Information Processing and Management 59 (2022) 103011

Contents lists available at ScienceDirect

Information Processing and Management

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

A text classification approach to detect psychological stress combining a lexicon-based feature framework with distributional representations

Sergio Muñoz *, Carlos A. Iglesias

Intelligent Systems Group, ETSI Telecomunicación, Universidad Politécnica de Madrid, Avenida Complutense 30, Madrid, 28040, Spain

ARTICLE INFO

Keywords:
Stress detection
Stress framework
Distributional representations
Text classification
Affective computing

ABSTRACT

Nowadays, stress has become a growing problem for society due to its high impact on individuals but also on health care systems and companies. In order to overcome this problem, early detection of stress is a key factor. Previous studies have shown the effectiveness of text analysis in the detection of sentiment, emotion, and mental illness. However, existing solutions for stress detection from text are focused on a specific corpus. There is still a lack of well-validated methods that provide good results in different datasets. We aim to advance state of the art by proposing a method to detect stress in textual data and evaluating it using multiple public English datasets. The proposed approach combines lexicon-based features with distributional representations to enhance classification performance. To help organize features for stress detection in text, we propose a lexicon-based feature framework that exploits affective, syntactic, social, and topic-related features. Also, three different word embedding techniques are studied for exploiting distributional representation. Our approach has been implemented with three machine learning models that have been evaluated in terms of performance through several experiments. This evaluation has been conducted using three public English datasets and provides a baseline for other researchers. The obtained results identify the combination of FastText embeddings with a selection of lexicon-based features as the best-performing model, achieving F-scores above 80%.

1. Introduction

Mental illness – and more specifically, stress – is a growing problem for modern society (Can, Chalabianloo, Ekiz and Ersoy, 2019). Stress has become part of our daily lives, impacting affected individuals and their families, health care systems, private and social insurers, employers, work colleagues, and the society at large (Kassymova et al., 2019).

The economic and health costs that stress entails worldwide are much more significant than expected. In Europe, a public survey identified stress as the second most prevalent work-related health problem (for Safety & at Work, 2013). According to this survey, the stress in the workplace is a frequent issue for 51% of European employees, and 40% of them feel that stress is not handled effectively (Parent-Thirion et al., 2012). In terms of economic impact, the annual cost to the European companies of work-related stress is estimated at 25 billion euros (Hassard et al., 2014). The problem is not particular to Europe since existing reports from countries worldwide show similar results. For example, in the United States, 40% of workers consider their job notably or vastly stressful, whereas 29% of them feel slightly or highly stressed at the workplace (Saute et al., 1999). Also, it is estimated that the

[*] Corresponding author.
E-mail address: sergio.munoz@upm.es (S. Muñoz).

https://doi.org/10.1016/j.ipm.2022.103011
Received 29 March 2022; Received in revised form 22 June 2022; Accepted 26 June 2022
Available online 1 July 2022

United States spends 300 billion USD per year on stress-related diseases (Can, Arnrich and Ersoy, 2019). Available reports from other countries such as China (Xiong, Skitmore, & Xia, 2015) or Australia (Society, 2015) show the global importance of the stress problem.

In the light of its high impact, preventing and regulating stress has become a critical health issue for populations (Greene, Thapliyal, & Caban-Holt, 2016). Early detection and monitoring of stress problems can significantly improve the efficiency of interventions, decreasing their costs and preventing stress from being chronic (Can, Chalabianloo et al., 2019). In this context, methods are needed to detect stress in time. Stress detection has been traditionally assessed using self-reports in response to standardized questionnaires (Andreou et al., 2011), such as Perceived Stress Scale (Chan & La Greca, 2020) or Depression Anxiety and Stress Scale (Osman et al., 2012). Although the validity of these methods has been proved, they suffer from two major drawbacks: their frequent delay in diagnosis, which makes them unsuitable for early detection, and their subjectivity and dependence on subjects' recall and situation awareness (Alberdi, Aztiria, Basarab, & Cook, 2018).

For this reason, there is still a need for objective measures to detect stress based on physical and physiological information (Greene et al., 2016). The great advances in affective computing open a range of possibilities for addressing these issues. In recent years, a great deal of effort has been devoted to research into methods and systems that use smart devices and affective computing algorithms for automatic stress detection. Some of the most popular approaches consist of the analysis of physiological signals (de Santos Sierra, Ávila, Casanova, & del Pozo, 2011), facial expression (Giannakakis et al., 2017), speech (Hansen & Patil, 2007), phone usage (Ferdous, Osmani, & Mayora, 2015), or keystroke-dynamics (Vizer, Zhou, & Sears, 2009). These methods avoid the subjective response bias from the individual introduced in self-report questionnaires, and their reliability has been proved. However, their intrusiveness and high implementation costs are often a major constraint (Novais & Carneiro, 2016).

This challenge has boosted the research on more economical and unobtrusive methods, such as the use of social media data (Lin, Jia, Nie, Shen, & Chua, 2016). Social media services where users communicate and share their thoughts or experiences have gained popularity and provide a vast amount of information related to people's emotions, daily moods, and worries (Pang et al., 2019). Textual data coming from these platforms have been successfully exploited by the research community in a wide variety of text classification applications such as detection of sentiment (Yue, Chen, Li, Zuo, & Yin, 2019), radicalization (Araque & Iglesias, 2020) and also mental illness (Banerjee & Shaikh, 2021; Chancellor & De Choudhury, 2020). The most popular approach in the literature concerning text classification is the use of machine learning techniques that represent texts as vectors in a feature space and classify them into categories (Bandhakavi, Wiratunga, Padmanabhan, & Massie, 2017).

Some studies have exploited these techniques for the task of stress detection from text (Cao et al., 2021; Lin et al., 2016; Winata, Kampman, & Fung, 2018). Existing works achieve great results, but they focus on specific data sources. Thus, there is still a lack of well-validated methods that show good results in different datasets. We aim to contribute to state of the art in stress classification from text by proposing a machine learning method to detect stress in corpora from different sources: personal interviews, Reddit social network, and Twitter social network. Also, our work aims at identifying which features and techniques perform better. Therefore, we focus on the following research questions (RQs) related to the task of stress classification from text:

- **RQ1**: Which kinds of lexicon-based features are more relevant and yield better results?
- **RQ2**: How do different machine learning models compare in terms of performance?
- **RQ3**: Can a machine learning approach achieve good results when evaluated on several corpora?

Motivated by these RQs, this paper proposes three different machine learning models for detecting stress in texts: the first consists of a lexicon-based feature extraction method, the second uses word embedding techniques for exploiting distributional representations, and the third combines distributional representations with lexicon-based features. For this purpose, our work proposes a lexicon-based feature framework that exploits affective, syntactic, social, and topic-related features. This framework aims at helping in the organization and characterization of features for stress detection in text. Besides, three different word embedding techniques are considered to analyze their suitability for the stress detection task. To evaluate the effectiveness of the proposed models, several experiments have been conducted using public English datasets from the three different sources mentioned before. In this way, a baseline for other researchers is provided. Furthermore, the relevance of the experiments is confirmed with a statistical study that allows us to further analyze the proposed models' performance. Finally, our best-performing model is compared with state-of-the-art stress detection methods in text. The obtained results identify the combination of FastText embeddings with a selection of lexicon-based features as the best-performing method. This method outperforms the existing works for every dataset, achieving F-scores that surpass 80%.

The rest of the paper is organized as follows. An overview of stress theory and text classification methods is given in Section 2. Following, Section 3 presents the used datasets along with a preliminary analysis performed on them. The feature framework for stress detection from text is presented in Section 4. In Section 5, the proposed stress detection models are described. Later, in Section 6, the experimental setup aimed at evaluating the proposed models is presented, along with the obtained results. A discussion regarding the main findings of the article is given in Section 7. Finally, the paper concludes in Section 8, where conclusions drawn from the work are depicted, and an outline of possible lines of future work is presented.

2. Related work

This section presents the background and related work of the concepts and technologies involved in the paper. First, Section 2.1 gives an overview of stress theory and detection methods. Then, Section 2.2 introduces popular approaches for text classification.

2.1. Psychological stress

Due to the subjectivity of stress and the different contexts where this concept is used, a universally recognized definition for stress is still lacking (The American Institute of Stress, 2013). One of the earlier and more generic stress definitions was proposed by Hans Selye, who defined stress as the non-specific response of the body to any demand (Selye, 1956). Over the last years, a vast number of extended and more specific stress definitions have been proposed (Burman & Goswami, 2018). For example, Kim and Diamond (Kim & Diamond, 2002) propose a stress definition that considers that stress requires three main components: heightened excitability or arousal, an experience perceived as aversive, and lack of control. Another perspective is proposed by Cox and Griffiths (1995a), Cox and Griffiths (1995b) who state that the definition of stress can be addressed from three different approaches: psychological, engineering, and physiological. From a psychological point of view, stress can be seen as a dynamic process that stems from the interaction between an individual and the environment. The engineering approach defines stress as a stimulus of the environment in the form of a demand level. Finally, regarding the physiological one, stress can be defined as the changes that occur in a human under pressure. According to this, stress can be non-formally defined as the reaction of the human body to any challenging or hazardous situation (Can, Arnrich et al., 2019).

The stress subjectivity and complexity have also resulted in numerous stress theories that form the basis for understanding stress (Dewe, O'Driscoll, & Cooper, 2012). One of the most relevant is the Person–Environment (P-E) fit theory (French, Caplan, & Van Harrison, 1982), founded on the studies by Lewin (1936) and Murray (1938). This theory has been the source for other approaches to stress and well-being and argues that stress arises from the lack of fit or congruence between the person and the environment. According to this theory, stress can be seen as a lack of match between a person's abilities and demands. In 1982, the Transactional Model of Stress (Holroyd & Lazarus, 1982) was proposed by Lazarus and Holroyd. This theory considers stress as a relationship between the person and the environment. According to the authors, the person appraises the environment as taxing, hence threatening well-being (Glanz, Rimer, & Viswanath, 2008).

These models have been the basis for understanding stress and have helped the development of prevention, detection, and regulation methods. Research in the field has shown excellent results in detecting stress using smart sensors and devices (Can, Arnrich et al., 2019). These devices measure physiological (e.g., brain or heart activity, skin response, and breath response) and physical features (e.g., facial expression, eye tracking, behavior, and gesturing). Greene et al. (2016) and Panicker and Gayathri (2019) performed exhaustive reviews about the usage of these techniques to detect stress. The most successful results have been yielded using Electro-Dermal Activity (EDA) (Affanni, Bernardini, Piras, Rinaldo, & Zontone, 2018), Electroencephalogram (EEG) (Vanitha & Krishnan, 2017), Blood Pressure (BP), Respiration, Blood Volume Pulse (BVP) (Widanti, Sumanto, Rosa, & Miftahudin, 2015), facial expressions (Gao, Yüce, & Thiran, 2014), speech (Tomba, Dumoulin, Mugellini, Abou Khaled, & Hawila, 2018), or mobile phone usage (Maxhuni et al., 2021). The accuracy obtained with these methods ranges between 70% and 90%, proving the reliability of these systems for detecting stress. However, they often have a major drawback when implementing these solutions in real scenarios: their intrusiveness and high costs (Novais & Carneiro, 2016). Besides, errors from incorrect placement, movements, or detached equipment are very common in daily life and lead to corrupted data (Can, Arnrich et al., 2019).

This fact, along with the significant advances in Natural Language Processing techniques, has given momentum to the approach of stress detection from text. Moreover, the vast amount of textual data contained nowadays in social networks (Statista, 2018) makes these techniques a promising approach.

2.2. Text classification methods applied to stress detection

Stress detection from text can be considered a text classification problem that aims to distinguish texts depending on whether they express stress. This can be shaped by exploiting certain syntactic and linguistic features using machine learning techniques or through lexicons. The lexicon-based approach compares words in the text with a dictionary to calculate the presence and frequency of specific bearer terms. Lexicons can provide an overall indication of specific features (e.g., sentiment, emotion, cognition, or topic) depending on the nature of the words they contain (Khoo & Johnkhan, 2018). This approach tends to be computationally fast and has yielded good performance on a wide variety of text classification applications such as detection of mental illness (Giuntini et al., 2020) or extreme opinions (Almatarneh & Gamallo, 2018) from social media. For example, Mike Thelwall used lexicons to develop TensiStrength (Thelwall, 2017), a system able to detect stress and relaxation in tweets. TensiStrength detects the expressions of stress and relaxation through a list of stress-related terms and a set of rules. This method yields reasonable accuracy levels, but the performance is lower than that obtained using machine learning methods.

In order to enhance the performance, a popular approach is to include lexicon-based features in machine learning methods. Lin et al. (2014) exploited this approach by proposing a deep neural model for detecting stress from Chinese micro-blogging posts like Sina Weibo or Tencent Weibo. Intending to overcome the problem of scarce stress-annotated data existing, they collected posts from these platforms. They used sentence patterns such as "The day was stressful" to obtain the ground truth labeled data. The social data extracted from the social network was combined with textual and image data obtained from each post and then used to train a deep neural network model.

Lexicon-based features can help to provide a general indication of a text in terms of sentiment, cognition, or topic. However, they fail to apprehend more refined attributes and contextual cues intrinsic to the human language (Giatsoglou et al., 2017). Word embedding-based approaches address this challenge, enabling the encoding of semantic and syntactic features present in words and their representation in a vector space as relation offsets. These vectors are named pre-trained word vectors and can be used for textual representation in text categorization tasks (Wang, Zhou and Jiang, 2020). Popular word embedding techniques are

Word2Vec (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013), Global Vectors for Word Representation (GloVe) (Pennington, Socher, & Manning, 2014) and FastText (Joulin et al., 2016). Winata et al. (2018) exploited Word2Vec word embeddings for detecting stress in interview questions. They used data from social networks (Twitter) to extend the Natural Stress Emotion corpus (Zuo, Lin, & Fung, 2012). This corpus contains transcriptions of 12 interview questions designed to be progressively stress-provoking answered by 25 students. Then, they used these data for feeding a neural model based on bidirectional Long Short-Term Memory (Bi-LSTM) model with an attention mechanism, obtaining great results for stress classification.

Although word embedding-based approaches effectively capture syntactic and semantic features in texts, they do not exploit the individual affective or social value of the words. Previous studies have demonstrated that combining word embeddings with lexicon-based features can enhance the quality of text representations, yielding higher prediction performance (Fu, Yang, Li, Fang, & Wang, 2018). E. Turcan and K. McKeown exploited this combination to detect stress from social media data (Turcan & McKeown, 2019). They proposed a new text corpus of long social media data coming from Reddit to identify more implicit indicators than those available in micro-blogging posts. The typically longer length of Reddit posts compared to those from micro-blogging platforms allows researchers to deeply analyze the causes and indicators of stress in text. They evaluated different supervised learning methods for stress detection in the corpus, achieving the best results by combining lexicon-based features and Word2Vec word embeddings. This method establishes the current benchmark on the problem of binary text classification of stress at 79.80% F-score.

All these works have shown excellent results for classifying stress. However, they are evaluated only on data from a specific dataset (coming from micro-blogging platforms, interviews, or Reddit). We aim to provide a solution that is validated across multiple public English datasets. In this way, we propose a cross-dataset model for detecting psychological stress from text. Our approach combines word embeddings with different kinds of lexical-based features: affective, topic, social, and syntactic related. Different lexicons are evaluated, and a feature framework is proposed to identify which kinds of features drive better performance. Moreover, several word embedding techniques are considered to analyze their impact on classification performance. To the best of our knowledge, this is the first paper on stress detection from text that proposes several machine learning models combining surface and deep features and evaluates them on different stress-based text corpora (Skaik & Inkpen, 2020; Su, Xu, Pathak, & Wang, 2020; Thieme, Belgrave, & Doherty, 2020). The following sections will deeply describe our approach and its evaluation through a set of experiments. The performance reached by the proposed method is comparable to the performance of stress detection methods that use physiological or physical data (Panicker & Gayathri, 2019), but the entailed costs and complexity are considerably lower. The applications of stress textual classifiers are manifold, from detecting stress at work to business applications such as customer management or marketing.

3. Materials

To perform the evaluation, we have used three English language datasets from three different sources: Dreaddit (Reddit) (Turcan & McKeown, 2019), Natural Stress Emotion (personal interviews) (Zuo et al., 2012), and TensiStrength (Twitter) (Thelwall, 2017). These datasets and some statistics drawn from them are described in the following lines. It should be noted that even if the datasets have a different nature, for commodity, we will refer to the instances of each dataset as “posts” in the rest of the article.

- The *Dreaddit* dataset was collected by E. Turcan and K. McKeown. It contains 3549 Reddit posts annotated using Amazon Mechanical Turk, resulting in 47.75% of non-stress and 52.75% of stress posts. The average post length in the dataset is 88 words. Besides, they proposed a subset of this dataset containing only those posts that obtained confidence greater than 0.8 in the annotation (80% of agreement between annotators). The authors demonstrated that this high-agreement subset led to higher reliability in stress detection, so we have used this subset in our study.
- The *Natural Stress Emotion (NSE)* dataset, collected by Zuo et al. (2012) and extended by Winata et al. (2018), consists of a set of 38 interviews where students answered questions designed to be progressively stress-provoking. After the extension, the data results in a set of 2243 instances, where 63.70% are labeled as non-stress and 36.30% as stress. This labeling was carried out by three judges, taking as ground truth label the majority vote between them.
- The *TensiStrength* dataset, collected by M. Thelwall, contains 6142 tweets labeled manually. First, the tweet collection was carried out using keywords from various sources, and then the collected tweets were labeled using a five-point scale system. A text is annotated as “−1” if its content is not related to stress and “−5” if it describes situations likely to cause high levels of stress. Thus, a number between −2 and −4 indicates that the text somehow describes stress-related situations or matters. During pre-processing, we have transformed the 1–5 annotation into a binary system to use with our binary classifier. To convert these levels into binary classes, we transformed only texts annotated as “−1” (i.e., those that did not contain any reference to stress or stressful situations) into “0”. Thus, all tweets describing stressful situations to a greater or lesser extent and therefore originally annotated by a number between “−2” and “−5” have been annotated as “1” (stress). This transformation results in 41.30% of non-stress posts and 58.70% of stress posts.

The same pre-processing has been applied to all the data: normalization of Uniform Resource Locators (URLs), capital letters, numbering, and contractions (e.g., It’s, we’ll), resulting in posts consisting of lower case tokens with the punctuation removed. Table 1 presents the used datasets along with a summary of some statistics.

Furthermore, a word frequency analysis has been carried out in order to identify stress patterns. This analysis is shown in Fig. 1, a word frequency scatter plot which depicts a visualization of the most used words in the data accordingly to the stress and non-stress categories. The figure has been generated using Scattertext library (Kessler, 2017), a tool for finding specific terms in corpora and visualizing them.

Table 1
Statistics of the used datasets.

	Dreaddit	TensiStrength	NSE
No. of posts	2294	6142	2243
No. posts w/Stress	1246	3605	813
Avg. no. of words	88.32	15.56	16.65
Avg. no. of chars	243.48	47.01	45.62

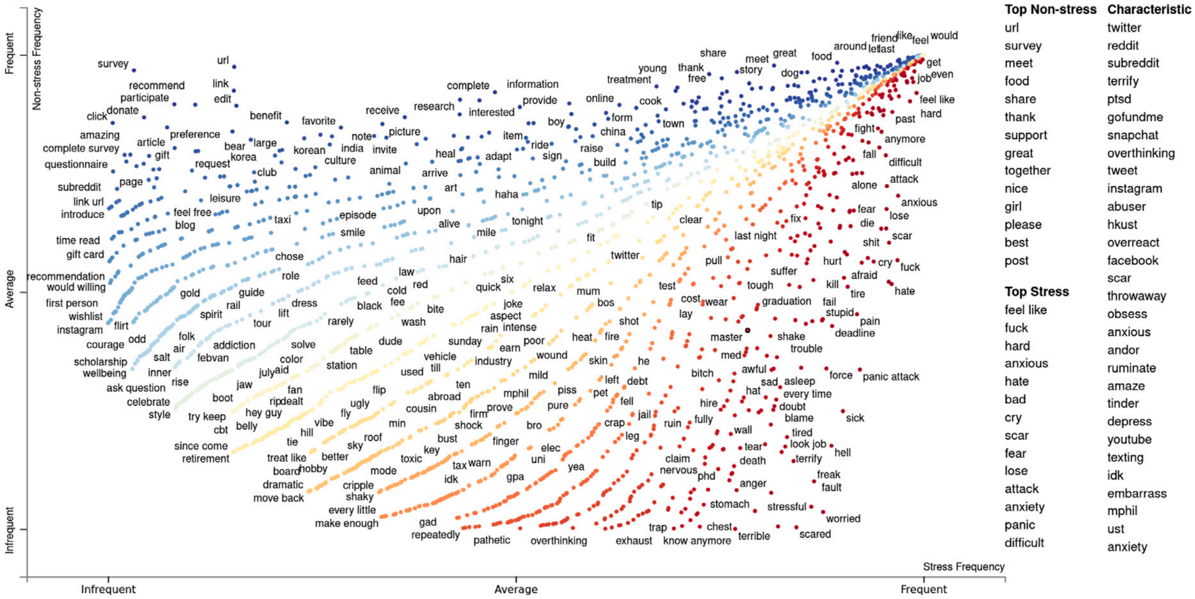


Fig. 1. Normalized word frequency for stress and non-stress categories for the data. On the right, a list containing most frequent words for non-stress (Top Non-stress), stress (Top Stress), and both (Characteristic) is shown. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The different colors indicate the frequency of each class: blue for non-stress and red for stress. The frequency of each different word for each category is computed. Thereby, the frequency of the words in the stress category is represented on the x-axis, and the frequency of the words in the non-stress category is pictured on the y-axis. In this manner, a word that frequently appears in non-stress texts will be placed in the top area, whereas a word that frequently appears in stress-annotated texts will be placed in the right area. Consequently, the most frequent words appear in three particularly interesting areas of the figure according to the category: bottom right (common in stress texts), top left (common in non-stress texts), and top right (common in both non-stress and stress texts). Those areas present the most characteristic words for the non-stress, stress, and both categories, offering a view of which words are commonly used in each category. For instance, common stress words are “hate”, “fear”, and “anxious”; whereas non-stress texts frequently contain words such as “thank”, “food”, and “support”.

Besides, an analysis has been carried out to identify topics. This allows us to visualize the most common topics present in our data depending on the stress level. We have performed topic identification using Scattertext (Kessler, 2017) with Empath (Fast, Chen, & Bernstein, 2016). This tool enables the on-demand generation and validation of new lexical categories and the analysis of text across 200 categories generated from common topics.

The results of this analysis can be seen in Fig. 2. The figure consists of a frequency scatter plot of the most frequent topics accordingly to the stress and non-stress categories. The color indicates the frequency of the topic with regard to each class. Those topics associated with non-stress are blue, and those more associated with stress are red. The position along the axis indicates the frequency in the classes. In the far upper right-hand corner, we can see topics highly associated with both classes, while in the bottom left corner, we see topics with low frequency in our data. The most frequent topics for non-stress and stress are listed on the right. For example, common stress topics are “timidity”, “weakness” and “anger”; and non-stress common topics are “internet”, “shopping” and “tourism”. The analysis points out that most topics do not present substantial frequency differences between stress and non-stress posts. The frequency of topics such as “religion”, “wealthy”, and “home” is similar in both cases. However, there are a few key topics where this frequency difference is significant. For example, “swearing_terms”, “horror”, and “aggression” are topics very frequent in stress posts but uncommon in non-stress posts. This fact indicates that some topic-related features may provide relevant information for the task of stress classification.

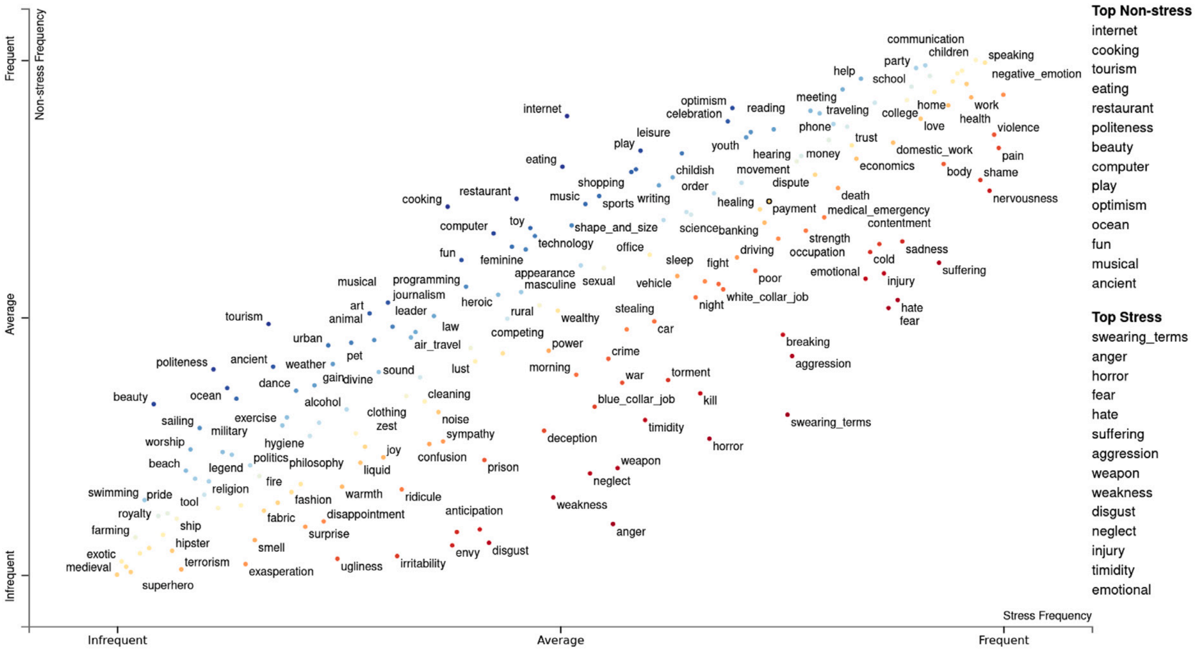


Fig. 2. Comparison of Empath topics for both stress and non-stress categories for the data. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4. A feature framework for stress detection from text

In this section we introduce a feature framework aiming at helping in the organization and characterization of features for stress detection in text. These features are based on existing lexicons and sentiment analysis methods that enable the extraction of information from textual data. Different lexicons have been used to identify which of them provide features more relevant for the task of stress detection: Linguistic Inquiry and Word Count (LIWC) (Pennebaker, Francis, & Booth, 2001), General Inquirer (GI) (Stone, Dunphy, & Smith, 1966), Lasswell (Lasswell & Namenwirth, 1969), Geneva Affect Label Coder (GALC) (Martin & Pu, 2014), Affective Norms for English Words (ANEW) (Bradley & Lang, 1999), EmoLex (Mohammad & Turney, 2010), SenticNet (Cambria, Liu, Decherchi, Xing, & Kwok, 2022), Valence Aware Dictionary for Sentiment Reasoning (VADER) (Gilbert & Hutto, 2014), Hu–Liu polarity (Liu, Zeng, Li, & Hu, 2004), and Empath (Fast et al., 2016):

- *GI* (Stone et al., 1966) includes 119 features regarding institutions, roles, semantic, lexical, and syntactic dimensions, pleasure, places, communication, or social categories.
- *Lasswell* (Lasswell & Namenwirth, 1969) includes 69 features related to affection, wealth, well-being, respect, or power.
- *GALC* (Martin & Pu, 2014) consists of word lists concerning to 36 specific emotions (such as anger, guilt, joy, or hope) and two general emotional states.
- *ANEW* (Bradley & Lang, 1999) includes a total of 6 affective norms for valence, arousal, dominance, and pleasure.
- *EmoLex* (Mohammad & Turney, 2010) contains 10 lists of words and bigrams evoking particular emotions (such as joy, sadness, anger, fear, or disgust).
- *SenticNet* (Cambria et al., 2022) is a database expansion of WordNet that contains norms for around 13 000 words related to sensitivity, aptitude, attention, and pleasantness.
- *VADER* (Gilbert & Hutto, 2014) consists of a rule-based sentiment analysis system which was particularly developed for shorter texts, making it very useful in social media contexts.
- *Hu–Liu* (Liu et al., 2004) includes two large polarity lists for the purposes of sentiment analysis.

All these lexicons have been used utilizing SEANCE (Crossley, Kyle, & McNamara, 2017), an automatic tool that enables the analysis of sentiment, social order, and social cognition in text. Besides, some stylistic features have been considered, such as the Automated Readability Index (ARI) and the Flesch-Kincaid Grade Level (FKG). Both scales are intended to measure the understandability and readability of a text. Finally, also LIWC (Pennebaker et al., 2001) and Empath (Fast et al., 2016) have been used. LIWC is a lexicon-based tool that provides scores for psychologically relevant categories such as joy, sadness, or certain cognitive processes. In contrast, Empath contains 200 pre-validated categories generated from common topics. Furthermore, we

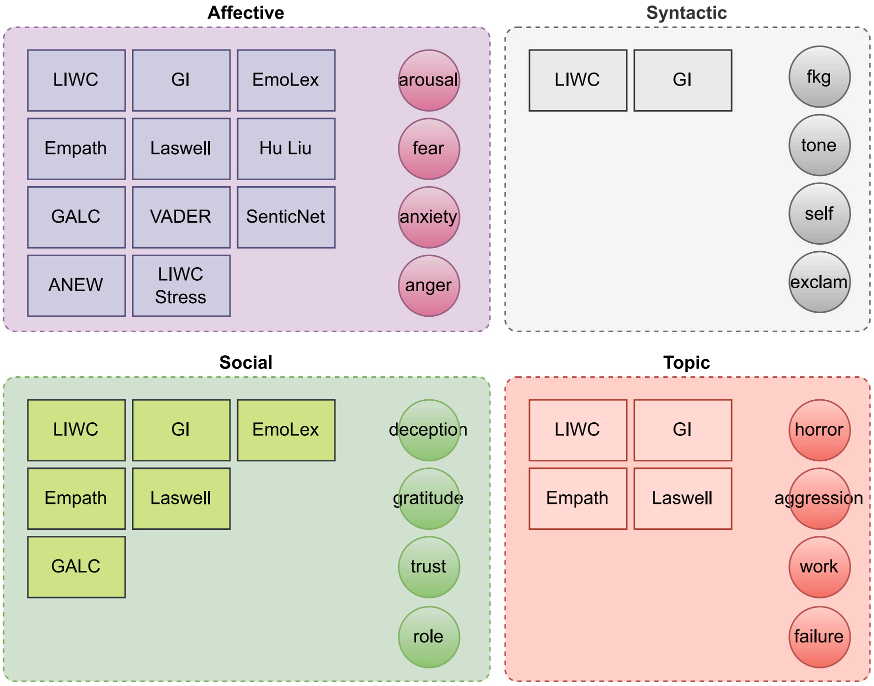


Fig. 3. Overview of the proposed feature framework. Blocks represent the lexicons which contribute to each set, whereas circles represent example features of each set.

have also included the LIWC stress dictionary (Wang, Hernandez, Newman, He, & Bian, 2016), a dictionary specifically intended to measure psychological stress.

This results in a total of more than 500 features with different types and natures. To analyze the influence of the features according to their nature, we have split them into four sets: affective, social, syntactic, and topic-related features. Affective features are those related to sentiment (positive, negative, or neutral), emotion (e.g., fear, sadness, joy), and mood (e.g., anxiety, attention, nervousness). All used lexicons contain categories of this nature, resulting in 173 features. Some examples of features belonging to this set are *fear*, *anxiety*, or *arousal*. Features related to social relations belong to the social set, which comprises categories such as *trust*, *deception* or *gratitude*. This set comprises 65 features extracted from the lexicons Empath, LIWC, GALC, GI, Laswell, and EmoLex. Only LIWC and GI contribute to the syntactic set, which also contains the previously mentioned stylistic features, ARI and FKG. Seventy-five features related to how the text has been written compound this set. Some examples are *tone*, *self* and *fkg*. Finally, the topic set comprises 272 categories to identify the different topics present in the text. Empath, LIWC, Laswell and GI dictionaries contain features of this nature, such as *horror*, *aggression* or *work*. Fig. 3 shows these feature sets, indicating which lexicon contributes to each set, along with some examples.

Finally, an analysis has been conducted to analyze how well each feature splits the posts from each corpus. This analysis allows us to gain insights into the differences between the different sources of the data and how each feature set performs on each source. The information gain criterion of each feature in each class has been computed for this purpose. This metric measures the entropy's reduction within each class once the best split induced by the feature has been conducted. Given a feature F and a class C, the information gain can be calculated as:

$$I_{gain}(F,C) = H(C) - H(C \mid F) \tag{1}$$

where $H(C)$ represents the entropy of the class and $H(C \mid F)$ is the conditional entropy of the class given the feature F. The minimum value of the information gain is achieved when $H(C \mid F) = 1$, that is, the feature F and the class C are unrelated. In contrast, a feature F that only appears in a specific class C would yield the maximum information gain value.

Fig. 4 shows the distribution of the achieved information gain values for each feature set. We can see that a significant part of the information gain values is close to 0. This indicates that a significant number of features do not provide enough information gain. However, we can also appreciate a long tail in the distribution for all feature sets. This elongated distribution, which is especially long for affective features, suggests that there are features that could potentially provide good results when being exploited by the classification model. For affective features, the elongated tail also has wider portions, indicating the existence of multiple features with high information gain. On the other hand, the tail is narrower for the social features set, indicating fewer social features with high information gain. We can observe that the information gain distribution shows a shorter tail for the syntactic features but presents wider sections for high information values. If we analyze the information gain ranges, we can see that they are greater for

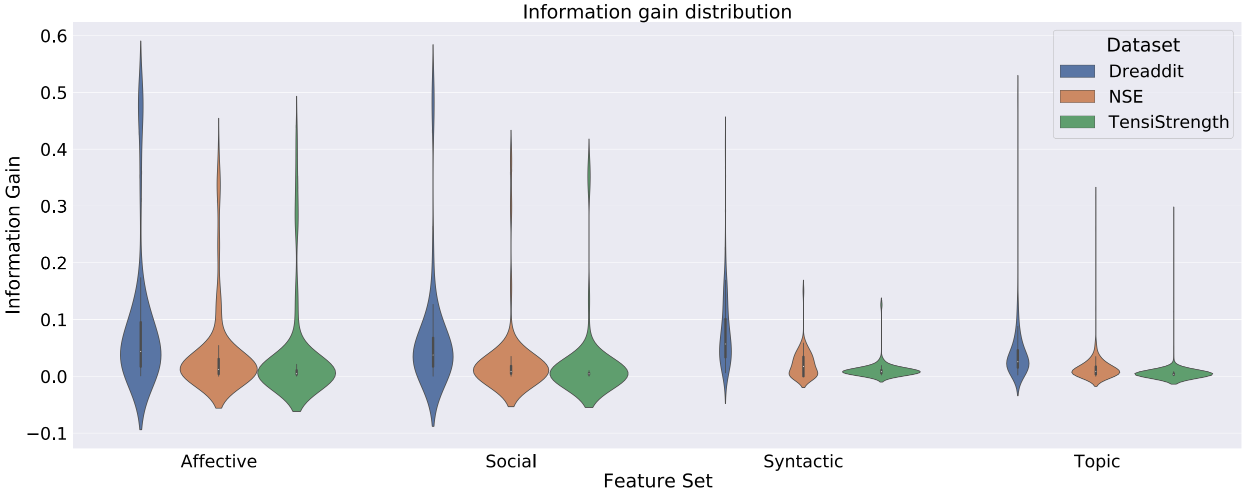


Fig. 4. Information gain distribution for features belonging to each set for the three datasets.

the affective set. This indicates that this set may contain a more significant number of relevant features for the stress classification task.

Moreover, when comparing among datasets, we can observe that the information gain achieved by all feature sets decreases with the length of the posts. Even so, analyzing the information gain ranges for NSE and TensiStrength, we can appreciate that affective features still obtain significant information values for corpora with a shorter length. Furthermore, we can observe how the information gain distribution form of affective features remains very similar for the three datasets, and the information gain range remains higher in all corpora.

By analyzing the scope, it can be noticed that the information provided by the lexicon-based features may be beneficial for the stress classification task. Specifically, the analysis points out the relevance of affective features (**RQ1**), given the promising information gain obtained by features from this set.

5. Stress classification models

Once we have analyzed the main features of the lexicons for stress characterization, we aim to use them to develop a stress classifier. Besides, we aim to explore the combination of lexicon-based features with distributional representations. With this purpose, three different models are proposed: a lexicon-based features model (M_{LF}), a distributional representation model (M_{DR}), and an ensemble model (Araque, Corcuera-Platas, Sánchez-Rada, & Iglesias, 2017) combining lexicon-based features with distributional representations (M_E). These models are described in the following subsections.

5.1. Lexicon-based features model (M_{LF})

The first model, M_{LF}, which is shown in Fig. 5, makes use of affective, social, syntactic, and topic features extracted from the text. These features have been extracted using the framework described in Section 4.

This model aims at investigating whether lexicon-based features are relevant for stress detection and to which extent. We propose a lexicon-based representation to encode the text into a fixed-length vector. Consider a set of c lexicons $L = \{l_1, \dots, l_i, \dots, l_c\}$. Each lexicon l_i is composed by a vocabulary of n words $W(l_i) = \{w_1, \dots, w_j, \dots, w_n\}$ and a set of m features $F(l) = \{f_1, \dots, f_k, \dots, f_m\}$. For each word w_j in the lexicon there is a feature vector $P(w_j) = [p^{w_j}_{f_1}, \dots, p^{w_j}_{f_k}, \dots, p^{w_j}_{f_m}]$ of numeric annotations that express the intensity of each feature f_k for this word. Thus, the lexicon contains n feature vectors of dimension m and the lexicon annotation matrix has dimension $n \times m$. Following, let $W(s) = \{w_1, \dots, w_h, \dots, w_S\}$ represent each post to analyze, with a length S and containing each input word w_h. For each word $w_h \in W(s)$, the associated feature vector P is extracted. In case the word w_h is not contained in the lexicon ($w_h \notin W(l_i)$), the resulting vector will have value zero in all positions. This process results in a matrix M containing the feature annotation for all the input words:

$$M(l_i) = \begin{pmatrix} p^{w_1}_{f_1} & \cdots & p^{w_1}_{f_k} & \cdots & p^{w_1}_{f_m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ p^{w_h}_{f_1} & \cdots & p^{w_h}_{f_k} & \cdots & p^{w_h}_{f_m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ p^{w_S}_{f_1} & \cdots & p^{w_S}_{f_k} & \cdots & p^{w_S}_{f_m} \end{pmatrix} \tag{2}$$

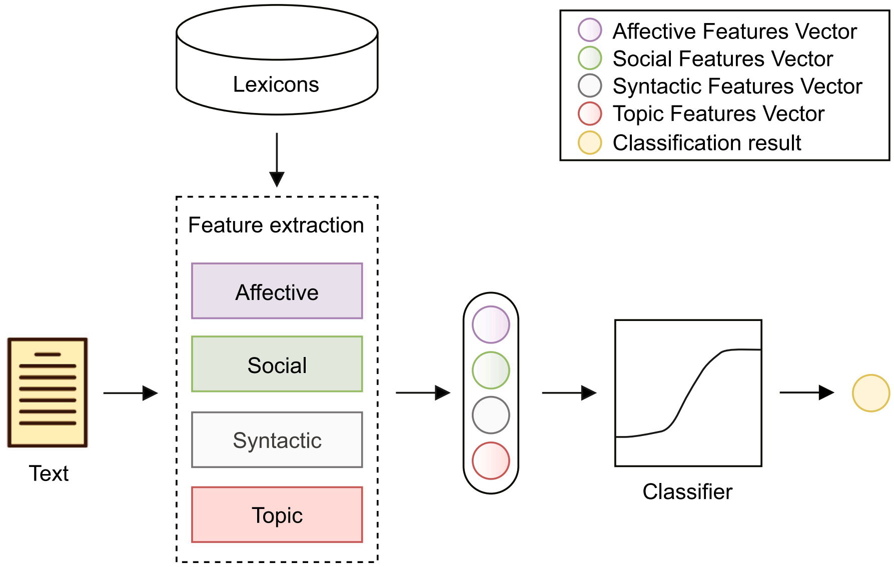


Fig. 5. General architecture representation of the M_{LF} model.

In order to compute the annotation of the post for this lexicon, the average of the feature annotation of all words is considered, resulting in a lexicon annotation vector $V(l_i) = [v_{f_1}, \dots, v_{f_k}, \dots, v_{f_m}]$, with length m, where:

$$v_{f_k} = \frac{1}{S}\sum_{h=1}^{S} p_{f_k}^{w_h} \tag{3}$$

In case no word in the post is contained in the lexicon, that is, $W(l_i) \cap W(s) = \emptyset$, the resulting matrix will have value zero in all positions, and consequently the lexicon annotation vector $V(l_i)$ will be a zero vector. Finally, the annotation vectors of all lexicons are concatenated, resulting in the annotation vector:

$$A = \bigoplus_{i=1}^{c} V(l_i) \tag{4}$$

Algorithm 1 shows the proposed feature extraction method. The function *annotation* extracts the feature vector corresponding to each input word w_h. If the word is not contained in the lexicon l_i, the resulting vector will be a null vector. This operation results in the matrix M containing the feature vector extracted from the lexicon for all the words in the post. The function *average* computes the average of the feature annotation of all words, resulting in the annotation vector V of the entire post for the lexicon l_i. Finally, the function *concat* concatenates the post's feature vectors of all the lexicons in L. The resulting vector A, containing all relevant information extracted from the lexicons, is fed to a machine learning classifier.

Algorithm 1 Lexicon-based feature extraction algorithm

```
Require: Set of lexicons $L$, each of them composed by a vocabulary $W(l_i)$ and the set of features $F(l)$; and an input post $W(s)$
Ensure: $A \in \mathbb{R}^m$
  for all $l_i \in L$ do
      for all $w_h \in W(s)$ do
          $M_{h,:} \leftarrow annotation(w_h, l_i)$
      end for
      $V(l_i) \leftarrow average(M)$
  end for
  for
      $i \leftarrow 1, c$ do
      $A \leftarrow concat(V(l_i))$
  end for
```

5.2. Distributional representation model (M_{DR})

The second model (M_{DR}) uses word embedding techniques to exploit distributional representations. An overview of the model is given in Fig. 6. Word embeddings are a type of computing distributed text representation that gives words with similar meanings

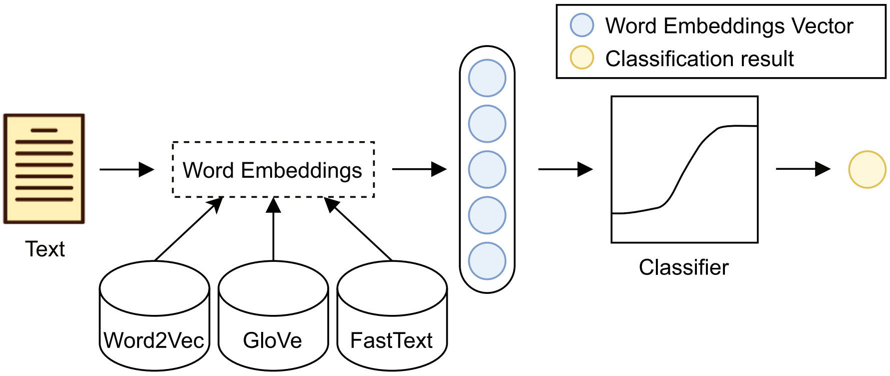


Fig. 6. General architecture representation of the M_{DR} model.

a similar representation (Mikolov et al., 2013). This enables the conversion of text into vector representations where semantic and syntactic information is encoded. However, these vectors do not enclose any information related to sentiment or cognition.

Consider a vocabulary of n words $W(e) = \{w_1, \dots, w_j, \dots, w_n\}$. Word embeddings are encoded by column vectors in an embedding matrix M with dimension $d \times n$. In this case, d is the dimension of the word vectors and n is the size of the vocabulary. Each column of the matrix M represents the embedding vector of a word existing in the vocabulary. The matrix components are parameters to be learned according to the word embedding technique used. Consider now a post to analyze $W(s) = \{w_1, \dots, w_h, \dots, w_S\}$ with a length S and containing each input word w_h. For each word w_h of the intersection $W(e) \cap W(i)$, its word embeddings vector Q_{w_h} results from the matrix–vector product $Q_{w_h} = M \cdot v_{w_j}$. In this case, v_{w_j} is the one-hot vector of word w_h. It has value one at j and zero in the rest. Finally, the embedding vector of each word in the post can be combined into a unique vector E representing the entire text:

$$E = \frac{1}{S} \sum_{h=1}^{S} Q_{w_h} \tag{5}$$

Thus, this vector E will be fed to the machine learning classifier. Some popular word embedding techniques are Word2Vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014) and FastText (Joulin et al., 2016). Each of them presents a different manner of learning the embedding matrix. Word2Vec learns it by connecting target words to their context, regardless of the frequency in which the words appear. A frequent co-occurrence of words in Word2Vec results in more training instances but does not give new information. GloVe, on the other hand, emphasizes the importance of taking into account the frequency of co-occurrence. In this way, GloVe learns embeddings so that a set of word vectors corresponds to the likelihood of these words co-occurring in the corpus. Also, whereas Word2Vec has a predictive nature, GloVe is count-based. Finally, FastText is intended to improve Word2Vec. It is based on the same principles, but instead of using words to build word embeddings, FastText uses a combination of lower-level embeddings of parts of words and characters. This reduces the amount of training data needed since each piece of text contains more information and enables generalization, as new words may contain the same characters as previously learned words. This fact allows FastText to obtain theoretically better vector representations than GloVe or Word2Vec in corpora with specific domain rare words. It constructs a word vector from its character n-grams even when the word is not contained in the training corpus.

To evaluate how each technique impacts the performance of the stress detection, the three described techniques have been used for extracting word vectors. However, to make GloVe more adequate to our data, the pre-trained model has been fine-tuned using our corpus. In this manner, the pre-trained model has been trained with the used datasets to learn the domain-specific vocabulary. This fine-tuning has been carried out with Mittens (Dingwall & Potts, 2018), an extension of GloVe that allows us to update general-purpose representations with data from a specialized domain. Thus, the unseen vocabularies are also added to the model.

5.3. Ensemble model (M_E)

Finally, the third model (M_E) takes into account both distributional representations and lexicon-based features. The general architecture representation of the model is found in Fig. 7. This model combines each text instance's word embedding vector representation with the lexicon-based feature vector drawn from it. Let $W(s) = \{w_1, \dots, w_h, \dots, w_S\}$ represent each post to analyze, with a length S and containing each input word w_h. The feature vector A and the embeddings vector E are computed using the above-mentioned methods. Then, these vectors are concatenated into a unified vector C containing information related to lexicon-based features and word embeddings:

$$C = A \oplus E \tag{6}$$

This vector is fed to the machine learning classifier. Thus, the information given by word embeddings is combined with the affective and lexical information given by the lexicon-based features. This information combination may improve the performance

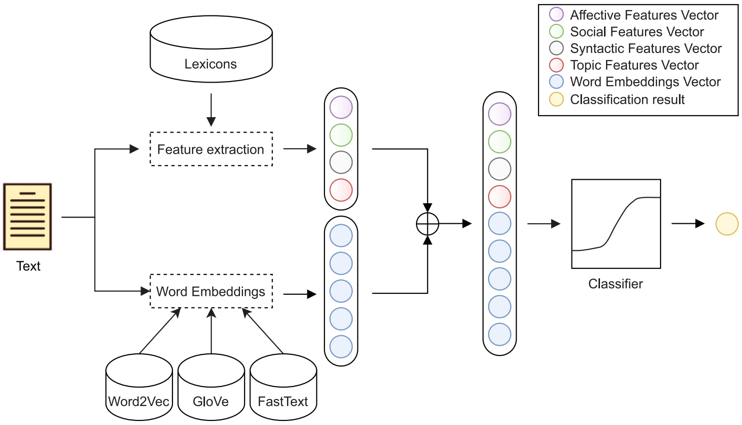


Fig. 7. General architecture representation of the M_E model.

of a classifier algorithm learning from this unified set compared to the one learning only from word embedding or lexicon-based features data.

The proposed models have been validated in three datasets from different sources (Reddit, Twitter, and personal interviews) using the data described in Section 3. Furthermore, the experiments allow us to analyze the performance of different lexicons or word embedding techniques, as described in the following section.

6. Evaluation

6.1. Methodology

To evaluate the effectiveness of the proposed models in stress classification, an experimental study has been designed according to the research questions described in Section 1. The main goal of the experiment is to provide insight into which models and techniques perform better for stress detection in different corpora. With this aim, we postulate the problem as a binary classification task that aims to detect stress evidence in text. This is accomplished by learning from the provided lexicon-based features and the computed word embeddings. The stress detection is conducted at the post level, classifying posts as either stress or non-stress. In order to analyze which lexicon-based features perform better for stress detection (**RQ1**), several experiments have been carried out, splitting them by nature or lexicon according to the feature framework proposed in Section 4. Also, the three proposed models are evaluated separately to investigate the performance of the different methods, and they have been compared with state-of-the-art solutions (**RQ2**). This evaluation has been conducted using three English public datasets to determine whether the proposed approach can achieve good results in different corpora (**RQ3**).

The proposed models have been evaluated using three different machine learning classifiers implemented with Python scikit-learn library (Pedregosa et al., 2011): Support Vector Machines (SVMs), logistic regression, and Stochastic gradient descend (SGD) classifier. SGD classifier implements regularized linear models (SVM, logistic regression, etc.) with Stochastic gradient descend (SGD) learning. In our experiments, SGD has been used with SVM as the model to fit. In addition, our work has been compared with the previous works in the field of stress detection from text that use any of the public datasets considered in this article. These works are: the lexicon approach (Thelwall, 2017); a combination of Word2Vec embeddings with LIWC features using a logistic regression classifier (Turcan & McKeown, 2019); and a Bidirectional Long-Short Term Memory (LSTM) with Attention and Word2Vec Embeddings (Winata et al., 2018). Since each method was validated only with one of the considered datasets in the original works, we evaluated them with the datasets considered in this article. In addition, we have applied Bidirectional Encoder Representations from Transformers (BERT) to the task of stress detection, using the pre-trained BERT-base (Kenton & Toutanova, 2019). We have used 10-fold validation and the weighted average of the F1-Score as the performance metric in all experiments. F1-Score is defined as the harmonic mean of the model precision and recall and enables the evaluation of a model accuracy:

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \tag{7}$$

Finally, a Friedman statistical test (Demšar, 2006) and a cross-dataset experiment have been carried out. The former allows us to further study the performance and impact of the proposed models. In contrast, the latter allows us to analyze the generalization performance. To sum up, the methodology followed for evaluating our models consists of the following steps:

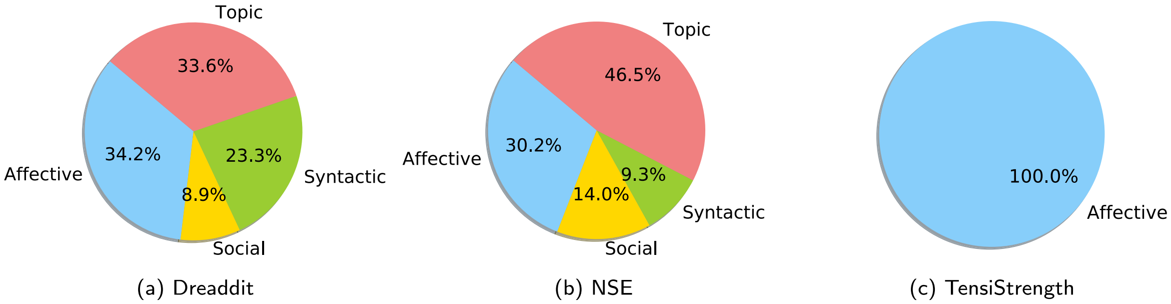
1. Evaluation of the M_{LF} model, analyzing the impact of the proposed feature sets and the different lexicons.
2. Evaluation of the M_{DR} model, identifying the best word embedding technique.
3. Evaluation of the M_E model, comparing the performance of our best-performing method with existing works in stress detection from text.

Table 2
Best performance achieved with the M_{LF} model with each feature set for each dataset.

Model	Features	Dreaddit	NSE	TensiStrength
M_{LF}	Affective	0.7908	0.7289	**0.6314**
	Social	0.6762	0.6532	0.5288
	Syntactic	0.7869	0.7188	0.5341
	Topic	0.7541	0.7223	0.5415
	All	**0.8210**	**0.7750**	0.6201

Table 3
Best performance achieved with the M_{LF} model with each feature set for each dataset using RFE.

Model	Features	Dreaddit	NSE	TensiStrength
M_{LF} (RFE)	Affective	0.7931	0.7295	0.6322
	Social	0.6851	0.6645	0.5395
	Syntactic	0.7911	0.7269	0.5897
	Topic	0.7622	0.7289	0.5443
	All	**0.8257**	**0.7777**	**0.6322**



(a) Dreaddit (b) NSE (c) TensiStrength

Fig. 8. Feature set representation between the selected features after running the RFE method for each dataset.

4. Comparison of the performance of all considered methods through a statistical test.
5. Analysis of the generalization performance in our best-performing method by a cross-dataset experiment.

6.2. Results

First of all, we proceed to evaluate the M_{LF} model in order to analyze whether the use of lexicon-based features can provide good results for stress classification. A first experiment has been carried out using all features of each set, and the results are shown in Table 2. The best results are obtained using Linear SVM for the NSE dataset and Logistic Regression for Dreaddit and TensiStrength datasets. As we can see, the best results are achieved when combining all the features, except for TensiStrength, where the best results are obtained using only affective features. This can be explained by observing the low results obtained in this dataset for the other sets and indicates that only affective features perform well on the microblogging source.

It is also logical to think that not all extracted features are useful in stress detection, and some might even decrease the classification accuracy. We have performed a feature selection to further study this fact and enhance our model. First, we have removed those features that presented a high correlation (greater than 0.95) with other features, as they would have almost the same effect on stress prediction. Then, we used RFE with cross-validation to obtain the best features. Table 3 shows the results of this experiment. The best results are obtained using Linear SVM for the NSE dataset and Logistic Regression for Dreaddit and TensiStrength datasets. As we can see, the results always improve after performing the feature selection.

One of the first points to highlight is that the best results are always achieved when performing a feature selection between all the feature sets combined. This shows that combining different kinds of features is a good approach to classifying stress. By analyzing which kind of features perform better, we find that the affective and syntactic features are those with better results in almost every case. Furthermore, we see that topic-related features also achieve good results for the NSE dataset, whereas social features present the lowest performance in all cases (**RQ1**). We can analyze this in more detail in Fig. 8, which shows the distribution of feature sets between the selected features for each dataset.

The figure shows that affective features present the highest rates for all datasets except for NSE. This exception can be explained by observing Table 3, which shows that the topic features presented promising results for this corpus. It is also interesting to see how for the TensiStrength dataset, all the selected features are from the affective set. The bad results obtained for this dataset using the other kinds of features explain this fact. Finally, the figure also shows the low representation of social-related features among the selected ones, as we could predict from Table 3. These results are coherent with the conclusions drawn from Fig. 4 in Section 3.

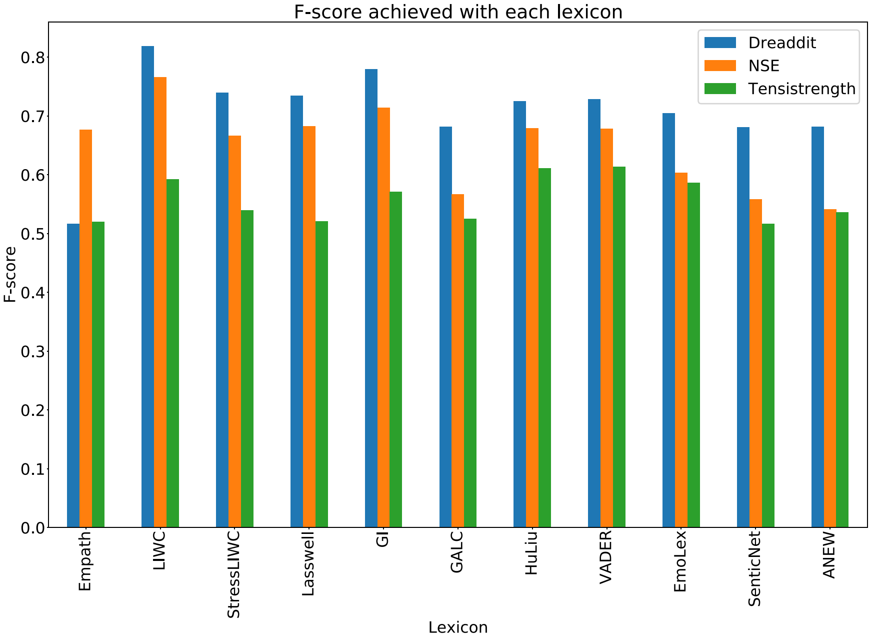


Fig. 9. Performance of the different lexicons used.

Table 4
Best performance achieved with the M_{DR} model for each word embedding technique and for each dataset.

Model	Embeddings	Dreaddit	NSE	TensiStrength
M_{DR}	Word2Vec	0.7901	0.6487	0.5822
	GloVe	0.8334	0.7813	0.7674
	FastText	**0.8487**	**0.8072**	**0.7705**

Besides, we have experimented with the features divided by the lexicon to which they belong to obtain insights into which lexicons are more interesting for stress classification. Fig. 9 shows the best performances of the M_{LF} model using only selected features from each lexicon. We can observe that LIWC and GI seem to be the lexicons that perform better for the Dreaddit and NSE datasets. If we analyze Fig. 9 along with Fig. 3, we can see that these lexicons are the only ones that provide features of all kinds. Also, we see that these lexicons are the only ones that contribute with syntactic features, one of the most relevant sets according to the previously commented results. However, in the TensiStrength dataset, better results are obtained with Hu–Liu and VADER lexicons. As previously mentioned, affective features perform much better than other types of features in this dataset. In addition, it contains shorter posts. This explains why the best results are obtained with lexicons more focused on affective features. Moreover, VADER is specifically designed for short texts and is particularly effective on texts coming from social networks.

The results of the M_{LF} model indicate that stress classification using affective, syntactic, social, and topic-related features can be successful for datasets coming from Reddit and interviews (achieving an F-score of 0.8257 for the Dreaddit dataset). However, these features do not perform so well for short text data coming from microblogging posts (as we can see from the results for the TensiStrength dataset). Still, the performance obtained using only lexicon-based features (M_{LF}) is comparable to more complex methods, which take into account the visual attributes existing in social media data (Winata et al., 2018). Besides, the results suggest that affective features are those which perform better for the stress classification; and that the best results come from their combination (**RQ1**).

Once obtained the results for the M_{LF} model, we proceed to evaluate the distributional representation model (M_{DR}). For this purpose, an experiment has been carried out using three different word embedding techniques: Word2Vec, GloVe, and FastText.

Results shown in Table 4 indicate that the use of distributional representations (M_{DR}) improves the performance over the M_{LF} model. Concerning the word embedding technique comparison, we can observe that the FastText method provides the best performance. Nevertheless, the differences between FastText and GloVe performances are pretty slight except for the NSE dataset. As explained in Section 5, FastText is supposed to achieve better results in corpora with rare domain-specific words. The fine-tuning of the GloVe embeddings to each dataset should enhance the performance, but we think that a greater amount of data would be needed to make it overtake the performance of FastText. We can support this hypothesis by analyzing the performance of Word2Vec. As can be observed, this method is more sensitive to corpora with domain-specific words. Therefore, it performs considerably worse than the other two, given that it has not been fine-tuned.

Analyzing the results from the M_{DR} model, we can observe that the use of distributional representations supposes an interesting approach. The improvement achieved by this method compared to the M_{LF} model is especially significant in short texts. The F-score

Table 5
Best performances achieved with the M_E model (lexicon-based features + distributional representation) and comparisons of these results with the state-of-the-art baselines in stress classification from text and with the other models proposed in this article.

Method	Dreaddit	NSE	TensiStrength
Lexicon approach (Thelwall, 2017)	0.7040	0.5921	0.7130
LogReg + LIWC + Word2Vec (Turcan & McKeown, 2019)	0.7980	0.6987	0.6270
BiLSTM + Word2Vec (Winata et al., 2018)	0.7460	0.7430	0.7410
BERT base	0.8479	0.7809	0.7562
M_{LF} (Features)	0.8257	0.7777	0.6322
M_{DR} (FastText)	0.8487	0.8072	0.7705
M_E (Features + FastText)	**0.8604**	**0.8372**	**0.7750**

of 77.05% achieved for the TensiStrength dataset supposes an improvement of more than 10 points compared to the previous model. For the NSE dataset, the improvement is also relatively significant. This model also improves the results for corpora with longer texts, as shown by the F-score of 84.87% achieved for Dreaddit. Regarding the performance comparison between classifiers, the best results are obtained using SGD classifier for the Dreaddit dataset, Logistic Regression for the NSE dataset, and Linear SVM for the TensiStrength dataset.

Once the results for the distributional representation and lexicon-based models have been analyzed separately, we proceed to analyze their combination (M_E model). For this experiment, we took those features that obtained the best results for each dataset and combined them with word embeddings. Some examples of the selected features are: "arousal", "disgust" and "deception".

Table 5 shows the best results obtained for the M_E model and compares them with other models proposed in this paper. The results show that the proposed method achieves considerably good performance for all datasets. Let us compare the proposed M_E model with the other models proposed in our work. We see that the combination of lexicon-based features with distributional representations enhances the performance in all datasets. This enhancement is especially high for data coming from personal interviews (NSE dataset), where the improvement in the performance reaches 3 points compared to the distributional representation model (M_{DR}) and 6 points compared to the lexicon-based features model (M_{LF}). Concerning the data coming from Reddit (Dreaddit dataset), the results show an improvement of 1 point compared to M_{DR}; and almost 3 points compared to M_{LF}. Finally, the improvement in the TensiStrength data is smaller compared to the M_{DR} model but very high compared to the M_{LF} model. This can be easily explained if we consider the lower performance of the lexicon-based features for this dataset.

The M_E model surpasses the 80% F-score in the Dreaddit (86.04%) and NSE datasets (83.72%) and yields to 77.50% in the TensiStrength dataset. Regarding classifiers, the best results are obtained using Logistic Regression for Dreaddit and Tensistrength datasets; and Linear SVM for the NSE dataset. The results obtained for each model with each classifier are publicly available online for the interested reader.[1] In addition, our work has been compared with previous works in the field of stress detection from text which uses any of the public datasets considered in this article. These works are: the lexicon approach (Thelwall, 2017); a combination of Word2Vec embeddings with LIWC features (Turcan & McKeown, 2019); and a Bidirectional LSTM with Attention and Word2Vec Embeddings (Winata et al., 2018). Furthermore, we have applied BERT to the task of stress detection, using the pre-trained BERT-base (Kenton & Toutanova, 2019). As a result, we can appreciate that our model outperforms the current state-of-the-art methods for all datasets. Whereas a greater amount of data could be needed to benefit from the advantages of deep neural networks, the proposed solution yields good results on small datasets.

To further study the performance and impact of the proposed models, a Friedman statistical test (Demšar, 2006) has been carried out. This test aims to determine if we may conclude from the sample of results that there is a difference between the classification methods. As a result, the Friedman test outputs a ranking of methods regarding their effectiveness in different datasets. A lower ranking indicates a better performance of the specific method than the rest.

The first step in calculating the Friedman test is to convert the original results to ranks. Let r_i^j be the rank of the jth algorithm on the ith dataset, and k and n the number of methods and datasets respectively. Friedman's test compares the average ranks of the methods $R_j = \frac{1}{n}\sum_i r_i^j$ and states that the Friedman statistic under the null hypothesis (that is, all the algorithms are equal, so their ranks are also equal) with $k-1$ degrees of freedom is:

$$X_F^2 = \frac{12n}{k(k+1)}(\sum_j R_j^2 - \frac{k(k+1)^2}{4}) \tag{8}$$

However, Iman and Davenport (1980) proposed a better static distributed based on the F-distribution with $k-1$ and $(k-1)(n-1)$ degrees of freedom:

$$F_F = \frac{(n-1)X_F^2}{n(k-1)-X_F^2} \tag{9}$$

We perform the test with an α value of 0.1, $k = 19$ (the number of methods included in the analysis), and $n = 3$ (number of datasets). With these data, $X_F^2 = 39.72$, $F_F = 5.56$, and the critical value $F(k-1,(k-1)(n-1)) = 1.64$. As $F_F > F(18,36)$, the null hypothesis of the Friedman test is rejected, and the results are statistically relevant. For simplicity, Table 6 shows the best five approaches according to their ranks, as computed by the Friedman test.

[1] https://gsi.upm.es/∼smunoz/stress-text/additional_material_stress_text.pdf.

Table 6
Friedman rank for the top-5 methods.

Method	Rank
M_E (Log. Reg. + Features + FastText embeddings)	2
M_E (Lin. SVM + Features + FastText embeddings)	3
M_{DR} (Log. Reg. + FastText embeddings)	5.5
M_E (Lin. SVM + Features + GloVe embeddings)	5.67
BERT base	7

Table 7
Averaged F1-Scores for the best-performing classification method in a cross-dataset evaluation.

	Dreaddit	TensiStrength	NSE
Dreaddit	0.8604	0.6883	0.6837
TensiStrength	0.7764	0.7750	0.6615
NSE	0.7370	0.6176	0.8373

As can be noted, Friedman's test confirms the combination of lexicon-based features with distributional representations as the best-performing approach for detecting stress in texts (**RQ2**). The two lower ranks are obtained when combining the extracted features with the FastText word embeddings using Logistic Regression and Linear SVM (ranks of 2 and 3, respectively). These two methods present results significantly better than the others. The results demonstrate the effectiveness of combining lexicon-based features with distributional representations for detecting stress from text, especially using FastText embeddings. Regarding the comparison with state-of-the-art works, only the use of BERT is ranked among the top-5 analyzed methods.

Once the best classification method has been identified, a cross-dataset experiment is conducted to analyze the generalization performance. This experiment has been performed using our best-performing method: the M_E model combining lexicon-based features and FastText embeddings with a logistic regression classifier. The classifier has been trained with data from a specific dataset and evaluated with other datasets. Results are shown in Table 7, where the rows indicate the training dataset and the columns indicate the test dataset.

As expected, the differences between the datasets led to a relatively high performance drop. This drop is exceptionally high when training with data from the Reddit corpus and evaluating with data from interviews or Twitter. However, reasonably good results are obtained when training with data from interviews or Twitter and evaluating with data from Reddit. The greater length in data from Reddit compared to the rest of the data can explain this difference: longer posts could contain a richer vocabulary and more relevant features which are not present in posts from interviews or micro-blogging platforms. The classifier can draw conclusions from these features when training with Reddit posts that are irrelevant in the other datasets. However, the relevant features existing in short posts are also present in longer posts. Besides, among the NSE and TensiStrength datasets, even if the length of the posts is similar for both corpora, the differences in the nature of the data also led to a performance drop. This can be due to the expressiveness difference between data coming from Twitter and data coming from interviews.

The results obtained from the evaluation confirm that an approach for stress classification from text combining lexicon-based features with distributional representations can achieve good performance on different datasets (**RQ3**). The yielded F-score with this method varies between 77% and 86%, demonstrating its effectiveness.

7. Discussion

Previously, we presented three research questions that drove this work (Section 1). The first question (**RQ1**) was concerned with comparing different kinds of lexicon-based features in terms of stress detection performance. In this respect, we propose a feature framework based on multiple existing lexicons for identifying and categorizing features among four sets: affective, topic, social, and syntactic related. The obtained results suggest that affective features perform better in text classification of stress. However, including syntactic and topic-related features can help enhance classification performance. Regarding the different lexicons, LIWC and GI are those which led to better performance. These lexicons provide features of all kinds and are the only ones that provide syntactic features.

In the second question (**RQ2**), we focused on which is the best-performing method for detecting stress of the individuals in different corpora. In this regard, we presented and evaluated three different stress classification models: a lexicon-based features model, a distributional representation model, and an ensemble model combining lexicon-based features with distributional representations. The evaluation of these models in terms of F-score points out the combination of lexicon-based features with distributional representations as the best-performing approach. This approach yields significant performance, as shown by the experiments. The obtained F-scores surpass the 80% in the Dreaddit (86.04%) and NSE datasets (83.72%) and yield 77.50% in the TensiStrength dataset. The Friedman test confirms these results and demonstrates that they are statistically relevant. Besides, to analyze how different word embedding techniques impact the text classification performance, three popular word embedding techniques have been exploited: FastText, GloVe, and Word2Vec. The statistical results also identify FastText embeddings as the best-performing distributional representation technique, given its better performance in corpora with rare domain-specific words. The fine-tuning of the GloVe embeddings achieves comparable performances, but a greater amount of data would be needed to make it surpass the outcomes of FastText. In this line, Word2Vec without fine-tuning performs considerably worse than the other two. These results point out the importance of domain-specific word embeddings.

Finally, the third question (**RQ3**) pertained to studying the capability of a machine learning approach to classify stress in texts with good performance on multiple datasets. Previous studies had demonstrated the effectiveness of this method on specific datasets (Turcan & McKeown, 2019; Winata et al., 2018), but none of them had performed a cross-dataset evaluation using multiple public English corpora. Our approach has been evaluated on three different public English datasets obtaining good performance metrics on all of them. These results confirm the potential of textual information for detecting stress in several scenarios.

Also, the experiments have shown that the proposed model outperforms previous works in the field of stress detection from text. Our method presents two significant advances compared with previous works: the evaluation and validation with data from multiple corpora and the exhaustive analysis of feature extraction techniques. Some of the previous works have reached good results with the use of deep neural networks for predicting stress from Sina Weibo posts (Cao et al., 2021; Lin et al., 2016; Wang, Zhang, Cao and Feng, 2020). However, these models require large quantities of labeled data and often suffer from higher inference times (Yang, Shou, Gong, Lin, & Jiang, 2020). To the aim of our knowledge, this is the first study that proposes a method for stress detection from text and validates it on different public English datasets. Besides, our proposal exploits a wide variety of feature extraction methods, including multiple lexicons and three word embeddings techniques. In this manner, apart from carrying out an exhaustive analysis of which techniques achieve better performance, the proposed method improves the performance of previous work in all the existing datasets considered (Thelwall, 2017; Turcan & McKeown, 2019; Winata et al., 2018).

When comparing with other stress detection methods or approaches (Alberdi, Aztiria, & Basarab, 2016; Can, Arnrich et al., 2019; Greene et al., 2016), we can see that stress classification from text achieves comparable performances while entailing less complex or expensive scenarios. Furthermore, the proposed techniques can be integrated into existing software solutions, enabling stress detection in real-time and reducing complex hardware implementations' costs. For example, these methods could be integrated into existing software solutions such as a company communication platform or even social messaging apps. This can allow managers to easily monitor workers' stress levels and users to have an insight into their state.

8. Conclusion

This work proposes a method for detecting psychological stress from texts. The presented approach uses two different kinds of information sources: (i) affective, syntactic, topic, and social-related features and (ii) distributional representations. The first method proposes to benefit from existing lexicons for generating features that can be used for stress detection. A feature framework is presented to provide insight into which kinds of features and lexicons perform better. According to this, the features have been split into four different sets depending on their nature. A total of eleven different lexicons have been used for extracting these features. As for the second method, word vectors have been generated for extracting semantic information from the text. We have analyzed the use of three different word embedding techniques: Word2Vec, GloVe, and FastText. Finally, the work proposes a combination of these approaches aiming to improve the performance scores achieved by a classifier algorithm that learns from this unified set instead of only distributional representations or lexicon-based feature data. In order to conduct a comparative experimental study that enables the analysis and evaluation of all the models, three English public datasets from different sources and nature have been used. Also, an exhaustive analysis of the data contained in these datasets has been carried out.

The experiments show that the affective, syntactic, topic, and social features can obtain considerably high scores, achieving an F-Score of 82.57% in the Dreaddit dataset. Therefore, it can be reasonable to assume the convenience of these features when performing stress detection. Regarding word embeddings, it has been stated that they can achieve accurate representations of the analyzed text, allowing simple classifiers to reach elevated classification scores: 84.87% for the Dreaddit dataset. Hence, given the promising results obtained, we conclude that the application of this method can be further studied. Besides, the combination of distributional representations with lexicon-based features significantly improves the performance in all the cases reaching an F-score of 86.04% for the Dreaddit dataset. A statistical analysis has been conducted to empirically verify that combining information from varied lexicon-based features with distributional representations is suitable for improving stress classification performance. This test verifies the conclusions drawn from the experiments and points out the enhancement compared with previous works. Finally, a cross-dataset study has been carried out in order to analyze how the proposed method performs when using data from different corpora.

In conclusion, this paper presents a machine learning method for classifying stress text from different sources, such as social media, microblogging sites, and interviews. The proposed approach exploits several lexicons and three word embedding techniques. Besides, the evaluation presented provides a baseline for other researchers. According to the results of this work, we think that it is worth advancing in stress classification from text. The presented advancement in automatic detection of stress from text can be beneficial for detecting stress early, which is crucial for fastening its diagnostic and reducing its high growing impact. Our approach provides an excellent compromise between performance and required computational resources since it is based on simple features and outperforms more complex architectures such as BERT. Moreover, our approach can be easily interpreted to understand better how social, personal, and environmental factors influence stress and find suitable regulation methods. Nevertheless, there exist more complex pre-trained models specially designed for mental health, such as MentalBERT (Ji et al., 2021), which outperform our performance by 4% of F-score in the Dreaddit and TensiStrength dataset and yield a slightly worse performance (2% less of F-score) in the NSE dataset. This result encourages us to explore two research directions. Firstly, our current solution can be easily deployed in real environments with good performance without requiring high demanding computational resources. Additionally, we aim to explore how pre-trained models such as MentalBERT can be trained taking into account our feature framework. An additional possible line of future work would be to extend the domain of the proposed method to other languages or even to different paradigms, like depression detection. Besides, to deepen the use of word embeddings for studying semantic similarity between texts could be another possible future line to enhance the classification performance.

CRediT authorship contribution statement

Sergio Muñoz: Conceptualization, Methodology, Software, Validation, Data curation, Writing – original draft, Visualization. **Carlos A. Iglesias:** Conceptualization, Writing – review & editing, Visualization, Supervision, Project administration, Funding acquisition.

Acknowledgments

This work is supported by the Spanish Ministry of Science and Innovation under the R&D project COGNOS (PID2019-105484RB-I00). We would also to thank the authors of the used datasets for their work and for providing free access to them, Turcan and McKeown (2019), Zuo et al. (2012), Winata et al. (2018) and Thelwall (2017).

References

Affanni, A., Bernardini, R., Piras, A., Rinaldo, R., & Zontone, P. (2018). Driver's stress detection using Skin potential response signals. *Measurement*, *122*, 264–274.

Alberdi, A., Aztiria, A., & Basarab, A. (2016). Towards an automatic early stress recognition system for office environments based on multimodal measurements: A review. *Journal of Biomedical Informatics*, *59*, 49–75.

Alberdi, A., Aztiria, A., Basarab, A., & Cook, D. J. (2018). Using smart offices to predict occupational stress. *International Journal of Industrial Ergonomics*, *67*, 13–26.

Almatarneh, S., & Gamallo, P. (2018). A lexicon based method to search for extreme opinions. *PLoS One*, *13*(5), Article e0197816.

Andreou, E., Alexopoulos, E. C., Lionis, C., Varvogli, L., Gnardellis, C., Chrousos, G. P., et al. (2011). Perceived stress scale: reliability and validity study in Greece. *International Journal of Environmental Research and Public Health*, *8*(8), 3287–3298.

Araque, O., Corcuera-Platas, I., Sánchez-Rada, J. F., & Iglesias, C. A. (2017). Enhancing deep learning sentiment analysis with ensemble techniques in social applications. *Expert Systems with Applications*, *77*, 236–246.

Araque, O., & Iglesias, C. A. (2020). An approach for radicalization detection based on emotion signals and semantic similarity. *IEEE Access*, *8*, 17877–17891.

Bandhakavi, A., Wiratunga, N., Padmanabhan, D., & Massie, S. (2017). Lexicon based feature extraction for emotion text classification. *Pattern Recognition Letters*, *93*, 133–142.

Banerjee, S., & Shaikh, N. F. (2021). A survey on mental health monitoring system via social media data using deep learning framework. In *Techno-societal 2020* (pp. 879–887). Cham: Springer International Publishing.

Bradley, M. M., & Lang, P. J. (1999). *Affective norms for English words (ANEW): Instruction manual and affective ratings*: *Technical report, Technical report C-1*, the center for research in psychophysiology.

Burman, R., & Goswami, T. G. (2018). A systematic literature review of work stress. *International Journal of Management Studies*, *5*(3–9), 112–132.

Cambria, E., Liu, Q., Decherchi, S., Xing, F., & Kwok, K. (2022). SenticNet 7: a commonsense-based neurosymbolic AI framework for explainable sentiment analysis. *Proceedings of LREC 2022*.

Can, Y. S., Arnrich, B., & Ersoy, C. (2019). Stress detection in daily life scenarios using smart phones and wearable sensors: A survey. *Journal of Biomedical Informatics*, *92*, Article 103139.

Can, Y. S., Chalabianloo, N., Ekiz, D., & Ersoy, C. (2019). Continuous stress detection using wearable sensors in real life: Algorithmic programming contest case study. *Sensors*, *19*(8), 1849.

Cao, L., Zhang, H., Li, N., Wang, X., Ri, W., & Feng, L. (2021). Category-aware chronic stress detection on microblogs. *IEEE Journal of Biomedical and Health Informatics*, *26*(2), 852–864.

Chan, S. F., & La Greca, A. M. (2020). Perceived stress scale (PSS). In *Encyclopedia of behavioral medicine* (pp. 1646–1648). Cham: Springer International Publishing.

Chancellor, S., & De Choudhury, M. (2020). Methods in predictive techniques for mental health status on social media: a critical review. *NPJ Digital Medicine*, *3*(1), 1–11.

Cox, T., & Griffiths, A. (1995a). The nature and measurement of work stress: theory and practice. In *The evaluation of human work: a practical ergonomics methodology*. London: Taylor & Francis.

Cox, T., & Griffiths, A. (1995b). Work-related stress: nature and assessment. In *IEE colloquium on stress and mistake-making in the operational workplace* (pp. 1/1–1/4).

Crossley, S. A., Kyle, K., & McNamara, D. S. (2017). Sentiment analysis and social cognition engine (SEANCE): An automatic tool for sentiment, social cognition, and social-order analysis. *Behavior Research Methods*, *49*(3), 803–821.

de Santos Sierra, A., Ávila, C. S., Casanova, J. G., & del Pozo, G. B. (2011). A stress-detection system based on physiological signals and fuzzy logic. *IEEE Transactions on Industrial Electronics*, *58*(10), 4857–4865.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, *7*(Jan), 1–30.

Dewe, P. J., O'Driscoll, M. P., & Cooper, C. L. (2012). Theories of psychological stress at work. In *Handbook of occupational health and wellness* (pp. 23–38). Boston, MA: Springer US.

Dingwall, N., & Potts, C. (2018). Mittens: an extension of glove for learning domain-specialized representations. In *NAACL-HLT (2)* (pp. 212–217).

Fast, E., Chen, B., & Bernstein, M. S. (2016). Empath: Understanding topic signals in large-scale text. In *Proceedings of the 2016 CHI conference on human factors in computing systems* (pp. 4647–4657).

Ferdous, R., Osmani, V., & Mayora, O. (2015). Smartphone app usage as a predictor of perceived stress levels at workplace. In *2015 9th international conference on pervasive computing technologies for healthcare (PervasiveHealth)* (pp. 225–228).

French, J. R., Caplan, R. D., & Van Harrison, R. (1982). *The mechanisms of job stress and strain, Vol. 7*. New York: Chichester, J. Wiley.

Fu, X., Yang, J., Li, J., Fang, M., & Wang, H. (2018). Lexicon-enhanced LSTM with attention for general sentiment analysis. *IEEE Access*, *6*, 71884–71891.

Gao, H., Yüce, A., & Thiran, J.-P. (2014). Detecting emotional stress from facial expressions for driving safety. In *2014 IEEE international conference on image processing (ICIP)* (pp. 5961–5965). IEEE.

Giannakakis, G., Pediaditis, M., Manousos, D., Kazantzaki, E., Chiarugi, F., Simos, P. G., et al. (2017). Stress and anxiety detection using facial cues from videos. *Biomedical Signal Processing and Control*, *31*, 89–101.

Giatsoglou, M., Vozalis, M. G., Diamantaras, K., Vakali, A., Sarigiannidis, G., & Chatzisavvas, K. C. (2017). Sentiment analysis leveraging emotions and word embeddings. *Expert Systems with Applications*, *69*, 214–224.

Gilbert, E., & Hutto, C. J. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. *81*, In *Eighth international conference on weblogs and social media (ICWSM-14)* (p. 82).

Giuntini, F. T., Cazzolato, M. T., dos Reis, M. d. J. D., Campbell, A. T., Traina, A. J., & Ueyama, J. (2020). A review on recognizing depression in social networks: challenges and opportunities. *Journal of Ambient Intelligence and Humanized Computing*, *11*(11), 4713–4729.

Glanz, K., Rimer, B. K., & Viswanath, K. (2008). *Health behavior and health education: theory, research, and practice*. San Francisco, CA: John Wiley & Sons.

Greene, S., Thapliyal, H., & Caban-Holt, A. (2016). A survey of affective computing for stress detection: Evaluating technologies in stress detection for better health. *IEEE Consumer Electronics Magazine*, *5*(4), 44–56.

Hansen, J. H. L., & Patil, S. (2007). Speech under stress: Analysis, modeling and recognition. In *Speaker classification i: fundamentals, features, and methods* (pp. 108–137). Springer Berlin Heidelberg.

Hassard, J., Teoh, K., Cox, T., Cosmar, M., Gründler, R., Flemming, D., et al. (2014). *Calculating the cost of work-related stress and psychosocial risks*: *Technical report*, European Agency for Safety and Health at Work.

Holroyd, K. A., & Lazarus, R. S. (1982). Stress, coping and somatic adaptation. *Handbook of stress: theoretical and clinical aspects*, 21–35.

Iman, R. L., & Davenport, J. M. (1980). Approximations of the critical region of the Friedman statistic. *Communications in Statistics. Theory and Methods*, *9*(6), 571–595.

Ji, S., Zhang, T., Ansari, L., Fu, J., Tiwari, P., & Cambria, E. (2021). MentalBERT: Publicly available pretrained language models for mental healthcare. arXiv preprint arXiv:2110.15621.

Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H., & Mikolov, T. (2016). Fasttext. zip: Compressing text classification models. arXiv preprint arXiv:1612.03651.

Kassymova, G., Tokar, O., Tashcheva, A., Gridneva, S., Bazhenova, N., Shpakovskaya, E., et al. (2019). Impact of stress on creative human resources and psychological counseling in crises. *International Journal of Education and Information Technologies*, *13*(1), 26–32.

Kenton, J. D. M.-W. C., & Toutanova, L. K. (2019). BERT: PRe-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT* (pp. 4171–4186).

Kessler, J. S. (2017). Scattertext: a browser-based tool for visualizing how corpora differ. In *Proceedings of ACL-2017 system demonstrations* (pp. 85–90). Vancouver, Canada: Association for Computational Linguistics.

Khoo, C. S., & Johnkhan, S. B. (2018). Lexicon-based sentiment analysis: Comparative evaluation of six sentiment lexicons. *Journal of Information Science*, *44*(4), 491–511.

Kim, J. J., & Diamond, D. M. (2002). The stressed hippocampus, synaptic plasticity and lost memories. *Nature Reviews Neuroscience*, *3*(6), 453–462.

Lasswell, H. D., & Namenwirth, J. Z. (1969). *The Lasswell value dictionary*. New Haven.

Lewin, K. (1936). A dynamic theory of personality. *Journal of Heredity*, *27*(11), 441–442.

Lin, H., Jia, J., Guo, Q., Xue, Y., Li, Q., Huang, J., et al. (2014). User-level psychological stress detection from social media using deep neural network. In *Proceedings of the 22nd ACM international conference on multimedia* (pp. 507–516).

Lin, H., Jia, J., Nie, L., Shen, G., & Chua, T.-S. (2016). What does social media say about your stress?. In *IJCAI* (pp. 3775–3781).

Liu, Y.-d., Zeng, H.-q., Li, R.-l., & Hu, Y.-f. (2004). Polarity text filtering based on semantic analysis. *Journal-China Institute of Communications*, *25*(7), 78–85.

Martin, L., & Pu, P. (2014). Prediction of helpful reviews using emotions extraction. In *Proceedings of the AAAI conference on artificial intelligence, Vol. 28* (pp. 1551–1557).

Maxhuni, A., Hernandez-Leal, P., Morales, E. F., Sucar, L. E., Osmani, V., & Mayora, O. (2021). Unobtrusive stress assessment using smartphones. *IEEE Transactions on Mobile Computing*, *20*(6), 2313–2325.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111–3119).

Mohammad, S., & Turney, P. (2010). Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text* (pp. 26–34).

Murray, H. A. (1938). *Explorations in personality*. Oxford: Oxford Univ. Press.

Novais, P., & Carneiro, D. (2016). The role of non-intrusive approaches in the development of people-aware systems. *Progress in Artificial Intelligence*, *5*(3), 215–220.

Osman, A., Wong, J. L., Bagge, C. L., Freedenthal, S., Gutierrez, P. M., & Lozano, G. (2012). The depression anxiety stress Scales—21 (DASS-21): further examination of dimensions, scale reliability, and correlates. *Journal of Clinical Psychology*, *68*(12), 1322–1338.

Pang, J., Rao, Y., Xie, H., Wang, X., Wang, F. L., Wong, T.-L., et al. (2019). Fast supervised topic models for short text emotion detection. *IEEE Transactions on Cybernetics*, *51*(2), 815–828.

Panicker, S. S., & Gayathri, P. (2019). A survey of machine learning techniques in physiology based mental stress detection systems. *Biocybernetics and Biomedical Engineering*, *39*(2), 444–469.

Parent-Thirion, A., Vermeylen, G., van Houten, G., Lyly-Yrjänäinen, M., Biletta, I., & Cabrita, J. (2012). *Fifth European working conditions survey*: *Technical report*, European Foundation for the Improvement of Living and Working Conditions.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Pennebaker, J. W., Francis, M. E., & Booth, R. J. (2001). *Linguistic inquiry and word count: LIWC 2001, Vol. 71* (p. 2001). Mahway: Lawrence Erlbaum Associates.

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

for Safety, E. A., & at Work, H. (2013). *European opinion poll on occupational safety and health*: *Technical report*, European Agency for Safety and Health at Work.

Saute, S., Murphy, L., Colligan, M., Swanson, N., Hurrell, J., Scharf, F., et al. (1999). *Stress at work*: *Technical Report*, National Institute for Occupational Safety and Health.

Selye, H. (1956). *The stress of life*. New York: McGraw-Hill.

Skaik, R., & Inkpen, D. (2020). Using social media for mental health surveillance: A review. *ACM Computing Surveys*, *53*(6), 1–31.

Society, A. P. (2015). Stress and wellbeing. How Australians are coping with life.

Statista, I. (2018). Number of social media users worldwide from 2010 to 2021 (in billions). *Statista*.

Stone, P. J., Dunphy, D. C., & Smith, M. S. (1966). *The general inquirer: a computer approach to content analysis*. MIT Press.

Su, C., Xu, Z., Pathak, J., & Wang, F. (2020). Deep learning in mental health outcome research: a scoping review. *Translational Psychiatry*, *10*(1), 1–26.

The American Institute of Stress (2013). Definition of stress. https://www.stress.org/what-is-stress/ Online; accessed 17 September 2020.

Thelwall, M. (2017). Tensistrength: Stress and relaxation magnitude detection for social media texts. *Information Processing & Management*, *53*(1), 106–121.

Thieme, A., Belgrave, D., & Doherty, G. (2020). Machine learning in mental health: A systematic review of the hci literature to support the development of effective and implementable ml systems. *ACM Transactions on Computer-Human Interaction*, *27*(5), 1–53.

Tomba, K., Dumoulin, J., Mugellini, E., Abou Khaled, O., & Hawila, S. (2018). Stress detection through speech analysis. In *ICETE (1)* (pp. 560–564).

Turcan, E., & McKeown, K. (2019). Dreaddit: A Reddit Dataset for Stress Analysis in Social Media. In *Proceedings of the tenth international workshop on health text mining and information analysis (LOUHI 2019)* (pp. 97–107).

Vanitha, V., & Krishnan, P. (2017). Real time stress detection system based on EEG signals. *Biomedical Research — Tokyo*, 271–275.

Vizer, L. M., Zhou, L., & Sears, A. (2009). Automated stress detection using keystroke and linguistic features: An exploratory study. *International Journal of Human-Computer Studies*, *67*(10), 870–886.

Wang, W., Hernandez, I., Newman, D. A., He, J., & Bian, J. (2016). Twitter analysis: Studying US weekly trends in work stress and emotion. *Applied Psychology*, *65*(2), 355–378.

Wang, X., Zhang, H., Cao, L., & Feng, L. (2020). Leverage social media for personalized stress detection. In *Proceedings of the 28th ACM international conference on multimedia* (pp. 2710–2718).

Wang, S., Zhou, W., & Jiang, C. (2020). A survey of word embeddings based on deep learning. *Computing*, *102*(3), 717–740.

Widanti, N., Sumanto, B., Rosa, P., & Miftahudin, M. F. (2015). Stress level detection using heart rate, blood pressure, and GSR and stress therapy by utilizing infrared. In *2015 international conference on industrial instrumentation and control (ICIC)* (pp. 275–279). Ieee.

Winata, G. I., Kampman, O. P., & Fung, P. (2018). Attention-based LSTM for psychological stress detection from spoken language using distant supervision. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)* (pp. 6204–6208).

Xiong, B., Skitmore, M., & Xia, B. (2015). Exploring and validating the internal dimensions of occupational stress: evidence from construction cost estimators in China. *Construction Management and Economics*, *33*(5–6), 495–507.

Yang, Z., Shou, L., Gong, M., Lin, W., & Jiang, D. (2020). Model compression with two-stage multi-teacher knowledge distillation for web question answering system. In *Proceedings of the 13th international conference on web search and data mining* (pp. 690–698).

Yue, L., Chen, W., Li, X., Zuo, W., & Yin, M. (2019). A survey of sentiment analysis in social media. *Knowledge and Information Systems*, *60*(2), 617–663.

Zuo, X., Lin, L., & Fung, P. (2012). A multilingual database of natural stress emotion. In *Proceedings of the eight international conference on language resources and evaluation (LREC'12)* (pp. 1174–1178).

Sergio Muñoz received the graduate and master's degrees in Telecommunications Engineering from the Universidad Politécnica de Madrid (UPM), Spain, in 2016 and 2017, respectively, where he is currently pursuing the Ph.D. degree. He is currently a Teaching Assistant with the Technical University of Madrid. His research interests include ambient intelligence and agent-based simulation. The main topic of his thesis is the adaptation of smart environments to users' emotions, to enhance well-being and performance.

Carlos A. Iglesias received the telecommunications engineering degree and the Ph.D. degree in telecommunications from the Universidad Politécnica de Madrid (UPM), Spain, in 1993 and 1998, respectively. He is currently a University Professor with the Telecommunications Engineering School, UPM, where he has been leading the Intelligent Systems Group, since 2014. He has been the Principal Investigator on numerous research grants and contracts in the field of advanced social and the IoT systems, funded by the regional, national, and European bodies. His primary research interests include social computing, multiagent systems, information retrieval, sentiment and emotion analysis, linked data, and web engineering.