Journal of Intelligent Information Systems
https://doi.org/10.1007/s10844-024-00845-0

RESEARCH

Leveraging distant supervision and deep learning for twitter sentiment and emotion classification

Muhamet Kastrati[1] · Zenun Kastrati[2] · Ali Shariq Imran[3] · Marenglen Biba[1]

Received: 14 May 2023 / Revised: 16 January 2024 / Accepted: 23 January 2024

Abstract

Nowadays, various applications across industries, healthcare, and security have begun adopting automatic sentiment analysis and emotion detection in short texts, such as posts from social media. Twitter stands out as one of the most popular online social media platforms due to its easy, unique, and advanced accessibility using the API. On the other hand, supervised learning is the most widely used paradigm for tasks involving sentiment polarity and fine-grained emotion detection in short and informal texts, such as Twitter posts. However, supervised learning models are data-hungry and heavily reliant on abundant labeled data, which remains a challenge. This study aims to address this challenge by creating a large-scale real-world dataset of 17.5 million tweets. A distant supervision approach relying on emojis available in tweets is applied to label tweets corresponding to Ekman's six basic emotions. Additionally, we conducted a series of experiments using various conventional machine learning models and deep learning, including transformer-based models, on our dataset to establish baseline results. The experimental results and an extensive ablation analysis on the dataset showed that BiLSTM with FastText and an attention mechanism outperforms other models in both classification tasks, achieving an F1-score of 70.92% for sentiment classification and 54.85% for emotion detection.

Keywords Distant supervision · Emotion detection · Sentiment analysis · Deep learning · Transformers · Twitter · Emojis

✉ Zenun Kastrati
zenun.kastrati@lnu.se

Ali Shariq Imran
ali.imran@ntnu.no

Marenglen Biba
marenglenbiba@unyt.edu.al

[1] Department of Computer Science, University of New York Tirana, Tirana 1046, Albania

[2] Department of Informatics, Linnaeus University, Växjö 351 95, Sweden

[3] Department of Computer Science, Norwegian University of Science and Technology (NTNU), Gjøvik 2815, Norway

Published online: 22 March 2024

1 Introduction

Microblogging and social networks wield significant influence today in a wide range of domains, encompassing daily communication, ideas sharing, opinions, emotions, reactions, shopping behaviors, political discourse, and responses to crises, to name a few (Kapoor et al., 2018). Over the past few years, researchers have shown a growing interest in text-based sentiment and emotion detection on online social networks, particularly Twitter and Facebook (Zimbra et al., 2018).

The vast amount of text generated by Twitter users serves as a rich source for capturing people's emotions, integral to human life, and strongly influencing people's behaviors and actions (Wang et al., 2012). Emotion detection in short texts, such as social media posts, has a high impact on different sectors including industries, health, security, or education with a wide range of applications such as e-learning environment, depression monitoring (Zucco et al., 2017), detecting mental disorders (Aragon et al., 2021), personality traits, detection suicide-related content and emotions (Schoene et al., 2022), hate speech detection, cyber-bullying identification, event detection, disease tracking, and cyber threat detection.

Moreover, detecting emotions in social network data poses a non-trivial task due to the brevity of the text, especially considering that Twitter users often employ non-standard language (irony, sarcasm, and humor) to express their emotional state (Canales et al., 2019). Additionally, social tweets are characterized by a prevalence of informal and slang words, misspellings, hashtags, emoticons, and abbreviations, making interpretation challenging for automated emotion detection models (Kusal et al., 2021).

Emotional models form the foundation of the emotion-sensing process, with three main modeling approaches being categorical, dimensional, and componential emotion models. The categorical emotion model assumes that only a small number of significant emotions are independent and not related to each other. Two predominant emotion models for emotion classification are Plutchik's model (Plutchik, 1980) with eight basic (primary) emotions and Ekman's model (Ekman, 1993) with six basic emotions.

Various learning approaches are employed for text emotion detection, including lexicon-based (Mohammad & Turney, 2013), rule-based (Krommyda et al., 2020), machine learning-based (Wood & Ruder, 2016; Yousaf et al., 2020), and deep learning-based approaches (Colnerič & Demšar, 2018; Polignano et al., 2019; Kastrati et al., 2022).

Conventional machine learning and deep learning models are widely used to build sentiment analysis and emotion recognition systems (Kastrati et al., 2022; Imran et al., 2020; Edalati et al., 2021). More recently, deep neural networks, including CNN and RNN (such as LSTM, BiLSTM, and GRU), have gained popularity for their state-of-the-art performance in various natural language processing (NLP) tasks. Kastrati and Biba (2021). Supervised learning is the most widely used approach in machine learning, including deep and shallow learning (LeCun et al., 2015). However, training supervised learning models requires a large amount of human-labeled data, which is not always available for real-world applications, and text emotion detection is no exception (Wood & Ruder, 2016). Furthermore, high-quality datasets for text emotion research have been scarce. Most existing datasets with multiclass emotion annotations are either too small or/and highly imbalanced to adequately support supervised emotion learning (Kang et al., 2020).

To address this challenge, we have collected a large-scale emotion dataset of tweets from Twitter. Inspired by the research study conducted in Batra et al. (2021), emotion-indicative emojis are used for the automatic labeling of the dataset. Then, several supervised conventional machine learning and deep learning, including transformer-based models are tested on the newly collected dataset to establish the baseline results and examine an approach to sentiment polarity and emotion detection that better suits the dataset, aiming to improve the performance of the classifier models.

1.1 Study objective and research questions

This study focuses on automatic labeling techniques for very large-scale tweet datasets for sentiment and emotion analysis tasks using distant supervision with emojis. It also investigates the training of deep neural networks on our large-scale dataset for classifying both sentiment polarity and emotions.

Therefore, with this background, we formulated the main research objective to improve the effectiveness of sentiment polarity and emotion classification using a very large-scale dataset automatically labeled through distant supervision with emojis and deep learning models.

According to the objective above, the following research questions were raised:

- RQ1: How can we automatically create a large-scale emotion dataset by utilizing emotion-indicative emojis available in tweets for sentiment polarity and emotion classification tasks?
- RQ2: How do the size of training data and class imbalance affect the performance of conventional machine learning algorithms and deep neural networks?
- RQ3: To what extent do pre-trained word embedding techniques and attention mechanisms improve sentiment and emotion classification performance?

1.2 Contribution

The core contributions of this work are:

- Collecting and curating a real-world large-scale dataset of tweets that are automatically labeled with categorical emotions based on Ekman's model using distant supervision with emotion-indicative emojis.
- The new knowledge concerning performance comparison of supervised conventional machine learning algorithms and deep neural networks for sentiment polarity and emotion classification on our created dataset.
- Proposed a multi-layer BiLSTM assessment model with pre-trained word embeddings and an attention mechanism for classifying both sentiment polarity and emotions (multiclass classification).
- Provide an ablation analysis on the effect of the size of the dataset and the number of classes, as well as on the effect of class imbalance in the classification performance.

2 Related work

During the past decade, several studies have been conducted with regard to the sentiment analysis tasks in Twitter posts. Most of these studies can generally be grouped into two main research directions based on their core contributions: i) data curation/labeling techniques for sentiment analysis tasks, and ii) polarity/emotion classification. The first group entails studies concerning data collection and (semi) automatic labeling techniques. For instance, the research work conducted in Go et al. (2009), introduced for the first time distant supervision labels (emoticons) for classifying the sentiment polarity of tweets. The study presents one of the most widely used Twitter sentiment datasets for sentiment analysis tasks known as Sentiment140. Another similar study that uses a distant supervision strategy for automatic labeling is presented in Davidov et al. (2010). In particular, hashtags and text emoticons for sentiment annotation are applied in both studies to generate labels. A similar study that applies not only emoticons and hashtags but also emojis, as distantly supervised labels to detect Plutchik's emotions is conducted in Suttles and Ide (2013).

There is another strand of research that focuses on creating datasets for the emotion detection task. For example, the research study in Mohammad and Kiritchenko (2015) presents Twitter Emotion Corpus annotated using distant supervision with emotion-specific hashtags for emotion annotation. An extended dataset called the Tweet Emotion Intensity dataset is presented later in Mohammad and Bravo-Marquez (2017) where the authors created the first dataset of tweets annotated for anger, fear, joy, and sadness intensities using the best-worst scaling technique. The researchers in Kralj Novak et al. (2015) present the first emoji sentiment lexicon, known as the Emoji Sentiment Ranking as well as a sentiment map that consists of the 751 most frequently used emojis. The sentiment of the emojis is computed from the sentiment of the tweets in which they occur.

A similar work was conducted in Batra et al. (2021), where the authors presented a dataset containing around 1.1 Million Urdu tweets distributed over two months. They employed a heuristics labeling approach that allowed multi-label emotion. Furthermore, the dataset is characterized by the presence of a high-class imbalance problem. In contrast to the study in Batra et al. (2021), our research work differs in both data collection and heuristic labeling. We collected tweets posted over the last 10 years with an almost proportional daily-based distribution, which helps to reduce the bias during data collection. Additionally, our collected dataset is balanced, with an equal number of samples among six basic emotion categories, even though some emotions are more representative than others on Twitter. Furthermore, our selection heuristic for determining the true label for tweets having more emojis that refer to different emotions maintained a strict one-emotion-per-tweet.

The second group of research works focuses on polarity and emotion classification using conventional machine learning algorithms and deep neural networks. For instance, such a study is conducted in Polignano et al. (2019), where the authors proposed a classification approach for emotion detection from text using deep neural networks including Bi-LSTM, and CNN, with self-attention and three pre-trained word embeddings for word encoding. Another similar example where LSTM models are used for estimating the sentiment polarity and emotions from Covid-19 related tweets is proposed in Imran et al. (2020) and in Batra et al. (2021). The later study also introduced a new approach employing emoticons as a unique and novel way to validate deep learning models on tweets extracted from Twitter. Another study focusing on emotion recognition using both emoticons and text with LSTM is conducted in Islam et al. (2020).

In Kastrati et al. (2022) authors conducted a set of experiments on their distant-supervised labeled dataset using conventional machine learning and deep learning models for sentiment polarity and multiclass emotion detection tasks. According to the authors, deep neural networks such as BiLSTM and CNN-BiLSTM outperformed other models in both sentiment polarity and multiclass emotion classification tasks.

From the literature reviewed above, we observed that there are numerous articles focused on distant supervision with hashtags, and emoticons and only a few of them use emojis as a noisy label for automatic labeling tweet datasets for sentiment and emotion analysis tasks. However, emojis are used far more extensively than hashtags and they present a more faithful representation of a user's emotional state. Moreover, most of those studies experimented with small and imbalanced tweet datasets, which are often domain-specific. Furthermore, in most of these studies, the researchers treated the multiclass problem of emotion classification as a binary problem. Our research work is different from the above-mentioned studies in many aspects including distant supervision with emojis, size of the dataset, timeline coverage, and variety of deep learning models. Additionally, we experimented mainly with the emotion-balanced dataset and treated the emotion classification as a multiclass classification task.

3 Design and research methodology

This study uses a quantitative approach composed of five major phases. The first phase entails the collection of emoji tweets on Twitter, belonging to the period from 01 January 2012 until 31 December 2021. To be able to collect enough tweets to meet our needs, we selected 41 emojis indicative of the emotion used in research from Batra et al. (2021) and then we collected tweets that contained at least one of the selected emojis, and only those tweets that were tagged by Twitter as English (retweets excluded). In the second phase of this study, text pre-processing is performed to remove extra attributes related to tweets (author id, date of creation, language, source, etc.), duplicate tweets, extract emojis from tweets, remove hashtags/mentions, URLs, emails, phone number, non-ASCII characters and tweets with length less or equal to five characters. Additionally, all tweets were converted to lowercase. In the third phase, the automatic labeling of collected tweets was carried out using distant supervision with emotion-indicative emojis. Consequently, all emoji tweets are properly classified into one of Ekman's six basic emotion categories, including anger, disgust, fear, joy, sadness, or surprise. In the fourth phase, a representation model to prepare and transform the tweets to an appropriate numerical format to be fed into the emotion classifiers is performed. More precisely, a bag-of-words approach (TF-IDF) with conventional machine learning algorithms, as well as three different pre-trained word embeddings (GloVe, Glove Twitter, and FastText) with deep learning neural networks, are used. The final phase of the study involves the sentiment analyzer for binary classification and the emotion analyzer for multiclass emotion classification. The analyzer involves several classifiers including conventional machine learning and deep neural networks for sentiment polarity and emotion classification. A high-level architecture of the proposed sentiment and emotion analyzer depicting all the five phases elaborated above is illustrated in Fig. 1.

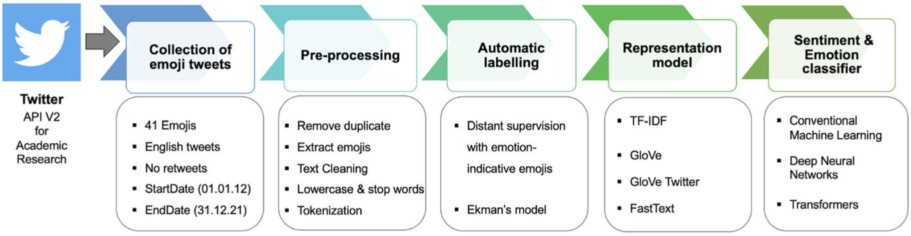


Fig. 1 High-level architecture of the proposed solution

Table 1 Number of tweets among emotion and sentiment classes (D1)

Emotion	# of instances	%	Sentiment	# of instances	%
Joy	6,369,299	36.4	Positive	8,452,049	48.3
Surprise	2,082,750	11.9			
Sadness	2,452,298	14.0	Negative	9,064,788	51.7
Disgust	2,180,383	12.4			
Anger	2,115,000	12.1			
Fear	2,317,107	13.2			
Total	17,516,837	100.0	Total	17,516,837	100.0

4 Experimental settings

This section briefly describes the dataset (emoji tweets) as well as the classifier models used to perform the sentiment and emotion classification tasks.

4.1 Dataset

The dataset utilized for carrying out the diverse range of experiments in this study consists of 17.5 million tweets (more precisely 17,516,837 tweets) posted within 10 years, respectively, between January 1, 2012, and December 31, 2021, with an almost proportional daily-based distribution. The whole data collection process was conducted through Twitter API v2 for academic research product track using Python 3.

Manually labeling the tweets would have been almost impossible even for a large team but also a labor-intensive, time-consuming, and error-prone task due to the quantity. We labeled the tweets by considering the distant supervision with emojis for emotion labeling, whereas the polarity associated with a tweet is inferred directly from the emotions. More precisely, the positive polarity class is comprised of two positive emotions (joy and surprise), and the negative polarity is derived from negative emotions (anger, fear, disgust, and sadness). Then conventional machine learning and deep neural networks including transformer-based models were employed for the binary classification of tweets into positive or negative classes and multiclass classification of emotions into one of the possible emotions such as anger, fear, joy, and sadness.

Table 2 Intentionally balanced among emotion classes (D2)

Emotion	# of instances	%
Joy	2,000,000	16.7
Surprise	2,000,000	16.7
Sadness	2,000,000	16.7
Disgust	2,000,000	16.7
Anger	2,000,000	16.7
Fear	2,000,000	16.7
Total	12,000,000	100.0

Table 3 Dataset statistics after removing disgust and surprise (D1a)

Sentiment	# of instances	%
Negative	6,884,405	48.1
Positive	6,369,299	51.9
Total	13,253,704	100.0

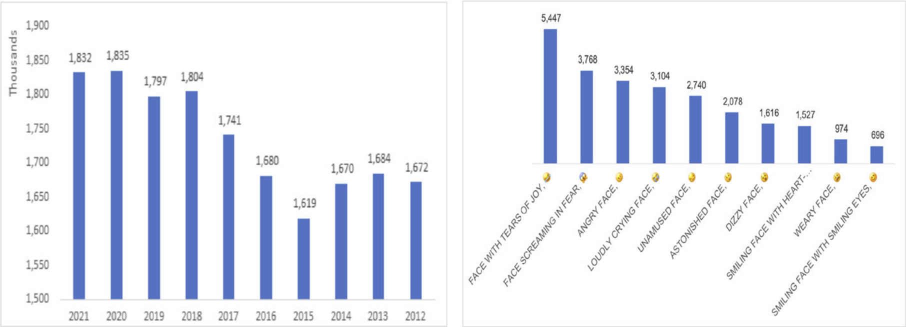
As shown in Table 1, in the original dataset each sentiment polarity class is represented by the same number of instances (tweets) - it is balanced for sentiment polarity classes (48.3% for positive and 51.7% for negative sentiment) but it is imbalanced for emotions.

Table 2 shows statistics of the intentionally emotion-balanced dataset (D2) that comprises 12 million tweets. We randomly selected 2 million tweets for each emotion class from the original D1 dataset and as a result, we obtained a well-balanced emotion dataset (16.7% for each emotion).

Table 3 shows statistics for the D1a dataset. It is a subset of the original D1 dataset without *disgust* and *surprise* emotions, as these two emotions are overlapped with other emotions. The D1a dataset remains balanced for sentiment polarity classes and was used in our experiments for the task of sentiment polarity classification (except Section 5.7 where the whole D1 dataset was used).

4.1.1 Dataset statistics

The number of tweets across years and the top 10 emojis are illustrated in Fig. 2. As shown, the number of tweets per year ranged from 1.6 and 1.8 million. Among the most commonly used emojis “Face with tears of Joy”, emerged as the dominant one with 5,447 thousand tweets followed by “Face screaming in Fear” with 3,768 thousand tweets, and last from the top 10 was “Smiling face with smiling eyes” with 696 thousand tweets (Fig. 2).



(a) per year

(b) per emojis

Fig. 2 Distribution of tweets (a) per year and (b) per emojis (values in thousands)

4.1.2 Distant supervision of tweets

The concept behind distant supervision involves the automatic labeling of data in order to be able to leverage large amounts of it. This type of data is referred to as distant supervised or weakly annotated, as the quality is not great, but the quantity is Byrkjeland et al. (2018).

The distant supervision used in this study employs the emoji heuristic labeling algorithm, enabling the automatic labeling of training sets for sentiment and emotion classification tasks.

To create our training labels of one emotion per tweet, we used the following simple heuristic: We used the emotion-indicative emojis to determine the emotion per tweet. In cases where a tweet contains multiple emojis expressing different emotions, the emoji that occurs more frequently is used for determining the emotion. In cases where ambiguity arises because of more emojis in a tweet with the same frequency but different emotions, the algorithm considers the sentiment scores calculated in Kralj Novak et al. (2015), selecting the emotion associated with the emoji possessing the largest sentiment score. By using this approach, we have efficiently and automatically labeled a large-scale dataset of 17.5 million tweets, facilitating the training of models for sentiment and emotion classification tasks.

4.1.3 Dataset tagging

The main purpose of this dataset was to collect only English tweets that contain emotion-indicative emojis and tag each tweet with emojis that are present in tweets for sentiment and emotion analysis. For our purpose, we designed a query that extracts tweets for each day for 10 years, containing emojis and text written in English and no retweets.

We intend to ensure that this dataset is suitable for the tasks of sentiment and emotion analysis. We have used the list of 41 emotion-indicative emojis to categorize tweets based on Ekman's emotion model. Consequently, sentiment polarity is derived from emotions, as positive or negative depending on the emotion category. For example, tweets that belong to the joy and surprise emotion category are labeled with a positive sentiment class, and other negative emotions (sad, disgust, fear, anger) with a negative sentiment polarity label. An excerpt of the dataset is shown in Table 4.

4.2 Architecture and parameter settings

This section presents a brief overview of the deep neural network and transformer-based architectures and their parameter/configuration settings applied in our experiments for this study.

4.2.1 Deep neural networks

To perform the tasks of sentiment polarity and emotion classification in our Twitter dataset, we employed five different supervised deep neural networks, including one-dimensional CNN, LSTM, GRU, BiLSTM, and a hybrid CNN_BiLSTM. The reasons that we have chosen these architectures are based on the specific nature of text modeling. Additionally, we have performed experiments with two other state-of-the-art transformer-based architectures, including BERT and RoBERTa. Table 5 presents various deep neural networks along with their model configurations (experiment settings).

Due to space limitations, we have opted to showcase and elaborate on the best-performing architecture, allowing us to delve deeper into the details of the most effective choice. Figure 3

Table 4 Example of mapping emojis to emotion labels

Tweet	Emoji Description	Emotion	Polarity
You better chill before I tell twitter how you was screaming out daddy" last night 😂 😂	[Face with tears of joy, 0.221]	Joy	Positive
NHS England makes plans for field hospitals in preparation for Omicron wave 😦	[Frowning face with open mouth, -0368]	Surprise	Positive
I tried to find the tweet but i think she deleted it? But i reported the acc. That was so uncalled for 😠	[Angry face, -0.299]	Anger	Negative
Wow you knew what he was and y'all STILL voted for him 😒	[Unamused face, -0.374]	Disgust	Negative
This is so cute but i wish what happened to this father and son doesn't happen to yujin and myah please 😭	[Loudly crying face, -0.093]	Sadness	Negative
We should have been okay until late september but now I have no clue what we'll do. I'm a little scared honestly 😨	[Fearful face, -0.14]	Fear	Negative

illustrates the BiLSTM architecture applied to our dataset for a sentiment polarity task. The architecture comprises the following components: an embedding layer with word embeddings of size 300D, a dropout layer, four BiLSTM layers with 256, 128, 64, and 32 units, each using a ReLU activation function, and an attention layer. The output layer consists of two neurons, one for each class (positive or negative), and employs a sigmoid activation function to produce probability-like predictions for each class. The classification model is trained using the logarithmic loss function and the efficient ADAM gradient-based optimization algorithm.

The same model is used for the emotion classification task. However, since we are dealing with a multiclass classification problem, we replace the classification loss function with 'categorical_crossentropy'. We also adjust the number of units in the output layer to 4, one for each emotion class (anger, fear, joy, or sadness). Additionally, we replace the sigmoid activation function with softmax to output probability-like predictions for each emotion. This approach, of adapting the binary classification model for a multiclass classification task, is applied in all other architectures presented in the following.

4.2.2 Parameter settings

All deep neural network models for sentiment and emotion classification in this study are implemented in Keras (https://keras.io, accessed on 15 May 2022). Keras is a simple and robust deep-learning library for Python used for constructing a neural network. It is a high-level framework based on TensorFlow developed at Google. Scikit-learn (https://scikit-learn.org/stable/, accessed on 10 May 2022), a simple, efficient, and open-source tool for predictive data analysis in Python, is used for developing conventional machine learning classifiers in this study. The maximum number of words to be used in the tokenizer model was set to 200,000 and the input comment sequence is padded to 30 words.

Table 5 Configuration and accuracy of the deep learning models

Classifier	Model Configuration / Parameters	Sentiment Polarity	Emotion Detection
DNN Baseline	Embedding Layer with 300 Dimension, GlobalMaxPooling1D, Layers with 128, 64, 32 with ReLU, Dense 2 with Sigmoid (Dense 4 with Softmax).	67.76%	50.52%
1D-CNN + FastText	Embedding Layer with crawl-300d-2M.vec Layers with 512 with ReLU, GlobalMaxPooling1D, Dense 32 with ReLU Dense 2 with Sigmoid (Dense 4 with Softmax).	69.43%	52.70%
LSTM + FastText	Embedding Layer with crawl-300d-2M.vec, BiLSTM Layers with 256, 128, 64, 32 with ReLU, GlobalMaxPooling1D Dense 2 with Sigmoid (Dense 4 with Softmax).	70.69%	54.45%
GRU + FastText	Embedding Layer with crawl-300d-2M.vec, GRU Layers with 256, 128, 64, 32 with ReLU, GlobalMaxPooling1D, Dense 2 with Sigmoid (Dense 4 with Softmax).	70.32%	53.93%
BiLSTM + GloVe	Embedding Layer with glove.6B.300d.txt, GRU Layers with 256, 128, 64, 32 with ReLU, GlobalMaxPooling1D, Dense 2 with Sigmoid (Dense 4 with Softmax).	70.05%	53.78%
BiLSTM + GloVe.Twitter	Embedding Layer with glove.twitter.27B.200d.txt, GRU Layers with 256, 128, 64, 32 with ReLU, GlobalMaxPooling1D, Dense 2 with Sigmoid (Dense 4 with Softmax).	70.84%	54.31%
BiLSTM + FastText	Embedding Layer with crawl-300d-2M.vec, BiLSTM Layers with 256, 128, 64, 32 with ReLU, GlobalMaxPooling1D, Dense 2 with Sigmoid (Dense 4 with Softmax).	70.91%	54.94%
CNN-BiLSTM + FastText	Embedding Layer with crawl-300d-2M.vec, SpatialDropout1D(0.3), Conv1D with 32 with ReLU, BiLSTM with 32 with ReLU, Flatten layer, Dense 64 with ReLU, Dense 2 with Sigmoid (Dense 4 with Softmax).	70.22%	53.63%
BERT	"bert-base-uncased", 12-layer, 768-hidden, 12-heads, 110M parameters. Further details can be found in Devlin et al. (2018).	69,87%	54,56%
RoBERTa	"roberta-base". 12-layer, 768-hidden, 12-heads, 125M parameters. RoBERTa using the BERT-base architecture. For further details see in Liu et al. (2019).	68,55%	53.90%

The following parameter settings are used to conduct experiments. The dataset is divided into two subsets: a training set and a test set. The training set consists of 90% of the data, while the remaining 10% is used for testing the model. Model training was set to 50 epochs and the “EarlyStopping” criteria with its arguments: monitor = “val_loss” and patience = 3, is used to

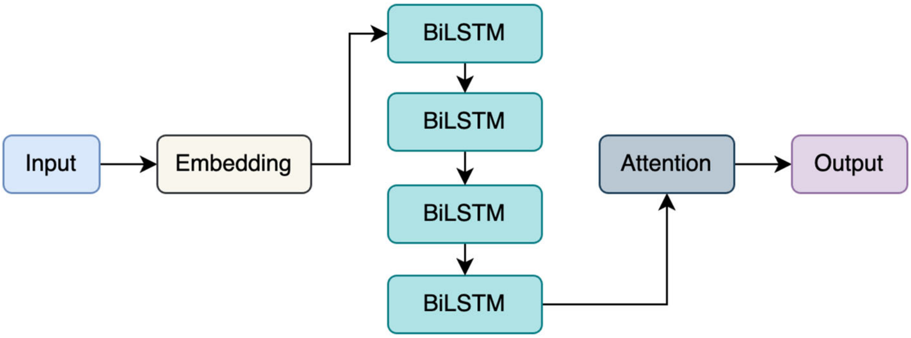


Fig. 3 BiLSTM Architecture

stop classifiers. The batch size of 2048 gave us the best results. Deep recurrent neural networks such as LSTM and BiLSTM, generally have the problem of overfitting. To avoid overfitting in our deep neural networks, we used a dropout strategy in which certain units (neurons) are temporarily removed from the network model along with incoming and outgoing connections. Dropout prevents model units from co-adapting too much to the training data and thus it leads to better generalization on the testing set as well (Srivastava et al., 2014). We have applied between layers using the Dropout Keras layer and the dropout rate was set to 0.3 (SpatialDropout1D(0.3)).

On the other hand, the hyperparameters used for pre-trained transformer-based models (BERT and RoBERTa) in our experiments are as follows: the activation function is ReLu, both models use the AdamW optimization algorithm, the batch size is set to 32 for BERT and 8 for RoBERTa, and the number of epochs is 3 for both models.

Fine-tuning transformer-based models on large-scale datasets such as ours, which includes million of instances poses a significant challenge due to their substantial computational complexity, extensive model size, and demanding memory requirements. Moreover, our experiments were conducted using Colab Pro+, where we encountered restrictions on the maximum runtime set at 24 hours and faced the high cost of acquiring additional compute units. To address these challenges in fine-tuning both transformer-based models (BERT and RoBERTa), we implemented a random sampling strategy. This involved creating two distinct subsets, each comprising one million tweets. One subset was designated for sentiment analysis, whereas the other subset served as the foundation for emotion classification task. By employing a random sampling strategy, we not only accommodated practical limitations but also ensured that the selected subsets maintained representative diversity.

The source code for the transformer-based architectures in our experiments is obtained from Teja (2021).

4.3 Pretrained word embeddings

In this study, we have compared the results of deep learning models obtained using three different pre-trained word embeddings such as GloVe, GloveTwitter, and FastText.

- GloVe stands for Global Vector for Word Representation proposed in Pennington et al. (2014). The model is an unsupervised learning-based algorithm developed by Stanford for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus. The resulting representations show interesting linear substructures of the word vector space. The GloVe version used in this work is one that is trained on a Wikipedia 2014 dump with Gigaword 5 that has 6 billion tokens, 400 thousand of vocab, uncased, 300d vectors. Further detailed information on the process of training GloVe word embedding is explained in Pennington et al. (2014).

- GloveTwitter is a pre-trained glove vector based on 2 billion tweets, 27 billion tokens, and 1.2 million vocab, uncased.
- FastText proposed in Bojanowski et al. (2017), is an extension of word2vec model, which represents words as n-grams of characters. It is composed of a vocabulary of 2 million words and n-grams of the words, case sensitive and obtained from 600 billion tokens trained on data crawled from generic Internet web pages by Common Crawl nonprofit organization (Polignano et al., 2019). Further detailed information on the process of training can be found in Bojanowski et al. (2017).

4.4 Attention mechanism

An attention mechanism is a feature that equips models with the ability to focus on specific words or phrases within the text. It works by assigning different weights to each word in the text, enabling the models to capture the context and most significant information. When it comes to analyzing tweets, the attention mechanism aims to assign more weight to sentiment-carrying words in order to better grasp the overall sentiment expressed in that particular tweet. Mathematically, an attention mechanism can be defined using (1).

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

where,

K represents a key vector. A key vector is a word embedding from a sequence of word embeddings constituting a tweet.

Q indicates a query vector tasked with understanding the sentiment expressed in the tweet.

V denotes value vectors.

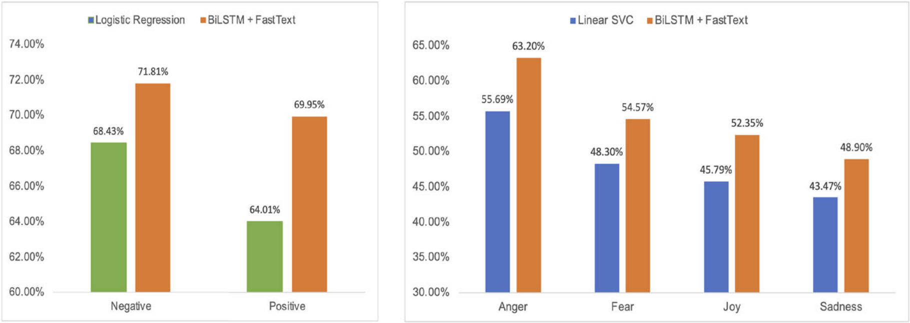
$\sqrt{d_k}$ denotes the dimension of a key vector.

The attention mechanism computes the attention scores by calculating a dot product between the key vector K, and the query vector Q, divided by values dimension $\sqrt{d_k}$ which is used to control the magnitude of the score. These attention scores are converted into weights using a softmax function.

After calculating the weights, a context vector C is obtained by computing a weighted sum of the value vectors V.

$$C = \sum_i Attention(Q, K_i, V_i) \tag{2}$$

The context vector C contains information that is contextually important for understanding the sentiment of the tweet. This context-aware representation is then used as an input to the sentiment analysis model to classify the sentiment of the tweet as positive, negative, or neutral based on the attended information.



(a) sentiment

(b) emotion

Fig. 4 F1 score of best-performing algorithms on (a) sentiment polarity and (b) emotion classification tasks

5 Experimental results

This section provides the experimental results obtained from various sentiment and emotion classifiers trained and validated on our dataset. The configuration settings for the deep learning models employed in our dataset are given in Table 5.

The findings illustrated in Fig. 4 show that the best performance is achieved by deep learning models in both sentiment polarity and emotion classification tasks. Class-wise performance with respect to the F1 score for sentiment polarity and emotion classification tasks is given in Fig. 4. For the sake of space, we present results obtained from only two best-performing models, including one from conventional machine learning and one from deep learning.

Table 6 Performance of ML and DL models for sentiment polarity classification

Classifier	P (%)	R (%)	F1 (%)	Acc (%)
Naive Bayes	65.69	65.67	65.54	63.27
Logistic Regression	66.34	66.37	66.31	66.37
Linear SVC	66.27	66.29	66.27	66.29
Decision Tree	61.80	54.62	57.99	53.86
AdaBoost	62.49	58.52	60.44	58.52
DNN Baseline	67.80	67.76	67.62	67.76
1D-CNN + FastText	69.70	69.43	69.42	69.43
LSTM + FastText	70.75	70.69	70.70	70.69
GRU + FastText	70.33	70.32	70.33	70.32
BiLSTM + GloVe	70.17	70.05	70.06	70.05
BiLSTM + GloVe Twitter	70.86	70.84	70.85	70.84
BiLSTM + FastText	70.93	70.91	70.92	70.91
CNN_BiLSTM + FastText	70.23	70.22	70.22	70.22
BERT	69.87	69.87	69.87	69.87
RoBERTa	67.07	69.55	68.30	69.55

5.1 Sentiment polarity classification

We used the D1a dataset (described in Section 4.1) for training conventional machine learning and deep learning models for sentiment polarity classification. Table 6 summarizes the models' performance on 10% test data.

The deep BiLSTM with FastText pre-trained word embeddings and an attention layer, outperforms other deep learning models in sentiment polarity classification in our dataset, achieving an F1 score of 70.92%. This represents an average performance improvement of 3.3 percentage points compared to the baseline results. Additionally, the deep BiLSTM + FastText model demonstrates superior overall accuracy compared to the RoBERTa model, surpassing it by 2.62 percentage points. Moreover, it outperforms the BERT model by 1.05 percentage points.

5.2 Emotion classification

We used the D2 dataset (described in Section 4.1), excluding disgust and surprise, to train various deep learning models for multiclass emotion classification. Table 7 summarizes the models' performance on 10% test data. The deep BiLSTM with FastText pre-trained word embeddings and an attention layer outperforms other deep learning models for multiclass emotion classification task, achieving an F1 score of 54.85%. This marks an average performance improvement of 4.4 percentage points over the baseline results. Moreover, the overall accuracy of the deep BiLSTM + FastText model surpasses that of the BERT model by 0.2 percentage points and the RoBERTa model by 1.0 percentage points.

Table 7 Performance of ML and DL models for emotion classification

Classifier	P (%)	R (%)	F1 (%)	Acc (%)
Naive Bayes	47.52	47.62	47.57	47.62
Logistic Regression	47.73	47.78	47.75	47.78
Linear SVC	48.31	48.45	48.38	48.45
Decision Tree	47.93	27.52	34.96	27.52
AdaBoost	42.77	35.67	38.90	35.66
DNN Baseline	50.38	50.52	50.40	50.52
1D-CNN + FastText	52.89	52.70	52.77	52.70
LSTM + FastText	54.32	54.45	54.34	54.45
GRU + FastText	53.83	53.93	53.85	53.93
BiLSTM + GloVe	53.64	53.79	53.71	53.78
BiLSTM + GloVe Twitter	54.10	54.32	54.21	54.31
BiLSTM + FastText	54.75	54.95	54.85	54.94
CNN_BiLSTM + FastText	53.82	53.96	53.89	53.96
BERT	54.74	54.56	54.65	54.56
RoBERTa	53.69	53.90	53.79	53.90

Table 8 Performance of BiLSTM w/o attention for sentiment polarity classification

Class	BiLSTM			BiLSTM + Att		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
Negative	71.05	68.97	70.00	70.72	69.73	70.22
Positive	67.51	69.65	68.57	67.79	68.81	68.30
Weighted avg	69.35	69.30	69.31	69.31	69.29	69.30

5.3 Effect of attention mechanism

In this section, we examine the impact of the attention mechanism on capturing the long-range dependencies in the collected tweets. For this purpose, an attention layer considering a global and local context is used on top of BiLSTM to extract high-level features. Global context characterizes the entire tweet, and it is too broad. On the other hand, local context is defined from a small window of different sizes. In our case, we have used the window size of 8 words based on the research work (Kastrati et al., 2021) as the optimal context to extract semantic features using the attention mechanism.

This section provides an overview of the impact of the attention mechanism on the performance of BiLSTM for the sentiment classification task. More precisely, we carried out experiments with two different classification parameters with regard to the network architecture used. In the first case, the network architecture consists of BiLSTM layers with 256, 128, 64, and 32 units with ReLU, and a Flatten layer. In the second case, the network architecture is extended with an attention layer integrated on top of BiLSTM. The obtained results for the sentiment classification task using both architectures (without and with attention mechanism) with respect to precision, recall, and F1 score are summarized in Table 8. As can be seen from Table 8, there is no performance improvement when the BiLSTM model is used with the attention mechanism and the results are almost the same. On the other hand, regarding the class-wise performance, there is a subtle shift in the performance, indicating that the performance of the negative sentiment class improved at the cost of the positive one. As can be seen from the table, the F1 score for negative sentiment increased from 70.00% to 70.22% but decreased for positive sentiment from 68.57% to 68.30%.

Table 9 shows the obtained results for the multiclass emotion classification task using both architectures (without and with attention mechanism) with respect to precision, recall, and F1 score. As can be seen from Table 9, there is no impact on the overall performance of the classifier for the task of emotion classification when the attention mechanism is used. However, subtle shifts were observed in the class-specific metrics, indicating that the perfor-

Table 9 Performance of BiLSTM w/o attention for emotion classification

	BiLSTM			BiLSTM + Att		
Class	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
Anger	58.83	62.50	60.61	59.35	61.52	60.42
Fear	52.47	51.63	52.04	51.84	52.90	52.36
Joy	49.54	50.46	50.00	50.54	47.68	49.07
Sadness	49.80	46.57	48.13	48.62	48.59	48.60
Weighted avg	52.66	52.79	52.69	52.58	52.67	52.61

Table 10 F1 score of BiLSTM model with different word embeddings for sentiment polarity classification

Classifier	Negative	Positive	Weighted avg
BiLSTM	70.00	68.57	69.31
BiLSTM + GloVe	70.38	69.70	70.06
BiLSTM + GloVe Twitter	71.74	69.88	70.85
BiLSTM + FastText	71.81	69.95	70.92

mance of certain classes improved at the cost of others. For example, for fear emotion, the F1 score increased from 52.04% to 52.36%, and for sadness from 48.13% to 48.60% but this was at the cost of anger and joy emotions.

5.4 Effect of static word embeddings

In this section, we present the results obtained from the set of experiments conducted to see the impact of general-purpose pre-trained word embeddings on the sentiment and emotion classification tasks.

Tables 10 and 11 show the impact of three pre-trained word embeddings such as GloVe, GloVe Twitter, and FastText on the best-performing model, that is, BiLSTM for sentiment polarity and emotion classification tasks. The results obtained showed that all three pre-trained word embeddings initialize word vectors for the datasets effectively. The F1 score was slightly different on different word vector methods. Furthermore, BiLSTM with FastText pre-trained word embeddings produced better results followed by BiLSTM with GloVe Twitter. This proved that pre-trained word embeddings and especially FastText substantially affected the accuracy of the entire model.

Similarly, the three pre-trained word embeddings used with the BiLSTM model for emotion classification had a substantial impact on performance improvement, even better than for the sentiment polarity classification task, as shown in Table 11.

5.5 Effect of having multiple classes

Recognizing that multiclass classification is characterized by several challenges, we aimed to delve deeper and get better insight into the effects of multiclass classification of emotions. To accomplish this, we initially performed a multiclass classification of emotions based on Ekman's six basic emotions, achieving an accuracy of 41.33%. Seeking for more comprehensive understanding, we performed the chi-squared test to identify the top 20 terms (top

Table 11 F1 score of BiLSTM model with different word embeddings for emotion classification

Classifier	Anger	Fear	Joy	Sadness	Weighted Avg
BiLSTM	60.61	52.04	50.00	48.13	52.69
BiLSTM + GloVe	62.02	53.44	51.26	47.71	53.71
BiLSTM + GloVe Twitter	62.50	54.16	50.74	48.86	54.21
BiLSTM + FastText	63.20	54.75	52.35	48.90	54.85

10 uni-grams and top 10 bi-grams) most correlated with each emotional class. The analysis revealed that many terms overlapped between the classes. Specifically, the anger and disgust emotion classes share a lot of common terms, making their distinction challenging. Similarly, fear and surprise shared numerous common terms, further complicating the differentiation between these emotional classes.

This way, we continued our experiment to explore the effects of removing these problematic classes. We started by removing instances belonging to surprise from our dataset. The decision was guided by its high level of overlap with fear, introducing ambiguity in classification. Additionally, we acknowledged the complexity of surprise, as some instances of this class are associated with positive valence and others with negative valence (Mohammad, 2021). As a result, our best-performing classifier (BiLSTM + FastText) demonstrated an improvement of almost 6.5 percentage points in accuracy compared to the 6-class results.

Observing this change in performance, we further reduced the number of emotional classes by removing disgust, which is also known for its complexity and higher overlap with anger according to the chi-square test. This decision led to an even more substantial improvement, with an additional 7.2 percentage points in accuracy compared to the 5-class results.

Continuing in this manner, our exploration culminated in binary classification, where we assessed the classifier's accuracy by comparing each emotional class against the others. The results for binary classification were promising, and the highest accuracy achieved was 79.22% for the [fear vs disgust] comparison. Almost all binary emotion classifications achieved an accuracy higher than 70%, except the two pairs, [anger vs disgust] with an accuracy of 68.4%, and [fear vs surprise] having the lowest accuracy of 60.7% because of their complexity and overlap problem mentioned above. The average accuracy obtained on all binary classification pairs was 73.0%. A detailed analysis of the effect of multiple classes is provided in the following subsections.

5.5.1 Six emotion classes

In our first experiment, we performed the multiclass classification of emotions based on Ekman's six basic emotions. For this experiment, we used the entire dataset of 12 million tweets (2 million tweets for each emotion class). Table 12 shows the precision, recall, and F1 score of the best-performing model BiLSTM + FastText on the balanced dataset on Ekman's six basic emotions such as anger, disgust, fear, joy, sadness, and surprise. The obtained weighted average F1 score is 41.33%.

Table 12 Precision, Recall, and F1 score for the 6-class emotion classification

Class	P (%)	R (%)	F1 (%)
Anger	47.92	49.37	48.63
Disgust	39.32	47.55	43.04
Fear	39.90	42.50	41.16
Joy	42.66	44.05	43.34
Sadness	41.74	35.06	38.11
Surprise	36.14	29.70	32.61
Weight avg	41.28	41.38	41.33

Table 13 Precision, Recall, and F1 score for the 5-class emotion classification

Class	P (%)	R (%)	F1 (%)
Anger	51.45	52.81	52.12
Disgust	42.74	47.44	44.97
Fear	50.40	53.77	52.03
Joy	48.57	46.40	47.46
Sadness	45.88	38.60	41.92
Weight avg	47.81	47.81	47.70

5.5.2 Five emotion classes

Table 13 shows the Precision, Recall, and F1 score of the best-performing model BiLSTM + FastText on the dataset with five discrete emotions such as anger, disgust, fear, joy, and sadness. Here we drop the sixth category - a surprise as it has the worst performance (overlapped with fear emotion) and causes weighted average performance degradation.

5.5.3 Four emotion classes

Table 14 shows the obtained results with regard to the Precision, Recall, and F1 score of the best-performing model BiLSTM + FastText on the emotion-balanced dataset on four discrete emotions such as anger, fear, joy, and sadness. Here we drop the second category - disgust as it is characterized by class overlapping with anger and causes weighted average performance degradation.

5.5.4 Three emotion classes

Table 15 shows the Precision, Recall, and F1 score of the best-performing model BiLSTM + FastText on the emotion-balanced dataset on three basic emotions such as anger, joy, and sadness. Here we drop the third category - fear to leave only three basic emotions.

5.5.5 Two emotion classes

Table 16 shows the Precision, Recall, and F1 score of the best-performing model BiLSTM + FastText on the emotion-balanced dataset on two basic emotions such as joy and sadness.

Table 17 shows the weighted average Precision, Recall, F1 score, and Accuracy of the best-performing model BiLSTM + FastText on the emotion-balanced dataset on two basic

Table 14 Precision, Recall, and F1 score for the 4-class emotion classification

Class	P (%)	R (%)	F1 (%)
Anger	60.18	66.54	63.20
Fear	54.69	54.45	54.57
Joy	51.69	53.02	52.35
Sadness	52.44	45.81	48.90
Weighted avg	54.75	54.95	54.85

Table 15 Precision, Recall, and F1 score for the 3-class emotion classification

Class	P (%)	R (%)	F1 (%)
Anger	68.80	68.19	68.50
Joy	60.72	66.13	63.31
Sadness	59.78	54.98	57.28
Weighted avg	63.11	63.11	63.03

emotions (binary classification) for all pairs of emotions. Almost all emotion binary classification tasks achieved an Accuracy higher than 70%, except the two pairs, [anger vs disgust] with an accuracy of 68.4%, and [fear vs surprise] having the lowest of 60.7%. The average Accuracy of classification is 73.0%. It is worth mentioning that each emotion is comprised of the same number of tweets - 2 million tweets per emotion.

5.6 Effect of the size of training data

In this section, we examine the effect of increasing the size of the training data on the accuracy of the best-performing classifier BiLSTM with FastText. Since most of the existing studies on sentiment analysis and emotion identification in tweets are conducted on datasets comprising a few thousand tweets, we expect to derive new insights and benefits from using large training data. In our case, we started with a small sample consisting of 20 thousand (20K) randomly selected tweets and continued this way increasing the number of tweets up to 13.3 million (13M) for sentiment classification and 8 million (8M) for the emotion classification task.

Figure 5 shows the result of training the BiLSTM with FastText classifies on each subset in the sequence and the F1 score achieved on 10% test instances by the model for the sentiment and emotion classification tasks.

From Fig. 5 we observe that as the size of training data increases from 20 thousand (20K) to 13.3 million (13M), we got an F1 score between 60.98% and 70.92% for the sentiment classification task, an average performance improvement of nearly 10 percentage points on the smallest subset results. On the other hand, for the emotion classification task, by increasing the size of the training data from 20 thousand (20K) to 8 million (8M), we got an F1 score between 44.45% and 54.85%, an average performance improvement of more than 10 percentage points on the smallest subset results.

5.7 Effect of class imbalance

In this section, we investigate the effects of class imbalance on the performance of deep neural network-based classifiers for sentiment polarity and emotion classification tasks. The investigation is performed using the best performing model i.e. BiLSTM with FastText model trained on our newly created Twitter dataset. The experimental results obtained show that

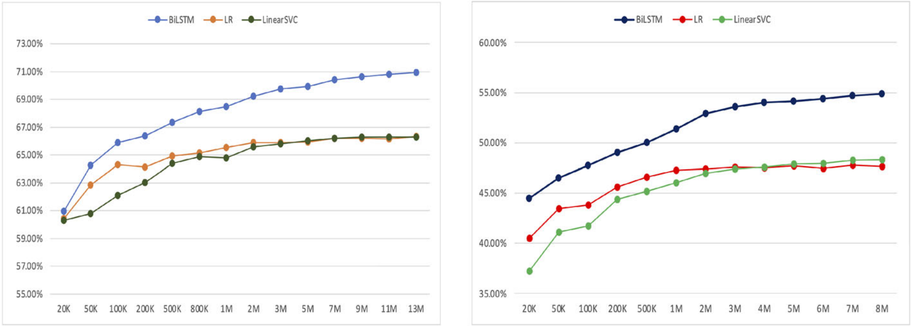
Table 16 Precision, Recall, and F1 score for the 2-class emotion classification

Class	P (%)	R (%)	F1 (%)
Joy	70.87	69.03	69.94
Sadness	69.97	71.78	70.86
Weighted avg	70.42	70.41	70.40

Table 17 Precision, Recall, F1 score, and Accuracy of best-performing model for binary emotion classification

Emotion classes	P (%)	R (%)	F1 score (%)	Acc (%)
Joy vs Anger	78.13	78.11	78.10	78.10
Joy vs Disgust	76.64	76.41	76.36	76.40
Joy vs Fear	70.85	70.82	70.81	70.81
Joy vs Sadness	70.42	70.41	70.40	70.41
Joy vs Surprise	70.70	70.69	70.69	70.69
Sadness vs Disgust	70.26	70.11	70.06	70.11
Fear vs Disgust	79.22	79.22	79.22	79.22
Anger vs Disgust	68.43	68.42	68.42	68.42
Anger vs Fear	78.80	78.78	78.78	78.78
Anger vs Sadness	75.80	75.80	75.80	75.80
Anger vs Surprise	76.45	76.45	76.44	76.45
Disgust vs Surprise	75.85	75.83	75.82	75.83
Fear vs Surprise	60.70	60.69	60.69	60.69
Sadness vs Surprise	70.85	70.74	70.70	70.70
Fear vs Sadness	73.58	73.57	73.57	73.56

the performance of the classifier deteriorates when a class imbalance exists within training data. Specifically, the performance obtained from our best-performing classifier showed a bias towards the majority class, respectively joy class. To overcome this issue, for the set of experiments conducted in this study, we intentionally balanced the dataset to have an equal number of instances among all classes. Table 18 summarizes classifiers' performance on 10% test data. The results are obtained using the BiLSTM + FastText classifier on the imbalanced dataset for the multiclass emotion classification task. Observe that the dataset has an unequal distribution of emotions. Furthermore, the Joy emotion class has a larger number of instances compared to other emotion classes, thus, as a consequence, the classifier exhibits bias towards the majority class. More precisely, the performance given in terms of the F1 score for the Joy emotion class is much higher (63.58%) compared to other emotion classes, but poor performance can be seen in the other two emotions (i.e., sadness and surprise), especially on the minority class where the F1 score achieved is 20.23%. The difference in the weighted



(a) Sentiment

(b) Emotion

Fig. 5 F1 score of the best-performing deep learning model with varied sizes of training data for (a) sentiment and (b) emotion classification

Table 18 Precision, Recall, and F1 score for emotion classification on the imbalanced dataset

Class	P(%)	R (%)	F1 (%)	# of instances
Anger	46.54	44.46	45.48	211,119
Disgust	39.24	33.60	36.20	218,027
Fear	40.77	31.24	35.37	232,209
Joy	53.18	79.04	63.58	636,382
Sadness	41.32	27.62	33.11	245,630
Surprise	38.05	13.77	20.23	208,317
Weight avg	45.54	47.91	44.82	1,751,684

average for the Precision and F1 score (Precision: 47.91% and F1 score: 44.82%) shown in Table 18 is a result of the class imbalance.

Table 19 summarizes classifiers' performance on 10% test data. The results are obtained using the BiLSTM + FastText classifier on the emotion-balanced dataset for the multiclass emotion classification task. The dataset used here is balanced, we randomly selected 2 million tweets for each emotion category from the original D1 dataset. As mentioned in the previous paragraph, here there is almost no difference between the weighted average of the precision and F1 score.

Tables 20 and 21 summarize classifiers' performance on imbalanced and well-balanced datasets for the sentiment polarity classification task. As can be seen from the right-most column in Table 20 (# of instances) the dataset has an unequal distribution of sentiment polarity classes. Furthermore, the negative class has a larger number of instances compared to positive classes, thus, as a consequence, the classifier exhibits bias towards the majority class. The difference in class-wise performance is around 14 percentage points with regard to the F1 score.

Table 21 summarizes classifiers' performance on 10% test data. The results are obtained using the BiLSTM + FastText classifier on the sentiment-balanced dataset for the sentiment classification task. The dataset used here is almost equally balanced after removing tweets belonging to the disgust and surprise emotion classes. As can be seen, here there is almost no difference between the weighted average of the Precision and F1 score.

Table 19 Precision, Recall, and F1 score for emotion classification on the balanced dataset

Class	P (%)	R (%)	F1 (%)	# of instances
Anger	47.92	49.37	48.63	199,823
Disgust	39.32	47.55	43.04	199,882
Fear	39.90	42.50	41.16	200,594
Joy	42.66	44.05	43.34	200,220
Sadness	41.74	35.06	38.11	199,790
Surprise	36.14	29.70	32.61	199,691
Weight avg	41.28	41.38	41.33	1,200,000

Table 20 Precision, Recall, and F1 score for sentiment classification on the imbalanced dataset

Class	P(%)	R (%)	F1 (%)	# of instances
Negative	73.91	81.13	77.35	905,841
Positive	68.87	59.32	63.74	637,568
Weight avg	71.83	72.12	71.73	1,543,409

6 Discussion

Returning to the research questions (RQs) posed at the beginning of this study, we can now affirm that it is possible to automatically create a large-scale dataset with emotion labels (i.e., emotions-indicative emojis) for sentiment polarity and emotion classification tasks. This approach demonstrates several advantages, as outlined below.

- RQ1: The emotion-indicative emojis in tweets are provided by the tweet's author, which is more natural and reliable as they represent the author's intended interpretation or emotional state, in contrast with emotion labels of other datasets given by a few annotators. In addition to that, another function of emojis could also be to emphasize or strengthen the emotion or sentiment conveyed by a message. Moreover, emojis can serve as explicit sentiment markers. In contrast, manually annotated datasets generally are expensive, in terms of time and money, which greatly limits the size of training data. Furthermore, manual annotation is often inefficient and error-prone as detecting emotions in tweets can be difficult even for humans.
- RQ2a: The size of the training data has a substantial effect on the performance of deep neural networks, which tend to require very large amounts of training data. On the other hand, it can provide comprehensive coverage of emotional moments in our daily lives. Based on our experiments with different sizes of training data (randomly sampled from our dataset), we demonstrated that by training deep neural networks with more data, their performance continues to improve for both sentiment polarity and emotion recognition tasks.
- RQ2b: Regarding the class imbalance issue, it is characteristic of almost all real-world datasets, and our dataset makes no exception. The number of emojis that belong to the joy emotion class is larger compared to the emojis used to query other emotion classes. As a result, we got a larger number of tweets for the joy class compared to other classes. As a consequence, the performance obtained from our experiments with the imbalanced dataset was biased by the high proportion of the dominant class (joy class). To overcome the imbalance problem, for the set of experiments conducted in this study, we intentionally balanced the dataset by randomly selecting an equal number of instances among all classes.
- RQ3: We demonstrated that pre-trained word embeddings such as Glove, Glove Twitter, and FastText have a substantial impact on the performance of deep neural networks.

Table 21 Precision, Recall, and F1 score for sentiment classification balanced dataset

Class	P(%)	R (%)	F1 (%)	# of instances
Negative	72.27	71.36	71.81	688,204
Positive	69.48	70.43	69.95	637,167
Weight avg	70.93	70.91	70.92	1,325,371

> Specifically, the findings reveal that BiLSTM with FastText pre-trained word embeddings and an attention layer provided the best performance on our dataset for sentiment polarity and emotion classification tasks, with an F1 score of 70.92% for sentiment and 54.85% for multiclass emotion classification (anger, fear, joy, and sadness). However, regarding the attention mechanism, the findings revealed that it has no (or less) impact on the performance of our models for sentiment and emotion classification tasks.

However, the study examines further possible improvements with regard to the quality of the collected data (tweets) along the following lines.

- (i) Because of the very large size of the dataset, we were not able to manually verify all the emotion tweets, and it is known that data obtained by distant supervision is often noisy. We should further investigate applying some heuristics to remove irrelevant tweets and incorrect annotations.
- (ii) Our newly created dataset does not contain tweets with neutral labels, which is a common problem faced by automatically collecting training data for emotion analysis, as there are tweets (text) that convey no emotion. We should further investigate to find a solution to automatically identify collected neutral tweets.
- (iii) The dataset collected in its original form is imbalanced. The number of emojis that convey joy emotion is a few times larger compared to the number of emojis for other emotion classes. To have a well-balanced dataset, one possible way is to design a more efficient collection approach that concentrates much more on collecting tweets from minority classes.

Regarding the performance of the classifiers, based on the experimental results, deep neural networks (1D-CNN, LSTM, GRU, BiLSTM, and CNN_BiLSTM) and transformer-based (BERT and RoBERTa) models generally outperform conventional machine learning models (NB, LR, Linear SVC, DT, and ADB). This advantage can be attributed to the capabilities of deep neural networks and transformer-based models to learn multiple layers of representations (multiple feature learning) that improve data mining results and classification modeling (Bengio et al., 2013, 2009).

It is worth mentioning that the performance of all the deep learning models utilized in this study is improved using pre-trained word embeddings such as Glove, Glove Twitter, and FastText, but there was no (or less) improvement in using the attention mechanism.

Despite the better classification performance that deep neural networks and transformer-based models offered on our sentiment and emotion classification task, there are still certain benefits of using conventional machine learning models for these tasks. Other benefits of using these models include being easier to implement, generally requiring less data for training, and being financially and computationally cheap as they can run on legacy CPUs.

These findings suggest that in general, the results are inspiring given the fact that the tweets are characterized by several challenges when it comes to automatic natural language processing tasks including sentiment and emotion analysis. These challenges include both technical and linguistic-related aspects such as short texts originally restricted to 140 characters (extended to 280 characters from Nov. 2017), creative uses of language (sarcasm, irony, humor, and metaphor), terms not seen in dictionaries, including misspellings, creatively spelled words, hashtagged words, emoticons, and abbreviations, etc., and many of these terms convey emotions (Mohammad, 2021).

7 Conclusion

In this study, we explored the tasks of sentiment polarity and multiclass emotion classification. We presented and evaluated the use of emotion-indicative emojis to automatically label a large-scale dataset of tweets with basic categorical emotions they express based on Ekman's model. We created this extensive dataset by selectively collecting only English tweets that contain emotion-indicative emojis and tagging each tweet using a distant supervision approach with emojis that are present in tweets for sentiment and emotion analysis purposes.

Supervised conventional machine learning (NB, LR, Linear SVC, DT, and ADB), deep neural networks (DNN, CNN, LSTM, GRU, BiLSTM, and hybrid CNN-BiLSTM), and transformer-based (BERT and RoBERTa) models were used for both sentiment polarity and emotion classification from users' tweets on the created dataset.

The experimental results showed that BiLSTM with FastText pre-trained word embeddings and an attention layer outperforms all other deep learning and conventional machine learning-based models in our dataset for both sentiment polarity and emotion classification. It yielded an F1 score of 70.92% for sentiment polarity classification and an F1 score of 54.85% for the multiclass emotion classification task.

In addition, we investigated the effect of pre-trained word embeddings such as Glove, Glove Twitter, and FastText on deep neural networks. It has been demonstrated that for the BiLSTM architecture, the FastText pre-trained word embeddings provide the best results for the task of sentiment and emotion classification.

We also investigated the effect of increasing the size of training data for deep neural networks and conventional machine learning. We demonstrated that for deep neural networks, training them with more data, their performance continues to increase for both sentiment and emotion classification tasks. These findings are in line with the results reported in Ng (2017).

Furthermore, we explored the effects of having multiple classes on classification performance. The study has confirmed that multiclass classification is difficult and associated with several challenges that dropped the accuracy from about 73% (weighted average for binary classifications) to 41.4% (multiclass classification on six basic emotions). These results are also in line with the findings presented in Bouazizi and Ohtsuki (2019). The findings demonstrate that there is a strong correlation between emojis and emotion annotations in tweets and our method used for automatic labeling was suitable for some emotions such as anger, fear, joy, and sadness, but less able to distinguish others such as surprise and disgust.

In future work, we will focus more on developing an efficient collection approach that would address the class imbalance issue during the data collection phase. We will also focus on introducing any heuristics or approach to further clean the dataset from irrelevant tweets and introduce a neutral emotion class. Additionally, experimenting with any new and larger deep learning architectures, and pre-trained word embedding models would be interesting to further investigate in the future. Furthermore, experimenting with any unsupervised or weak supervised learning paradigms would be of interest to explore in the future.

Author Contributions Muhamet Kastrati: Conceived the idea, performed all experiments related to the study, and reported results. Zenun Kastrati: Supervised experiments, and contributed to writing. Ali Shariq Imran: Contributed to writing and improved the overall language of the paper. Marenglen Biba: Provided a review of the paper and improved the overall language of the paper.

Funding Open access funding provided by Linnaeus University.

Data Availability The dataset is available upon request.

Declarations

Competing interests The authors declare no competing interests.

References

Aragon, M.E., Lopez-Monroy, A.P., Gonzalez-Gurrola, L.-C.G., & Montes, M. (2021) Detecting mental disorders in social media through emotional patterns-the case of anorexia and depression. *IEEE Transactions on Affective Computing*

Batra, R., Kastrati, Z., Imran, A.S., Daudpota, S.M., & Ghafoor, A. (2021). A large-scale tweet dataset for urdu text sentiment analysis. arXiv:2021.03057

Batra, R., Imran, A. S., Kastrati, Z., Ghafoor, A., Daudpota, S. M., & Shaikh, S. (2021). Evaluating polarity trend amidst the coronavirus crisis in peoples' attitudes toward the vaccination drive. *Sustainability, 13*(10), 5344.

Bengio, Y., et al. (2009). Learning deep architectures for ai. *Foundations and Trends® in Machine Learning, 2*(1), 1–127.

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(8), 1798–1828.

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics, 5*, 135–146.

Bouazizi, M., & Ohtsuki, T. (2019). Multi-class sentiment analysis on twitter: Classification performance and challenges. *Big Data Mining and Analytics, 2*(3), 181–194.

Byrkjeland, M., Lichtenberg, F. G., & Gambäck, B. (2018). Ternary twitter sentiment classification with distant supervision and sentiment-specific word embeddings. In: Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 97–106

Canales, L., Daelemans, W., Boldrini, E., & Martínez-Barco, P. (2019). Emolabel: semi-automatic methodology for emotion annotation of social media text. *IEEE Transactions on Affective Computing*

Colnerič, N., & Demšar, J. (2018). Emotion recognition on twitter: Comparative study and training a unison model. *IEEE Transactions on Affective Computing, 11*(3), 433–446.

Davidov, D., Tsur, O., & Rappoport, A. (2010). Enhanced sentiment learning using twitter hashtags and smileys. In: Coling 2010: Posters, pp. 241–249

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805

Edalati, M., Imran, A.S., Kastrati, Z., & Daudpota, S.M. (2021). The potential of machine learning algorithms for sentiment classification of students' feedback on mooc. In: Proceedings of SAI Intelligent Systems Conference, pp. 11–22. Springer

Ekman, P. (1993). Facial expression and emotion. *American Psychologist, 48*(4), 384.

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford, 1*(12), 2009.

Imran, A. S., Daudpota, S. M., Kastrati, Z., & Batra, R. (2020). Cross-cultural polarity and emotion detection using sentiment analysis and deep learning on covid-19 related tweets. *IEEE Access, 8*, 181074–181090.

Islam, J., Ahmed, S., Akhand, M., & Siddique, N. (2020). Improved emotion recognition from microblog focusing on both emoticon and text. In: 2020 IEEE Region 10 Symposium (TENSYMP), pp. 778–782. IEEE

Kang, X., Shi, X., Wu, Y., & Ren, F. (2020). Active learning with complementary sampling for instructing class-biased multi-label text emotion classification. *IEEE Transactions on Affective Computing*

Kapoor, K. K., Tamilmani, K., Rana, N. P., Patil, P., Dwivedi, Y. K., & Nerur, S. (2018). Advances in social media research: Past, present and future. *Information Systems Frontiers, 20*(3), 531–558.

Kastrati, M., & Biba, M. (2021). A state-of-the-art survey on deep learning methods and applications. *International Journal of Computer Science and Information Security (IJCSIS),19*(7)

Kastrati, M., Biba, M., Imran, A.S., & Kastrati, Z. (2022). Sentiment polarity and emotion detection from tweets using distant supervision and deep learning models. In: International Symposium on Methodologies for Intelligent Systems, pp. 13–23. Springer

Kastrati, Z., Ahmedi, L., Kurti, A., Kadriu, F., Murtezaj, D., & Gashi, F. (2021). A deep learning sentiment analyser for social media comments in low-resource languages. *Electronics, 10*(10), 1–19.

Kralj Novak, P., Smailović, J., Sluban, B., & Mozetič, I. (2015). Sentiment of emojis. *PloS One, 10*(12), 0144296.

Krommyda, M., Rigos, A., Bouklas, K., & Amditis, A. (2020). Emotion detection in twitter posts: a rule-based algorithm for annotated data acquisition. In: 2020 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 257–262. IEEE

Kusal, S., Patil, S., Kotecha, K., Aluvalu, R., & Varadarajan, V. (2021). Ai based emotion detection for textual big data: Techniques and contribution. *Big Data and Cognitive Computing, 5*(3), 43.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv:1907.11692

Mohammad, S.M. (2021). Sentiment analysis: Automatically detecting valence, emotions, and other affectual states from text. In: *Emotion Measurement* (pp. 323–379). Elsevier

Mohammad, S.M., & Bravo-Marquez, F. (2017). Wassa-2017 shared task on emotion intensity. arXiv:1708.03700

Mohammad, S. M., & Kiritchenko, S. (2015). Using hashtags to capture fine emotion categories from tweets. *Computational Intelligence, 31*(2), 301–326.

Mohammad, S. M., & Turney, P. D. (2013). Crowdsourcing a word-emotion association lexicon. *Computational Intelligence, 29*(3), 436–465.

Ng, A. (2017). Machine learning yearning. 139.http://www.mlyearning.org/(96)

Pennington, J., Socher, R., Manning, C.D. (2014). Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543

Plutchik, R. (1980). *A general psychoevolutionary theory of emotion.* (pp. 3–33) Elsevier

Polignano, M., Basile, P., Gemmis, M., & Semeraro, G. (2019). A comparison of word-embeddings in emotion detection from text using bilstm, cnn and self-attention. In: Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization, pp. 63–68

Schoene, A.M., Bojanić, L., Nghiem, M.-Q., Hunt, I.M., & Ananiadou, S. (2022). Classifying suicide-related content and emotions on twitter using graph convolutional neural networks. *IEEE Transactions on Affective Computing*

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research, 15*(1), 1929–1958.

Suttles, J., & Ide, N. (2013). Distant supervision for emotion classification with discrete binary values. In: International Conference on Intelligent Text Processing and Computational Linguistics, pp. 121–136. Springer

Teja, R. (2021). Twitter-Sentiment-Analysis-and-Tweet-Extraction. GitHub

Wang, W., Chen, L., Thirunarayan, K., & Sheth, A.P. (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. IEEE

Wood, I., & Ruder, S. (2016). Emoji as emotion tags for tweets. In: Proc. of the Emotion and Sentiment Analysis Workshop, Portorož, pp. 76–79

Yousaf, A., Umer, M., Sadiq, S., Ullah, S., Mirjalili, S., Rupapara, V., & Nappi, M. (2020). Emotion recognition by textual tweets classification using voting classifier (lr-sgd). *IEEE Access, 9*, 6286–6295.

Zimbra, D., Abbasi, A., Zeng, D., & Chen, H. (2018). The state-of-the-art in twitter sentiment analysis: A review and benchmark evaluation. *ACM Transactions on Management Information Systems (TMIS), 9*(2), 1–29.

Zucco, C., Calabrese, B., & Cannataro, M. (2017). Sentiment analysis and affective computing for depression monitoring. In: 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 1988–1995. IEEE

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