Received: 22 March 2021 | Revised: 1 June 2021 | Accepted: 26 June 2021

DOI: 10.1002/int.22575

RESEARCH ARTICLE

WILEY

Improving the emotion-based classification by exploiting the fuzzy entropy in FCM clustering

Barbara Cardone[1] | **Ferdinando Di Martino**[1,2] | **Sabrina Senatore**[3]

[1]Dipartimento di Architettura, Università degli Studi di Napoli Federico II, Napoli, Italy

[2]Centro Interdipartimentale di Ricerca “Alberto Calza Bini”, Università degli Studi di Napoli Federico II, Napoli, Italy

[3]Dipartimento di Ingegneria dell'Informazione ed Elettrica e Matematica applicata, Università degli Studi di Salerno, Fisciano (Salerno), Italy

Correspondence
Ferdinando Di Martino, Dipartimento di Architettura, Università degli Studi di Napoli Federico II, Via Toledo 402, 80134 Napoli, Italy.
Email: fdimarti@unina.it

Abstract

Emotion detection in the natural language text has drawn the attention of several scientific communities as well as commercial/marketing companies: analyzing human feelings expressed in the opinions and feedback of web users helps understand general moods and support market strategies for product advertising and market predictions. This paper proposes a framework for emotion-based classification from social streams, such as Twitter, according to Plutchik's wheel of emotions. An entropy-based weighted version of the fuzzy c-means (FCM) clustering algorithm, called EwFCM, to classify the data collected from streams has been proposed, improved by a fuzzy entropy method for the FCM center cluster initialization. Experimental results show that the proposed framework provides high accuracy in the classification of tweets according to Plutchik's primary emotions; moreover, the framework also allows the detection of secondary emotions, which, as defined by Plutchik, are the combination of the primary emotions. Finally, a comparative analysis with a similar fuzzy clustering-based approach for emotion classification shows that EwFCM converges more quickly with better performance in terms of accuracy, precision, and runtime. Finally, a straightforward mapping between the computed clusters and the emotion-based classes allows the assessment of the classification quality, reporting coherent and consistent results.

KEYWORDS
emotion extraction, fuzzy clustering, fuzzy entropy, Plutchik's wheel of emotions, sentiment analysis

1 | INTRODUCTION

The recent advent of unexpected worldwide pandemic situations had unquestionably pushed to social media adoption and shifted online user behavior. Millions of people record and share their daily lives in social applications and, as a consequent result, a large number of social network data describe emergencies, incidents, disasters, and some other hot events, including people's opinions and sentiments. However, in the last 20 years, the role of end-users in the Social Web has completely changed: from consumers of accessible web resources to producers of web content accessible on the global network by sharing posts, tweets, and messages. The volunteer contribution is a valuable resource that creates innovative content and helps other users to make decisions, express opinions, suggestions, and advice. They may unpredictably influence decision-making tasks ranging from buying an item simply to social and political events.

Emotion detection in the written text has drawn the attention of commercial/marketing companies: analyzing human feelings expressed in the web-user opinions and feedbacks contribute to understanding the general moods and supporting market strategies for product advertising and market predictions. The influence of human moods in Social Web applications has also attracted more researchers' attention in the field of natural language processing. In early studies, research was focused on determining the polarity of the given word, sentence, or in general a text for positive or negative orientation.[1] But the sentiment polarity does not provide the several emotional shades expressed in human opinions. The emotion detection from natural language needs enhanced computational linguistics methods aimed at text processing to actually grasp the effective sentiments and emotions behind the words. Natural language processing helps distill knowledge and capture feelings from a huge amount of textual data disseminated on the Web, although the natural language is often very difficult to analyze especially when emotions are not explicitly associated with words and the emotional concept is expressed figuratively. Classifying the emotions of humans has interested several researchers, especially psychologists, such as Plutchik[2] that identified eight primary emotions, each one has a polar opposite. Several studies have been developed on the several nuances of emotion and how emotions contrast with each other. Marvin Minsky's conception of emotions, for instance, considers each emotional state as a result of turning some resources (which compound our mind) on and turning off some others.[3]

At the same time, Machine Learning (ML) techniques have been exploited and integrated to extract the meaning and the emotions from the text. The synergy between ML and NLP methods presents promising solutions by low-dimensional continuous representations of sentences and words[4] for predicting emotional categories,[5] even though some performance degradations are registered especially when the topic is not centered on common sense, but domain-oriented.[6]

In Reference [7], a straightforward framework for classifying emotions from message streams is proposed. The rationale behind the approach is to exploit emotional terms as features. Each emotional term is a natural language word expressing an emotion. Emotional terms with a similar meaning (synonyms) represent the same emotion, that is, they constitute the same emotional category. In Reference [7], an extension of the FCM algorithm, called extended fuzzy c-means (EFCM),[8] is proposed to classify emotions from documents extracted by data streams.

This paper proposes Fuzzy Entropy Light Emotion Classification (FELEC), a novel framework for the classification of message streams, achieved by defining categories representing Plutchik's emotions. It is based on the work presented in Reference [7] but exploits a weighted variation of FCM, called entropy weighted fuzzy c-means (EwFCM),[9] in substitution of the EFCM algorithm proposed in the previous approach. EwFCM uses the Fuzzy Entropy measures[10] to optimize the selection of initial cluster centers, which, due to the random cluster initialization, represents a known drawback of the traditional FCM method.

Empirical evidence shows that the introduction of fuzzy entropy as a variation of the EFCM algorithm enhances the performance of the proposed classifier in terms of accuracy and runtime, compared with the previous approach.[7]

The remainder of the paper is organized as follows: Section 2 provides recent literature on Sentiment Analysis (SA) and Emotion Extraction, with a focus on Plutchik's theory of emotions; then, Section 3 introduces some preliminaries and the theoretical background on the EwFCM algorithm. The framework is described in Section 4. Comparative experiments are shown in Section 5. Final considerations close the paper.

2 | RELATED WORKS

Over the past decade, the processing of emotions in natural language text has attracted the interest of researchers and entrepreneurs for statistics and market analysis.

Several approaches focusing on NLP techniques have been developed to detect and classify texts extracted from social environments to analyze the moods of users.

The early research was oriented to investigate the polarity of the given text for positive or negative orientation.[1] SA has been extensively studied at various levels of abstraction: document, sentence, and aspect to capture the sentiment polarity in the phrase, document, and entity (aspect), respectively. Deep syntactic analysis for the phrase-level SA allows extracting more different sentiments from a document,[11] identifying the sentiment locally and with more reliability than the global document sentiment.[8] Moreover, knowing what particular aspects of the entity/item the user is commenting on and also whether he likes them or not can be important information to capture in the text analysis. Also, ML methods have been employed to accomplish text-categorization techniques to subject-driven parts of the document, using techniques for finding minimum cuts in graphs.[12]

Despite the SA progress in capturing sentiments in the written language, enhanced natural language processing techniques to extract the finest-grained feelings become crucial to capture the wide range of emotions.[7] Distinguishing the sentiment polarity in fact may not be enough, compared with the different nuances of emotions describing the human feeling and expressed in text from the Social Web. A careful understanding of the emotions and opinions from social networks, such as Twitter and Facebook, requires the integration of expertise and knowledge from different fields, such as linguistic, psychology, cognitive science, sociology, and ethics.

For this purpose, Sentic Computing is a multidisciplinary domain that bridges computer and social sciences to capture opinions and sentiments over the Web.[13] Emotion-oriented ontologies exploit social sciences for opinion and sentiment interpretation and inspect collective emotions affecting human behavior.[14]

Several approaches have been developed based on psychological, social, and linguistic studies to recognize not just sentiment polarity but emotions from the text.

In Reference [15], an approach for emotion extraction from natural language in news headlines is presented; it is mainly based on the construction of a large data set, annotated for six basic emotions: anger, disgust, fear, joy, sadness, and surprise. A class sequential rules[16] classify the text into seven different emotion types. Rule-based approaches, probabilistic models, and ML methods (e.g., decision tree, support vector machine (SVM), and Naïve Bayes classifier) are widely employed for emotion extraction.[9,17]

ML techniques are widely applied for SA[17,18]; in Reference [19], Twitter messages are classified depending on critical situations over specific events, thanks to a combination of manually annotated and automatically extracted linguistic features; tweets are also analyzed to capture sentiments from product-related brands.[20]

Noteworthy is SentiWordNet,[21] a lexical resource designed for supporting sentiment classification and opinion mining applications, such as in Reference [22], where it supports document-level sentiment classification of Movie reviews and Blog posts. In Reference [23], the authors propose an unsupervised tweet opinion retrieval method based on a SVM algorithm; they show that the approach is more effective than other supervised search and classification methods for information retrieval. In Reference [24], a hybrid SA framework based on hadoop distributed file system MapReduce and the Gradient Boosted Decision Tree classification method is applied to classify sentiments in the tweets. An automated neural network-based SA model is proposed in Reference [25] to explore Twitter data. Also, a deep learning neural classification for polarity classification is proposed in Reference [26], based on a Sentiment Treebank; while in Reference [27] a word embedding method has been exploited for deep convolution neural networks. In Reference [11], an SA model based on the Naïve Bayes, Maximum Entropy, and Negation algorithms is developed to classify tweets.

Several hybrid approaches proposed in the literature integrate ML and linguistic models[16,28] for social stream classification. They often suffer from a high computational effort that significantly can affect the performance of the algorithm in terms of memory consumption and execution times.

2.1 | Plutchik's main emotions

Emotion extraction and classification are essentially driven by emotion models that generally set the basic emotions that can be identified.

The most used theories[2,29–31] converge on a set of six predominant emotions; other studies proposed different sets of basic emotions, based on a psychological brain–emotion correspondence[32–34] and also depending on different cultures and educations.[33] In particular, Ekman's model is based on six basics emotions: sadness, happiness, anger, fear, disgust, and surprise. These emotions are coded by facial expressions and neurobiological.[29] Plutchik's multidimensional model of emotions[2] is based on the psycho-evolutionary theory of emotions, which established the foundation for conceptualizing the domain of emotion (primary and secondary). Emotional processes are part of chains of events, perceptions, moods, and actions whose evolution guarantees to maintain behavioral homeostasis. According to Plutchik, humans acquire and gain experience in their lives with eight different primary emotions, which are divided into four distinct pairs of opposite feelings. Plutchik puts the eight emotions in a circular model, in which the opposite emotions are placed across from one another, while adjacent emotions can fuse, generating novel blended emotions. Figure 1 shows the so-called Plutchik's wheel. In detail, the model divided emotions into eight main categories. Half of these emotions are positive emotions, and the other half are negative ones: joy–sadness, trust–disgust, anticipation–surprise, and anger–fear (see Figure 2).

For Plutchik, these human primary emotions are biologically primitive and culturally independent; they have evolved to allow species to survive.

Moreover, each emotion is divided into subgroups, treating them as secondary and tertiary emotions in the wheel-shaped mechanism, associating intensities and polarities with them.[10,35,36] Precisely the intensity of an emotion is high going towards the center of the wheel and it decreases as the distance from the center increases. Plutchik's wheel of emotions summarizes simply the emotions that a human can feel, highlighting that they are the result of "mixtures" and events of various kinds.

Like colors, primary emotions can be expressed at different intensities and can blend to form new emotions. Blending primary emotions can generate secondary emotions, in the outside circle of the wheel; Table 1 shows the several emotions as a blending of two adjacent primary ones. Blended emotions can be remorse, as a fusion of sadness and disgust, or love as a combination of joy and trust. They, in turn, can give rise to more complex emotions.

Plutchik emphasizes the role of primary emotions that govern the instincts of humans and animals and at the same time, the secondary or blended emotions are acquired by socializing response to stimuli of some psychological conditions or triggered by purely cognitive events.

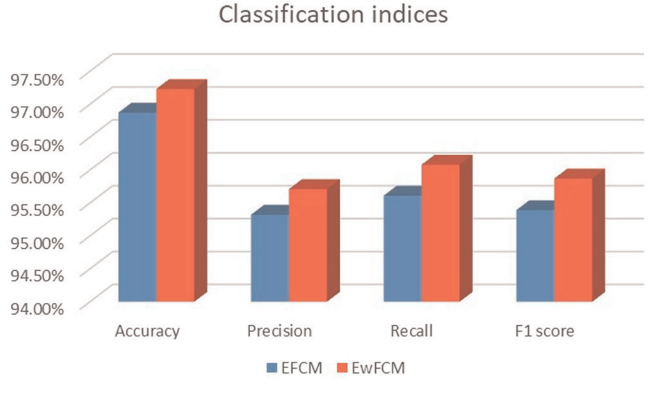

FIGURE 1 Comparative performances EFCM versus EwFCM: mean value of accuracy, precision, recall, and F1-score measures considering the 52 data sets from UCI Machine Learning repository. EFCM, extended fuzzy c-means; EwFCM, entropy weighted fuzzy c-means [Color figure can be viewed at wileyonlinelibrary.com]

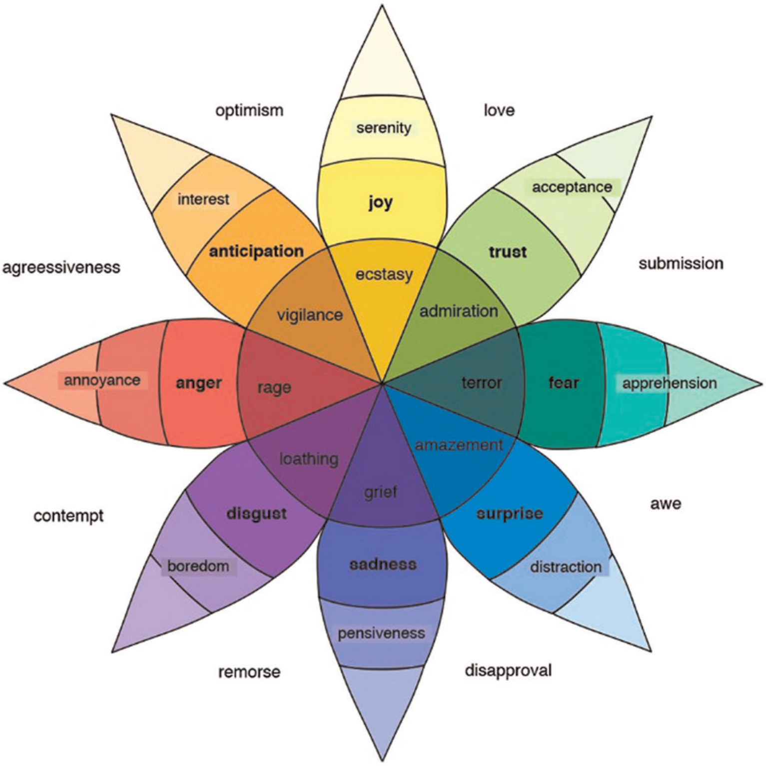

FIGURE 2 Plutchik's wheel of emotions [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 1 Secondary emotional categories as a blending of primary ones

	Secondary emotions (blended)	Basic emotions	Basic emotions
Positive/pleasant	Awe	Fear	Surprise
	Love	Joy	Trust
	Optimism	Anticipation	Joy
	Content	Anger	Disgust
Negative/unpleasant	Aggression	Anger	Anticipation
	Disapproval	Sadness	Surprise
	Remorse	Disgust	Sadness
	Submission	Fear	Trust

3 | FORMAL BACKGROUND

3.1 | From EFCM to EwFCM

The approach presented in Reference [7] achieves an emotion-based classification of social data streams. It uses an extended version of the well-known FCM algorithm, called EFCM.[8]

EFCM builds hyperspheric cluster prototypes and iteratively determines the optimal number of clusters in the running phase. This algorithm allows overcoming some known FCM drawbacks, such as the a priori setting of the number of clusters: EFCM finds the optimal number of clusters by applying a heuristic process for merging similar clusters; moreover, it improves the performance in terms of robustness to noise and outliers, and independence on the initialization.

In addition to the a priori selection of the number of clusters, the additional flaw of FCM is the random assignment of cluster centers which can lead to local minima and increase the number of iterations. To avoid convergence to local minima, EFCM uses cluster prototypes consisting of hyperspheres in feature space and minimizes the objective function, where the distance of each data point from the cluster is determined by the distance to the hypersphere, rather than the cluster center.

Although EFCM is computationally slower than FCM, one of its advantages is the determination of the optimal number of clusters performed at runtime; FCM instead needs to know in advance the number of clusters to run. When required, the validity index allows calculating a priori the optimal number of clusters, also measuring the compactness of clusters and the separability between clusters.

However, in the current approach to emotion-based document classification, the optimal number of clusters is set equal to the number of defined emotion categories.

An additional advantage of EFCM over FCM is its robustness to the initial selection of cluster centers. Let us notice that EwFCM presents the same advantage by determining the cluster centers by measuring the fuzzy entropy of the initial fuzzy clusters; the initial clusters are such that they have low fuzziness, this ensures an optimal selection of the initial cluster centers and allows reducing the number of iterations and avoiding convergence to local minima. In a nutshell:

- EwFCM is computationally faster than EFCM; it has the same computational complexity as FCM and in particular, in this approach, it does not need to determine the optimal number of clusters since it is assigned equal to the number of emotional categories.
- By exploiting the fuzziness of clustering through the use of De Luca and Termini's fuzzy entropy,[10] EwFCM is robust with respect to the choice of initial cluster centers and can achieve convergence in a minimum number of iterations.

3.2 | The EwFCM algorithm

The EwFCM algorithm[9] applies a variation of the weighted FCM algorithm[37] (wFCM) to find the initial values of the cluster centers; the weights assigned to the data points are calculated using the De Luca and Termini fuzzy entropy measure in Reference [10]. In the wFCM algorithm the weight is assigned to each data point to measure its influence on the formation of the final clusters.

Formally, let $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\} \subset R^n$ be a set of N data points in the n-dimensional space R^n with $\mathbf{x}_k = (x_{k1}, ..., x_{kn})$ and $\mathbf{V} = \{\mathbf{v}_1, ..., \mathbf{v}_C\} \subset R^n$ is the set of centers of the C clusters. Let $\mathbf{U}$ be the $C \times N$ partition matrix where u_{hk} is the membership degree of the kth data point $\mathbf{x_k}$ to the hth cluster $\mathbf{v}_k$.

The objective function in wFCM is given by

$$J_w(\boldsymbol{U}, \boldsymbol{V}) = \sum_{h=1}^{C} \sum_{k=1}^{N} w_k u_{hk}^m d_{hk}^2 = \sum_{h=1}^{C} \sum_{k=1}^{N} w_k u_{hk}^m \, || \boldsymbol{x}_k - \boldsymbol{v}_h \, ||^2, \tag{1}$$

where m is the fuzzifier parameter.

The solutions for the centers of the clusters v_h and the membership degrees components u_{hk} are given, respectively, by

$$\boldsymbol{v}_h = \frac{\sum_{k=1}^{N} w_k u_{hk}^m \boldsymbol{x_k}}{\sum_{k=1}^{N} w_k u_{hk}^m}, \quad h \in \{1, \ldots, C\} \tag{2}$$

and

$$u_{hk} = \frac{1}{\sum_{l=1}^{C} \left(\frac{d_{hk}}{d_{lk}}\right)^{\frac{2}{m-1}}}, \quad h \in \{1, \ldots, C\}, \quad k \in \{1, \ldots, N\}, \tag{3}$$

where d_{hk} is the Euclidean distance between the hth cluster and the kth data point.

An iterative process is applied to find the partition matrix and the cluster centers: initially, the membership degrees are assigned randomly; in each iteration the cluster centers are calculated by (2) and the weight w_k of each data point $\mathbf{x_k}$ is calculated by using a weight function $w(\mathbf{x})$; then the membership degree components are calculated by (3). The iterative process stops at the tth iteration when $|\boldsymbol{U}^{(t)} - \boldsymbol{U}^{(t-1)}| < \varepsilon$ where $\varepsilon > 0$ is a parameter assigned a priori to stop the iteration and $|\boldsymbol{U}^{(t)} - \boldsymbol{U}^{(t-1)}| = \max_{\substack{h=1, \ldots, C \\ k=1, \ldots, N}} \{|u_{hk}^{(t)} - u_{hk}^{(t-1)}|\}$.

Algorithm 1 shows the pseudocode of wFCM algorithm.

Algorithm 1. wFCM algorithm

1. *Set* m, ε, C
2. *Initialize* randomly the partition matrix $\mathbf{U}$
3. **Repeat**
4. *Calculate* w_k $k = 1, \ldots, N$
5. *Calculate* $\boldsymbol{v}_h$ $h = 1, \ldots, C$ by using (2)
6. *Calculate* u_{hk} $h = 1, \ldots, C$ $k = 1, \ldots, N$ by using (3)
7. **Until** $|\boldsymbol{U}^{(t)} - \boldsymbol{U}^{(t-1)}| > \varepsilon$

To optimize the initialization of cluster centers in Reference [9], a variation of wFCM called EwFCM is proposed, in which De Luca and Termini's fuzzy entropy measure is applied to minimize clustering fuzziness.

In Reference [9], the mean fuzziness $H(\mathbf{x}_k)$ assigned to the kth object is given by

$$H(\boldsymbol{x}_k) = \frac{1}{C} \sum_{h=1}^{C} h(u_{hk}), \tag{4}$$

where $h(u_{hk})$ is the De Luca and Termini fuzzy entropy function:

$$h(u_{hk}) = \begin{cases} 0 & \text{if } u_{hk} = 0, \\ -u_{hk} \ln(u_{hk}) - (1 - u_{hk}) \ln(1 - u_{hk}) & \text{if } 0 < u_{hk} < 1, \\ 0 & \text{if } u_{hk} = 1. \end{cases} \tag{5}$$

The mean fuzziness of clustering $\bar{\mathrm{H}}$ is given by

$$\bar{H} = \frac{1}{N}\sum_{k=1}^{N} H(\boldsymbol{x_k}). \tag{6}$$

$\bar{H}$ takes values in $[0, h_{\max}]$; its value is 0 if the clusters are crisp set and any object belongs to only a cluster with membership degree equal to 1 (i.e., null fuzziness), and is $h_{\max} \leq 1$ if the membership degree of each object to each cluster is equal to $1/C$ (maximum fuzziness).

EwFCM calculate the following values for the weight of the kth data point:

$$w_k = 1 - H(\boldsymbol{x_k}). \tag{7}$$

After assigning randomly the cluster centers, the wFCM algorithm with the weights in (7) is applied in an iterative process; the initial cluster centers are found when the absolute difference between the mean fuzziness (6) calculated in the current cycle and the one calculated in the previous cycle is below a fixed threshold η or when the number of iterations is equal to a maximal number of iterations $i_{\max}$.

After calculating the initial cluster centers, EwFCM runs FCM using these cluster centers as initial cluster centers.

The parameters to set a priori in EwFCM are the fuzzifier m, the threshold η, the error threshold ε, the maximum number of iterations $i_{\max}$ and the number of cluster C. The pseudocode of EwFCM algorithm is given in Algorithm 2.

Algorithm 2. EwFCM algorithm

1. *Set* m, η, ε, C, $i_{\max}$
2. *Initialize* randomly the partition matrix **U**
3. $n_{\text{iter}} := 1$//number of iterations
4. **Repeat**
5. **For** $k = 1$ to N
6. *Calculate* w_k with $k = 1, \ldots, N$ using (7)
7. **For** $\mathrm{h} = 1$ to C
8. *Calculate* $\mathbf{v_h}$ with $h = 1, \ldots, C$ by using (2)
9. *Calculate* u_{hk} with $h = 1, \ldots, C$ $k = 1, \ldots, N$ by using (3)
10. **Next** h
11. **Next** k
12. **Until** $|\bar{H}^{(t)} - \bar{H}^{(t-1)}| > \eta$ OR $n_{\text{iter}} = i_{\max}$
13. **Call** FCM using the initial cluster center $\mathbf{v}_1, \ldots, \mathbf{v}_C$.

4 | THE FELEC FRAMEWORK

Figure 3 shows the logical overview of the proposed framework. It provides a general, high-level description of the data flow across the main designed tasks, taking as input natural language resources, and processing them to get as a final output an emotion-based classification of the textual collection.

The framework is designed to process a wide range of Web resources (posts, tweets, web pages, etc.) enclosing textual content; an initial scraping activity allows selection of only appropriate textual information, by discard useless markups from the textual content. Tweets are generally parsed to discard meaningless words (stop word, slangs, etc.); then, since tweets are

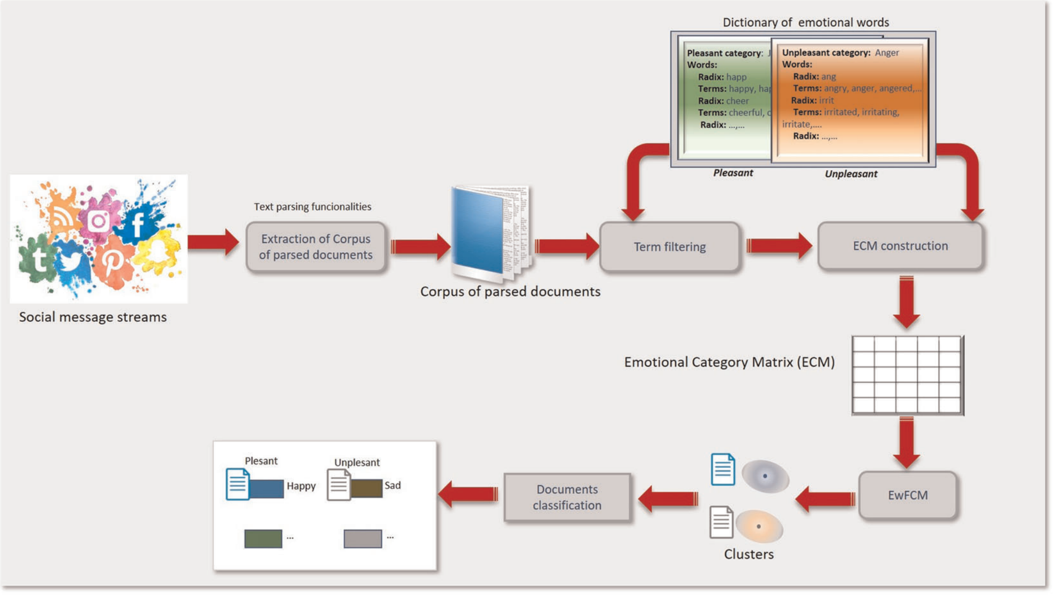

FIGURE 3 The FELEC framework. FELEC, Fuzzy Entropy Light Emotion Classification [Color figure can be viewed at wileyonlinelibrary.com]

short text, including words with hashtags to emphasize a certain topic, they are grouped by word hashtag, to compose a document representing a social trend, associated with that hashtag. A tweet with more hashtags will appear in each document for each hashtag.

Our approach focuses specifically on the tweet stream processing. After the initial parsing task, the remaining cleaned text is furthermore processed to extract relevant terms to build the term-document matrix. The idea is to exploit the emotions from Plutchik's wheel as the feature space; documents describing tweet trends can contain some emotions, expressed in the text words, such as nouns, adjectives, and adverbs. During the parsing of documents, each term is reduced in its stemmed form (i.e., cutting off the end of the word) and compared with terms from a dictionary of emotional words.

Building this dictionary allows us to collect a set of words that describes emotions: the selection of these words has been accomplished manually, starting from Plutchik's classification of emotions. Each one of the primary and secondary emotions (Table 1) became an emotional category, extended by synonyms or additional terms with similar meanings. An example of words belonging to the emotional category *joy* is shown in Table 2: words in this category are reduced to their stemmed form, so derivative terms, such as *happy, happiness*, and *happily,* hierarchically arranged, are attributable to a (root) word *happi* in the *joy* category.

According to Plutchik's psycho-evolutionary theory of emotions, the dictionary has been composed of exactly 16 emotional categories: eight primary emotions, constituting the four pairs of opposites: *joy–sadness, trust–disgust, anticipation–surprise, anger–fear*, and eight secondary emotions which are from the blending of the primary ones (see Table 1): *awe, love, optimism, content, aggression, disapproval, remorse, and submission.*

In Figure 3, the *Term Filtering* task is in charge to accomplish the emotion-related term selection: each document term appearing in the dictionary is assigned to the emotional

TABLE 2 Dictionary structure of emotional category: an example of words associated with the emotion *joy*

14. Pleasant category: Joy		
15. Words:	**16.** Radix: happi	
	18. Terms:	**19.** ...
	21. Adjective	**22.** Happy
	24. Adverb	**25.** Happily
	27. Noun	**28.** Happiness
	30. ...	**31.** ...
32. ...	**33.** Radix: cheer	
	35. Terms:	**36.** ...
	38. Substantive	**39.** Cheerful
	41. Verb	**42.** Cheer
	44. Adverb	**45.** Cheerfully
	47. ...	**48.** ...

category name the term occurs in. Precisely, each stemmed word from a document is searched in the dictionary: if it appears in an emotional category of the dictionary means that it is representative for that category. Selected terms from each document are associated with the corresponding emotional category name and contribute to building the Emotional Category Matrix (ECM), which will feed EwFCM. This (transposed) term-document matrix contains a weight that represents the emotion relevance in each document. More precisely, an entry (i, j) of this matrix provides the weight of the jth emotion in the ith document; in particular, the weight is cumulative of all the terms of the ith document, associated with the jth emotional category. The *ECM Construction* task achieves this aim: it indeed calculates all the matrix entries by summing up the weight of terms of the ith document to the category whose emotional meaning is associated with. The weight of each term is computed by the well-known term frequency-inverse document frequency (TF-IDF) measure.[36] This measure evaluates the importance of a term in a document; in fact the relevance increases as the number of times that that term appears in a document, compared with the inverse proportion of the same term in the whole collection of documents.

More formally, let $D = \{d_1, d_2, ..., d_N\}$ be the collection of documents, the ECM is a matrix with size $N \times 16$, where N is the cardinality of D. The rows represent the documents from D and the columns the emotional categories, according to Plutchik's theory.

An entry ECM_{ij} of such matrix is the weight related to the document d_i in correspondence of the jth emotion and is expressed as follows:

$$\mathrm{ECM}_{ij} = \sum_{t \in \mathrm{Dict}(j)} \text{TF-IDF}(t, d_i), \tag{8}$$

where t is a term of the document d_i, Dict(j) is the set of all the terms associated with the jth emotion in the dictionary; the TF-IDF is the measure calculated for the term t in the document d_i.

The dictionary is structured in a hierarchical form. Each emotional category collects stemmed terms that are semantically related to the category meaning. More specifically, as shown in Table 2, each term is associated with all the derivative words, formed by its own linguistic radix. For example, the emotional category name *joy* has as a synonym term, *happy*, which appears in a stemmed form *happi*. A list of derivative terms connected to *happi* (i.e., *happy, happily*, etc.) are shown in Table 2.

As stated, the final matrix will have the number of rows (data samples) equals to the number of documents in the collections and 16 columns (features) corresponding to the 16 emotional categories (i.e., Plutchik's primary and secondary emotions).

This matrix is given as input to the EwFCM algorithm. The number of clusters was set equal to the number of emotional categories, to classify the documents accordingly. Let us notice that once the clustering algorithm terminated, the resulting clusters were mapped with one of the emotional categories.

The final clustering partitioning finds a straightforward correspondence with the selected Plutchik's emotional categories: each cluster will be associated with the emotional category corresponding to the feature whose component value in the coordinates of the cluster center is the highest. Similarly, every document from input collection will be assigned to the emotional category corresponding to the cluster where it has the highest membership degree.

5 | EXPERIMENTAL RESULTS

Two sets of experiments have been conducted to evaluate the performance of the proposed approach, particularly, in comparison with the EFCM algorithm proposed in Reference [7]. The first tests were performed using well-known classification data sets extracted from the UCI ML repository (https://archive.ics.uci.edu/ml/datasets.php). Then, a collection of tweets in trend-based documents has been built to accomplish effective text for emotion-based classification. In Reference [7], similar tests were performed but comparing the performance of EFCM and EwFCM; the authors showed that EFCM performance overcomes that produced by EFCM. In light of the benefits presented by EwFCM, our goal is to show how this algorithm outperforms EFCM, with better performance in terms of classification accuracy, precision, and runtime.

Section 5.1 shows the comparative analysis of selected UCI ML data sets. Then, Section 5.2 introduces the experiments on documents extracted from social streams as well.

5.1 | Performance comparison tests applied using UCI machine data sets

Experiments have been carried out on some classification data sets from the UCI ML repository. The performance has been evaluated by considering the known measures, such as accuracy, precision, and recall, and for the sake of completeness, F1-score, which provides the harmonic mean of precision and recall. Moreover, the number of iterations and the execution times of the two algorithms are also provided.

The metrics are mainly based on the calculus of the TP, TN, FP, and FN parameters; more specifically:

- True Positive (TP) is the number of documents (patterns) correctly assigned to the expected emotional category (class).
- True Negative (TN) is the number of documents (patterns) that actually are not part of the class and consequently are not assigned to that class.
- False Positive (FP) is the number of documents (patterns) wrongly assigned to an emotional category (class).
- False Negative (FN) is the number of documents (patterns) not assigned to the right class appropriately.

In detail, the introduced performance measures are expressed as follows.

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP+TN} + \text{FP+FN}}, \quad \text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}},$$
$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{F1-score} = 2\cdot\frac{\text{precision}\cdot\text{recall}}{\text{precision} + \text{recall}}. \tag{9}$$

The values set for the parameters of EFCM and EwFCM are shown in Table 3. In EwFCM the number of clusters C is set to the number of classes.

For the sake of brevity, below are the results obtained using the Iris and wine data sets. The Iris data set consists of 150 patterns with four numerical features describing three types of iris flowers: Iris setosa, Iris virginica, and Iris versicolor, forming three classes. The four attributes describe the varieties of this flower in terms of sepal and petal length and weight. The first class is linearly separable from the other two, whereas the other two classes are not linearly separable from each other. For each class, the metrics accuracy, precision, recall, and F1-score are shown, finally, an average value calculated for all classes for each metric is shown.

Table 4 shows the results obtained by applying EFCM and EwFCM, respectively.

EwFCM improves the classification quality; results are slightly better that EFCM, as highlighted in bold in the table. The last column in the table shows that on average, all four performance measures obtained from EwFCM are higher to those obtained by EFCM.

Let us notice that all the metrics performance yields 100% in the classification of Iris setosa, as often happens for some classification models (e.g., References [38,39]) since due to the nature of the data set, this class is linearly separable from the other two classes.

In Table 5, the number of iterations and the running time of the two algorithms are also shown. EwFCM converges quickly compared with EFCM, due to the fuzzy entropy-based weights exploited to the initialization of cluster centers, as stated in Section 3.2.

TABLE 3 EFCM and EwFCM parameters used in the comparison tests

Parameter	**EFCM**	**EwFCM**
Initial number of clusters	10	–
Fuzzifier (m)	2	2
Stop iteration error (ε)	$1\cdot10^{-3}$	$1\cdot10^{-3}$
Cluster merging error (η)	$1\cdot10^{-2}$	–
Max. number of iterations ($i_{\max}$)	–	50

Abbreviations: EFCM, extended fuzzy c-means; EwFCM, entropy weighted fuzzy c-means.

TABLE 4 Data set Iris—EFCM versus EwFCM—classification performance

Parameter		Iris setosa (%)	Iris versicolor (%)	Iris virginica (%)	Mean (%)
Accuracy	EFCM	100.00	96.00	96.00	97.33
	EwFCM	100.00	96.00	**97.33**	**97.78**
Precision	EFCM	100.00	92.31	95.83	96.05
	EwFCM	100.00	92.31	**96.00**	**96.10**
Recall	EFCM	100.00	96.00	92.00	96.00
	EwFCM	100.00	96.00	**96.00**	**97.33**
F1-score	EFCM	100.00	94.10	93.88	96.00
	EwFCM	100.00	94.10	**96.00**	**96.71**

Abbreviations: EFCM, extended fuzzy c-means; EwFCM, entropy weighted fuzzy c-means.

TABLE 5 Data set Iris—number of iterations and running time

	Iterations	Running time (s)
EFCM	10	0.15
EwFCM	**8**	**0.11**

Abbreviations: EFCM, extended fuzzy c-means; EwFCM, entropy weighted fuzzy c-means.

These results confirm that EwFCM overwhelms EFCM in terms of both classification quality and runtime.

A similar result has been found for the *wine* data set.

The wine data set is composed of 178 patterns with 13 numerical features representing the chemical composition of wine. The data set contains data of chemical composition of Italian wine derived from three different crops, and thus three classes are given to which 59, 71, and 48 data points belong, respectively.

Like the previous experiment, Table 6 shows the performance obtained by using the EFCM and EwFCM algorithms.

Table 7 also shows the number of iterations and the execution time taken by the two algorithms.

Moreover, in this experiment, the results confirm that the EwFCM algorithm overwhelms the performance of EFCM in terms of both classification quality and running time.

Table 8 shows the average values of the accuracy, precision, recall, and F1-score metrics obtained by running the two algorithms on the 52 data sets enclosed in the UCI ML repository. Besides, the average difference between the number of iterations obtained by running EwFCM and EFCM as well as the average difference between the execution times obtained by running EwFCM and EFCM is given.

A graphical version of Table 8 relating to the mean classification indices has been provided in Figure 1. It is evident that on average, EwFCM shows better performances in terms of accuracy, precision, recall, and F1-score measures, which always overcome the corresponding ones evaluated on EFCM.

TABLE 6 Data set wine—EFCM versus EwFCM—classification performance

Parameter	Class 1 (%)	Class 2 (%)	Class 3 (%)	Mean (%)
Accuracy				
EFCM	97.74	96.05	97.14	96.98
EwFCM	**97.75**	**96.63**	**97.75**	**97.38**
Precision				
EFCM	93.65	98.48	93.88	95.34
EwFCM	**94.92**	**98.48**	**93.88**	**95.76**
Recall				
EFCM	100.00	91.55	95.83	95.79
EwFCM	100.00	**92.86**	95.83	**96.23**
F1-score				
EFCM	96.72	94.89	94.84	95.48
EwFCM	**97.39**	**95.59**	**94.85**	**95.94**

Abbreviations: EFCM, extended fuzzy c-means; EwFCM, entropy weighted fuzzy c-means.

TABLE 7 Data set wine—number of iterations and running time

	Iterations	Running time (s)
EFCM	11	0.16
EwFCM	**10**	**0.14**

Abbreviations: EFCM, extended fuzzy c-means; EwFCM, entropy weighted fuzzy c-means.

TABLE 8 Classification results in terms of accuracy, precision, recall, and F1-score measures evaluated on average on all the 52 data sets from UCI Machine Learning repository; mean difference of iterations and of runtime obtained executing EFCM and EwFCM on these data sets

	EFCM (%)	EwFCM (%)
Accuracy	96.87	**97.23**
Precision	95.32	**95.71**
Recall	95.61	**96.08**
F1-score	95.39	**95.87**
Mean iterations difference	–	**−2.04**
Mean running time difference (s)	–	**−0.05**

Abbreviations: EFCM, extended fuzzy c-means; EwFCM, entropy weighted fuzzy c-means.

5.2 | Tests on emotional Twitter data sets

The data set for emotion extraction from message streams was built by collecting 400,000 public tweets posted by users from March 2020 to December 2020 in the metropolitan Italian cities of Rome, Milan, Naples, Turin, Bologna, Florence, Bari, and Palermo. Tweets are selected by hashtag related to the COVID-19 pandemic (some examples of hashtags *#Covid19, #Covid-19, #Coronavirus, etc.*). Most of the analyzed tweets are in Italian; they are grouped by the city of origin and extracted in a day to form a document, constructing a corpus formed by 2104 documents.

The dictionary presented in previous sections encompasses all terms that can be assimilated to specific emotional categories, as shown in Table 2.

Before building the corpus, noisy tweets were eliminated, which, although including one or more of the selected hashtags, do not have terms connected to emotional categories and instead, relevant terms have been returned to their inflectional form.

In the *Term Filtering* task (Figure 3), document terms matching dictionary words related to an emotional category were selected, while words with incorrect, slang, or irregular syntactic forms were discarded: more than 60% of the extracted tweets, had such words to be discarded.

At the end of this process, the *ECM Construction* task builds the ECM matrix by calculating the TF-IDF-based weight values of each matrix entry, as detailed in Section 4 (see Equation 8).

The ECM matrix is given as an input to the EwFCM algorithm.

EwFCM is run by setting the initial number of clusters to 16, which is the number of emotion categories. The fuzzifier parameter is set to 2, the stop iteration error is set to 0.001 and the merging error is set to 0.01.

Tables 9 and 10 show the cluster center values, for the positive/pleasant and the negative/unpleasant features, respectively. As stated, each cluster is associated with the emotional

TABLE 9 Values of the components of the cluster centers for the pleasant category features

	Cluster centers—pleasant features							
Cluster	**Awe**	**Content**	**Expectation**	**Joy**	**Love**	**Optimism**	**Surprise**	**Trust**
C1	−0.001	−0.001	0.000	0.010	−0.001	−0.003	0.004	−0.002
C2	0.000	0.000	0.001	0.018	0.003	−0.002	0.004	**3.726**
C3	−0.001	−0.002	0.000	0.015	0.003	−0.005	0.007	−0.001
C4	0.001	0.000	−0.002	−0.044	−0.016	0.004	−0.011	−0.010
C5	−0.001	0.000	0.001	0.032	0.000	−0.005	0.007	0.002
C6	0.001	0.000	−0.001	−0.033	−0.005	**3.267**	−0.007	−0.005
C7	0.000	0.000	0.000	0.002	−0.002	0.001	−0.001	0.000
C8	0.000	0.001	0.001	0.011	−0.002	0.000	0.001	0.002
C9	0.000	0.000	−0.001	−0.018	−0.005	0.001	−0.004	−0.004
C10	0.000	0.000	0.000	0.001	−0.001	−0.001	**2.924**	0.000
C11	0.001	**4.741**	0.001	−0.007	0.000	0.004	−0.004	0.002
C1	0.001	0.001	−0.001	−0.027	−0.007	0.003	−0.007	−0.004
C13	0.001	0.002	**5.19**	0.006	0.008	0.003	0.000	0.006
C14	0.000	0.002	0.003	**4.011**	0.014	−0.002	0.008	0.013
C15	**2.515**	−0.003	−0.002	−0.018	0.002	−0.001	0.001	−0.005
C16	0.000	0.000	0.001	0.005	**4.598**	0.000	0.001	0.003

TABLE 10 Values of the components of the cluster centers for the unpleasant category features

	Cluster centers—unpleasant features							
Cluster	Aggression	Anger	Disapproval	Disgust	Fear	Remorse	Sadness	Submission
C1	−0.005	−0.003	**4.738**	−0.005	−0.014	−0.002	−0.007	−0.006
C2	−0.005	−0.002	−0.009	−0.004	−0.014	−0.002	−0.008	−0.004
C3	−0.008	−0.005	−0.019	−0.007	−0.039	−0.004	−0.018	**4.753**
C4	0.014	0.006	0.027	0.009	**6.385**	0.006	0.027	0.009
C5	−0.009	−0.004	−0.016	−0.007	−0.021	**1.987**	−0.011	−0.007
C6	0.010	0.004	0.017	0.007	0.023	0.004	0.014	0.006
C7	0.000	**2.451**	−0.001	0.000	0.007	0.000	−0.001	0.001
C8	−0.002	0.000	−0.003	−0.002	0.004	0.000	**4.261**	0.000
C9	0.005	0.002	0.008	**4.752**	0.010	0.002	0.008	0.002
C10	−0.001	0.000	−0.002	0.000	−0.005	0.000	−0.001	0.000
C11	0.005	0.003	0.01	0.005	0.027	0.003	0.011	0.007
C1	**3.109**	0.004	0.018	0.007	0.032	0.004	0.018	0.007
C13	0.000	0.001	0.002	0.001	0.009	0.001	0.000	0.004
C14	−0.011	−0.004	−0.019	−0.007	−0.028	−0.004	−0.017	−0.004
C15	0.000	−0.002	−0.006	−0.001	−0.027	−0.001	−0.009	−0.006
C16	−0.001	0.000	−0.001	0.000	−0.008	0.000	−0.005	0.000

category whose value of the cluster center component corresponding to the emotional category of the feature is highest. These values are highlighted in bold in Tables 9 and 10.

Table 11 shows the mapping between clusters and the emotional categories.

To compare EFCM and EwFCM, the setting considers for EFCM to the initial number of clusters to 50; for EwFCM the number of clusters is set to 16, that is, the number of emotional categories. In detail, the values set for the parameters of EFCM and EwFCM are given in Table 12.

After running EFCM, according to the algorithm definition, the final number of clusters obtained from the iterative process is 16; then a direct mapping between each cluster and an emotional category is defined as described.

The EwFCM algorithm performs 14 iterations before stopping, while EFCM needs 17 iterations before satisfying the stop condition.

Table 13 shows some comparative statistics about the number of documents assigned to each emotional category for both approaches; in particular, the column labels of Table 13 are detailed as follows.

- n_1 is the number of documents classified in an emotional category by using EWFCM in FELEC framework.
- n_2 is the number of documents classified in an emotional category by the framework presented in Reference [7].
- $n_\cap$ is the number of documents that fall into the same category for both frameworks;

TABLE 11 Assignment of clusters to emotional categories

Emotional category	Type (pleasant/unpleasant)	Cluster
Awe	Pleasant	C15
Content	Pleasant	C11
Expectation	Pleasant	C13
Joy	Pleasant	C14
Love	Pleasant	C16
Optimism	Pleasant	C6
Surprise	Pleasant	C10
Trust	Pleasant	C2
Aggression	Unpleasant	C1
Anger	Unpleasant	C7
Disapproval	Unpleasant	C1
Disgust	Unpleasant	C9
Fear	Unpleasant	C4
Remorse	Unpleasant	C5
Sadness	Unpleasant	C8
Submission	Unpleasant	C3

TABLE 12 EFCM versus EwFCM: parameter configuration for experiments

Parameter	EFCM	EwFCM
Initial number of clusters	50	16
Fuzzifier (m)	2	2
Stop iteration error (ε)	$1{\cdot}10^{-3}$	$1{\cdot}10^{-3}$
Cluster merging error (η)	$1{\cdot}10^{-2}$	–
Max. number of iterations ($i_{\max}$)	–	100

Abbreviations: EFCM, extended fuzzy c-means; EwFCM, entropy weighted fuzzy c-means.

- perc_1 is the ratio $n_\cap/n_1$ expressed in percentage.
- perc_2 is the ratio $n_\cap/n_2$ expressed in percentage.

Table 13 shows that on average about 70% of documents are consistently classified in the same emotional category by the two approaches.

A further investigation consists of studying classification consistency, that is, observing how these percentages vary by considering only those documents associated with an emotional category whose degree of membership in the corresponding cluster is greater than a given threshold. It is expected that as this threshold increases, the percentage of documents correctly assigned to the same category by both frameworks should also increase.

TABLE 13 Comparison between FELEC and the framework in Reference [7]: message stream-based document classification

Emotional category	n_1	n_2	$n_\cap$	$perc_1$ (%)	$perc_2$ (%)
Awe	104	112	70	67.31	62.50
Content	123	134	84	68.29	62.69
Expectation	122	113	83	68.03	73.45
Joy	159	172	112	70.44	65.1
Love	157	151	107	68.15	70.86
Optimism	130	114	89	68.46	78.07
Surprise	102	98	71	69.61	72.45
Trust	125	137	86	68.80	62.77
Aggression	146	139	100	68.49	71.94
Anger	128	139	91	71.09	65.47
Disapproval	113	109	77	68.14	70.64
Disgust	144	132	102	70.83	77.27
Fear	165	168	116	70.30	69.05
Remorse	105	103	73	69.52	70.87
Sadness	145	158	101	69.66	63.92
Submission	136	125	94	69.1	75.20

Abbreviation: FELEC, Fuzzy Entropy Light Emotion Classification.

Table 14 underscores this by reporting the results of emotion-based classification by both approaches, only for documents whose degree of membership in the cluster corresponding to the prevailing emotional category is greater than or equal to a threshold value $\sigma = 0.5$.

As shown in the table, on average, 90% of documents classified in an emotional category by one of the two frameworks are in turn classified in the same emotional category by the other framework.

Further comparative analysis is reported in Table 15, setting the degree of cluster membership corresponding to the prevailing emotional category greater than or equal to a threshold value $\sigma = 0.3$.

Also in this further comparative analysis, in which the threshold value σ was set to 0.3, about 80% of documents are classified into the same emotional categories (i.e., a document assigned to an emotional category by one of the two frameworks is assigned to the same emotional category by the other framework).

Figure 4 shows the mean values of $perc_1$ and $perc_2$ obtained considering all the documents (Table 13) and fixing the threshold σ to 0.3 (Table 14) and 0.5 (Table 15), respectively.

These results reveal that the higher the document membership degree to the cluster corresponding to the prevailing emotional category, the higher the percentage of documents assigned to the same category by the two frameworks. In other words, the more prevalent an emotion expressed in a document is, the more certain is the attribution of the corresponding emotional category to the document.

TABLE 14 Comparison results between the classification of the documents from Twitter streams by applying FELEC and the framework in Reference [7] where $\sigma = 0.5$

Emotional category	n_1	n_2	$n_\cap$	perc_1 (%)	perc_2 (%)
Awe	36	38	35	97.22	92.11
Content	40	44	38	95.00	86.36
Expectation	39	37	35	89.74	94.59
Joy	56	58	52	92.86	89.66
Love	50	46	45	90.00	97.83
Optimism	43	39	37	86.05	94.87
Surprise	36	33	30	83.33	90.91
Trust	41	44	37	90.24	84.09
Aggression	48	47	45	93.75	95.74
Anger	40	46	36	90.00	78.26
Disapproval	37	35	33	89.19	94.29
Disgust	48	45	44	91.67	97.78
Fear	51	53	49	96.08	92.45
Remorse	34	36	32	94.1	88.89
Sadness	46	48	42	91.30	87.50
Submission	44	42	40	90.91	95.24

Abbreviation: FELEC, Fuzzy Entropy Light Emotion Classification.

When documents appear in more than one emotional category in a nonnegligible percentage, classifying the document to an individual emotional category as described could be tricky and undecidable. In this case, it might be useful to consider, instead of the cluster whose membership degree is higher, a set of clusters the document might belong to. This way it could be assigned and then classified in more than one emotional category, generating a more complex blended emotion.

To this purpose, a further investigation considers those documents that have close values for the two higher membership degrees and at the same time, the remaining membership degrees to other clusters are irrelevant values.

Specifically, the focus is on those documents whose two higher membership values are greater than 0.2, with a mutual difference (in absolute value) lesser than 0.05 and finally, the difference from the other degrees of membership is greater than 0.15. Documents that meet these conditions are assigned to the prevailing pair of emotional categories. This test has been executed by FELEC and the framework in Reference [8].

Results are shown in Table 16. The first two columns show the two categories to which a document is assigned, the other column contain, respectively, the number of documents assigned to this category pair by FELEC (n_1) and[7] (n_2), the number of documents assigned to the category pair by both frameworks ($n_\cap$), and the percentages calculated for n_1 (perc_1) and n_2 (perc_2).

Let us observe that when a document is assigned to the category pair (due to a similar membership degree, that is, much greater than the membership degrees to the other clusters),

TABLE 15 Comparison results between the classification of the documents from Twitter streams by applying FELEC and the framework in Reference [7], where $\sigma = 0.3$

Emotional category	n_1	n_2	$n_\cap$	$perc_1$ (%)	$perc_2$ (%)
Awe	48	52	40	83.33	76.92
Content	56	60	45	80.36	75.00
Expectation	55	53	43	78.18	81.13
Joy	72	76	61	84.72	80.26
Love	68	70	58	85.29	82.86
Optimism	59	53	43	72.88	81.13
Surprise	46	46	37	80.43	80.43
Trust	55	64	46	83.64	71.88
Aggression	63	65	53	84.13	81.54
Anger	58	65	49	84.48	75.38
Disapproval	50	47	40	80.00	85.11
Disgust	63	60	51	80.95	85.00
Fear	74	76	62	83.78	81.58
Remorse	45	43	35	77.78	81.40
Sadness	67	69	57	85.07	82.61
Submission	60	58	49	81.67	84.48

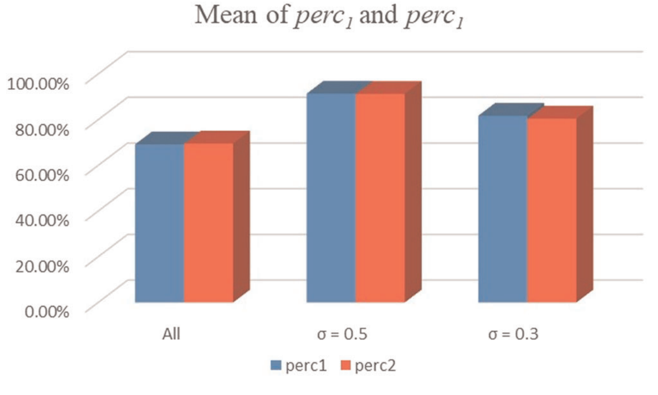

FIGURE 4 Histogram of the average values of $perc_1$ and $perc_2$ varying the threshold value σ [Color figure can be viewed at wileyonlinelibrary.com]

both the frameworks assign the document to the same pair of emotional categories. Therefore, the emotion-based classification of the documents is more accurate and complete.

Anyway, assigning a document to some emotional categories can become complex when the membership degrees are very similar to each other.

Let us notice that the last row of Table 16 presents a composition of two opposite emotions, one positive, *love,* and the other negative, *sadness*: the documents contain indeed two prevailing

TABLE 16 Comparison results of documents assigned to pairs of emotional categories

Emotional categories		n_1	n_2	$n_\cap$	perc$_1$ (%)	perc$_2$ (%)
Joy	Love	8	8	8	100.00	100.00
Optimism	Trust	5	6	5	100.00	83.33
Fear	Sadness	10	10	10	100.00	100.00
Aggression	Anger	6	5	5	83.33	100.00
Love	Sadness	5	5	5	100.00	100.00

and contrasting emotions that significantly express a clear feeling, experienced in the world pandemic situation.

6 | FINAL CONSIDERATIONS AND FUTURE PERSPECTIVES

The paper presents an emotion-based classification of documents extracted from social streams, by exploiting a fuzzy entropy-based variation of the FCM algorithm, called EwFCM. This algorithm optimizes the assignment of initial cluster centers by reducing the number of iterations and runtime. The proposed framework FELEC has been compared with the approach described in Reference [7], which exploits the EFCM clustering algorithm.

While EFCM finds the optimal number of clusters during iterations, EwFCM sets the number of clusters equal to the number of emotional categories, getting a lower computational complexity than EFCM; then the use of the fuzzy entropy measure to determine the initial cluster centers allows convergence to be achieved in fewer iterations. Comparative analysis shows better performance of EwFCM compared with EFCM in terms of result quality and runtime.

Experiments accomplished on a Twitter data set have shown that the two frameworks classify documents in the proper emotional categories, designed starting from Plutchik's wheel of emotions. FELEC associates indeed the right emotional categories to each document. Classification quality is confirmed by evaluating the accuracy, precision, recall, and F1-score measures.

A further investigation shows that the approaches also correctly detect documents expressing secondary emotions, that is, blended emotions according to Plutchik's emotion theory, coming from the combination of the two predominant emotions (calculated in the corresponding clusters by considering the membership degrees of the documents that are significantly higher than the other clusters).

A possible future investigation will focus on a different approach to emotion-based classification: the classification will be exclusively driven by all and only the data, that is, exploiting the emotional content enclosed into words and linguistic expressions of the tweets/documents; no preliminary emotional categories will be taken into account for leading the classification. This way the clustering result will reveal exactly the native mix of emotions from the document collection, pulling out the full range of possible shades of emotions in terms of their weight (i.e., their influence) in the combination of emotions.

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interests.

AUTHOR CONTRIBUTIONS

All authors contributed substantially and equally to the conception, design, drafting, and revision of the manuscript.

ORCID

Barbara Cardone http://orcid.org/0000-0002-9844-9513
Ferdinando Di Martino https://orcid.org/0000-0001-5690-5384
Sabrina Senatore http://orcid.org/0000-0002-7127-4290

REFERENCES

1. Liu B. Sentiment analysis and subjectivity. In: Indurkhya N, Damerau FJ, eds. *Handbook of Natural Language Processing*. Vol 2. New York, NY: Chapman and Hall/CRC; 2010:627-666. https://doi.org/10.1201/9781420085938
2. Plutchik R. The nature of emotions. *Am Sci*. 2001;89(4):344-350.
3. Minsky ML. *The Society of Mind*. New York, NY: Simon and Schuster; 1986:339 pp. isbn: 978-0671657130.
4. Arora M, Kansa V. Character level embedding with deep convolutional neural network for text normalization of unstructured data for Twitter sentiment analysis. *Soc Network Anal Min*. 2019;9(1). Art. No. 12. 14.
5. Dongliang X, Zhihong T, Rufeng L, Xiangtao K, Zhiyuan T, Wei S. Deep learning-based emotion analysis of microblog texts. *Inf Fusion*. 2020;64:1-11.
6. Saif MM. Sentiment analysis: detecting valence, emotions, and other affectual states from text. In: Herbert LM, ed. *Emotion Measurement*. Cambridge, UK: Woodhead Publishing; 2016:201-237. https://doi.org/10.1016/C2014-0-03427-2
7. Di Martino F, Senatore S, Sessa S. A Lightweight clustering-based approach to discover different emotional shades from social message streams. *Int J Intell Syst*. 2019;34(7):1505-1523.
8. Kaymak U, Setnes M. Fuzzy clustering with volume prototype and adaptive cluster merging. *IEEE Trans Fuzzy Syst*. 2002;10(6):705-712.
9. Cardone B, Di Martino F, Novel A. Fuzzy entropy-based method to improve the performance of the fuzzy c-means algorithm. *Electronics*. 2020;9:554.
10. De Luca A, Termini S. Entropy and energy measures of fuzzy sets. In: Gupta MM, Ragade RK, Yager RR, eds. *Advances in Fuzzy Set Theory and Applications*. Amsterdam, The Netherlands: North-Holland; 1979:321-338.
11. Kale S, Padmadas V. Sentiment analysis of tweets using semantic analysis. In: 2017 *International Conference on Computing, Communication, Control and Automation (ICCUBEA), Pune*; 2017:1-3.
12. Pang B, Lee L. A sentimental education: sentiment analysis using subjectivity based on minimum cuts. In: *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics ACL '04*. Stroudsburg, PA, USA: Association for Computational Linguistics; 2004:271-277. https://doi.org/10.3115/1218955.1218990
13. Poria S, Cambria E, Winterstein G, Huang GB. Sentic patterns: dependency-based rules for concept-level sentiment analysis. *Knowl-Based Syst*. 2014;69:45-63.
14. Cavaliere D, Senatore S. An ontology-driven perspective on the emotional human reactions to social events. *Int J Metadata Semantics Ontol*. 2021 in press
15. Strapparava C, Mihalcea R. Learning to identify emotions in text. In: *Proceedings of the* 2008 *ACM Symposium on Applied Computing, SAC '08*. New York, NY, USA: ACM; 2008:1556-1560.
16. Bandana R. Sentiment analysis of movie reviews using heterogeneous features. In: 2018 2*nd International Conference on Electronics, Materials Engineering & Nano-Technology* (*IEMENTech*), *Kolkata*; 2018:1-4.
17. Barbosa L, Feng J. Robust sentiment detection on Twitter from biased and noisy data. In: *Proceedings of the* 23*rd International Conference on Computational Linguistics* (*COLING* 2010), *Beijing, China*; 2010:36-44.
18. Pak A, Paroubek P. Twitter as a corpus for sentiment analysis and opinion mining. In: *Proceedings of the Seventh International Conference on Language Resources and Evaluation, LREC*2010, *Valletta, Malta*; 2010:1320-1326.
19. Verma S, Vieweg S, Corvey WJ, et al. Natural language processing to the rescue? Extracting "situational awareness" tweets during mass emergency. In: *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media* (*Barcelona, Spain,* 2011); 2011:385-392.

20. Zimbra D, Ghiassi M, Lee S. Brand-related Twitter sentiment analysis using feature engineering and the dynamic architecture for artificial neural networks. In: *49th Hawaii International Conference on System Sciences* (*HICSS*), *Koloa, HI*; 2016:1930-1938.
21. Baccianella S, Esul AA, Sebastiani F. SentiWordNet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In: *Proceedings of the Seventh Conference on International Language Resources and Evaluation* (*LREC'10*); 2010:2200-2204.
22. Singh VK, Piryani R, Uddin A, Waila P. Sentiment analysis of movie reviews and blog posts. In: *3rd IEEE International Advance Computing Conference, IACC, Ghaziabad*. IEEE: Piscataway, NJ, United States; 2013:893-898.
23. Luo Z, Osborne M, Wang T. An effective approach to tweets opinion retrieval. *World Wide Web*. 2013;18: 545-566.
24. Neelakandan S, Paulraj D. A gradient boosted decision tree-based sentiment classification of Twitter data. *Int J Wavelets Multiresolution Inf Process World Sci*. 2020;18(4):2050027. pp. 1-21.
25. Neelakandan S, Paulraj D. An automated learning model of conventional neural network based sentiment analysis on Twitter data. *J Comput Theor Nano Sci*. 2020;17(5):2230-2236.
26. Jianqiang Z, Xiaolin G, Xuejun Z. Deep Convolution Neural Networks for Twitter Sentiment Analysis. *IEEE Access*. 2018;6:23253-23260.
27. Socher R, Perelygin A, Wu J, et al. Recursive deep models for semantic compositionality over a sentiment treebank. In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*; 2013: 1631-1642.
28. Fersini E. A machine learning perspective. In: Pozzi FA, Fersini E, Messina E, Liu B, eds. *Sentiment Analysis in Social Networks*. Burlington, MA, United States: Morgan Kaufmann; 2017:91-111. (Chapter 6). https://doi.org/10.1016/B978-0-12-804412-4.00006-1
29. Ekman P, Friesen WV. *Unmasking the Face: A Guide to Recognizing Emotions from Facial Clues*. San Jose, CA, United States: Malor Books; 2003:23pp. isbn: 978-1883536367.
30. Frijda NH. The laws of emotion. *Am Psychol*. 1988;43(5):349-58.
31. Parrot W. *Emotions in Social Psychology*. Philadelphia, PA, United States: Psychology Press; 2001:392pp. isbn: 978-0863776830.
32. Barrett LF. Are emotions natural kinds? *Perspect Psychol Sci*. 2006;1(1):28-58.
33. De Leersnyder J, Boiger M, Mesquita B. Cultural differences in emotions. *Emerging Trends Soc Behav Sci: Interdiscip Searchable Linkable Resour*. 2015;121:22. https://doi.org/10.1002/9781118900772.etrds0060
34. Lindquist KA, Wager TD, Kober H, Bliss-Moreau E, Barrett LF. The brain basis of emotion: a meta-analytic review. *Behav Brain Sci*. 2012;35(3):121-143.
35. Plutchik R. A general psychoevolutionary theory of emotion. In: Plutchik R, Kellerman H, eds. *Emotion: Theory, Research, and Experience: Theories of emotion*. Vol 1. New York: Academic Press; 1980:3-33. https://doi.org/10.1016/B978-0-12-558701-3.50007-7
36. Salton G, Buckley C. Term-weighting approaches in automatic text retrieval. *Inf Process Manage*. 1988; 24(5):513-523.
37. Hathaway RJ, Hu Y. Density-weighted fuzzy c-means clustering. *IEEE Trans Fuzzy Syst*. 2009;17(1):243-252.
38. Chang H, Astolfi A. Gaussian based classification with application to the Iris data set. *IFAC Proc Vol*. 2011; 44(1):14271-14276. ISSN 1474-6670. https://doi.org/10.3182/20110828-6-IT-1002.02644
39. Eldem A, Eldem H, Üstün D. A model of deep neural network for Iris classification with different activation functions. In: 2018 *International Conference on Artificial Intelligence and Data Processing (IDAP)*; 2018:1-4. https://doi.org/10.1109/IDAP.2018.8620866

How to cite this article: Cardone B, Di Martino F, Senatore S. Improving the emotion-based classification by exploiting the fuzzy entropy in FCM clustering. *Int J Intell Syst*. 2021;1-24. https://doi.org/10.1002/int.22575