added precision and recall scores to present more accurate results.

We can see in Table 3 that joy and anger have the highest F1-score, both equal to 81%, while on the other hand, surprise gets the lowest (63%). The overall F1-score in phase 1 is 73%. Here, we used only 0.1% of the dataset, so it is absolutely normal to not get the best results. In phase 2, we trained the model with 250,000 tweets. Here again, anger and joy have the highest F1-score and surprise and fear have the lowest (Table 4). We saw a 2% improvement in the overall F1-score (75%). But, this is not yet the optimal result, compared to the baseline models. So, we needed to keep feeding more data to the model. This time we decided to double the amount of data in phase 3, and repeat the experiment.

In phase 3, we used 500,000 tweets for training the model. The results for phase 3 are depicted in Table 5. The most significant point in this table is that F1-score is 100% for four classes. That means the model predicted and classified all of them correctly. Here, as we have the optimal result, we ended the experiment with three phases. Although the model has a great performance for four classes, it did not perform well for surprise and fear, as they show descending F1-scores throughout the experiment.

Table 7 Top 30 common words for classes surprise and fear

Fear	Numbers	Surprise	Numbers
0 Get	1898	0 actually	464
1 I'm	1830	1 i'm	272
2 Hope	1211	2 good	267
3 Going	1209	3 got	233
4 First	1120	4 like	181
5 Tomorrow	1091	5 know	181
6 Can't	1019	6 #surprise	165
7 Like	966	7 really	153
8 Wait	906	8 day	146
9 Time	880	9 today	144
10 Today	869	10 thought	144
11 Go	11	11 lol	138
12 Day	825	12 going	133
13 Know	796	13 see	132
14 Really	754	14:)	131
15 Got	751	15 see	128
16 See	729	16 time	125
17 Getting	668	17 one	124
18 One	656	18 get	118
19 Ready	645	19 u	116
20 Good	633	20 would	113
21 Gonna	625	21 think	109
22 Need	612	22 never	105
23 Im	608	23 23 can't	104
24 Walking	594	24 pretty	102
25 Back	541	25 christmas	95
26 Want	534	26 &	94
27 Final	532	27 first	94
28 Last	528	28 even	91
29 #nervous	512	29 #surprised	90

This is possibly happening due to two reasons. First, as we mentioned earlier, the dataset is not balanced and surprise and fear have the lowest amount of data compared to other classes. As given in Table 1, there are only around 13000 and 73000 tweets labeled with surprise and fear, respectively. Also, we used only 36% of the dataset, so this amount is even less than what was previously mentioned. The distribution of data for each class in 36% of the dataset is demonstrated in Table 6.

According to Table 6, in 500,000 tweets, there are 4,962 tweets with surprise labels and 26,944 for fear. As we can see, surprise has far fewer data compared to some classes such as joy or sadness. This lack of balance in the dataset can significantly affect the performance of the model for this class. So, the reason for the model not performing well on surprise is not having enough data.

Table 8 The comparison between EmotionalBERT, LSTM, and bidirectional GRU dataset

Emotion	BERT-based	Bidir-GRU	LSTM
Fear	0.70	**0.78**	0.13
Sadness	**1**	0.79	0.33
Love	**1**	0.80	0.28
Joy	0.82	0.82	0.60
Anger	**1**	0.83	0.52
Thankfulness	**1**	0.83	0.47
Surprise	0.55	**0.75**	0.03
Average	**0.86**	**0.80**	**0.33**

The bold numbers denote the superior performance or higher f1-score achieved by the model

Table 9 The comparison between EmotionalBERT and LSTM performance on the MELD dataset

Emotion	EmotionalBERT	LSTM
Neutral	**0.79**	0.54
Joy	**0.58**	0.08
Anger	**0.47**	0.00
Surprise	**0.62**	0.3
Sadness	**0.52**	0.5
Disgust	**0.55**	0.00
Fear	**0.50**	0.00
Average	**0.57**	**0.10**

The bold numbers denote the superior performance or higher f1-score achieved by the model

Fear has not also a significant amount of data (26,944), but it is almost as large as thankfulness (29,152). So, there should be some other reason besides the insufficient amount of data. We decided to see whether there is any other reason for the model not performing well on fear. We checked the top 30 common words for each class in the dataset and found out that, unlike other classes, there are not enough representative words for this class (Table 7). The most common words for fear are mostly irrelevant and cannot be representative of this emotion.

The only one that can be considered related to fear is the hashtag “nervous” with only 512 frequency. This issue is also true for the surprise class. The word “surprise” with 165 frequency and the hashtag ”surprised” are the only representative words for this class. This makes it difficult for the model to predict the tweets correctly, and it may classify them into other classes by mistake.

4.3.2 First experiment results

Despite the fact that the model did not perform great for the fear and surprise classes, it got a far better overall F1-score compared to the RNNS. The comparison of the results is shown in Table 8. The bidirectional GRU outperformed the EmotionalBERT in classifying fear and surprise, but EmotionalBERT had a better performance in five other classes.

Table 10 The utterance instances in the MELD dataset that can be classified into other classes without considering the running dialog

Utterance	Emotion	Other possible emotions
You're welcome. I'm sorry. Did I hurt you?	Fear	Disgust/anger
Will you marry me?	Fear	Joy/surprise
I've never lived like this before	Disgust	Anger/sadness
You have no idea how loud they are!	Disgust	Anger/surprise
Man, this is gonna be kinda weird	Sadness	Disgust/anger
No. No, not at all, that's-that's ridiculous	Sadness	Disgust/anger
You wouldn't believe what people put in here!	Anger	Surprise/disgust

As we can see the EmotionalBERT model got 86% F1-score, while the bidirectional GRU and LSTM got 80% and 33%, respectively. The results from our model show a significant improvement in F1-score, despite the fact that it used only 36% of the dataset.

4.4 Second experiment

In the second experiment, we adopted the MELD dataset. Since the second dataset is small itself, we conduct this experiment in only one phase, using the whole data at once. The batch size and learning rate set for both models are 16 and 2e-5, respectively. The EmotionalBERT and the LSTM were trained for 5 and 20 epochs, respectively. The results in Table 9 depict the EmotionalBERT and the LSTM model F1-scores. Seyeditabari et al. did not test their bidirectional GRU model on the MELD dataset.

As the results demonstrate, EmotionalBERT outperforms the LSTM model in all classes. The LSTM model failed to capture three classes: disgust, fear, and anger. It is not surprising, since the amount of data for these classes was not enough for the model to learn them. Moreover, this dataset consists of utterances that can be more meaningful while considering the dialogs. The utterance before and after could help capture the context even better. Here, the model is only assessing a single utterance and it can affect the results. Table 10 demonstrates some examples in the dataset where the utterance could be classified into some other classes, without considering the running dialog. Despite the fact that the training data is so small and contains only 16K utterances, and the similarity between classes discussed above, the EmotionalBERT still achieved far better results compared to the LSTM model.

5 Conclusion

In this paper, we tried to rely on transfer learning techniques that are now widely used in the field of natural language processing and developed an architecture based on the pre-trained BERT model, called EmotionalBERT, so that it can be used to detect emotions in the textual data with higher accuracy compared to RNN-based models, with considerably reduced training material [32, 33]. We demonstrated the benefit of transfer learning in dealing with small datasets, in classifying text based on their emotions.

For future work, the reasons the model has performed poorly in some classes can be explored using explainable AI. We can also work on dialog-based emotion detection to give the model a richer context and see how it can improve the model's performance. Also, the effectiveness of employing other pre-trained models such as TinyBERT [32], DistilBERT, XLNet, and MobileBERT [33] on classifying texts adopting small datasets can be compared, and using explainable AI, we can analyze the superior models to find out the reasons they can perform better on small datasets and how we can introduce new models focusing on this aspect.

Author Contributions The first author (Mahsa Hadikhah Mozhdehi) conceived of the presented idea under the supervision of the second author, carried out the experiment, designed the model, and wrote the manuscript. The second author (AmirMasoud Eftekhari Moghadam) discussed the results, supervised the findings of this work, provided recommendations, and commented on the manuscript.

Funding Not applicable.

Availability of data and materials The data analyzed during the presented study are available publicly from these links: https://github.com/armintabari/Emotion-Detection-RNN/tree/master/data and https://affective-meld.github.io.

Declarations

Conflict of interest The authors declare that there are no competing interests regarding the publication of this paper.

Ethical Approval Not applicable.

References

1. Wang W, Chen L, Thirunarayan K, Sheth AP (2012) Harnessing twitter" big data" for automatic emotion identification. In: 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Conference on Social Computing, pp 587–592. https://doi.org/10.1109/SocialCom-PASSAT.2012.119. IEEE
2. Sailunaz K, Dhaliwal M, Rokne J, Alhajj R (2018) Emotion detection from text and speech: a survey. Social Netw Anal Mining 8(1):1–26. https://doi.org/10.1007/s13278-018-0505-2
3. Wakamiya S, Belouaer L, Brosset D, Lee R, Kawai Y, Sumiya K, Claramunt C (2015) Measuring crowd mood in city space through twitter. In: International Symposium on Web and Wireless Geographical Information Systems, pp 37–49. https://doi.org/10.1007/978-3-319-18251-3_3. Springer
4. Garcia K, Berton L (2021) Topic detection and sentiment analysis in twitter content related to covid-19 from brazil and the usa. Appl soft Comput 101:107057. https://doi.org/10.1016/j.asoc.2020.107057
5. Fung P, Bertero D, Xu P, Park JH, Wu C-S, Madotto A (2018) Empathetic dialog systems. In: The International Conference on Language Resources and Evaluation. European Language Resources Association
6. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

7. Cho K, Van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using rnn encoder-decoder for statistical machine translation. Conference on Empirical Methods in Natural Language Processing, EMNLP. https://doi.org/10.3115/v1/D14-1179
8. Devlin J, Chang M, Lee K, Toutanova K (2018) BERT: pre-training of deep bidirectional transformers for language understanding. Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423. arXiv:1810.04805
9. Jain DK, Shamsolmoali P, Sehdev P (2019) Extended deep neural network for facial emotion recognition. Pattern Recognit Lett 120:69–74. https://doi.org/10.1016/j.patrec.2019.01.008
10. Mehendale N (2020) Facial emotion recognition using convolutional neural networks (ferc). SN Appl Sci 2(3):1–8. https://doi.org/10.1007/s42452-020-2234-1
11. Avots E, Sapiński T, Bachmann M, Kamińska D (2019) Audiovisual emotion recognition in wild. Machine Vision Appl 30(5):975–985. https://doi.org/10.1007/s00138-018-0960-9
12. Hossain MS, Muhammad G (2019) Emotion recognition using deep learning approach from audio-visual emotional big data. Inform Fusion 49:69–78. https://doi.org/10.1016/j.inffus.2018.09.008
13. Hasan M, Rundensteiner E, Agu E (2019) Automatic emotion detection in text streams by analyzing twitter data. Int J Data Sci Anal 7(1):35–51. https://doi.org/10.1007/s41060-018-0096-z
14. Asghar MZ, Subhan F, Imran M, Kundi FM, Shamshirband S, Mosavi A, Csiba P, Várkonyi-Kóczy AR (2019) Performance evaluation of supervised machine learning techniques for efficient detection of emotions from online content. https://doi.org/10.20944/preprints201908.0019.v1
15. Acheampong FA, Nunoo-Mensah H, Chen W (2021) Transformer models for text-based emotion detection: a review of bert-based approaches. Artif Intell Rev 54(8):5789–5829. https://doi.org/10.1007/s10462-021-09958-2
16. Kratzwald B, Ilić S, Kraus M, Feuerriegel S, Prendinger H (2018) Deep learning for affective computing: Text-based emotion recognition in decision support. Decision Support Syst 115:24–35. https://doi.org/10.1016/j.dss.2018.09.002
17. Chatterjee A, Gupta U, Chinnakotla MK, Srikanth R, Galley M, Agrawal P (2019) Understanding emotions in text using deep learning and big data. Comput Human Behav 93:309–317. https://doi.org/10.1016/j.chb.2018.12.029
18. Xu G, Li W, Liu J (2020) A social emotion classification approach using multi-model fusion. Future Generat Comput Syst 102:347–356. https://doi.org/10.1016/j.future.2019.07.007
19. Du K-L, Swamy M (2019) Recurrent neural networks. In: Neural Networks and Statistical Learning, pp 351–371. Springer, https://doi.org/10.1007/978-1-4471-5571-3_2
20. Hochreiter S (1998) The vanishing gradient problem during learning recurrent neural nets and problem solutions. Int J Uncertainty, Fuzziness Knowl-Based Syst 6(02):107–116. https://doi.org/10.1142/S0218488598000094
21. Radford A, Narasimhan K, Salimans T, Sutskever I, et al. (2018) Improving language understanding by generative pre-training
22. Dai Z, Yang Z, Yang Y, Carbonell J, Le QV, Salakhutdinov R (2019) Transformer-xl: Attentive language models beyond a fixed-length context. Associat Comput Linguist. https://doi.org/10.18653/v1/P19-1285
23. Prottasha NJ, Sami AA, Kowsher M, Murad SA, Bairagi AK, Masud M, Baz M (2022) Transfer learning for sentiment analysis using bert based supervised fine-tuning. Sensors 22(11):4157
24. Adoma AF, Henry N-M, Chen W (2020) Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition. In: 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), pp 117–121. IEEE
25. Yang Z, Dai Z, Yang Y, Carbonell JG, Salakhutdinov R, Le QV (2019) Xlnet: Generalized autoregressive pretraining for language understanding. In: Neural Information Processing Systems
26. Hasan M, Rundensteiner E, Agu E (2021) Deepemotex: Classifying emotion in text messages using deep transfer learning. In: 2021 IEEE International Conference on Big Data (Big Data), pp 5143–5152. IEEE
27. Cer D, Yang Y, Kong S-y, Hua N, Limtiaco N, John RS, Constant N, Guajardo-Cespedes M, Yuan S, Tar C, et al. (2018) Universal sentence encoder for english. In: Proceedings of the 2018

Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp 169–174

28. Shaver P, Schwartz J, Kirson D, O'connor C (1987) Emotion knowledge: further exploration of a prototype approach. J Personal Soc Psychol 52(6):1061. https://doi.org/10.1037//0022-3514.52.6.1061
29. Poria S, Hazarika D, Majumder N, Naik G, Cambria E, Mihalcea R (2019) MELD: A multimodal multi-party dataset for emotion recognition in conversations. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp 527–536. Association for Computational Linguistics, Florence, Italy. https://doi.org/10.18653/v1/P19-1050. https://aclanthology.org/P19-1050
30. Speer R, Chin J, Havasi C (2017) Conceptnet 5.5: An open multilingual graph of general knowledge. In: Thirty-first AAAI Conference on Artificial Intelligence. https://doi.org/10.1609/aaai.v31i1.11164
31. Bojanowski P, Grave E, Joulin A, Mikolov T (2016) Enriching Word Vectors with Subword Information. arXiv. https://doi.org/10.48550/ARXIV.1607.04606. arXiv:1607.04606
32. Jiao X, Yin Y, Shang L, Jiang X, Chen X, Li L, Wang F, Liu Q (2020) Tinybert: Distilling bert for natural language understanding. Conference: Findings of the Association for Computational Linguistics: EMNLP. https://doi.org/10.18653/v1/2020.findings-emnlp.372
33. Sun Z, Yu H, Song X, Liu R, Yang Y, Zhou, D (2020) Mobilebert: a compact task-agnostic bert for resource-limited devices. Annual Conference of the Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.195

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