Personalised Emotion Detection from Text Using Machine Learning

A. V. Bhavya, R. H. Dhanush, J. Sangeetha, and Arun Cyril Jose

Abstract This research discusses an emotion recognition system, which is an important component of many effective computing technologies for natural language processing, based on open-source platforms with automatic speech recognition (ASR) and text analysis, and which is used for user-based customised sentiment analysis. PocketSphinx as ASR and Word2vec model, K-means clustering, and TfidfVectoriser for text automatic analysis are used to design the proposed framework. Further, the dataset that is used for testing and training the model is from International Survey on Emotion Antecedents and Reactions (ISEAR). This research yields a user-dependent system that will function as a tailored assistant for identifying emotional responses and discovering innovative applications. The suggested model greatly outperforms the prior models, with an efficiency of 81% and an f-measure of 89%.

Keywords Emotion detection · Text analysis · Machine learning · Word2vec model

1 Introduction

Models and classifiers fail to generalise when training and testing settings differ, which is one of the major challenges in emotion recognition [1]. This challenge is seen when the data from a person in the testing dataset is not confined when training. In reality, earlier studies have shown that speaker-dependent classifiers perform better than speaker-independent classifiers [1]. This statement implies that speaker dependencies are present in the articulation of emotions. Although there

A. V. Bhavya · R. H. Dhanush · J. Sangeetha · A. C. Jose (✉)
Department of Computer Science and Engineering, Indian Institute of Information Technology, Kottayam, Kerala, India
e-mail: avbhavya18bcs@iiitkottayam.ac.in; sangeethaj2017@iiitkottayam.ac.in; aruncyril@iiitkottayam.ac.in

J. A. Marmolejo-Saucedo et al. (eds.), *Computer Science and Engineering in Health Services*, EAI/Springer Innovations in Communication and Computing,
https://doi.org/10.1007/978-3-031-34750-4_10

are common patterns among speakers, these traits are insufficient to create reliable emotion identification systems [2]. The issue is that practical procedures have strong classifiers that can generalise to the suggestive speech of unseen speakers. The paper discusses the issue of adapting an emotion identification system to a specific user. The use of feature and/or model adaptation for emotion recognition is an intriguing strategy [2]. The fact that the model adaption frame requests data from the user with emotional labelling limits their usage in many practical operations [3]. This study introduces an unsupervised model to adapt an emotion recognition system to a particular user in this setting. The suggested technique seeks to reduce the discrepancy between aural information retrieved from the targeted speaker utilised during testing and those features used during training. A validated emotion model is required to increase the realism of the recognition process since it can effectively handle the variety of emotions that can arise from different scenarios depending on the individual [4]. The accuracy of traditional styles in the environment of SER (speech emotion recognition) is comparatively low [4]. Supervised learning requires manually annotated data, which is frequently time-consuming and not always possible. We're proposing a hybrid model addressing these issues. The ISEAR (International Survey on Emotion Antecedents and Reactions) dataset is used to build the emotion detection system. Videos from a well-liked videotape-participating website, containing colourful interviews from a targeted subject, are downloaded, and 1.5 hours of speech from the targeted speaker is extracted [5]. The experimental results indicate that the proposed system gives accuracy of 80% (approx.). This paper has the following contents: In Sect. 2, the related works for this project are discussed. Section 3 contains the architecture of the proposed model. Section 4 discusses the technique used to implement the architecture. Section 5 details about the dataset, and Sects. 6 and 7 discuss how the implementation is done and the results obtained.

1.1 Contributions

Here, we implement unsupervised training, without any prior assumptions. We have used transfer learning method (Word2vec algorithm), which increased the accuracy to 80%. In our work, we made emotion personalised to each individual. In this work, primarily, text is used for emotion detection, and if the provided speech as input, it is handled by converting speech to text.

2 Literature Review

Semantic information may be extracted using semantic analysis, ontologies can be created using emotion models, and case-based reasoning can be used to adopt new

keywords [5]. Their suggested architecture aims to offer improved flexibility for various domains and systematic processing of text input.

Semantic analysis: To retrieve extra information about phrases, comparable to dependency trees, semantic analysis uses methodologies of natural language processing such as statistic-based parsing. On the basis of sentence verbal information, keyword extraction was suggested [5].

Sentiment models: It defines the underpinning research required to link the results of semantics to plausible emotions. An OCC framework, which covers 22 emotion types, is employed. Implementing these algorithms into semantics resulted in a more methodical framework for analysing various linguistic feeds [5].

Case-based reasoning: It was advised for embracing new concepts and applications. The source is compared with every case stored in storage in a case-based reasoning system, and the distances of both the intake and every case are calculated using techniques to determine the amount of resemblance. The approximation approach was not created, and the information and instance criteria were not completed manually, which impacted the efficiency [5].

The keyword-based component of a hybrid architecture [6] is mostly based on the knowledge engineering approach for information extraction. Using the sentence separator, tokeniser, and POS tagger to capture syntactically and semantically data, hidden phrase patterns may be discovered and contextual analysis improved. Furthermore, the training module builds selected features and predicts accuracy using the LibSVM standard. The machine learning method is split into two stages. The initial stage is to develop an emotion prediction training model. The learnt model will be used to predict the sentiment class on a testing dataset. However, they then separated people based on industry-specific prejudice.

The goal of normalising features is to reduce speaker variability while keeping the capacity to differentiate between various emotional classes. The discrepancies between the audio characteristics utilised to train and test the emotion identification system were reduced [6], resulting in the anticipated outcome. They also used the front-end of the iterative feature normalisation (IFN) approach. Even after using the IFN, however, many samples were wrongly labelled. Shaheen and Hajj [14] performed a difficult syntactic and semantic analysis of the text and utilised a range of ontologies, including WordNet and ConceptNet, to detect emotions. WordNet and ConceptNet are used to help their classifier generalise the training data, which enhances emotion coverage. Their classifier is context sensitive due to syntactic and semantic examination of the phrase.

Based on the process of detection, it is separated into three types:

Approach according to keywords: Methods according to keywords are used at the fundamental word level. They require a lexicon of emotional words that pairs words with labels for the associated emotions [7]. Approach based on rules: Using this strategy, different rules are developed to form a linguistic structure that is helpful in determining emotion [8]. Approach based on learning: This technique articulates the challenge of identifying sentiment from message as a set of input texts with feelings as categories. The supervised technique trains a classifier using

hand-labelled datasets, which is then applied to other groups. Unsupervised methods do not require any labelled data [8].

Based on the data source used, it is separated into three types:

Knowledge-based approach: These approaches use synonyms of lexical resources to ascertain the mood or polarity in terms, phrases, and documents [9]. Approach based on corpora: It is a repository of vast and organised sets of information, typically tied to a specific topic or author [10]. Fusion approach: It is a type of hybrid method which employs both the previously mentioned approaches [10].

Based on the user perspective, it is divided into two types:

Reader perspective: From a reader's perspective, one particular text segment can evoke multiple emotions [11]. From a writer's perspective: In the perspective of the writer where one text segment portrays only one emotion [11].

To recognise the effect as six fundamental emotions, Anusha and Sandhya [1] combined machine learning and natural language processing methods. But lengthy sentences cannot be utilised with it. Feature selection and extraction, feature categorisation, acoustic modelling, unit-based recognition, and language-based modelling are some of the components that make up a SER system [12]. The different nonlinear parts that make up deep learning techniques carry out computation in parallel. To address the drawbacks of other procedures, these approaches still need to be more deeply layered in their framework. Convolutional neural networks (CNN), deep belief networks (DBNs), recurrent neural networks (RNN), recursive neural networks (RvNN), deep Boltzmann machines (DBMs), and auto encoders (AE) are some examples of deep learning techniques, used for SER [13] that significantly improves the overall performance of the designed system.

Deep Boltzmann machines (DBMs) are made up of a variety of hidden layers and are primarily derived from Markov random fields. These layers are based on variables that were selected at random and linked to stochastic elements. The primary benefits of DBM are its propensity to learn quickly and provide useful representation. This is accomplished through layer-by-layer pretraining [13].

An RNN is a type of neural network that uses sequential information and has interdependent outputs and inputs. This interdependency is typically helpful in anticipating the input's future state. The RNN's susceptibility to gradient disappearance is the primary issue affecting its overall performance [14].

Recursive neural network (RvNN) is a hierarchical deep learning algorithm that does not rely on a tree-structured input sequence. By breaking up the input into manageable bits, it may quickly learn the parse tree of the supplied data [14].

Deep belief network (DBN) is constructed from cascading RBM structures and has a significantly more complex structure. RBMs are extended into DBNs, where RBMs are bottom-up taught layer by layer. Due to their capacity to learn the recognition parameters fast, regardless of how many parameters there are, DBNs are typically utilised for speech emotion recognition. Additionally, it prevents layer nonlinearity.

These layer-wise frameworks for deep learning algorithms are succinctly created based on the classification of various natural sentiments. These methods provide

simple model training in addition to the effectiveness of added weights. Deep learning approaches include limitations such as being their huge internal layer-by-layer architecture, poorer effectiveness for temporally variable input data, and overlearning during layer-by-layer memory [15].

Iliev and Peter [5] applied data retrieval based on artwork by using sentiment from voice. The ability to extract sentiment from speech and use it to identify and recommend art has been proven to be quite useful. To locate and suggest digital content, they used speaker sentiment. They started the process of creating a conceptual model of an ecosystem for digital culture that serves several purposes. It is founded on pre-defined text queries for metadata. For multiclass text analysis and classification, Ramya H. R. and Dr. Mahabaleswara Ram Bhatt employed PocketSphinx [16] as an automatic speech recogniser, human-computer interaction (HCI) [16], and linear support vector machine (LineaSVM). Future directions of the study could involve applying ASR based on neural networks and convolution neural network (CNN) techniques to noisy environments. More data, including a database of annotated emotions, Twitter data, and other sources, should be used to train the text analyser. By dividing sentiment into positive and negative categories, CNN with sentiment analysis can be utilised to broaden research investigation. Using hybrid classifiers, such as NB-SVM and long-/short-term memory techniques, improves accuracy. A dialog-based or interaction-based emotion recognition system can be developed for usage in practical applications to execute the HMI cycle. Sentiment analysis levels, different emotion models, and the process of sentiment analysis and emotion identification from text were all employed by Pansy Nandwani and Rupali Verma [9]. Deep learning approaches include limitations such as their huge internal layer-based design, lower efficacy for temporally variable input data, and overlearning during layer-based information memory. Another significant issue is that it might be challenging to infer polarity from comparison phrases.

3 Methodology

We provide a customised emotion detection model to improve the realism of the recognition process. The model's flow is as follows: If the source exists in the shape of voice signals, it tries to classify emotional reactions collected via ASR in conjunction with textual analysis, and if the intake is text, it may be categorised straight by this concept.

In this paper, PocketSphinx is used for speech recogniser, while a word embedding model is used for categorisation. PocketSphinx is optimum lightweight recogniser developed in C language, and it promotes portable, memory optimisation, ease of use, and wide vocabulary continuous voice recognition. PocketSphinx can be used in android and in offline mode too. PocketSphinx ASR requires an audio input in .wav file format, and it is transformed into feature sets of MFCC. This MFCC are utilised in a knowledge base to map and create specialised text output, which includes a phonetic thesaurus, learning algorithm, and acoustics models.

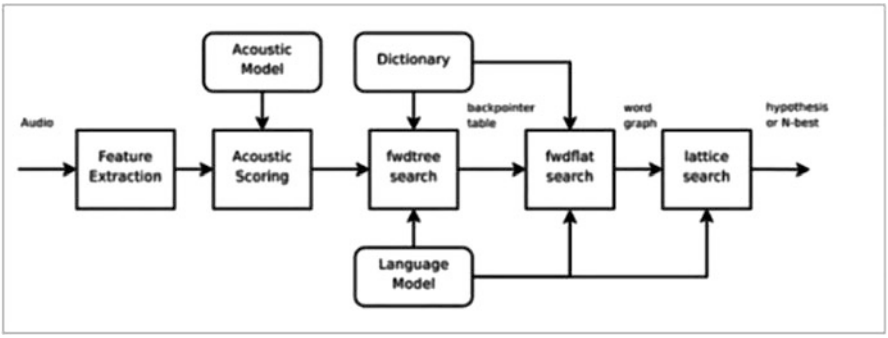

Fig. 1 PocketSphinx [11]

The three primary characteristics of the ASR model are as follows:

A phonetic dictionary in the system provides the semantics mapping terms to patterns of phonemes. A dictionary should have all of the terms you're looking for, or else the classifier will be unable to do so to recognise them. Even putting the terms in a thesaurus is insufficient. The recognition system looks up a word in the lexicon and the learning algorithm. Even if a term is in the dictionary, it will not be detected without the learning algorithm [17] (Fig. 1).

The linguistic model plays a crucial component of the structure since it tells the decoder which word sequences are feasible to identify. There are various varieties of models, each with its own set of capabilities and performance characteristics. Any decoding mode that meets your needs can be chosen, and you can switch between them at any moment [17].

Acoustic model is a hidden Markov framework statistical model. Again, the Gaussian distribution is used to represent the states' HMM output. PocketSphinx offers a variety of acoustic adaption techniques, including maximum a posteriori probability (MAP) and maximum likelihood linear regression (MLLR), which were used for our investigation. We can see that both MAP and MLLR are capable of designing speech systems. Furthermore, as the size of data expands, MAP adaptability outperforms MLLR. MAP is defined in this work as in equation:

$$\text{MAP} =_{\theta}^{\arg\max} f\,(x|\theta)\, g\,(\theta) \tag{1}$$

Word2vec learns word associations from a vast corpus of text using a neural network model. It can identify synonyms after being educated. Each word is represented by a specific collection of figures known as a vector by Word2vec. Word2vec supports two models [18]. The design of the CBOW predicts the present term from a window of words in the surrounding. The arrangement of the environmental terms has a little impact on prediction. Given the current word, the model in the skip structure estimates the neighbouring window of surrounding terms.

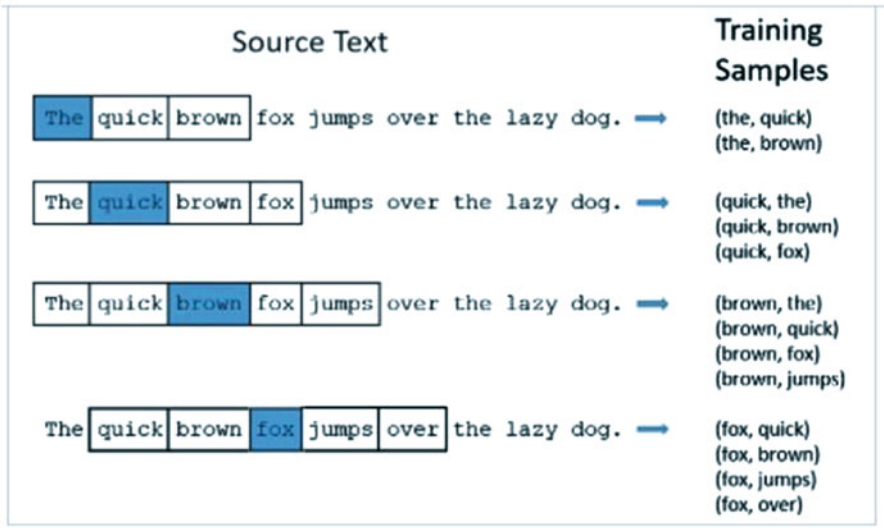

Fig. 2 Word2vec [8]

The fake task: We will train the neural network to look at the words around a certain word in the centre of a phrase (the input word) and choose one at random. The system will provide the likelihood of each term in our lexicon containing the "proximate term" that was selected. This is taught to neural network by feeding it terms from our training content. The picture below depicts some of the phrase-derived training instances (word pairs). The window size is set to 2. The entered word is highlighted in blue [18].

The neural network architecture:

Because it is difficult to feed a word to a neural network as a text string, a method for representing the words to the network is required. To do this, a lexicon of terms from our training manuals is constructed. Assume we have a vocabulary of 10,000 distinct words [18]. A single-word input, such as "books", is represented as a one-hot vector. The above vector will have roughly 20,000 components, with a "1" corresponding to the phrase "books" but a "0" in any other locations. The output of the network is a single vector expressing the chance that a randomly chosen neighbouring word fits each word in our lexicon (Figs. 2 and 3).

Following the Word2vec model, K-means clustering is conducted on the previously generated word vectors, and the assigned values are based on the cluster towards which they relate. By multiplying the values by their closeness to the cluster, the weighted sentiment coefficient is calculated. Each word now has two vectors: one for tf-idf score and one for weighted sentiment ratings. Finally, the tf-idf score is computed, and the detected emotion is determined by identifying the dot product of the two vectors for each utterance. K-means is employed because it scales well to big datasets, will always converge, and adjusts to new examples and

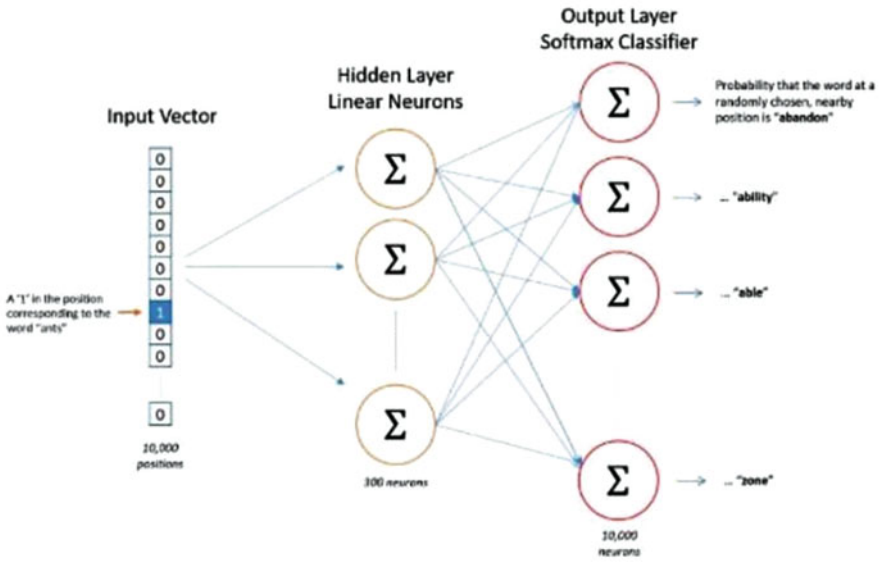

Fig. 3 Neural network [8]

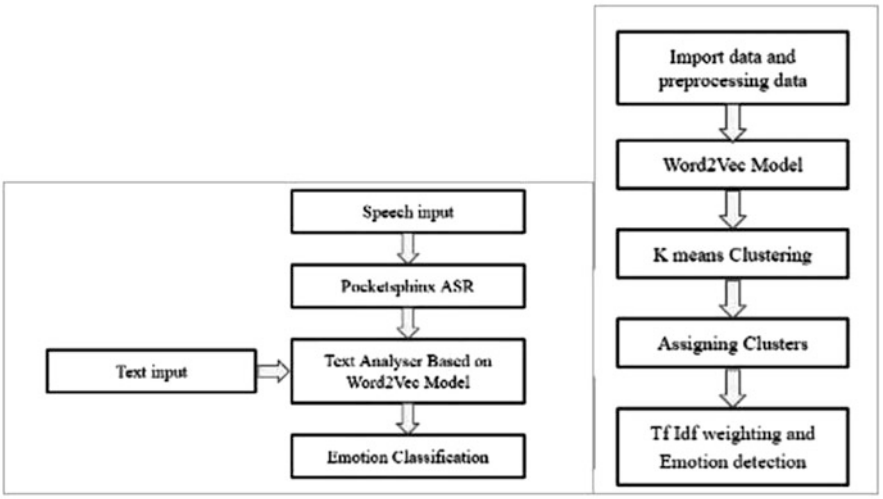

Fig. 4 Proposed architecture

TF. Idf is utilised because it assigns a value to a term depending on its significance in a paper, which is then scaled by its significance through all sources. Figure 4 depicts the suggested model.

4 Dataset

ISEAR is used to evaluate the proposed architecture. The ISEAR (International Survey on Emotion Antecedents and Reactions) has seven basic emotion groups. Anger, contempt, fear, guilt, pleasure, embarrassment, and grief are among them. There is just one categorisation label for each sentence. In the ISEAR project, coordinated by Klaus R. Scherer and Harald Wallbott, a varied collection of psychologists throughout the world provided data.

5 Implementation

The set of data that was utilised in this study was ISEAR. The Word2vec algorithm's gensim implementation with skip gram architecture was used. It was trained with a 6-word lookup window, 20-word negative sampling, 1e-5 subsampling, and a learning rate declining from 0.03 to 0.0007. The K-means method was implemented using sklearn, using 50 repeated beginning points and 1000 rounds of re-assignments of the points to clusters. To get the score, we multiply it by the distance between them and their cluster centres. Following these procedures, a comprehensive thesaurus (as in shape of a pandas DataFrame) was created, for each term given its own score. The tf-idf score of every term in each phrase was then obtained using sklearn's TfidfVectoriser. This was done to examine how each word differed in each phrase. The intersection of these two-word vectors showed whether the overall feeling was positive or negative.

6 Results and Discussion

To anticipate human emotions, we introduced an unsupervised and tailored emotion detection architecture in this work. We are training and evaluating the suggested sentiment recogniser with a combination of ASR and text categorisation, and we are doing so with the ISEAR datasets. Table 1 is a summary of the outcomes. We proposed a tailored emotion detection architecture for emotion categorisation. This algorithm is fed a human phrase as feed, which is subsequently fed into the classification algorithm, which produces the anticipated emotion. Figure 5 shows an

Table 1 Various performance measures

Performance analysis	Model
Accuracy measure	0.80
Precision measure	0.99
Recall measure	0.79
F1 score measure	0.89

```
Enter User text:
 For instance, giving a kiss to your younger sibling daily after waking up
 in the morning and showing him how much you love them. For some
 happiness means loving life and seeing others happy. While some finds
 happiness in writing stories. Some conquer happiness in being
 simple yet the best person they can ever be. Everyone has their own
 unique way to feel happy by finding things that they never expected to find.
 PREDICTION: Happy
Enter User text:
 I am so angry at you!
 PREDICTION: Anger
Enter User text:
 I think i'm gonna be sick
 PREDICTION: Sad
```

Fig. 5 Predicted emotion

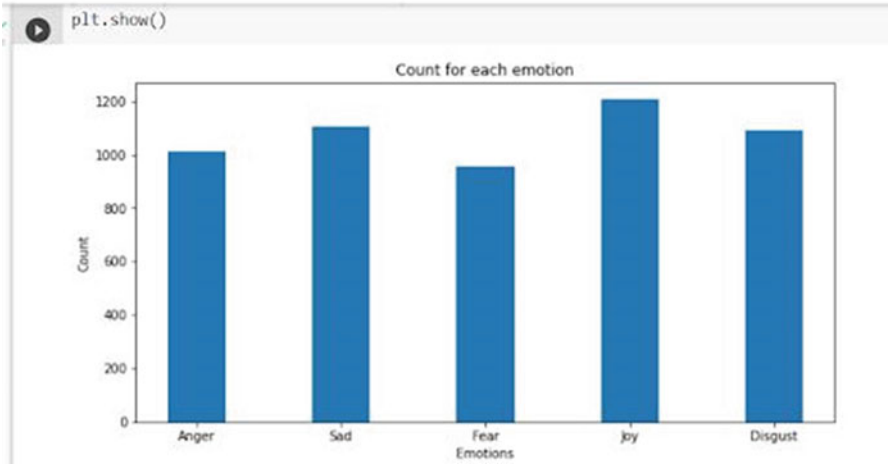

Fig. 6 Count for emotions

example of the obtained result. Figure 6 depicts the number of emotions we were able to achieve. The resulting prediction accuracy using the Word2vec approach was 80%, which is 10% higher than the model provided using the unsupervised feature adaptation scheme [8]. The achieved ac-curacy indicates that Word2vec produces the best results when applied with the appropriate settings as described previously. As it turned out, our model had a precision of 99%, indicating that it was quite effective at distinguishing negative sentiment observations. The algorithm also attained over 80% recall, implying that most of the positive samples were correctly recognised as positive, and an F1 score of 0.88, which is among the best. Based on the findings, we anticipate that when the data source grows more acquainted with user information, the feeling categorisation from text will be expanded to incorporate all of the users' feelings. We want to utilise it in an interactive journal,

and it could also be used as a virtual assistant in the future, with the obtained emotions functioning as a driver for the recommendation system, making the suggestion process more particular.

7 Conclusion

This research provides a model for emotion detection that combines ASR and text categorisation. For voice to text conversion, the open-source application PocketSphinx-based ASR and Word2vec-based text analyser were utilised. ISEAR (International Survey on Emotion Antecedents and Reactions) is the dataset used for emotion text analysis since it offers a global collection of emotion data. K-means and TfIdfVectoriser algorithms were employed. Our algorithm takes user text as input and predicts user emotion based on the content. According to the experimental data, the suggested system has an accuracy of (approx.) 80%. The supervised learning technique necessitates manually labelled data, which may be time-consuming and not always practical. Second, for underutilised NLP languages, such as Polish vocabulary, because there are no pretrained models to work with, repositories that have previously trained on a large prediction model are inaccessible. These concerns are addressed by our suggested hybrid algorithm.

8 Future Work

In the future, this algorithm might be integrated into a recommendation tool that uses emotion. Improvements include a third neutral cluster or assigning certain words that end up midway between the positive and negative clusters with a score of 0. Additionally, hyperparameter adjustment of the Word2vec method, depending on, for example, F1 score attained on dataset (albeit this would involve separating the dataset into train and test datasets, since the training would become supervised), may be done to enhance the model's accuracy above 80%.

References

1. Anusha, V., Sandhya, B.: A learning based emotion classifier with semantic text processing. In: Advances in Intelligent Informatics. Springer International Publishing (2015)
2. Binali, H., Wu, C., Potdar, V.: Computational approaches for emotion detection in text. In: 4th IEEE International Conference on Digital Ecosys-tems and Technologies (DEST) (2010)
3. Rachman, F.H., Sarno, R., Fatichah, C.: Corpus-based of emotion for emotion detection in text document. In: 3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE) (2016)

4. Aydin, C.R., Güngör, T.: Combination of recursive and recurrent neural networks for aspect-based sentiment analysis using inter-aspect relations. IEEE Access. **8**, 77820–77832 (2020)
5. Iliev, A., Stanchev, P.: Information retrieval and recommendation using emotion from speech signals. In: IEEE Conference on Multimedia Information Processing and Retrieval (MIPR) (2018)
6. Cho, K.H., Raiko, T., Ilin, A.: Gaussian-Bernoulli deep Boltz-mann machine. In: The 2013 International Joint Conference on Neural Networks (IJCNN) (2013)
7. Khalil, R.A., Jones, E., Babar, M.I., Jan, T., Zafar, M.H., Alhussain, T.: Speech emotion recognition using deep learning techniques: a review. IEEE Access. **7**, 117327–117345 (2017)
8. McCormick, C.: Word2vec tutorial – the skip-gram model (2016)
9. Nandwani, P., Verma, R.: A review on sentiment analysis and emotion detection from text. Social Network Analysis and Mining. **11**(1), 81 (2021)
10. Pantazoglou, F., Papadakis, N., Kladis, G.: Implementation of the generic Greek model for CMU Sphinx speech recognition toolkit (2017)
11. Rahman, T., Busso, C.: A personalized emotion recognition system using an unsupervised feature adaptation scheme. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2012)
12. Ramya, H.R., Bhatt, M.R.: Personalised emotion recognition utilising speech signal and linguistic cues. In: IEEE 11th International Conference on Communication Systems Networks (COMSNETS) (2019)
13. Rashid, U., Iqbal, M.W., Skiandar, M.A., Raiz, M.Q., Naqvi, M.R., Shahzad, S.K.: Emotion detection of contextual text using deep learning. In: IEEE 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT) (2020)
14. Shaheen, S., El-Hajj, W., Hajj, H., Elbassuoni, S.: Emotion recognition from text based on automatically generated rules. In: IEEE International Conference on Data Mining Workshop (ICDMW) (2014)
15. Kao, E.C.-C., Liu, C.-C., Yang, T.-H., Hsieh, C.-T., Soo, V.-W.: Towards Text-based emotion detection a survey and possible improvements. In: IEEE International Conference on Information Management and Engineering (2009)
16. He, Y., Udochukwu, O.: A rule-based approach to implicit emotion detection in text. In: Natural Language Processing and Information Systems. NLDB 2015. Lecture Notes in Computer Science, vol. 9103. Springer, Cham (2015)
17. Hua, Y., Guo, J., Zhao, H.: Deep belief networks and deep learning. In: IEEE Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things (2015)
18. Mikolov, T., Kombrink, S., Burget, L., Černocký, J., Khudanpur, S.: Extensions of recurrent neural network language model. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2011)