Expert Systems With Applications 198 (2022) 116686

Contents lists available at ScienceDirect

Expert Systems With Applications

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

A dual framework for implicit and explicit emotion recognition: An ensemble of language models and computational linguistics

Fereshteh Khoshnam [a], Ahmad Baraani-Dastjerdi [b,*]

[a] Department of Software Engineering, Phd Candidate, University of Isfahan, Isfahan, Iran
[b] Department of Software Engineering, Dr, University of Isfahan, Hezar-Jerib Ave., Isfahan, 81746-73441, Iran

ARTICLE INFO

Keywords:
Sentiment analysis (SA)
Opinion mining
Explicit emotion recognition (EER)
Implicit emotion recognition (IER)
Language model (LM)
Feature weighing
Dual framework
Ensemble method
Computational linguistics
Machine learning

ABSTRACT

One of the research domains in the field of sentiment analysis is automatic emotion recognition in texts which is a worthy topic in human-computer interaction. Text processing has always faced many challenges. The main one is the structural and semantic differences of sentences which have had a significant impact on the malfunction of auto-recognition systems. This problem becomes more prominent in short texts in which words and their concurrences are limited and insufficient. As a result of this, word frequency and TF-IDF weighing cannot well represent the relationship between words and the appropriate feature vector, leading to an undesirable accuracy of emotion recognition. Thus, different strategies should be applied to improve the feature vector and to formulate the features properly. The desired strategy should be able to identify the words that can distinguish between classes well and also to find the relationships between words and meaningful phrases using natural language processing concepts. In this paper, a combination of emotional models, categorical and hierarchical, are used for an emotional text recognition which could discover simultaneously explicit and implicit emotion in a short text. Our approach called DuFER, proposed a weighed method which improves the feature vector using language models and computational linguistics through applying a modified TF-IDF weighing to words as well as Maximum Likelihood Estimation weighing to expressions. Four implicit and explicit emotion datasets are used for the experiments. The results show that the accuracy of both implicit and explicit emotion recognition has increased and DuFER is actually the first successful dual framework in recognizing implicit and explicit emotions from text.

1. Introduction

As a prominent field in natural language processing, emotion recognition has an interesting and practical role in social communications and social media for publishing an idea through rapid verbal expression. It has many applications in academic and commercial fields. For instance, the results of emotion recognition systems can be used both as inputs to other systems and for recognizing the role of emotions in the human decision-making process. These kinds of applications make it necessary to create better tools for automatically recognizing emotions. Designing these tools has different challenges and issues including the following ones:.

1- **Informal writing in social media, especially in short text messaging:** because of the informality of most messages, they may contain spelling and grammatical errors. Although informal language has previously been studied in sentiment analysis, the use of such a language in the field of emotion recognition has been much less studied (Hasan, Rundensteiner, & Agu, 2019).
2- **Metaphors:** texts often contain metaphors and implicit language (Seyeditabari et al. 2017). Therefore, the introduction of appropriate approaches to discover and use these concepts is an important challenge.
3- **Monolingual frameworks:** generally, most metonymies and metaphors belong to particular languages and cultures (Kim & Klinger, 2018). Thus, providing a multilingual model is challenging.
4- **Inference conditions:** most emotional inferences can be made by interpreting the conditions in which they occur ((Kim & Klinger, 2018; Balahur, Hermida, & Montoyo, 2012). Recognizing these conditions has a significant impact on achieving the desired result.

Abbreviation: DuFER, Dual Framework for Emotion Recognition.

* Corresponding author.
E-mail addresses: f.khoshnam94@eng.ui.ac.ir (F. Khoshnam), ahmadb@eng.ui.ac.ir (A. Baraani-Dastjerdi).

https://doi.org/10.1016/j.eswa.2022.116686
Received 15 July 2021; Received in revised form 29 October 2021; Accepted 15 February 2022
Available online 4 March 2022

5- **Lack of sufficient labeled datasets** (Gaind et al., 2019).
6- **The complexity of human emotions:** human emotions are complex with close and common boundaries. Therefore, modeling and analyzing human emotional behavior are complex problems for automated systems (Gunes, Schuller, Pantic, & Cowie, 2011).
7- **Embedded emotions in the text:** many researchers in the field of cognition and emotion have confirmed the theoretical possibility that emotion regulation works at implicit levels. Implicit emotions can be defined as the process of acting without explicit and conscious intentions. The purpose of implicit emotions is to modify the quality, intensity, or duration of the emotional response (Koole & Rothermund, 2011). As a result, analyzing the implicit emotional process as embedded emotions of the text and its integration with explicitly expressed emotions infer complex emotional states (Siciliano & Clausi, 2020).

Considering these challenges, the style of writing may make the reader feel different from what the author intended. It can be said that the major challenge in automatic emotion recognition is to improve the recognition accuracy by solving each of the above challenges.

Researchers have presented different methods for the fifth case (Hasan, Rundensteiner, & Agu, 2019; Gaind et al., 2019; Strapparava, & Mihalcea, 2008). The third and the fourth cases have rarely been addressed so far and the second case has received very little attention. This study aims to investigate the second, fourth, sixth and seventh cases and develop a framework for improving accuracy.

Previous studies can be considered mainly in two different categories. The first group is defined solely on the basis of explicit emotion recognition (EER) from the text (Hasan, Rundensteiner, & Agu, 2019; Agrawal & An, 2012; Kiritchenko, Zhu, & Mohammad, 2014). The text contains emotional words and has one or more emotions.

The second group is based on implicit emotion recognition (IER). In this group, sentences do not contain any explicit emotional words (Gaind et al., 2019; Zhou & Wu, 2018; Adarsh, 2019).

Many studies have been done on the recognizing explicit emotions and less on implicit ones. The latest research studies using the deep learning method have surpassed machine learning tasks with acceptable and better results (Park, Bae, & Cheong, 2020; Feng et al, 2020; Chriqui & Yahav, 2021).

The model designed in this research, despite using the machine learning method, by considering the psychological and linguistic issues of emotion recognition, has succeeded in achieving equal and sometimes better results than deep-learning methods. Also, the simultaneous recognition of emotions, less complexity and process has made this model thought-provoking and expandable. A hybrid approach is considered to face the emotion recognition problem in this research. The issue has been addressed from two different and new perspectives:.

1- **We discover simultaneously explicit and implicit emotion in a short text.** EER methods cannot recognize implicit emotions in the text and act weakly if they are used for recognizing the implicit emotions (Kim & Klinger, 2018). On the other hand, considering psychological concepts, explicit and implicit emotions are not polar or quite apart from each other but they have porous boundaries. Emotions may be expressed in different ways at various times or conditions. The repeated use of an explicit emotion can motivate a person to express the sense implicitly. Implicit emotions are very reliable and less flexible (Gyurak, Gross & Etkin, 2011). Hence, the simultaneous discovery of these two groups (explicit and implicit emotions) in the text could appropriately reflect one's true feelings. Therefore, new insights into the emotion recognition of the text can be created and improved.
2- **We use combination of emotional models, categorical and hierarchical, for an emotional text recognition.** In Psychological literature emotions can be classified as categorical, dimensional or hierarchical model. Categorical approaches (e.g., Plutchik) suggest basic emotions. They are the most useful classification scheme for emotive language analysis in terms of training supervised machine learning algorithms (Williams, Arribas-Ayllon, Artemiou, & Spasić, 2019). Hierarchical approaches (lexical schemes e.g. WordNet), cover a wide range of words, different levels of synonyms for words and emotive expressions. Thus, they can provide more word-based analysis. Due to the ambiguity of natural language, hierarchical approaches can be useful in referring a specific emotion when a suitable generic category is not available in other schemes and this helps to minimize the ambiguity (Williams, Arribas-Ayllon, Artemiou, & Spasić, 2019; Seyeditabari et al., 2018). Given the advantages of the hierarchical model in resolving the ambiguity of natural language, the method presented in this study can recognize most of the words that implicitly express emotion. Words that express explicit emotions can also be identified. On the other hand, considering the appropriateness of the categorical emotional model in classification, the use of a combination of these models will greatly help to improve the recognition of textual emotions.

A two-stage framework for the emotional text recognition of the text is provided. In the first stage, for the first time, two types of weighing are applied to the text features (word or phrase). Secondly, a supervised hybrid classification is performed. The proposed method is an appropriate framework to identify the explicit and implicit emotions of short texts by taking advantage of linguistic concepts, special weighing features, extending existing vocabularies, combining language models, and machine learning. The proposed framework can be a new step towards finding the link between explicit and implicit emotions and improving the accuracy of the emotional text recognition. Hence, the appropriateness of the proposed framework was evaluated. For this purpose, the proposed approach was applied to several datasets and its performance was compared with those of earlier works in both explicit and implicit fields. The results showed that the performance of the proposed method was desirable.

This paper is structured as follows: Section 2 summarizes previous studies on emotional analysis. Details of the proposed method to analyze and recognize emotion in text are illustrated in Section 3. Section 4 explains our extensive experimental results of the proposed method on four datasets. Section 5 discusses the validity of the results. Finally, we conclude our paper in Section 6.

2. Related works

Emotional modeling is done with the aim of simulating human responses. Related works in this field have approached the issue from different perspectives. In this study, it is attempted to investigate previous studies from the perspective of explicit or implicit emotion recognition.

Most methods are defined in terms of recognizing explicit emotions from the text. In the work presented by Hasan, Rundensteiner, and Agu (2019), they used unigram, punctuation, negation, and symbols as the features to recognize explicit emotions with a supervised approach based on the Circumplex emotional model. Based on the results, only unigram was effective and the other features did not have any effect on improving the classification.

Strapparava and Mihalcea (2008), described the construction of a large dataset for the six Ekman emotions and provided several knowledge-based and corpus-based methods to identify the ones that work best for the annotation of emotions. The authors believe that the best evaluation results are obtained by a system that has deep syntactic analysis, and this may be the reason for the better results. In this study, the lexical structure of emotions has not been considered, it seems that the integration of deeper semantic processing of the text into the knowledge-based and corpus-based classification methods can lead to better results.

In another study, Agrawal and An (2012), presented an unsupervised

approach based on the Ekman model in which an emotional vector was calculated for each potentially effective word considering the semantic relation between various emotional terms and concepts. This method does not use annotated datasets and affect lexicons. According to the authors, one of the weaknesses of this approach is that the semantic relatedness scores depend on the text corpus from which they are derived. As a result, semantic relatedness between various emotional terms and concepts should be obtained from multiple measures which are not considered in this work.

Kiritchenko, Zhu and Mohammad (2014) tried to recognize emotions at two levels of short-message and word. Using a supervised method, they employed sentiments, surface-form, and semantic features in their system to discover explicit emotions. The authors were interested in applying and evaluating the lexicons generated from tweets on data from other kinds of text such as blogs and news articles.

Adarsh (2019) proposed an approach to improve emotional performance. In this supervised method, messages were classified using "word clusters" instead of "words" alone. In his study, a word clustering algorithm and weighed clusters were used as the features. This study suggests, thinking the intensity of the emotive words may improve the emotion recognition performances.

Zhong and Miao (2019), present a model on the task of textual emotion detection in SemEval-2019. Their method extends the Recurrent Convolutional Neural Network by using external fine-tuned word representations and DeepMoji sentence representations. They explored several pre-trained word and sentence representations including ELMo, BERT and InferSent but found inferior performance. Model requires no handcrafted features or emotion lexicons and achieved micro-F1 score of 0.7463.

Agrawal and Suri (2019), proposed NELEC, a system which combines textual and deep-learning based methods for sentiment classification in third task of SemEval-2019. System performs better than their deep-learning model benchmarks. It achieved a micro-averaged F1 score of 0.7765, ranking 3rd on the test-set leader-board. The authors emphasized that existing machine learning techniques have close performance to that of human on text-based classification tasks. The presence of multi-modal noise in chat data makes existing deep-learning solutions perform poorly. The inability of deep-learning systems to capture covariates has negative effect on their performances.

Recently a new trend of studies focusing on recognizing implicit emotions from the text has appeared in the field of emotion recognition. These studies are truly important in the field although the accuracy of their results is lower than those of EER approaches.

Zhou and Wu (2018) proposed a deep learning approach for implicit emotion analysis based on Ekman's theory. In order to make a system, they designed a BiLSTM-Attention model and an LSTM-Attention model. The final results were predicted by combining these two models with a soft voting strategy. The system developed in (Chronopoulou et al., 2018) provided a learning transfer method for recognizing implicit emotions. For this purpose, a set of untagged Twitter messages was used to learn different language models and word2vec features. The weight of pre-tested models was used to initialize the specific network layers. This model was complemented by a self-test mechanism based on LSTM networks. BrainT system in (Gratian & Haid, 2018) used the BOW feature, language models, and POS to predict the tweets' implicit emotions by testing a multi-class perceptron network. In this study, the researchers believed that textual preprocesses were not useful including stemming, lowercasing, filtering stopwords, emoji conversion, and emoticon.

EmotiKLUE is a deep learning system proposed in (Proisl et al., 2018) and was trained with the words at the left and right sides of the emotional word in the tweets. In this system, it is assumed that the distribution of emotions depends on the topic of the tweets. Therefore, at first, an LDA model was exclusively created to discover the subject matter of the Twitter messages. Then, to predict the implicit emotions, different ways were tested to combine the text topics with the neighboring words of the deleted emotional word. Ren et al (2017) proposed an inference-based method to recognize implicit emotional expressions. They made a logical system that broke down the process of recognition into a series of logical inference processes. The system aimed to extract emotions by applying inference knowledge.

Having carefully studying the related works in these two above-mentioned aspects and their methods, as well as checking existing emotional datasets, we can conclude that the sentences have always been considered as explicit emotional data or implicit ones. In daily life, however, sentences do not have this characteristic. People express their sentences without thinking explicitly or implicitly about their emotions. Therefore, it is very important to deal with the problem of recognizing emotions from this point of view. In fact, we should recognize the emotion of each sentence without paying particular attention to these two different kinds of emotions. In this study, we have explored the emotion of sentences from this perspective. In addition, Examination of machine learning methods and deep learning used in this field shows that finding specific features in a special subject and combining them with the learning method, will lead to better performance. Deep learning method does not always lead to better results and we should think twice before going deep (Agrawal and Suri, 2019).

3. Methodology

Emotion recognition is modeled as a text classification problem which could assign one or more emotional labels to a sentence. Accordingly, the proposed framework will provide a solution to the classification problem for recognizing emotions in short texts.

3.1. Overview of the framework

The proposed approach attempts to use feature-based and knowledge-based systems. As stated in Section 1, the main goal of the current study is to design a method that could infer emotions from the text even without using explicit emotional terms and assign the best emotional label to it. To better understand the problem, examples of both types of sentences are given.

Instances of the text contain emotional words and has one or more emotions:.

Example 1. *Can we go back two weeks and start again? This is seriously dreadful.*

Example 2. *Being stuck on the roof of your house provides an amazing view and the sheer terror of falling down.*

The second group is based on implicit emotion. In this group, sentences do not contain any explicit emotional words. For instance:.

Example 3. *I have a job interview in Loughborough next month.*

Example 4. *He broke the glass by throwing it to the wall.*

In the third example, the reader realizes that there is a low or high sense of concern even though there is no explicit emotional word. Example 4 delivers the feeling of anger.

This method works well for every sentence that combines two types of explicit and implicit emotions or for the texts that contain only one of these. The proposed framework, named DuFER, is presented in Fig. 1. Three main modules of the DuFER, preprocessing module, analysis module, and classification module are explained in sections 3.1.1, 3.1.2, and 3.1.3 respectively.

3.1.1. The preprocessing module

As you can see in Appendix, the preprocessing task, Preprocess (), consists of two procedures, text preprocessing and extending emotional lexicon, which are named Text-preprocess() and Extended-dictionary() respectively. The inputs of Preprocess () are dataset of tweets and three lexicons, NRC Emotion Lexicon, WordNet and NRC Hashtag

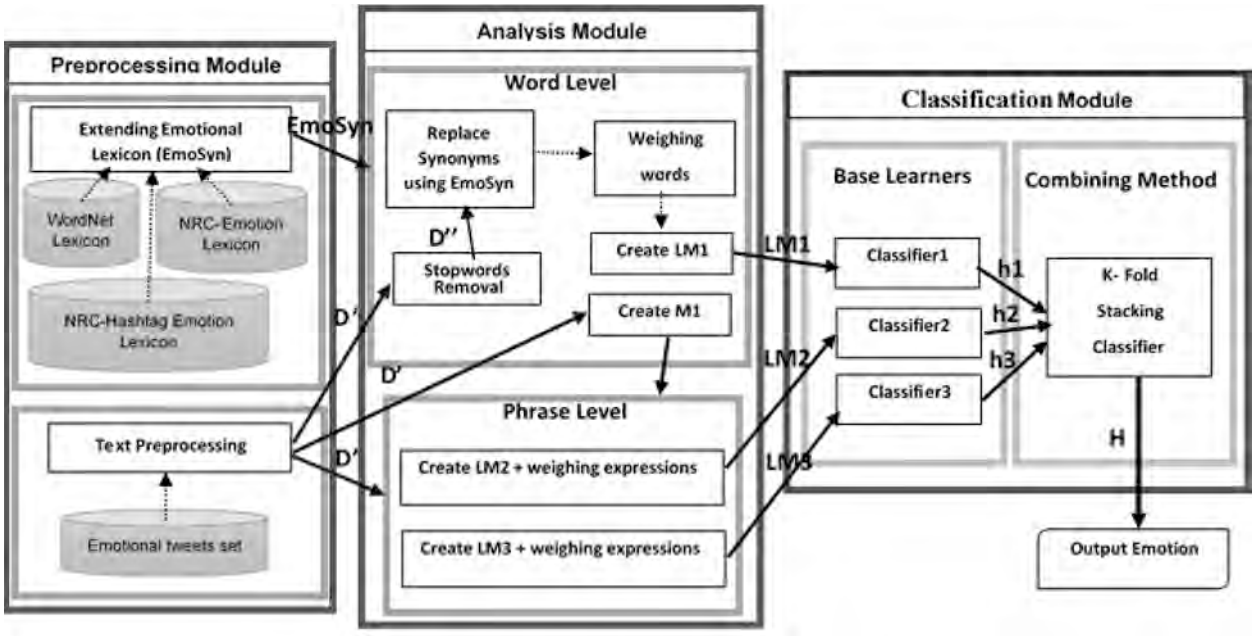

Fig. 1. Framework of DuFER, D' is the preprocessed dataset and D" is the preprocessed dataset which stopwords are removed, LM refers to the language model and the number written next to the LM indicates the n-gram of that language model, for example, LM1 means Unigram Language Model. hi is the result of the i^{th} base classifier. H is the final result.

Emotion Lexicon. How the procedures, Text-preprocess() and Extended-dictionary(), are worked are explained in the following.

1. **Text pre-processing (Text-preprocess()).** This procedure does tokenization, lemmatization, and text normalization for each input tweets. At first every word that does not belong to the main part of a tweet is removed and then sentence normalization is done for that. At the next step, all words are lemmatized. Finally, the preprocessed dataset, D', is returned back as an output.
2. **Extending emotional lexicon (Extended-dictionary()).** As mentioned in Section1, hierarchical[1] and categorical emotional models are used in the proposed DuFER. In order to discover implicit and explicit emotions of tweets, we need a proper emotional lexicon. Three emotional lexicons, NRC Hashtag Emotion Lexicon, NRC Emotion Lexicon (based on Plutchik which is Categorical model), and WordNet (Hierarchical model) are used as inputs of the procedure to create a new lexicon as output of the procedure named EmoSyn. In the following, the reason of using these lexicons and the process of creating the new lexicon and its properties are explained.

Several dictionaries have been described as the resources of the emotional domain. However, it can be difficult to achieve an acceptable emotional recognition merely using these resources, especially if the sentences do not have explicit emotional words and the emotions are expressed implicitly. For example, angry thoughts may be accompanied by verbal or physical expressions. Verbal expressions include *yelling, arguing, and cursing* which indicate explicit emotions. However, anger can also be expressed physically or implicitly by raising a clenched fist or by throwing, breaking, or hitting something (Kassinove & Sukhodolsky, 1995). As in the Example 4, there are no explicit emotional words.

Thus, it is difficult to identify the emotions of the person in the above sentence by emotional vocabulary or to find features to recognize emotions. However, it can be said that breaking can result from the hitting action and hitting shows anger. Such a deductive process will help to identify the emotions. Based on this idea, the extension of an existing emotional vocabulary is proposed so that the emotion of the sentence can be better undrestood by deducing its non-emotional words. For instance, in Example 4, searching for the word,'break' in this dictionary leads to Anger emotion. In making EmoSyn, the sense of "break" is deduced as follows:

Synset level in WordNet:	Level 1	→	Level 2	→	Level 3
	anger		hit		break

Which means that "break" has an implicit anger emotion. So, the sentence implicitly shows a sense of anger which is the successful result for EmoSyn. Or, in the Example 3 words 'interview' or 'job interview' are not in the EmoSyn dictionary and none of the other words in the sentence are expressed a special emotion.

Therefore, by using three existing dictionaries: NRC Hashtag Emotion Lexicon, NRC Emotion Lexicon and WordNet, a new dictionary is created named EmoSyn. For this purpose, emotional words from the Plutchik emotional model were taken and then synonymous words, up to three different levels were extracted from the WordNet dictionary. The steps are as follows:.

(a) Finding representative words of each emotion:

The Plutchik emotional model uses three words to express each of the eight introduced emotion, each of which representing the different intensity of that emotion. These words were originally intended to create EmoSyn.

(b) Hierarchical extraction of Synsets from Wordent:

1. By finding the closest Synsets to the representative words, first level synonyms were obtained. Note that in extracting data from Wordent, the threshold parameter was considered to select the most similar Synsets.
2. To extract the second level, for each of the synonymous words obtained in level one, new Synsets were extracted and its words were added to EmoSyn. The third level was obtained in the same way.

[1] The hierarchical models are also called lexicons schemes.

WordNet links words to semantic relations including synonyms, hyponyms, and meronyms. Synonyms are grouped into sets with brief definitions and practical examples. WordNet has 117 k Synset or category of synonyms (Global WordNet Association, 2020). Each category may have nouns, verbs, adjectives, and adverbs. Knowing this, our dictionary contains different types of words. For example, after examining up to three levels, Happy Emotion has 121 Synsets. Each has between 2 and 29 synonymous words. A total of 1107 synonyms are included in its list. Table 1 shows the number of Synsets and words extracted per emotion.

(c) The words in the NRC emotional lexicon are added to EmoSyn based on their emotion. Of course, duplicate words that resembled words extracted from WordNet were removed.
(d) Since it was intended to use datasets consisting of tweets, some hashtags from the NRC Hashtag Emotion Lexicon were also added. This extended dictionary, EmoSyn, has been used in this study.

The use of EmoSyn has increased performance by 2% which can be seen in Section 4.1.

3.1.2. The analysis module

Language models assign a probability to a word or sequence of text words (Jurafsky & Martin, 2019) which are known as high-level features. It can be said that some sentiment encoding features – hashtags, emoticons, and elongated words- which are suitable discriminating information are also could captured by them (Kiritchenko, Zhu, & Mohammad, 2014). As stated in Section 2 punctuation, negation, and symbols features did not have meaningful effect on emotion classification. Therefore, the n-gram language models are used in DuFER and their variants are named as LM1, LM2, and LM3 corresponding to unigram, bigram, and trigram language models. Basically, in language models, every sentence is a vector of all words or phrases that make it up. Generally speaking, text vectors can be categorized into two types: indexing and term-weighing. Indexing provides a meta language for describing a document, whereas weighing plays a significant role in correct text classification. Hence, DuFER will be an index-based weighing method which can generate information-rich terms and assign appropriate weights for each term. To explore the text more precisely in two levels: word-level and phrase-level which lead to the extraction of concepts and text relations, we use dataset of processed tweets D', dictionary of emotional words EmoSyn and three language models (LM1, LM2 and LM3) as inputs of the analysis module. The following is further description of what is done.

3.1.2.1. Word-level: Creating unigram language model (LM1). In the unigram language model, each sentence is a vector of its constituent words along with their term frequency (Jurafsky & Martin, 2019). In LM1, stopwords are removed from the dataset so that the repetitive words that do not provide proper information are not processed. Then by using the EmoSyn, words are replaced by their related emotions. Once the LM1 is constructed, the sentence vectors can be used for training and recognizing. However, in order to get better results, terms have been weighed.

In this research, we use a tweet as a short text which has a maximum of 140 characters and all datasets contain tweets. A tweet has little contextual information because of its length limitation. Therefore, a suitable criterion must be found for weighting words and phrases. TF-IDF criterion which stands for "Term Frequency – Inverse Document Frequency" is a technique to quantify a word in documents. TF represents the number of times that a term occurs in a document (Cambridge, 2009). IDF is the inverse of the document frequency which measures the informativeness of each term. Considering the short length of each tweet, TF-IDF criterion is very low for the most occurring words which reduces the accuracy of the classifiers. On the other hand, class labels are not considered in the TF-IDF weighing method during the calculation of the weight of a particular term. Thus, it has blind spots and is not a good enough criterion for weighing short texts (Ren & Sohrab, 2013).

In pursuit of a more suitable weighing criterion, we use contextual information for creating an information-rich term weighing method which can provide an effective classification. The method is used the significance or discriminatory power of each term for different classes and incorporated them into the weighing process. General terms may therefore be distributed almost uniformly across classes. However, a large number of discriminative terms occur in a particular class but rarely in other classes. In fact, it is expected that the distributions of the terms that inherently belong to a particular class are focused on that class. Hence, the weighing formula for LM1 consists of two parts: general terms weighing as calculating by Equation (1) and discriminative terms weighing which is computed by Equation (2). The weighing of general terms is explained by Equation (1).

$$w_s ti = \begin{cases} \dfrac{tf_s^{ti} - Min(tf_{tweet}^{ti})}{Max(tf_{tweet}^{ti}) - Min(tf_{tweet}^{ti})} & if\ Max tf_{tweet}^{ti} > 1 \\ \dfrac{tf_s^{ti}}{tf_D^{ti}} & if\ Max\ tf_{tweet}^{ti} = 1 \end{cases} \quad (1)$$

Where ti represents the ith term of dictionary of dataset D and tf_s^{ti} is the term frequency of term ti in tweets. $Min(tf_{tweet}^{ti})$ and $Max(tf_{tweet}^{ti})$ are the minimum and maximum term frequency of term ti in each tweet in dataset D respectively.

$Max(tf_{tweet}^{ti}) > 1$, means that term ti appears more than one time in at least one tweet in dataset D, so, the value of tf_s^{ti} is normalized by min–max method. There are synonymous words that represent one emotion in a tweet and its hashtags. Therefore, finding synonyms and replacing them by their emotion word in one tweet (as in the construction of LM1), increases tf_s^{ti} in tweet s and make $W_s(ti)$ better for weighing emotional word in each tweet. If$Max(tf_{tweet}^{ti}) = 1$, the term ti appears at most one time in a tweet and we weigh it by dividing to the number of occurrences in dataset D. For further understanding of Equation (1), consider the following example.

Example 5: *He broke the glass by throwing it to the wall #angry.*

Taking into account the inference method described in section 3.2, the words 'angry' and 'broke' are considered as the synonyms of the emotional word 'anger' and they are replaced by it in LM1. Therefore, the word 'anger' appears twice in this sentence. Now, assuming that $Max(tf_{tweet}^{anger})$ in the data set is equal to 3, then Ws(anger) = 2/3 in the Example 5.

Using contextual information in such a deductive way gives a higher weight to terms. Sometimes, even though the sentences are short, by replacing the synonyms from EmoSyn, we can observe that the term frequency of one emotional word (tf_s^{ti}) can reach to eight. On the other hand, the word 'glass' appears once in the example. Assuming that there is no tweet in the dataset D that contains word glass more than once, it

Table 1
Number of extracted Synsets, words from WordNet, NRC Emotion lexicon and NRC-Hashtag lexicon.

Emotion	Synset in WordNet	Number of synonym words			
		Extracted from WordNet – Repeated in NRC	NRC	NRC-Hashtag	Total
Anger	55	542	1247	5613	**7402**
Fear	49	876	1520	3813	**6209**
Joy	20	351	689	3434	**4474**
Sadness	18	493	1191	2561	**4245**
Trust	72	528	1231	1667	**3426**
Anticipation	61	514	839	3908	**5261**
Surprise	32	249	534	6031	**6814**
Disgust	15	304	1054	5362	**6720**

means$Max(tf^{ti}_{tweet}) = 1$, then there will be two different cases for weighing term (word) ti:.

1- The word is repetitive and has little information. Therefore, it will be given little weight.
2- The word is rare which has effective information and will help classification. In this case, a higher weight is assigned to that. This weighing positively discriminates rare terms from frequent terms.

In the Example 5, if 'glass' is a repetitive word and occurs 5000 times in dataset then we have Ws(glass) = 1/5000. Otherwise, if 'glass' is rare and occur 20 times in dataset then Ws(glass) = 1/20.

If a term is more likely to occur in a particular class, it is discriminative (Agrawal, & An, 2012). Thus, the probability value of the term in each class is calculated by Eq. (2) in order to add the importance degree of each term in different classes.

$$w_p(ti) = \frac{tf_c^{ti}}{tf_D^{ti}} \tag{2}$$

Where tf_c^{ti} is the number of ti replicates in class C and tf_D^{ti} represents the number of ti replicates in dataset D. W_P is calculated as a weight for each term ti to distinguish between classes. In the Example 5, if 'glass' is repeated 6 times in a specific class, then tf_c^{glass} is 6 and if 'glass' is a repetitive word in D, $tf_D^{glass}$is 5000. Hence, the Wp(glass) = 6/5000. But if 'glass' is a discriminative(rare) word, then Wp(glass) = 6/20. The difference is clear and this happens for the word anger too.

Finally, the weighing equation in LM1 is obtained by Eq. (3).

$$w(ti) = w_s(ti) + w_p(ti) \tag{3}$$

Therefore, in the Example 5, the final weight of 'glass' will be w (glass) = 1/5000 + 6/5000 if it is a repetitive word or w(glass) is equal to 1/20 + 6/20 if it is a discriminative(rare) word. Weighing words in LM1 by Eq. (3), is very effective to have a better classification.

To make LM1 and M1, matrices of unigram language model, we first remove stopwords from the dataset D' and nameD''. EmoSyn is used to find and replace the synonym words of each word ofD''. We then use Python libraries (n-gram, nltk.util) to create the matrix LM1 for D'' dataset and the matrix M1 for D' dataset (with stopwords). M1 will be used to create LM2 and LM3 matrices which explained in Section 3.3.2.

According to the author's point of view, stopwords can be helpful in extracting effective expressions in LM2 and LM3. Columns of LM2 and LM3 represent terms or unigram, bigram or trigram of a language model of the ith term. Each row represents one tweet of dataset. For each row and column of matrix LM1, *Ws* is calculated as shown the word importance in every sentence (tweet). The discriminative power, W_p, of each word for different emotional classes is also calculated. Finally, the weight of each word, W, is calculated and applied to every word in LM1. We now have the dataset LM1 with new weighted terms to use for classification. The following section shows how LM2 and LM3 are created.

3.1.2.2. Phrase-level: Creating bigram and trigram language models. Creating bigram and trigram language models is explained in the following:.

Bigram Language model.

Words are embedded in a larger structure such as sentences, phrases and expressions. In fact, if two words are repeated many times together, they tend to have semantic relevance (Strapparava & Mihalcea, 2008). A word or words around a term are usually used to better understand the meaning and sense of that term.

Automated systems may be able to detect some of the most basic rules. For example, subordinate words are immediately adjacent to the target word (Balahur et al., 2018). Considering this principle, the bigrams of each sentence were built with the help of Python libraries. In building the second language model, LM2, the aim is to create a model that can obtain the emotion of each sentence by discovering the common expressions in each emotional category. Human beings inherently think syntactically as what appears after "to" is usually a verb. The other rules may be facts such as the high probability of sentences starting with the word 'I' and some may even be cultural rather than linguistic (Jurafsky & Martin, 2019). As a result, these computational facts (finding and computing n-gram probabilities) are used for weighing the expressions to help improve the process of emotion classification.

In calculating these probabilities, the Maximum Likelihood Estimation (MLE) is estimated. This probability determines the co-occurrence degree of the terms. The MLE of a bigram is calculated by obtaining its number in the corpus and normalizing it so that it lies between 0 and 1. To calculate the probability of a particular bigram, Eq. (4) from Jurafsky and Martin (2019) can be used:.

$$P(w_n \mid w_{n-1}) = \frac{C(w_{n-1}w_n)}{\sum_w C(w_{n-1}w_n)} \tag{4}$$

Where W_n and W_{n-1} are n^{th} and n-1th words in sentence respectively. $P(w_n|w_{n-1})$ represents the probability of W_n given the previous wordW_{n-1}. $C(W_{n-1}W_n)$ is the frequency of bigram $W_{n-1}W_n$ and the sum of all bigrams that share the first word W_{n-1} is represented by$\sum_w c(w_{n-1}w)$. Eq. (4) can be simplified as follows (Jurafsky & Martin, 2019):

$$P(w_n \mid w_{n-1}) = \frac{C(w_{n-1}w_n)}{C(w_{n-1})} \tag{5}$$

Where $C(w_{n-1})$ represents the frequency of unigram W_{n-1} in dataset.

There are 10 different bigrams like 'throwing it' in sentence of the Example 5. Each bigram is weighed by Eq. (5). For instance, in the Example 5 at first, the number of repetitions of bigram ('throwing it') in all dataset is calculated which means $C(W_{n-1}W_n)$. Then the number of occurring the first word of bigram ('throwing') in all dataset, is calculated that is $C(W_{n-1})$. Finally, $P(w_n|w_{n-1})$ is obtained for bigram 'throwing it' by dividing these two numbers and placed in LM2 matrix which contains all the bigrams per tweet.

These calculations have been done for weighing the existing bigrams. Meanwhile, in the construction of this linguistic model, stopwords were not eliminated and synonyms were not replaced so that the author's syntactic thinking process could be accurately modeled and the facts be extracted and deduced.

Trigram Language model.

The language model that has less perplexity can give the greatest amount of information about the sequence of words. For this reason, trigram is generally used (Jurafsky & Martin, 2019). Using this linguistic model, the common emotional expressions in each category can be obtained. Therefore, with this knowledge in hand, LM3 was built and the expressions were weighed by MLE. For the general estimation of MLE for n-grams, in Eq. (6) Jurafsky and Martin, (2019) we have:.

$$P(w_n \mid w_{n-N+1}^{n-1}) = \frac{C(w_{n-N+1}^{n-1}w_n)}{C(w_{n-N+1}^{n-1})} \tag{6}$$

Eq. (6) estimates the n-gram probability by dividing the observed frequency of a particular sequence by the observed frequency of the prefix. The stopwords are not deleted and the synonyms are not replaced in constructing LM3. For instance, to weight the trigram "*throwing it to*" in the Example 5, at first the number of occurrences of this trigram is calculated. Then, the number of occurrences of its bigram prefix "*throwing it*" is obtain from LM2. Finally, by using Eq. (6), the MLE weight is calculated.

Considering three language models (LM1, LM2, and LM3) in DuFER, the use of the n-gram hierarchy and calculating the probability of them, help generalize model in unseen contexts (Jurafsky & Martin, 2019). So, two language models, LM2 and LM3, are built and weighed as follow.

Matrices of bigram and trigram language models, LM2 and LM3 respectively, are created for D' using Python libraries. The MLE weight of each trigram expression (W_1W_2W3) is calculated using count of W_1W_2W3 from LM3 and its bigram prefix (W_1W_2) count from LM2 and then these weights are applied to LM3 expressions. The MLE weight of bigram, W_1W_2, is also calculated by using counts of bigram in LM2 and unigram, W_1, in M1 and LM2 is updated.

Each row of matrices, LM2 and LM3, represents a tweet. Each column of the matrices, LM2 and LM3, is a bigram and trigram correspondingly. The weight of each n-gram, after calculation according to 3.3.2, is stored in matrix.

3.1.3. The classification module

The rule-based, classical learning-based, deep learning, and hybrid approaches have been introduced to recognize emotions in text. Studies show that hybrid approaches and learning-based approaches that utilize traditional text representation with distributed word representation outperform the other approaches on benchmark corpora (Alswaidan & Menai, 2020). So, in this research, a hybrid approach is considered which is used lexicon and classification method.

Classification is the data analysis task, where a model or classifier is constructed to predict class labels. Since single classification techniques generally have some limitations, in order to get better performance, ensemble models have been developed using a variety of techniques (Han, Pei, & Kamber, 2011). An ensemble combines a series of base classifiers with the aim of creating an improved composite classification model. Ensembles tend to be more accurate than their base classifiers. Each base classifier can be allocated to a different CPU and so ensemble methods can be parallelizable (Han, Pei, & Kamber, 2011). In the parallel method, multiple classifiers operate simultaneously. The results are then integrated with a combination algorithm on which the performance of the system depends (Kim, Kim, & Lee, 2003). We also use a parallel ensemble classifier in DuFER because it can reduce errors. The classification task consists of two parts, Base Classifiers and Ensemble classifier.

Base Classifiers Phase.

In the first phase of the classification module, three classifiers are used in parallel. Three feature matrices that have been built in the analysis module LM1, LM2, and LM3 are used by three classifiers. The result of each classifier assigns an emotional class to each tweet and produces the sub models.

Ensemble Phase.

In the second phase, it is attempted to combine the sub models to obtain the original model. Stacking which is a blended learning method is used to improve prediction. In fact, by using base classifiers as sub-models in the previous phase, the main predictive model is built at this phase.

In this phase, we first construct an n*4 matrix, D^h, where n indicates n tweets as rows of D^h and column 1, 2, and 3 of D^h are the emotional labels which classifier 1, 2, and 3 assigns to each tweet (rows) respectively. The column 4 of D^h is the actual emotional label of each tweet. Then, an ensemble classifier is trained on the matrix D^h.

4. Experimental results

In this section, the evaluation results of DuFER are reported and discussed in four standard datasets. In this regard, two different issues are examined and evaluated.

- The first issue is to evaluate the performance of DuFER on different corpora and to determine whether it is capable of recognizing explicit and implicit emotions as claimed.
- The second issue is to compare the accuracy of DuFER with those of other recent methods. It should also be determined whether DuFER can be trusted as an acceptable method for recognizing emotions.

4.1. Datasets

Four different bench datasets are used for evaluating the proposed method DuFER.

1. The dataset EmoTex was collected by Hasan, Rundensteiner, and Agu (2019), from Twitter and the tweets were automatically labeled based on the Circumplex emotional model according to the emotions of their authors. The properties of the dataset are shown in Table 2. We refer to this as EmoTex-dataset.
2. The dataset E-c at SemEval-2018 (Mohammad et al, 2018) with twelve different classes (anger, anticipation, disgust, fear, joy, love, optimism, pessimism, sadness, surprise, trust, neutral or no emotion) is prepared for the English part of the task that is the best presentation of the mental state of the Twitter. Each tweet can have one or more gold emotion labels. The first row of Table 3 shows structure of the dataset E-c at SemEval-2018.

The distribution of emotional classes on the E-c dataset is presented in Table 4. This dataset (Mohammad et al, 2018) assigns more than one label per tweet, hence, the total amount of labels will be more than 100%.

1. The Emotion Intensity Dataset, second row of Table 3, which its data are categorized into four emotional classes (joy, sadness, fear, and anger) (Mohammad, & Bravo-Marquez, 2017). Each tweet contains an emotional and numerical label that expresses the intensity of the emotion. This dataset contains four different emotions, data distributions are given in Table 5.
2. The Fourth dataset is prepared for the WASSA 2018 IEST (Klinger et al, 2018). For the first time, the proposed task is such that systems have to predict the emotions in a large automatically labeled dataset of tweets without access to words denoting emotions. Based on this intention, it is called the Implicit Emotion Shared Task (IEST) because the systems have to infer the emotion mostly from the context. Every tweet has an occurrence of an explicit emotion word that is masked. The tweets are collected in a manner such that they are likely to include a description of the cause of the emotion Data distribution is in accordance with Table 3. Since the training data have been accessible only to participants with usernames and passwords in this competition, this study only used the test data containing 28,757 labeled tweets for both training and testing DuFER.

Before we present the experiment results in the following sections, two points should be considered:.

1- In all experiments, k-fold cross-validation is used as a model validation method. It is mainly used when the goal is prediction (Kohavi, 1995). Given this, the k-fold cross validation stacking method in Perlich & Świrszcz, (2011) is used as the algorithm and since it is known that obtaining the optimal k-fold response depends on the value of k as well as the data distribution, the choice of k will be an important factor (Gutierrez-Osuna, 2005).
2- In order to choose ensemble classifier in DuFER, firstly different algorithms were tested on the task at hand. According to the results, for the final selection of classifiers, decision tree, Naïve Bayes, and Support Vector Machine (SVM), were selected and tested on three language models (LM1, LM2, and LM3) that are three of four text classification algorithms which introduce in Guia et al. (2019) as best

Table 2
Number of tweets collected as labeled data after preprocessing.

Class	Happy-Active	Happy-Inactive	Unhappy-Active	Unhappy-Inactive	Total
#Tweets	34,000	30,000	37,000	34,000	135,000

Table 3
The number of tweets.

Dataset	Train	Dev	Test	Total
SemEval2018 Affect Task E-c dataset	6838	886	3259	10,983
Emotion Intensity dataset	3613	342	3142	7097
Implicit Emotion Shared Task	153,383	9591	28,757	191,731

ones. Two dictionaries of emotional words – EmoSyn and Circumplex words – were used. 10-fold cross validation (Perlich & Świrszcz, 2011) is also used in the test. A part of test results is presented in Table 6.

As it shows, decision tree achieved the highest accuracy using LM1. SVM achieved the highest accuracy using LM2 and LM3. Decision tree provides high accuracy using LM2, LM3 too. Although its results are comparable to SVM, it is low. Therefore, we use Decision tree as classifier1 and SVM as second and third classifiers. Since SVM runs fast and provides the highest accuracy, we choose SVM as the ensemble classifier in our experimental.

4.2. Evaluation results of DuFER on the EmoTex-dataset

Emotex is the method name which presented in Hasan, Rundensteiner, and Agu (2019), and is similar to the name of its dataset. In this article, to avoid ambiguity, this method is called Emotex-method. Different emotional models have been used in DuFER and Emotex-method. In our emotion classification work, we utilized the Plutchik emotional model which has eight primary bipolar emotions: joy versus sadness; anger versus fear; trust versus disgust; and surprise versus anticipation. We also added love, optimism and pessimism emotions that can be seen in many tweets. In Emotex-method the Circumplex emotional model is used by considering four major classes of emotion: Happy-Active, Happy-Inactive, Unhappy-Active, and Unhappy-Inactive, including 28 emotional words. In order to compare the results of DuFER and Emotex-method on EmoTex-dataset, their equivalent emotional classes are presented in Table 7 using (Williams, Arribas-Ayllon, Artemiou, & Spasić, 2019). This table is a tabular representation of "Taxonomy Versus Folksonomy in (Williams, Arribas-Ayllon, Artemiou, & Spasić, 2019) which compare the wheel of Plutchik and Circumplex emotional models.

Emotex-method uses four features (unigram, emoticon, punctuation and negation) to recognize emotional classes. The classification results based on these features show that the three extracted features (except unigram) had no effect on improving the outcome. The evaluation results of Emotex-method and DuFER methods on the EmoTex-dataset are presented in Table 8 and Table 9, respectively.

As we can see that the best F1-Measure values is 90 which is related to SVM with using only unigram feature and Decision tree with using all features. These results show that the use of three features other than unigram has had little effect on improving recognition results.

Since there are three base classifiers and one ensemble classifier in DuFER, we evaluated DuFER by one decision tree and two SVM as base classifiers for the reason mentioned in Section 4 and SVM as the ensemble classifier and k-folds cross validation. First three columns of Table 9 show three base classifiers and corresponding language model which are used in experiments and the fourth column is ensemble classifier and k-fold. The results of evaluation of DuFER on EmoTex-dataset with two dictionaries Circumplex and EmoSyn are shown in the fifth and sixth columns of Table 9.

Emotex-method used Circumplex words to recognize emotions. For better comparison of Emotex-method and DuFER results, we also used the Circumplex words instead of EmoSyn dictionary in DuFER which is obtained better results than Emotex-method when it is used Circumplex words.

As we can see the results in Fig. 2, best results are obtained when DuFER used the EmoSyn dictionary. Furthermore, we experiment the Emotex-method with unigram feature and DuFER with classifier1 with weighing the unigram feature.

As we can see in Fig. 2, using just unigram feature in DuFER, we have better recognition than the Emotex-method. We can also see that the result of using the Circumplex words to classify tweets in DuFER is better than the Emotex-method. Using EmoSyn has made the recognition emotion even better.

Table 5
Percentage of tweets were labeled with a given emotion in EID dataset.

Emotion	Anger	Fear	Joy	Sadness	Grand Total
	23.97%	31.73%	22.70%	21.60%	100%

Table 6
F1-measure of Naïve Bayes, SVM, and Decision tree using different language models.

Classifiers		F1-measure		
		Naïve Bayes	SVM	Decision tree
Classifier1(using LM1)	Using Circumplex Words	89.4	90.63	91.01
	Using EmoSyn	90.01	92.05	92.24
Classifier2 (using LM2)		84.15	88.12	87.93
Classifier3 (using LM3)		83.4	86.46	86.17

Table 7
Equalizing emotional classes in Circumplex model and Emotional Classes (Williams, Arribas-Ayllon, Artemiou, & Spasić, 2019).

EmoTex Emotional Classes	DuFER Emotional Classes
Unhappy Active	Fear, Disgust, Anger
Unhappy Inactive	Sadness, Pessimism
Happy Active	Surprise, Joy, love
Happy Inactive	Trust, Optimism, Anticipation

Table 8
Results of Emotex-method on EmoTex-dataset (Hasan, Rundensteiner, & Agu, 2019).

Features	F1-Measure		
	Naïve Bayes	SVM	Decision tree
Unigram	86.3	**90**	89.5
Unigram + Emoticon	86.4	89	89.6
Unigram + Punctuation	86.6	89.9	89.7
Unigram + Negation	86.9	89.1	89.7
All Features	86.9	89.9	**90**

Table 4
Percentage of tweets were labeled with a given emotion.

Emotion	Neutral	Trust	Surp.	Sadn.	Pessi.	Optim.	Love	Joy	Fear	Disg.	Antic.	Anger
	2.7	5.0	5.2	29.4	11.6	31.3	12.3	39.3	16.8	36.6	13.9	36.1

Table 9
Results of DuFER on EmoTex-dataset.

Classifier1 & LM1	Classifier2 & LM2	Classifier3 & LM3	K-Fold Ensemble Classifier	F1-Measure	
				Using Circumplex Words	Using EmoSyn Words
Decision tree	Decision tree	Decision tree	SVM 5-Fold	90.87	91.6
Decision tree	Decision tree	Decision tree	SVM 10-Fold	92	92.8
Decision tree	Decision tree	Decision tree	SVM 20-Fold	91.76	92.3
SVM	SVM	SVM	SVM 5-Fold	91.1	91.76
SVM	SVM	SVM	SVM 10-Fold	93.1	94.3
SVM	SVM	SVM	SVM 20-Fold	93.05	93.91
Decision tree	SVM	SVM	SVM 5-Fold	91.29	91.87
Decision tree	**SVM**	**SVM**	**SVM 10-Fold**	**93.7**	**95.12**
Decision tree	SVM	SVM	SVM 20-Fold	93.02	94.77

4.3. Evaluation results of DuFER on the SemEval-2018, task: E-c dataset

In SemEval-2018, the Jaccard index was used as an evaluation criterion. In Equation (7) (Mohammad, Bravo-Marquez, Salameh, & Kiritchenko, 2018), G_t is the set of golden labels for tweet t, P_t is the set of predicted labels for tweet t, and T is a set of tweets.

$$Accuracy = \frac{1}{|T|}\sum_{t\in T}\frac{Gt \cap Pt}{Gt \cup Pt} \tag{7}$$

Since DuFER provides just one label for each tweet, the Jaccard similarity coefficient which is shown in Eq. (8) from Jaccard, (1912) was used instead of the Jaccard index to evaluate the results. For this purpose, DuFER is first used to estimate an emotion for each tweet and the set P is prepared by results. Then the new set P' is made of G (golden labels). The amount of the equivalent emotion class of each tweet t in P was extracted from G and placed inP'. Therefore, there is just one emotion for every tweet t in P' as in P. Obtaining the similarity of these two sets P and P' means obtaining the accuracy of the DuFER. Eq. (8) is used to calculate the accuracy of the similarity of the two sets.

$$Accuracy(P) = sim\ Jaccard\ (P, P') = \frac{q}{q+r+s} \tag{8}$$

Where q is the number of rows in P and P' which the value of the emotion are equal to one, r is the number of rows which emotion value are one in P and zero inP', and s is the number of rows which emotion value is zero in P and one inP'.

To compare the provided method with other tasks on this dataset, it is necessary to have the same measure. Micro averaging (micro F1) pools per-document decisions across classes, and then computes an effectiveness measure on the pooled contingency table (Cambridge, 2009). Therefore, the micro f1 rate was calculated.

Since DuFER is a single label algorithm, it is expected to be highly accurate in recognizing one correct class among several presented classes for each tweet in this dataset. The evaluation results prove it. The accuracies of the best multi-label methods in SemEval2018, and the proposed method are shown in Table 10.

4.4. Evaluation results of DuFER on the emotion intensity dataset (EID)

All research on EID dataset has been found the intensity of the tweets. In this experiment, the goal is to find the emotional label of each tweet. Thus, there is no need to use emotion intensity values. Only tweets and their relevant emotional labels are used in DuFER.

The motivation of using EID in testing DuFER, is that each tweet contains several words with similar emotion and these words have different degrees of emotional intensity. Some words that explicitly express an emotion, have a greater degree of emotional intensity, and words that implicitly involve an emotion are lower. With the discovery of explicit and implicit emotions in the tweets, DuFER tries to categorize them correctly. Due to the repetition of different emotional words in a tweet and the presence of more emotional words in the dataset, the results obtained by DuFER from this dataset are better than the results of EmoTex- dataset and IEST datasets.

As a result, given the differences in approaches and the use of the

Table 10
Results of SemEval2018, Task5 (E-c): NTUA-SLP,TCS,PlusEmo2vec and DuFER.

Team Name	Accuracy (Jaccard index)	Accuracy (Jaccard Sim)	Micro F1	Rank
NTUA-SLP	58.8	–	70.1	1
TCS Research	58.2	–	69.3	2
PlusEmo2Vec	57.6	–	69.2	3
DuFER	–	***94.81***	***96.23***	–

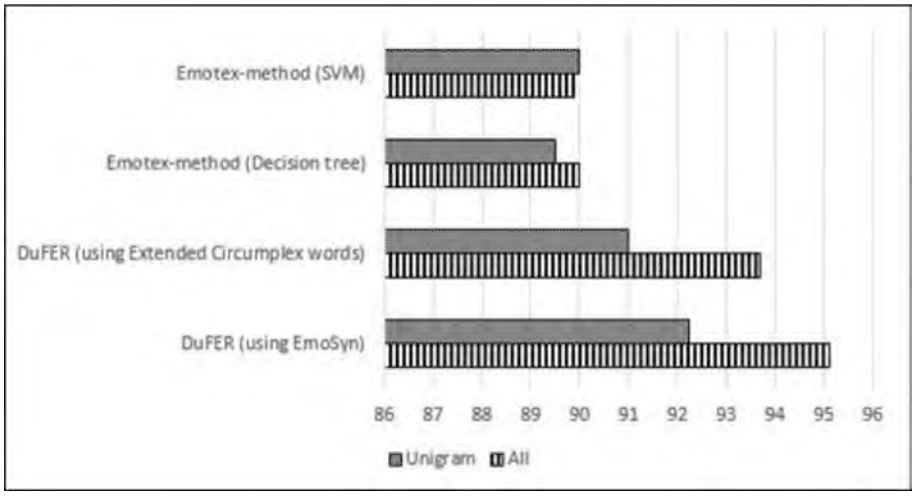

Fig. 2. Comparison of Emotex-method and DuFER accuracy using different features and dictionaries.

dataset, there is no related work to compare the results of DuFER with it. The results of DuFER on this dataset are presented in Table 11.

SVM 10-Fold classifier made better recognition. Within the defined framework due to the use of emotional words and dictionary, repetition of explicit emotional words in each tweet had positive effect on improvement of results.

In (Polignano, Basile, de Gemmis, & Semeraro, 2019) a classification approach is proposed based on deep neural networks. Moreover, 3 pre-trained word-embeddings are compared. The best result belongs to FastText which is shown in Table 15.

4.5. Evaluation results of DuFER on the WASSA 2018 implicit emotion Shared task (IEST) dataset

The IEST (Klinger et al, 2018) dataset is designed for implicit emotions and contains two columns. The first column contains the emotional class of the word that is omitted from the tweet in the second column. In IEST, emotional words are deleted and the position of the deleted word is indicated by [# TARGETWORD #]. The proposed method is trained and tested using 28,757 tweets. 20% of these tweets are intended for testing DuFER. Table 12 shows the confusion matrix and the accuracy of DuFER. As we can see that besides having an optimum performance in recognizing explicit emotions, DuFER has a relatively acceptable performance in recognizing implicit emotions, too.

Thirty participants took part in IEST competition. Results show that DuFER offers an acceptable recognition of implicit emotions compared to other participated methods. Three higher-ranked models are briefly described and compared with DuFER.

Amobee (Rozental et al., 2018), which is reached 1st place with macro F1 of 71.45, consists of a multi-level ensemble. It is combining a novel use of language models and two external embeddings. The language model is specifically trained to maximize the likelihood of matching a word to a given sentence. DuFER also used maximum likelihood to weigh two of three language models. External emotional lexicons and ensemble classifiers are used in DuFER. Amobee used deep learning approach and DuFER used machine learning.

IIIDYT, with macro F1 of 71.05, described the model that got second place in the WASSA 2018 Implicit Emotion Shared Task. Despite its low number of dependencies on libraries and external features, it performed almost as well as the system that obtained the first place (Balazs et al., 2018).

NTUA-SLP is a deep-learning method, whom achieved competitive results in the IEST competition with macro F1 of 70.29, ranking 3rd/30 teams. The proposed approach is based on an ensemble of transfer learning techniques. It demonstrates that the use of refined, high-level features of text, as the ones encoded in language models, yields a higher performance (Chronopoulou et al., 2018).

By examining the 30 presented methods which participated in IEST-2018 competition and comparing the results, Duffer can be ranked sixth, which represents the acceptable performance of DuFER in IER field as EER. The accuracy results of top ten methods show that more research studies in implicit emotion recognition are needed. It is interesting to note the task organizers have tested human performance on a subset sample, achieving macro F1 of 0.45, which is much lower than the automated systems (Rozental et al., 2018).

Table 11
Results of experiments performed by DuFER on EID.

Classifier1 on LM1	Classifier2 on LM2	Classifier3 on LM3	K-Fold Stacking Classifier	F-Measure
Decision tree	Decision tree	Decision tree	SVM 5-Fold	89.84
Decision tree	Decision tree	Decision tree	SVM 10-Fold	91.45
Decision tree	Decision tree	Decision tree	SVM 20-Fold	91.26
SVM	SVM	SVM	SVM 5-Fold	90.3
SVM	SVM	SVM	SVM 10-Fold	94.12
SVM	SVM	SVM	SVM 20-Fold	92.73
Decision tree	SVM	SVM	SVM 5-Fold	92.42
Decision tree	SVM	SVM	SVM 10-Fold	**95.27**
Decision tree	SVM	SVM	SVM 20-Fold	93.51

5. Discussion

The statistical test significance of DuFER is presented in section 5.1. Comparing DuFER with other emotional recognition frameworks is shown in section 5.2.

5.1. Statistical test significance of DuFER

The *t*-test is an inferential statistics test that is used to examine the meaningfulness of the difference between two models. If the difference between the two models is statistically significant, the performance of model with the lower error rate is better. The performance of DuFER is evaluated by this test in two cases.

- **DuFER test on EmoTex-dataset and E-c datasets**

The mean results of two independent sample *t*-test analyzing are obtained from two different datasets by a method. DuFER is applied on EmoTex-dataset and E-c datasets, in Section 4.2 and Section 4.3 respectively. Both datasets contain data with explicit emotions. The value of **t** is calculated by Eq. (9) Han, Pei and Kamber, (2011) because two different datasets are used for this test.

$$t = \frac{err(M1) - err(M2)}{\sqrt{\frac{var(M1-M2)}{k}}} \quad and \quad var(M1 - M2) = \sqrt{\frac{var(M1)}{k1} + \frac{var(M2)}{k2}} \tag{9}$$

Where M1, M2 are two classification models (the DuFER's performance) on two datasets. k1 and k2 are the number of cross-validation samples (in our case, 10-fold cross validation rounds) used for M1 and M2, respectively. The error rates for M1 are averaged to obtain a mean error rate for M1, denoted err(M1). Similarly, we can obtain err(M2).

The test was performed using the Excel software and the 'two-sample unequal variance' was considered as the type of *t*-test to show the differences in the number of samples and variance in the two datasets. In this experiment, the null hypothesis was that the M1 and M2 are the same.

To determine whether M1 and M2 are significantly different, we compute t and select a = 0.01. T-value is obtained using Eq. (9) and compared to suitable amount in the t-distribution table (T-Distribution, 2019) which are shown in Table 13. Results show that there was no significant difference between the mean of the first group of data in EmoTex-dataset (M1) and that of the second one in E-c dataset (M2). Therefore, the results of DuFER in recognizing explicit emotions are non-random and the null hypothesis is proved.

- **DuFER vs Top ten methods in IEST-2018**

This type of test is performed on one dataset while the researcher wants to know if the mean exceeds a certain limit. This test was performed on the Implicit Emotion dataset, which is introduced in Section 4.5, using Eq. (10) (Null Hypothesis Significance Testing III, 2019) Testing (2019). The null hypothesis is that the average accuracy of DuFER is not significantly different from the average performances of the top ten participated methods in IEST-2018.

$$T = \frac{x - \mu_0}{\frac{s}{\sqrt{n}}} \tag{10}$$

x represents the mean of the 10-fold results of DuFER, $\boldsymbol{\mu_0}$ is the average of the results of the 10-top participated methods, and **n** represents the number of samples. The value of **p** is obtained from the distribution of the t-table with the (k − 1) freedom degree ($\mathbf{k = 10}$ and

Table 12
Confusion Matrix on Test Data of IEST by DuFER.

		Anger	Disgust	Fear	Joy	Sadness	Surprise	**F-Measure**
Predicted Labels	Anger	611	81	78	59	86	99	**68.93**
	Disgust	70	645	34	30	61	82	
	Fear	81	17	691	27	25	72	
	Joy	69	14	40	806	58	59	
	Sadness	78	71	25	52	599	50	
	Surprise	95	89	82	59	47	609	

Table 13
Confidence limit results for two T-test types.

Type of T-test	Dataset	Obtained T-value	Related p in t-distribution table
None paired (Two-sample unequal variance)	1-EmoTex-Dataset 2-SemEval-2018, Task: E-c Dataset	2.050355	2.821
One sample	IEST Dataset	1.384993	3.250

a = 0.005). According to the distribution table, **p = 3.250**. After calculating Eq. (10) by Excel, it can be seen that **T < p** as shown in Table 13. This shows that the null hypothesis is verified and the estimation results of DuFER on the Implicit Emotion dataset are acceptable.

The results of these t-tests on the proposed method using explicit and implicit emotion datasets show that DuFER is reasonably capable of recognizing explicit and implicit emotions from the text and these estimates are reliable.

5.2. Comparing DuFER with other emotion recognition frameworks

DuFER can perform both IER and EER. It could apply on any dataset with explicit and/or implicit emotions. This feature is differentiating it from other frameworks which are designed for just one of these two kinds of emotions and DuFER is actually the first successful framework in this field. On the other hand, as stated in Section 4.2, DuFER is capable of adapting to several emotional models. Since this framework does not use unique features of a particular language, it can also be applied to other languages by having an emotional dictionary, which is expected to be accurate in recognizing emotions.

The comparison results of DuFER with a number of works are shown in Table 14 and Table 15. The important point is that in all the following cases, they have only recognized one type of emotions. While DuFER has the power to recognize both explicit and implicit emotions. Also, the results obtained from DuFER were acceptable and sometimes better.

IIIDYT system (Balazs et al., 2018) is designed for IEST, which obtained 2nd place out of 30 teams. However, the ablation study revealed that increased performance can be obtained by incorporating linguistic information as additional inputs such as POS embeddings. This shows the importance of using linguistic information. In (Chronopoulou et al., 2018), the submitted model consists of an ensemble of the aforementioned transfer learning models and ranked 3rd/30. This work demonstrated that the use of features which encoded in language models, yields a higher performance. The NLP method achieves 7th position out of 30 teams and outperforms the baseline method by 12.5% in terms of macro F1 (Zhou, Q & Wu, 2018). Authors believe that using more textual features, as well as the use of different weights and trying different ensemble methods like hard voting and stacking, to gain better performance.

Table 14
Compare DuFER & Other implicit Methods.

Proposed Work	Dataset	Contribution	Best prediction
IIIDYT (Balazs, Marrese-Taylor, & Matsuo, 2018)	WASSA 2018 IEST	Pre-trained ELMo + BiLSTM	71.05
NTUA-SLP (Chronopoulou et al., 2018**)**		Ensemble of transfer learning techniques + LSTM	70.9
DuFER		2 kinds of feature sets, lexicon-based, 3language models weighting + SVM, Decision tree	68.93
HUMIR (Naderalvojoud, Ucan, & Sezer, 2018**)**		3 kinds of feature sets, lexicon-based, emotion-weight and context-sensitive BiLSTM + MLP	68.8
NLP (Zhou & Wu, 2018)		BiLSTM-Attention + LSTM-Attention	68.48

Table 15
Compare DuFER & Other explicit Methods.

Proposed Work	Dataset	Contribution	Best prediction
Emotex-method (Hasan, Rundensteiner, & Agu, 2019)	EmoTex-dataset	Unigram, emoticons, punctuation, and negation as features + Naïve Bayes, SVM	90
DuFER	EmoTex-dataset	2 kinds of feature sets, lexicon-based, 3language models weighting + SVM, Decision tree	95.12
DuFER	EID		95.27
Proposed method in (Polignano, Basile, de Gemmis, & Semeraro, 2019)	EID	Deep neural networks, Bi-LSTM, CNN + FastText word-embedding	94

Polignano, Basile, de Gemmis, & Semeraro (2019), have proposed a text identification model based on the use of deep neural networks LSTM and CNN mediated through the use of a level of attention. The results demonstrated the effectiveness of the approach but based on (Acheampong et al., 2020) proposed model has a high complexity. Hasan, Rundensteiner, and Agu (2019), presented a model using a supervised learning method and emotion dictionaries. Their approach consisted of two methods: an offline and an online classification task. The model yielded an accuracy of 90% but it has loose semantic features (Acheampong et al., 2020).

Proposed points and the results obtained from other articles (which have been worked on other datasets) were reviewed and summarized (Acheampong et al., 2020). We believe that the proposed framework has been successful in extracting features and using linguistic information.

6. Conclusion

In this paper, we proposed a dual framework for recognizing emotion in short texts. We developed and evaluated a supervised machine learning system called DuFER to automatically classify short texts with implicit and/or explicit emotions. Our experiments showed that DuFER correctly classify emotion in more than 95% of text messages with explicit emotions. It achieves 68.93% of accuracy in classification of texts with implicit emotion which is very competitive result with the proposed methods in IEST competition. DuFER is based on an ensemble of language models (word-level, phrase-level) and computational

linguistics. Moreover, we propose a weighing method for word-level language model of text to identify discriminative words in short texts. We demonstrate that the use of weighed, high-level features of text, as the ones encoded in language models, yields a higher performance. A two-stage classification module was designed in DuFER. First three parallel classifiers using three different language models separated tweets with various emotional classes. Then the results were combined to conduct a fine-grained emotion classification on tweets in second stage. Our future plan is to change DuFER framework which we can use deep learning methods and different word embeddings.

Declaration of Competing Interest

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

Appendix A

Algorithm D. uFER (Dual Framework for Emotion Recognition), Classify short texts with implicit and/or explicit emotions**.**

Input:

- D: Dataset of tweets
- LX1: NRC Emotion Lexicon
- LX2: NRC Hashtag Emotion Lexicon
- Lx3: WordNet

Output: H, a collection of one emotion label for each tweet**Method:**//Preprocessing module(1) Preprocess (D, LX1, LX2, LX3, D', EmoSyn)//D, LX1, LX2, and LX3 are inputs.//Outputs are D' and EmoSyn.//Analysis module(2) Analysis (D', EmoSyn, LM1, LM2, LM3)//D', and EmoSyn are inputs.//Outputs are LM1, LM2, and LM3.//Classification module(3) l = Ensemble-Classifier (LM1, LM2, LM3,Y)(4) Return (l)**Procedure Preprocess (D:** dataset of tweets, **LX1:** NRC Emotion Lexicon, **LX2:** WordNet, **LX3:** NRC Hashtag Emotion Lexicon, **D':** dataset of processed tweets, **EmoSyn:** dictionary of emotional words**)**//Text preprocessing2. D' = Text-Preprocess (D)//D and D' are an input and Outputs respectively.// Extending emotional lexicon3. EmoSyn = Extended-Dictionary (LX1, LX2, LX3)4. Return (D', EmoSyn)**Procedure Text-preprocess (D:** dataset of tweets**)**(1) D' = Delete Usernames and every word that doesn't belong to the main part of a tweet in D and Normalize sentences(2) D' =Lemmatize every word in D'(3) Return (D')**Procedure Extended-Dictionary (LX1:** NRC Emotion Lexicon, **LX2:** WordNet, **LX3:** NRC Hashtag Emotion Lexicon**)**(1) M= {Anger, Anticipation, Disgust, Fear, Joy, Sadness, Surprise, Trust, Love, Pessimism, Optimism}//8 emotions of Plutchik emotional model + {Love, Pessimism, Optimism}(2) For each emotion e in M{(3) E = {}(4) For each itemset w ∈ LX1 {(5) Find b as second-level Seynset of w from LX3(6) Find t as third-level Seynset of w in LX3(7) E = E ∪ {w, b, t}(8) }(9) For each V in LX2//V is a member of LX2(10) If (e in V) then E = E ∪ V(11) EmoSyn = EmoSyn ∪ E//E is the set of words with emotion e.(12) }(13) Return (EmoSyn)**Procedure Analysis (D':** dataset of processed tweets, **EmoSyn:** dictionary of emotional words**)**(1) Word-level (D', EmoSyn, LM1, M1)//D' and EmoSyn are inputs.//Outputs are LM1 and M1.(2) Phrase-level (D', M1, LM2, LM3)(3) Return (LM1, LM2, LM3)**Procedure Word-level (D':** dataset of processed tweets, **EmoSyn:** dictionary of emotional words**, LM1:** unigram language model without stop word, **M1:** unigram language model with stop word**)**(1) D'' = Remove stopwords of D'(2) LM1 = Making Unigram Matrix for D'' dataset//Unigram Language Model without stop-words(3) M1 = Making Unigram Matrix for D' dataset//Unigram Language Model with stopwords(4) For each w ∈ D''(5) If w is in EmoSyn then w is replaced by its Emotion-word in D''(6) For each column ti ∈ LM1(7) For each row s in LM1 {(8) If Maxtf^{ti}_{tweet} greater than 1(9) W_s (ti) = ((tf^{ti}_s- Min tf^{ti}_{tweet})/(Max tf^{ti}_{tweet}- Min tf^{ti}_{tweet}))(10) Elseif Maxtf^{ti}_{tweet} = 1 (11) W_s (ti) = $tf^{ti}_s/tf^{ti}_{D''}$}(12) For each emotional class C in D''(13) W_p (ti) = $tf^{ti}_s/tf^{ti}_{D''}$ (14) For each column ti in LM1 {(15) For each row s in LM1(16) W(ti) = W_s (ti) + W_p(ti)(17) tf^{ti}_sW(ti)* tf^{ti}_s//replace weighed tf^{ti}in each tweet instead of its previous amount in LM1(18) }(19) Return (LM1, M1)**Procedure Phrase-level (D':** dataset of processed tweets, **M1:** unigram Language Model with stopwords**, LM2:** bigram Language Model**, LM3:** trigram Language Model**)**

(1). LM2 = Making Bigram Matrix for D' dataset//bigram Language Model
(2). LM3 = Making trigram Matrix for D' dataset//trigram Language Model
(3). For each column ti ∈ LM3
(4). W (ti) = C (W_1, W_2, W3)/C (W_1, W2)//Compute MLE for ti by using M1, LM2 and LM3
(5). LM3 = Replace tf^{ti} each row with **W(ti)** * tf^{ti} in LM3
(6). For each column ti ∈ LM2
(7). W(ti) = C (W_1, W_2)/C(W_1)//Compute MLE for ti by using M1 and LM2
(8). LM2 = Replace tf^{ti} each row with **W(ti)** * tf^{ti} in LM2
(9). Return (LM2, LM3)

Procedure Ensemble-Classifier (LM1: Unigram Language Model without stopwords**, LM2:** Bigram Language Model**, LM3:** Trigram Language Model, **Y:** The matrix of actual emotional label of each tweet, **l:** Ensemble-Classifier**)**//Base classifiers phase(1) For i = 1 to 3//Step1: Train base-level classifiers (h) in parallel(2) Train classifier h_i on LM $_i$//the output of each classifier h_i is a (n*1) matrix//Ensemble phase//constructing new matrix D^h base on three trained classifiers(3) For j = 1 to 3//3 is number of base-classifiers(4) For i = 1 to n//n is the number of tweets(5) $d^h_{i,j} = h_j(x)_i$//x_i is the predicted label for i^{th} tweet by a base-level classifier (h)(6) For i = 1 to n(7) $d_{i,4} = y_i$//Y is a (n*1) matrix which contains the emotional label of each tweet//Matrix (n*4) D^h is constructed//Train Ensemble-Classifier to achieve better predictions(8) Train l as Ensemble-Classifier on D^h(9) Return (l)

References

Acheampong, F. A., Wenyu, C., & Nunoo-Mensah, H. (2020). Text-based emotion detection: Advances, challenges, and opportunities. *Engineering Reports, 2*(7), Article e12189.

Adarsh, S. (2019). Enhancement of text-based emotion recognition performances using word clusters. *International Journal of Research-GRANTHAALAYAH, 7*(1), 238–250.

Agrawal, A., & An, A. (2012). Unsupervised emotion detection from text using semantic and syntactic relations. In *In 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology* (vol. 1, pp. 346–353). IEEE.

Agrawal, P., & Suri, A. (2019). NELEC at SemEval-2019 task 3: think twice before going deep. arXiv preprint arXiv:1904.03223.

Alswaidan, N., & Menai, M. E. B. (2020). A survey of state-of-the-art approaches for emotion recognition in text. *Knowledge & Information Systems, 62*(8).

Balahur, A., Hermida, J. M., & Montoyo, A. (2012). Detecting implicit expressions of emotion in text: A comparative analysis. *Decision Support Systems, 53*(4), 742–753.
Balahur, A., Klinger, R., De Clercq, O., & Mohammad, S. M. (2018). Iest: Wassa-2018 implicit emotions shared task, Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis. arXiv preprint arXiv:1809.01083, 31-42.
Balazs, J. A., Marrese-Taylor, E., & Matsuo, Y. (2018). Iiidyt at iest 2018: Implicit emotion classification with deep contextualized word representations. arXiv preprint arXiv:1808.08672.
Cambridge, U. (2009). *Online edition (c) 2009 Cambridge UP An Introduction to Information Retrieval Christopher D*. In): Manning Prabhakar Raghavan Hinrich Schütze Cambridge University Press.
Chriqui, A., & Yahav, I. (2021). Hebert & hebemo: a hebrew bert model and a tool for polarity analysis and emotion recognition. arXiv preprint arXiv:2102.01909.
Chronopoulou, A., Margatina, A., Baziotis, C., & Potamianos, A. (2018). NTUA-SLP at IEST 2018: Ensemble of neural transfer methods for implicit emotion classification. arXiv preprint arXiv:1809.00717.
Feng, X., Wei, Y., Pan, X., Qiu, L., & Ma, Y. (2020). Academic emotion classification and recognition method for large-scale online learning environment—Based on A-CNN and LSTM-ATT deep learning pipeline method. *International journal of environmental research and public health, 17*(6), 1941.
Gaind, B., Syal, V., & Padgalwar, S. (2019). Emotion detection and analysis on social media. arXiv preprint arXiv:1901.08458.
Global WordNet Association (2020). Retrieved from http://globalwordnet.org/resources/wordnets-in-the-world/Accessed January 19, 2020.
Gratian, V., & Haid, M. (2018). Braint at iest 2018: Fine-tuning multiclass perceptron for implicit emotion classification. In *In Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis* (pp. 243–247).
Guia, M., Silva, R. R., & Bernardino, J. (2019). Comparison of Naïve Bayes, Support Vector Machine, Decision Trees and Random Forest on Sentiment Analysis. In KDIR, (pp. 525-531).
Gunes, H., Schuller, B., Pantic, M., & Cowie, R. (2011). In *Emotion representation, analysis and synthesis in continuous space: A survey* (pp. 827–834). IEEE.
Gutierrez-Osuna, R. (2005). Introduction to pattern analysis. Lecture Notes, Texas A&M University.
Gyurak, A., Gross, J. J., & Etkin, A. (2011). Explicit and implicit emotion regulation: A dual-process framework. *Cognition and emotion, 25*(3), 400–412.
Han, J., Pei, J., & Kamber, M. (2011). *Data mining: Concepts and techniques*. Elsevier.
Hasan, M., Rundensteiner, E., & Agu, E. (2019). Automatic emotion detection in text streams by analyzing Twitter data. *International Journal of Data Science and Analytics, 7*(1), 35–51.
Jaccard, P. (1912). The distribution of the flora in the alpine zone. 1. *New phytologist, 11*(2), 37–50.
Jurafsky, D., & Martin, J. H. (2019). Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition (third ed. draft).
Kassinove, H., & Sukhodolsky, D. G. (1995). Anger disorders: Basic science and practice issues. *Issues in comprehensive pediatric nursing, 18*(3), 173–205.
Kim, E., Kim, W., & Lee, Y. (2003). Combination of multiple classifiers for the customer's purchase behavior prediction. *Decision Support Systems, 34*(2), 167–175.
Kim, E., & Klinger, R. (2018). A survey on sentiment and emotion analysis for computational literary studies. arXiv preprint arXiv:1808.03137.
Kiritchenko, S., Zhu, X., & Mohammad, S. M. (2014). Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research, 50*, 723–762.
R. Klinger O. Clercq S.M. Mohammad A. Balahur The data consists of the emotion class of the word which has been removed in the text, IEST: WASSA-2018 Implicit Emotions Shared Task 2018 .
Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In Ijcai, vol. 14 (pp. 1137-1145): Montreal, Canada.
Koole, S. L., & Rothermund, K. (2011). *The psychology of implicit emotion regulation*. Psychology Press.
[dataset] Mohammad, S., Bravo-Marquez, F., Salameh, M., & Kiritchenko, S. (2018)., English tweets for the eight basic emotions as per Plutchik , as well as a few other emotions that are common in tweets (love, optimism, and pessimism), SemEval-2018 Task1: Affect in tweets. https://competitions.codalab.org/competitions/17751#learn_the_details-datasets.
[dataset] Mohammad, S. M., & Bravo-Marquez, F. (2017). Emotion Intensities in Tweets for four emotions: joy, sadness, fear, and anger, In Proceedings of the 6th joint conference on lexical and computational semantics(*Sem). arXiv preprint arXiv:1708.03696.
Mohammad, S. M., Bravo-Marquez, F., Salameh, M., & Kiritchenko, S. (2018). Semeval-2018 task 1: Affect in tweets. In *In Proceedings of the 12th international workshop on semantic evaluation* (pp. 1–17).
Naderalvojoud, B., Ucan, A., & Sezer, E. A. (2018). HUMIR at IEST-2018: Lexicon-Sensitive and Left-Right Context-Sensitive BiLSTM for Implicit Emotion Recognition. In *In Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis* (pp. 182–188).
Null Hypothesis Significance Testing III. (2019). Retrieved from https://ocw.mit.edu/courses/mathematics/18-05-introduction-to-probability-and-statistics-spring-2014/readings/MIT18_05S14_Reading19.pdf. Accessed August 21, 2019.
Park, S.-H., Bae, B.-C., & Cheong, Y.-G. (2020). In *Emotion recognition from text stories using an emotion embedding model* (pp. 579–583). IEEE.
Perlich, C., & Świrszcz, G. (2011). On cross-validation and stacking: Building seemingly predictive models on random data. *ACM SIGKDD Explorations Newsletter, 12*(2), 11–15.
Polignano, M., Basile, P., de Gemmis, M., & Semeraro, G. (2019). In *A comparison of word-embeddings in emotion detection from text using bilstm, cnn and self-attention* (pp. 63–68). Adaptation and Personalization.
Proisl, T., Heinrich, P., Kabashi, B., & Evert, S. (2018). EmotiKLUE at IEST 2018: Topic-Informed Classification of Implicit Emotions. In *In Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis* (pp. 235–242).
Ren, F., & Sohrab, M. G. (2013). Class-indexing-based term weighting for automatic text classification. *Information Sciences, 236*, 109–125.
Ren, H., Ren, Y., Li, X., Feng, W., & Liu, M. (2017). In *Natural logic inference for emotion detection* (pp. 424–436). Springer.
Rozental, A., Fleischer, D., & Kelrich, Z. (2018). Amobee at IEST 2018: Transfer learning from language models. arXiv preprint arXiv:1808.08782.
Seyeditabari, A., Levens, S., Maestas, C. D., Shaikh, S., Walsh, J. I., Zadrozny, W., Danis, C., & Thompson, O. P. (2017). Cross corpus emotion classification using survey data. This paper was presented at AISB.
Seyeditabari, A., Tabari, N., & Zadrozny, W. (2018). Emotion detection in text: a review. arXiv preprint arXiv:1806.00674.
Siciliano, L., & Clausi, S. (2020). Implicit vs. explicit emotion processing in autism spectrum disorders: An opinion on the role of the cerebellum. *Frontiers in psychology, 11*, 96.
Strapparava, C., & Mihalcea, R. (2008). Learning to identify emotions in text. In *In Proceedings of the 2008 ACM symposium on Applied computing* (pp. 1556–1560).
T-Distribution. (2019). Table, Retrieved from http://math.mit.edu/~vebrunel/Additional/table.pdf. Accessed August 21, 2019.
Williams, L., Arribas-Ayllon, M., Artemiou, A., & Spasić, I. (2019). Comparing the utility of different classification schemes for emotive language analysis. *Journal of Classification, 36*(3), 619–648.
Zhong, P., & Miao, C. (2019). ntuer at semeval-2019 task 3: Emotion classification with word and sentence representations in rcnn. arXiv preprint arXiv:1902.07867.
Zhou, Q., & Wu, H. (2018). NLP at IEST 2018: BiLSTM-attention and LSTM-attention via soft voting in emotion classification. In *In Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis* (pp. 189–194).