2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT) | 978-1-7281-8595-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICCCNT51525.2021.9579844

Emotion Detection Using Bi-directional LSTM with an Effective Text Pre-processing Method

Sumanathilaka TGDK
Dept. of Software Engineering
Informatics Institute of Technology
Colombo 06, Sri Lanka
deshankoshala@gmail.com

Viggnah Selvaraj
CSED
NIT, Calicut,
India
viggnahselvaraj@hotmail.com

Uddav Raj
CSED
NIT, Calicut,
India
raj.uddhav2509@gmail.com

Venkatesh Raju P
CSED
NIT, Calicut,
India
vennevan@gmail.com

Jay Prakash
CSED
NIT, Calicut,
India
jayprakash@nitc.ac.in

***Abstract*—In a real-life scenario, extracting emotion from unstructured text is an active and challenging area of research. It has diverse applications in various aspects of our daily life To overcome various challenges involved in detecting emotion from text, researchers from diverse fields applied various machine learning algorithms. However, deep learning methods such as long short-term memory is effective to detect emotion by maintaining the sequence structure of the text. In this work, we use Bi-directional long short-term memory with attention layer for emotion detection for better accuracy for prediction. In addition, we employ a text preprocessing method to improve further results. We perform the experiments on three data sets and the models are evaluated based on the classification accuracy.**

***Index Terms*—Bi-directional Long Short-term memory, Text to Speech Synthesis, Emotion Detection, LSTM.**

I. Introduction

In affective computing, the field of involving emotions in computing, emotion detection from text has emerged as an important domain. Its applications are wide-ranging, including measuring the emotional closeness of interpersonal ties using affective language in social networks, in marketing, predicting purchase intentions of customers and gauging brand reputation using emotional states, removing inappropriate posts from social media [4]. Emotion detection from text neatly fits as a crucial intermediate step in many applications, one application which we would like to highlight is emotional text-to-speech (TTS) synthesis [6] since there is growing interest in the community about this task. The idea of speech synthesis has been around for a long time. Text-to-speech synthesis (TTS) is to date a challenging task because voice produced by these systems sound robotic and is easily distinguishable from human voices. Even though there has been some considerable research to generate natural sounding voice, generating emotional speech is still a relatively new field. Emotional TTS has many applications like assisting the visually impaired, and emotion detection from text is an important module in this process. The process of emotion detection starts from defining what emotions are exactly. There is no consensus among psychologists as to how to define and categorize emotions. There are various models like the categorical and dimensional models of emotions [14]. We will attempt to generalize our model such that with only minor modifications it could be used on datasets that use both models. Over the years many computational approaches for this problem have been proposed, such as lexicon-based, and machine learning based [7][1]. However, these models are not capable of maintaining the sequence structure of text to detect emotion effectively with better accuracy. A deep learning model Long Short-term memory (LSTM) has the capability to maintain sequence structure of the text. In addition, Deep learning methods [15] are becoming popular, since the rapid rise in computing power, and have shown promising results. In this work, we employ the Bi-directional Long Short Term Memory (Bi-LSTM) [5] to detect emotion from text as it is capable to perform bi-directional learning. In addition, we apply a text preprocessing method to for the further improvement of the model performance.

The rest of this paper is organized as follows. Section II presents a brief literature survey. Section III includes methodologies to detect emotion from text. Section IV presents results and discussion. Finally, conclusion, limitation, and future direction of research are presented in Section V.

II. LITERATURE REVIEW

A. Emotion detection from text

The emotion detection from text pipeline primarily consists of three parts, choosing an emotion model to follow, identifying and aggregating relevant datasets for the emotion model chosen and applying a computational approach to perform the task of accurately determining emotions on given text. The emotional models in use today in the field of emotion detection can be broadly categorized into three types [2].

- Categorical: Assumes humans capable of only a limited, finite set of emotions.

- Dimensional: Emotions represented as points in dimensional space, subject to variables like valence and arousal. Eg: Russells Circumplex model (valence and arousal), Mehrabians model (pleasure, arousal,and dominance) [4].
- Extended models: Considers personality, targets and desires of the communicating party, helpful for individual-based emotion recognition.

In the literature, many computational methods have been proposed to detect emotions accurately from text. These methods can be broadly categorized into three approaches, Lexicon-based, Machine learning and Deep learning [7][1][15].

1) Lexicon-based Methods: The detection of the emotion of words is done primarily using words related to different emotions. In a keyword-based approach, an emotion has particular words associated with it that help in classifying the sentence to one of the emotions. Another method assigns a probabilistic affinity for an emotion to arbitrary words rather than detecting predefined emotional keywords from a text. This method called "lexical affinity" is an extension of the keyword-based method. These methods are quite basic and face challenges such as: Ambiguity in the way keywords are defined, inability to recognize sentences without keywords and lack of linguistic information. [14] [7] are examples of a keyword-based implementation using POS (part of speech) tagging as well.

2) Machine Learning: Machine learning can be broadly defined as inference of decision rules from a database of labeled training samples for the task of recognizing emotions. This overcomes challenges posed by lexicon-based methods. Random forests and support vector machines are commonly used as models in this approach. Usually, these models are used in conjunction with linguistic features to increase accuracy.Supervised learning approaches rely on a labelled training data. The supervised learning algorithm analyses the training data and infers a function, which we use for mapping new examples. In unsupervised learning, the algorithms attempt to detect hidden structures in unlabeled data in order to build models for emotion classification.It tries to compute an emotional vector for words based on semantic relatedness to other words [1][8].

3) Deep Learning: This a relatively new approach as the emergence of deep learning in the recent past has motivated researchers to try it out in a variety of domains, including natural language processing (NLP). Kratzwald et al. [9] attempt to use deep learning to improve the accuracy of emotion detection across various datasets. They use a recurrent neural network architecture (LSTM in uni and bidirectional), with dropout and a weighted loss function, trained on word embeddings (GloVe). Note that these word embeddings do not take sentiment into account unlike works such as [11][10] which attempt to incorporate sentiment features also into the embedding. They introduce a novel method in transfer learning where they transfer neural network parameters trained on sentiment analysis (only positive or negative emotion) to the task of emotion detection.

Majumder et al. [12] use word-level to sentence-level to document-level aggregation to classify documents into one of five personalities. They use a CNN to extract features, it is important to note that they trained five different networks for each of the personality types.

B. Text to speech Synthesis

The typical TTS pipeline consists of two parts: 1) Text analysis and Speech synthesis. Text analysis, also called the front end deals with Natural Language Processing (NLP) to extract the semantic and syntactic meaning of the text to convert it into phonemes and other intermediate representations that can be consumed in the next stage to produce speech waveforms. The next stage, the back end, is done with a vocoder, which usually takes parametric input and generates speech. Over the years many techniques have been used for TTS. Some of the major techniques are as follows.

- Concatenative speech synthesis with unit selection, where small units of pre-recorded waveforms are tied together to form a single coherent waveform. Usually phonemes, diaphones, or phrase-based audio act as units. These are retrieved from a database according to the requirement of the input text.
- Statistical parametric speech synthesis directly generates smooth trajectories of speech features to be synthesized by a vocoder. This attempts to solve boundary artifacts (at the point of concatenation) which occur with the use of concatenative speech synthesis.
- Since we now possess very high computing power, deep learning methods are be-coming popular, and have shown promising results.

After detecting the emotion, an approach for expressing emotion in the speech synthesizer. Initially the model uses several linguistic resources which can be used to recognize emotions in a text and assigns appropriate parameters to the synthesizer to carry out a suitable speech synthesis. To incorporate the linguistic information an XML based markup language is used for audio script building. In the paper [6] the authors review the current methods of emotional speech synthesis, with particular focus on explicit control. Speech synthesis is done to obtain neutral speech which is then "explicitly controlled" to get emotional speech. The parameters upon which such speech can be modified are mainly prosodic and excitation parameters. The paper [13] proposes a generative model for the second part of the TTS pipeline, speech synthesis. The model,"Wavenet", is conditioned on linguistic features such as phone, syllable, word, phrase, and utterance-level features (e.g. syllable stress, position of the current syllable in a phrase, the number of syllables in a word, and phone identities) which are derived from input texts and also on the logarithmic fundamental frequency (log F0.). Its architecture consisted of dilated causal convolutions to increase the receptive field and also to take care of long-range temporal dependencies [13]. There have been recent attempts to incorporate a neural network architecture for the whole pipline. Char2Wav [13] is one of the first attempts at an end-to-end speech synthesis model that can produce speech from text directly. It consists of two parts: Reader and a Neural vocoder. The main point of deviation from the classical pipeline of TTS, is that this model does not generate intermediate linguistic features, rather it directly generates vocoder acoustic features which arefed to the neural vocoder. The reader is a bidirectional recurrent neural network and the neural vocoder is a recurrent neural network (RNN) with attention [1].

Here, we employed Bi-directional Long Short Term Memory (Bi-LSTM) [5] for emotion detection from text. To improve the performance, we performed the text pre-processing before training the Bi-LSTM model.

III. METHODOLOGY

Our design was flexible enough take into account two of the predominant emotional models: categorical and dimensional. We used a deep learning architecture to accurately identify emotions in datasets. In order to convert the words into a suitable intermediate representation that can be fed into the neural network, we used word embeddings. We also incorporated latent sentiment features into these embeddings as in [11][10] to improve performance. It is important to note that in text processing usually stop words (commonly occurring words) are removed and stemming is applied to standardize word representation. We did not follow this, instead, we used approach mentioned in [11], because certain stop words may indicate sentiment and the model learns similar representations for words of the same stem when the data suggests it. We identified that a recurrent neural network is suited to consider temporal dependencies, but due to potential instabilities during optimization, we followed common choices that advocate the use of long short-term memory networks. We extended the recurrent neural network architecture presented in [3] as in [9], to include dropout layers for regularization and use a weighted loss function to deal with class imbalance. We also incorporated the attention mechanism [18], to increase the accuracy. Finally, a fully connected dense layer was used for the voting process, the output of which was then passed through an activation function. Changes in this layer made our model flexible enough to adapt to both categorical (using a SoftMax activation function) and dimensional (using an affine transformation) emotion models.

IV. RESULTS AND DISCUSSION

We conducted the experiments using Python [1] programming Language.

A. Data sets

We used three datasets for evaluating our model; the ISEAR dataset and two twitter tweet datasets. ISEAR dataset contains emotional sentences categorized into 7 emotions whereas the Twitter dataset contains tweets about election categorized in to 8 emotions. Data sets are taken from https://www.kaggle.com for experimental purpose.

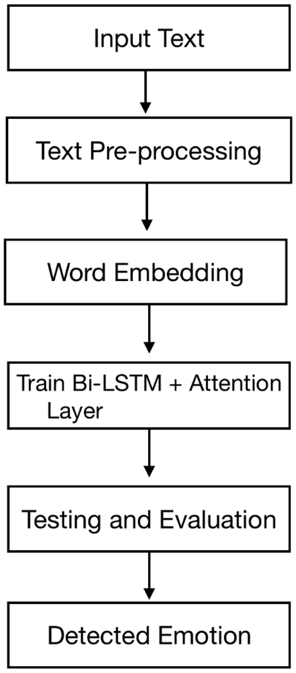

Fig. 1. Proposed Method

B. Performance Measure

We used Classification Accuracy [17] as the evaluation measure. The accuracy can be defined as the number of correct predictions divided by total number of instances in the test dataset.

C. Experimental results and Analysis

We selected Ekman's model [14] of six basic emotions (happiness, sadness, anger, disgust, surprise, and fear) for the experimental purpose. Table I and Table II show the obtained experimental results on the data sets. We use classification accuracy to measure the performance of the model. In Table I, we compare the performance of a machine learning model K-nearest Neighbor (K-NN), and two deep learning models Convolutional Neural Network (CNN) [16] and Long Short-Term Memory (LSTM). To improve the model performance, we pre-process the text data before training the model. Here, the Performance of models is measured with text preprocessing and without text preprocessing. We can observe that LSTM based model is performing better with respect to other competing algorithms. We can also observe that K-NN performs poorly as it is not capable of maintaining the sequence structure of the text.

Further, We compare the performance of Bi-directional Long Short Term Memory model without text pre-processing and with text pre-processing. As a result, with text pre-processing we observe significant improvement in Bi-directional Long Short Term Memory for all the selected data sets which are shown in Table II. In addition, Bi-directional Long Short Term Memory with text preprocessing outperforms

[1] https://www.python.org/

TABLE I
MODELS PERFORMANCE FOR DETECTING EMOTION

Method (Data set)	Accuracy (without preprocessing)	Accuracy (with preprocessing)
KNN (ISEAR dataset)	55.06	59.81
CNN (ISEAR dataset)	63.5	65.8
LSTM with CNN phase (ISEAR dataset)	64.6	65.9

over methods mentioned in the Table I. This indicates that bi-directional learning with a forward and backward pass of Bi-LSTM is more effective over LSTM with unidirectional learning.

TABLE II
PERFORMANCE OF BI-LSTM OVER VARIOUS DATASETS

Dataset	Accuracy of Bi-LSTM (without preprocessing)	Accuracy of Bi-LSTM (with preprocessing)
ISEAR	75.83	82.53
Twitter Tweet	71.06	79.81
Twitter Tweet (Old)	59.22	62.47

V. CONCLUSION AND FUTURE DIRECTIONS

Emotion detection from unstructured text is diverse applications in real life. We employed Bi-LSTM to perform emotion detection effectively over other selected deep learning methods as BiLSTM has bi-directional learning capability. We further observe that a suitable text preprocessing gives a high impact on the performance of the model.

As a limitation of this work, the detection of emotions is mainly focused on Ekman's model of six emotion categories, the work may be extended further for more categories of emotion detection. The model may be improved further for emotion detection and the work may be further used to improve the performance of text to speech synthesis.

REFERENCES

[1] Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat. "Emotions from text: machine learning for text-based emotion prediction". In: *Proceedings of human language technology conference and conference on empirical methods in natural language processing*. 2005, pp. 579–586.

[2] Ondřej Bruna, Hakob Avetisyan, and Jan Holub. "Emotion models for textual emotion classification". In: *Journal of physics: conference series*. Vol. 772. 1. IOP Publishing. 2016, p. 012063.

[3] Huimin Chen et al. "Neural sentiment classification with user and product attention". In: *Proceedings of the 2016 conference on empirical methods in natural language processing*. 2016, pp. 1650–1659.

[4] Salma Elgayar, Abdel ElAziz A Abdelhamid, and Zaki T Fayed. "Emotion Detection from Text: Survey". In: (2017).

[5] Lidia Ghosh, Sriparna Saha, and Amit Konar. "Bi-directional Long Short-Term Memory model to analyze psychological effects on gamers". In: *Applied Soft Computing* 95 (2020), p. 106573.

[6] D Govind and SR Mahadeva Prasanna. "Expressive speech synthesis: a review". In: *International Journal of Speech Technology* 16.2 (2013), pp. 237–260.

[7] Abdul Hannan. "Emotion Detection from Text". In: *International Journal of Engineering Research and Development* 11.7 (2015), pp. 23–34.

[8] Ubeeka Jain and Amandeep Sandhu. "A review on the emotion detection from text using machine learning techniques". In: *Int J Curr Eng Technol* 5.4 (2015), pp. 2645–2650.

[9] Bernhard Kratzwald et al. "Deep learning for affective computing: Text-based emotion recognition in decision support". In: *Decision Support Systems* 115 (2018), pp. 24–35.

[10] Igor Labutov and Hod Lipson. "Re-embedding words". In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 2013, pp. 489–493.

[11] Andrew Maas et al. "Learning word vectors for sentiment analysis". In: *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*. 2011, pp. 142–150.

[12] Navonil Majumder et al. "Deep learning-based document modeling for personality detection from text". In: *IEEE Intelligent Systems* 32.2 (2017), pp. 74–79.

[13] Aaron van den Oord et al. "Wavenet: A generative model for raw audio". In: *arXiv preprint arXiv:1609.03499* (2016).

[14] Kashfia Sailunaz et al. "Emotion detection from text and speech: a survey". In: *Social Network Analysis and Mining* 8.1 (2018), pp. 1–26.

[15] Elvis Saravia, Hsien-Chi Toby Liu, and Yi-Shin Chen. "DeepEmo: Learning and Enriching Pattern-Based Emotion Representations". In: *arXiv preprint arXiv:1804.08847* (2018).

[16] Matla Suhasini and Badugu Srinivasu. "Emotion detection framework for twitter data using supervised classifiers". In: *Data Engineering and Communication Technology*. Springer, 2020, pp. 565–576.

[17] CM Suneera and Jay Prakash. "Performance Analysis of Machine Learning and Deep Learning Models for Text Classification". In: *2020 IEEE 17th India Council International Conference (INDICON)*. IEEE. 2020, pp. 1–6.

[18] Lei Zhang, Shuai Wang, and Bing Liu. "Deep learning for sentiment analysis: A survey". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8.4 (2018), e1253.