



# Improving Sentiment Classification Performance through Coaching Architectures

Alberto Fernández-Isabel<sup>1</sup> · Javier Cabezas<sup>1</sup> · Daniela Moctezuma<sup>2</sup>  · Isaac Martín de Diego<sup>1</sup>

Received: 5 August 2021 / Accepted: 31 March 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

## Abstract

Intelligent systems have been developed for years to solve specific tasks automatically. An important issue emerges when the information used by these systems exhibits a dynamic nature and evolves. This fact adds a level of complexity that makes these systems prone to a noticeable worsening of their performance. Thus, their capabilities have to be upgraded to address these new requirements. Furthermore, this problem is even more challenging when the information comes from human individuals and their interactions through language. This issue happens more easily and forcefully in the specific domain of Sentiment Analysis, where feelings and opinions of humans are in constant evolution. In this context, systems are trained with an enormous corpus of textual content, or they include an extensive set of words and their related sentiment values. These solutions are usually static and generic, making their manual upgrading almost unworkable. In this paper, an automatic and interactive coaching architecture is proposed. It includes a ML framework and a dictionary-based system both trained for a specific domain. These systems converse about the outcomes obtained during their respective learning stages by simulating human interactive coaching sessions. This leads to an Active Learning process where the dictionary-based system acquires new information and improves its performance. More than 800, 000 tweets have been gathered and processed for experiments. Outstanding results were obtained when the proposed architecture was used. Also, the lexicon was updated with the prior and new words related to the corpus used which is important to reach a better sentiment analysis classification.

**Keywords** Sentiment analysis · Automatic coaching · Active learning · Combination of information · Continuous dynamical system

## Introduction

Human beings are animals able to feel emotions and express them to other like-minded individuals [1]. We have several channels to communicate sentiments, but the most typical is through language, producing expressions with the body, and using objects with special meaning.

However, all of them present some drawbacks [2]. The language can indicate complex ideas or topics but needs a receiver in a nearby space to communicate the information. Body expressions are usually very simple and concise, and they need a relatively close receiver. Finally, objects are usually easy to transport and cover long distances, but they are not able to express several feelings and their possible combination. Instances of these latter could be smoke signals and colors. In the first case, they are typical to attract attention or raise an alarm. In the second case, colors can indicate the mood of individuals (it is well-known in several cultures

---

✉ Daniela Moctezuma  
dmoctezuma@centrogeo.edu.mx

Alberto Fernández-Isabel  
alberto.fernandez.isabel@urjc.es

Javier Cabezas  
javier.cabezas@urjc.es

Isaac Martín de Diego  
isaac.martin@urjc.es

<sup>1</sup> Data Science Laboratory, Rey Juan Carlos University, C/ Tulipán, s/n, 28933 Móstoles, Spain

<sup>2</sup> Centro de Investigación en Ciencias de Información Geoespacial, Circuito Tecnopolo Norte 117, Col. Fraccionamiento Tecnopolo Pocitos, 20313 Aguascalientes, México

that light colors represent positive feelings and dark colors represent negative feelings).

As a consequence of these limitations in human communication channels, we improved the language by translating it to text [3]. Thus, the main handicap of language (a nearby receiver) is solved, since the text can be written in a manuscript which can be easily transported and for sure, it usually endures for a long time. Nonetheless, language usually needs the support of gestures and body expression to be understood as it is non-deterministic. Common examples are irony (where the sentence, in this case means, different), synonyms (where several words share the same meaning), and polysemy (where a word has multiple meanings) [4].

Moreover, emotions transmitted by words can fluctuate according to the context or the moment of time [5]. This produces more difficulties to understand the feelings and emotions translated into textual contents.

Sentiment Analysis appears as a key issue to address these problems with texts [6]. Thus, it includes a set of techniques to measure the sentiment polarity by analyzing textual information.

The majority of the proposed solutions are generic and static [7]. This produces a problem in specific and variant contexts [8]. Moreover, the upgrade of these systems takes a big effort in time and resources. For this reason, the development of an automatic, dynamic, and adaptable framework for fickle contexts becomes relevant.

In this paper, it is proposed a novel architecture based on Active Learning. It includes a dynamic sentiment framework based on a dictionary (i.e., a lexicon) called *EmoWeb 2.0*. It is a social media adaptation of a first prototype initially focused on digital newspapers [9]. This system is able to learn new words and adapt sentiment values to the context over time. However, this dynamic learning can lead to a long-term degradation in the quality of the learned information due to lexicons are only based on individual words. A Machine Learning (ML) framework called *micro* Text Classification ( $\mu$ TC) [10] intervenes to mitigate this effect.

Acting as a coach ( $\mu$ TC) and a disciple (*EmoWeb 2.0*), both systems converse about the outcomes obtained during their respective learning stages by simulating human interactive coaching sessions. The ML system provides semantic knowledge to the *EmoWeb 2.0* framework (which is a lexical system), facilitating the automatic acquisition of new relevant information, and applying corrections to possible errors. This solution significantly improves the ability of *EmoWeb 2.0* to learn new words and adjust their sentiment values.

Several experiments have been designed to validate the proposal. Twitter has been selected as the reference social network [11] and the ongoing COVID-19 pandemic as the focus topic [12]. These choices are motivated by the broad effects on global society and the continuous variation of

several words over time which usually do not present such relevance (e.g., isolation, vaccines, or holidays).

The rest of the paper is organized as follows. **Background** introduces the foundations, relevant literature, and some works related to the proposal. **Proposed Framework** details the solution, the different systems and their interactions. **Experiments** details a set of experiments to illustrate the manner in which the presented approach works and its performance. Finally, **Conclusions** concludes and provides further research guidelines.

## Background

The main concepts related to this proposal are Knowledge-Based Systems (KBSs), Sentiment Analysis, and Active Learning. In this section the theoretical aspects related to them are presented. First, a literature review of the KBSs domain is addressed (see **Knowledge-Based Systems**). Thus, these systems are introduced showing their typical designs and technologies, and the most common approaches that make use of them. Subsequently, the Sentiment Analysis field is described, explaining the different approaches and configurations (see **Sentiment Analysis**). The Active Learning technique is also explained by highlighting automatic process configurations where a machine is able to train another machine without human supervision (see **Active Learning**). Finally, special attention is paid to similar works using the previous approaches and due comparisons are made with the proposal of this manuscript (see **State-of-the-art Approaches**).

## Knowledge-Based Systems

KBSs are considered a major branch of Artificial Intelligence (AI). They can be defined as a computer system fed by different sources of data with the aim of giving shape to an internal knowledge called knowledge base. That is, KBSs deal with knowledge, they can justify their decisions and have the ability to learn [13].

This knowledge confers the system a certain degree of expertise that is used by a reasoning engine to solve relevant problems or make required decisions depending on the particular requirements arising from the context on which the system is inserted [14]. Therefore, a proper understanding of the context as well as having effective learning processes at the system's disposal undoubtedly represent two demanding requisites that these types of systems must meet.

Delving into the knowledge, it may include facts, concepts, procedures, models, heuristics, or examples and it may be specific or general, exact or fuzzy, procedural or declarative [15]. The actual internal representation of this knowledge relies on the particular design concept and provides the

system with a solid base from which new potential knowledge may be inferred through the reasoning engine.

Regarding the categories to classify KBSs, they can be organized into expert systems, linked systems, intelligent tutoring systems, case-based systems, and database management systems [13]. Expert systems are approaches that emulate the decision-making process of human experts [16]. Linked systems, also known as hypermedia systems, are approaches that use chunks of media to generate knowledge [17]. Intelligent tutoring systems are approaches specifically dedicated to teaching and training the user in specific matters by using their internal knowledge for this purpose [18]. CASE-based systems, called systems for Computing-Aided Software Engineering, are approaches that guide the development of other systems for better effectiveness [19]. Finally, database management systems are approaches that provide an abstraction layer through specific query languages and visual interfaces [20]. Thus, they can simplify the use of databases.

The approach proposed in this paper consists of a software architecture which is made up of an expert system (*EmoWeb 2.0*) which is able to learn from another expert. This expert is a computer system instead of a human, but the essence of the expert systems is implicit.

## Sentiment Analysis

Sentiment Analysis constitutes an extensive field of research under the Natural Language Processing (NLP) following the objective of extracting subjective information expressed in texts written by humans [21]. Among the broad scope on which its responsibilities may fall, the inspection of the potential influence that texts might exert on readers and their respective feelings show to be two leading aspects.

In this regard, the outcomes arising from Sentiment Analysis may be represented by emotions (anger, fear, joy, repulsion, sadness, and surprise) [22] or by the polarity of emotions themselves (positive, neutral, and negative) [23]. The former shows a plausible level of complexity and subjectivity that turns the latter into the most preferred option.

As for the main approaches followed to implement Sentiment Analysis solutions, the ones based on dictionaries and the ones focused on ML techniques are the most common options.

Dictionaries can be defined as a collection of pre-stored words which appear associated with a sentiment score or a polarity. These dictionaries are usually referred to as lexicons. By means of these dictionaries, the polarity of a sentence could be calculated by averaging the polarity of the words matching the dictionary contents. A clear shortcoming of this approach gets defined by the limited set of words stored in the lexicons. Some instances of general purpose lexicons are *SentiWordNet* [23] and *SenticNet* [24].

With regard to ML approaches, sentiment values are predicted by using statistical models based on distributional semantics. These models firstly follow a training phase on which a collection of tagged corpora is used (also named *Ground Truth*). Subsequently, the testing phase is triggered to perform the actual classification of texts by the model. In this case, the generated ML model learns from a subset of the corpus and tests from another subset (different from the previous one). It allows evaluating the degree of effectiveness in the learning process.

There exist multiple ML solutions to tackle the Sentiment Analysis task [25]. For instance, there are solutions using NLP at the beginning of the process and a ML classifier at the end of it to build the model [26]. Other well-known solutions are those that only rely on deep learning approaches without rendering textual content. This is especially useful when the objective is to detect the syntax and semantic patterns of the text (e.g., in conversations between two parties [27]). Convolutional Neural Networks (CNN) [28], and Bidirectional Encoder Representations from Transformers (BERT) [29] are commonly used techniques in this perspective. Delving into these last techniques, attention-based models have arisen in the context [30]. These systems usually use two attention models to compute the weights of the model: intra-attention and global attention [31]. The first one is focused on estimating the similarity between any two words in a sentence, while the second one considers the whole textual content from a global perspective.

Notice that both strategies (i.e., dictionaries and ML approaches) are able to work together in hybrid approaches. Thus, the polarity of a given text gets estimated by firstly applying dictionary-based techniques and, subsequently, pertinent ML strategies are applied to predict sentences presenting words not detected in the lexicon. Typical instances of these approaches can be found in [32] and [33].

On the other hand, modern approaches have been focused on collecting multimodal data from several sources of information to estimate the emotion recognition [34]. Typical elements analyzed in addition to text are: images, videos, and voice records. Therefore, the fusion of this information and the development of a complete framework [35] to deal with the issue are still open challenges in the domain.

Other trends have also appeared in recent times. Knowledge-based systems that contain relationships between concepts to address the semantic level of language in the Sentiment Analysis domain are well-known approaches. These architectures usually make use of ontologies [36] or graph-based networks [37] to achieve the sentiment recognition task.

Regarding the limitations of the usual Sentiment Analysis approaches, the most important is the lack of dynamic learning or evolution over time [38]. This problem is related to the prefixed sentiment values of a lexicon or the

learned values by a ML technique. Therefore, retraining these systems and also making the lexicons dynamical are core proposals to mitigate the issue [9]. Another relevant limitation consists in the ambivalence of the language. Thus, in the Sentiment Analysis domain, it is common to find words capable of transmitting different emotions according to the context [39].

This paper makes use of a new release of a previous dynamic system that is now automatically trained over time by using a ML framework acting as a coach. This approach allows updating complex and static architectures based on Sentiment Analysis without human supervision.

## Active Learning

One of the main problems of any ML solution gets represented by the absence of adaptability to changes occurring in phenomena or data over time. Usually, ML models are trained just once and no re-training process is usually addressed.

Active Learning emerges as a possible solution to mitigate this fact. The key idea resides in the noticeable performance improvement of a ML algorithm when more training data are available (in any way and any time) [40].

Another way to conduct Active Learning is based on the idea of having a committee for sampling selection. It is crucial that this committee is arranged in an appropriate way in order to make available several points of view of the issue [41]. These multiple views (i.e., independent classifiers evaluating different aspects are used to reach a final decision) allow the system to encompass pieces of information resulting from distinct opinions. This technique is known as co-training [42].

Moreover, Active Learning can be adapted to coaching approaches [43] when the opinions of an expert lead the learning process of a system. This approach is based on the underlying coaching concept present in human interactions. This concept consists of a set of guidelines covering a specific area to provide new skills and knowledge, facilitating an observable improvement in performance, capabilities, and competencies. Usual examples illustrating this concept can be found in employee development programs of organizations or individuals [44].

In this paper, a coaching-based architecture based on Active Learning is used to improve the learning ability of a dynamic sentiment framework. Both *EmoWeb 2.0* and the ML framework are previously trained, and interchange information and opinions, establishing a conversation like a disciple (also called coachee) and a coach, respectively. The outcome reveals an improvement in the performance of the disciple system which acquires relevant knowledge from the coaching system.

## State-of-the-art Approaches

Approaches previously introduced are combined in this proposal. It includes an Active Learning coach-based architecture composed of two different systems focused on dynamic Sentiment Analysis.

In the literature, there are some related works whose objective is to generate an updated lexicon or an updated Sentiment Analysis method. In [45], an unsupervised learning approach for updating sentiment lexicons is proposed. This approach works according to contextual semantics between words to capture the relationship between tweets and hence update their sentiment scores. In [46], it is introduced a lexical updating algorithm capable of increasing the number of words considered.

Another typical approach usually included in the lexicon updating task refers to the use of genetic algorithms [47]. These algorithms are responsible for optimizing the values of the lexicon when it is built through a labeled corpus. This allows improving the quality of the lexicon, adapting it perfectly to the training instance.

Regarding the generation of domain-specific lexicons, there are techniques focused on obtaining specific words related to the addressed field [48]. In this case, Active Learning becomes very useful to gather information from documents once a basic lexicon is generated from document-level annotations. This process could be considered as a seed (i.e., the initial general purpose lexicon) that grows and adapts to a specific environment (i.e., the domain-specific lexicon). An instance of a domain where specific lexicons are very helpful is the healthcare field [49].

Combinations of lexicons and ML approaches are very typical and present some relevant strengths (see [Sentiment Analysis](#)). However, these combinations can be oriented to update a lexicon by using the knowledge captured by the ML method (usually a neural network). Relevant instances could be [50] and [51]. Both works use the Active Learning concept to improve the quality of the system over time.

In conclusion, it can be stated that there are two main issues to be addressed in the dynamic Sentiment Analysis domain. Firstly, the updating process of the lexicon over time, and secondly, the learning process of new words related to the context. The latter is more relevant when a domain-specific lexicon is considered.

In this paper, both concepts are included. The *EmoWeb 2.0* framework is able to learn new words and modify the sentiment values of the words. However, lexicon-based systems usually lose accuracy when they are adapted on several occasions over time. For this reason, an automatic coaching architecture has been included in the proposal. This architecture includes a previously trained ML framework which has the ability to correct those mistakes made by *EmoWeb*

2.0. The solution devised makes the system domain-specific and improves the adaptation over time.

## Proposed Framework

This section details the proposed architecture based on Active Learning by using a coaching interactive integration. This coaching process consists in evaluating the labeling results (i.e., the sentiment polarity of texts) offered by *EmoWeb 2.0* by considering the criteria of the  $\mu$ TC framework [10].  $\mu$ TC is a ML solution that adopts a set of text transformations and a Support Vector Machine (SVM) method as the central core, where both lead its internal evaluation process to effectively provide advice on tweet labels.

Next, the foundations of *EmoWeb 2.0*, the  $\mu$ TC framework, and the coaching architecture (conceived to make both systems work together) are detailed.

### EmoWeb 2.0: A Dynamic Lexicon-based Approach

The *EmoWeb 2.0* framework responds to an adaptation of a former prototype called *EmoWeb* [9]. This first version was designed to analyze online newspapers, while *EmoWeb 2.0* is specifically focused on texts coming from Twitter (i.e., tweets) [52].

Delving into the features of the original *EmoWeb*, it is a framework focused on dynamic Sentiment Analysis. It uses a well-known lexicon as a seed. It applies text analysis techniques followed by an unsupervised learning algorithm to textual content in order to further incorporate those new words detected into the lexicon along with their associated sentiment values. Lexicon words are joined to an updating process of their associated sentiment values according to the trends detected which utterly determines their strength and relevance over time.

The transition of the former *EmoWeb* framework to the new *EmoWeb 2.0* involves specific adjustments to adhere the latter to the intrinsic nature of Twitter data while preserving, at the same time, the original working philosophy. Thus, all thresholds used in the previous release are included in this new version maintaining the functionalities of the original system. These thresholds are  $\alpha$  to control the forgetting factor, *upper threshold* to indicate the limit for relevant positive values, *lower threshold* to indicate the relevant negative values and the *number of days* a word needs to overcome the thresholds to become relevant [9].

The new version sets as the initial general purpose well-known lexicon the English version of *SenticNet* [24]. This lexicon offers about 200, 000 English lemmatized words along with associated numerical values in  $[-1, 1]$  representing sentiment polarities ( $-1$  completely negative,  $0$  neutral, and  $1$  completely positive). The starting lexicon

is further enriched with the new words learned during the data processing. These new words receive or update their sentiment value according to a calculation based on the sentiment scores of the tweets to which they belong. This process reflects the dynamic nature of sentiments by rectifying the word values stored in the lexicon according to the trends detected in input tweets over time.

The system is organized into two separate and top-down sequential workflows, being *SenticNet* and *Twitter* their triggering sources, respectively. The internal processes of *EmoWeb 2.0* are three: setting the initial seed, estimating the sentiment value of the tweets using the words in the lexicon, and updating the lexicon with new words and the new sentiment values before processing another calendar day.

The architecture of the system that meets all requirements consists of six modules: the *Seed Retrieval Module*, the *Data Retrieval Module*, the *Data Processing Module*, the *Sentiment Evaluation Module*, the *Visualization Module*, and the *Rest Services Module* (see Fig. 1). The architecture is adapted to the specifications of the Twitter platform, and it is completed with a knowledge base. This knowledge base takes responsibility not only as a proper storage resource but also as a critical passive coordinator and cohesive member of the system.

The *Seed Retrieval Module* is in charge of the first process. This process is executed just once at the initial stage to incorporate *SenticNet* data into the internal lexicon by storing its words along with their associated polarity.

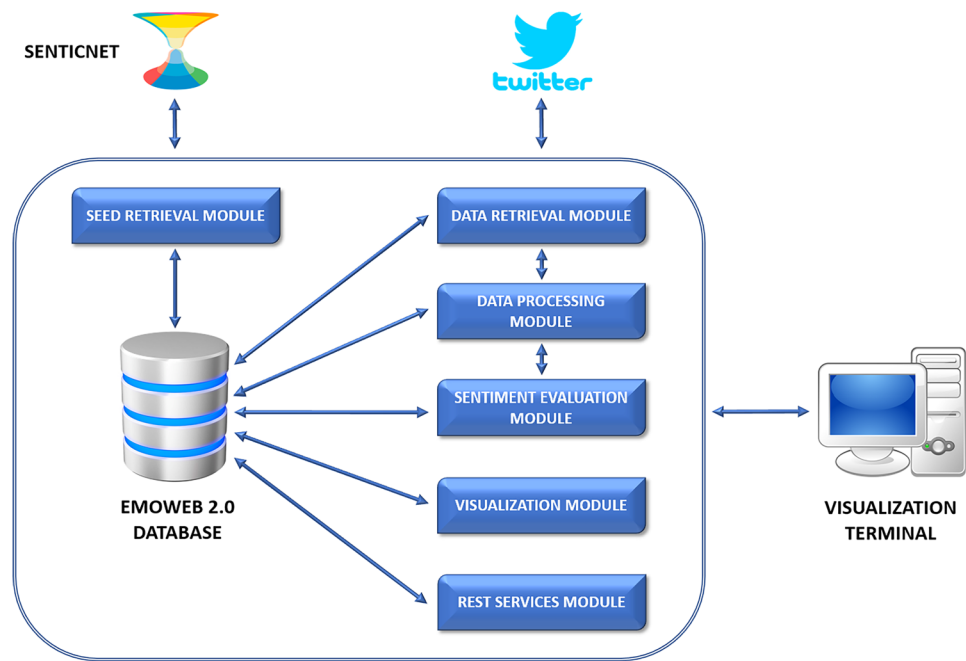
Following this initial one-off phase, the second process is triggered. Firstly, the *Data Retrieval Module* gathers a set of tweets created on a particular calendar day  $d$  and conducts all the necessary formatting adaptations (e.g., tweet hydration task [53]) before presenting it to the *Data Processing Module* (see Fig. 2). This computes the sentiment scores for each tweet along with a label (positive, neutral, or negative) and registers the new words detected in the internal lexicon.

After completion, the *Sentiment Evaluation Module* initiates the third process (see Fig. 3). It involves updating the sentiment values stored for the lexicon words by considering the trends detected until the calendar day under analysis. Finally, the *Data Retrieval Module* receives a notification to trigger the whole process again for the next consecutive calendar day (i.e.,  $d+1$ ) if still required. That is, the whole workflow is repeated as many times as there are days to process.

The *Visualization Module* maintains the functionalities of the initial prototype but now adapting the requests to the information gathered on Twitter.

Lastly, the framework uses the *Rest Services Module* to publish an ecosystem of *REST* services. They allow the data stored in the knowledge base to be made available to interested external entities.

**Fig. 1** Overview of the *EmoWeb 2.0* framework architecture



Delving into the internal way of working of the system, it preserves the same three flags with similar functionalities to the ones of the first prototype: *Modified*, *State*, and *Accumulated*.

The first flag is set to 1 when a word (either new or already present) is detected during the processing of tweets on a particular calendar day  $d$ . This flag is consulted by the *Sentiment Evaluation Module* to divide the lexicon in two separate groups (the one containing the words detected during the day and the one including those not present, i.e., showing their *Modified* flag equal to 0). These two groups receive different treatment during the updating processes concerning word sentiment scores. Next, all *Modified* flags are set to 0 with the aim of preparing the lexicon for the next calendar day  $d+1$ .

The second flag determines which words are influential. A word is considered as influential (i.e., *State* flag equal to 1) when its sentiment score has been exceeding the *upper threshold* or the *lower threshold* for more than a number of days. This flag is calculated by the *Sentiment Evaluation Module* during the word sentiment review procedure. It is consulted by the *Data Processing Module* during the tweet sentiment calculations occurring the following calendar day.

The third flag is referred to as *Accumulated* and registers the number of consecutive days that the word sentiment has been exceeding the limits drawn by the thresholds. It is likewise managed by the *Sentiment Evaluation Module*.

Finally, in reference to the text processing tasks, *EmoWeb 2.0* differs from the initial release since it manages the content of tweets. The *Data Processing Module* is responsible for this task. Here, tweets are processed by performing a

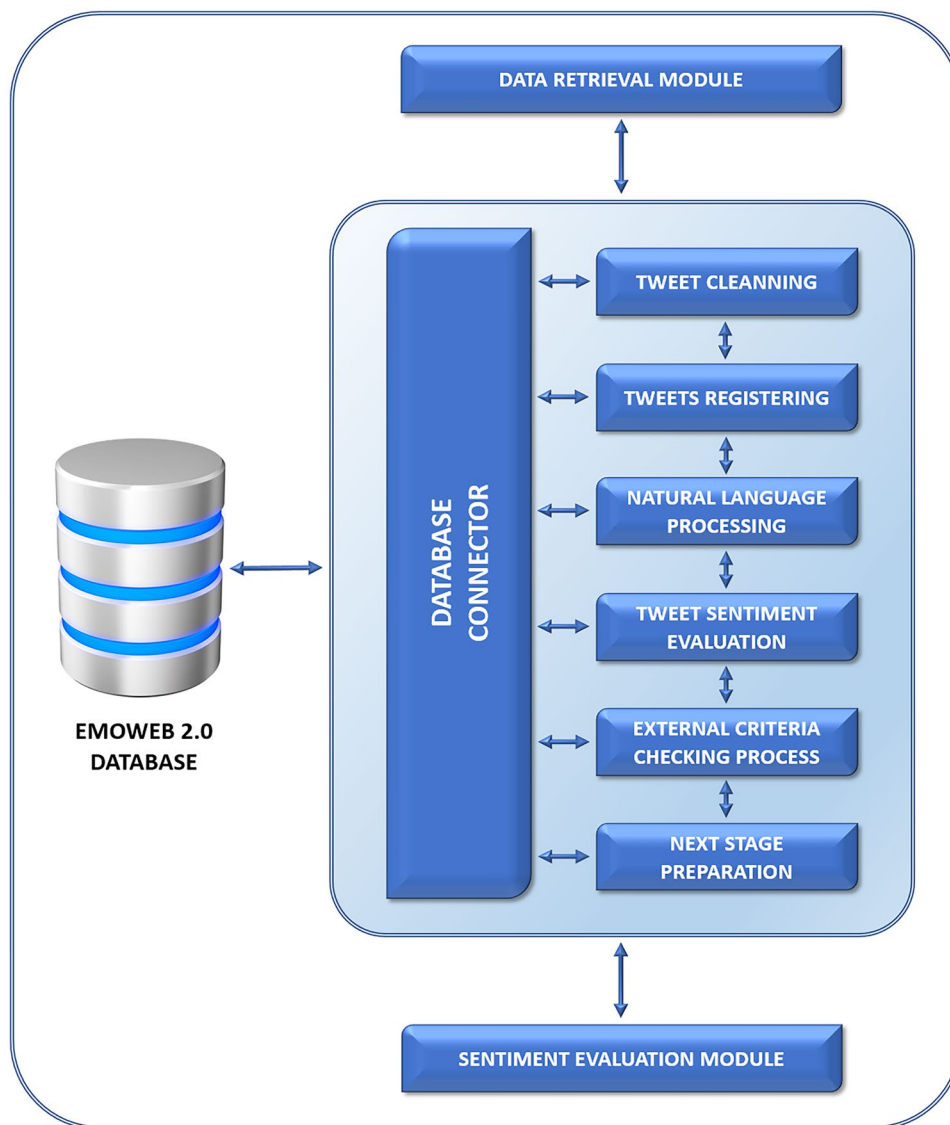
cleaning step which includes the removal of non-relevant information such as smileys, emojis, special characters, mentions, and hashtags, among others. The resulting cleaned tweets along with some metadata information (tweet ID, creation date, etc.) are stored in the database. Subsequently, simple NLP activities are conducted. The scope of related actions encompasses tokenization, lemmatization, and Part of Speech (*PoS*) tagging methods to select base forms of adverbs, adjectives, verbs, and nouns. Stopwords of the language are also deleted.

### The $\mu$ TC Framework

$\mu$ TC is a ML framework consisting of several easy-to-implement text transformations and text representations [10] aimed at classifying and estimating the sentiment associated with texts. The process is driven by four main steps (referred to as pre-processing, tokenization, weighting, and classification) and makes use of a SVM classifier at the end of it. Configurations required for each stage of the process are dynamic and get determined by a combinatorial optimization algorithm responsible for selecting the best possible set of text transformation and representation settings depending on the actual input text to be processed.

In the pre-processing step all the classical and well-known steps such as lower-casing, url, hashtag, and user (user) handlers are included. These handlers have, in general, three different options: *remove*, *group*, and *none*. The *remove* handler involves the deletion of the most recent processed token, *group* entails that all the similar tokens

**Fig. 2** Excerpt of the *Data Processing Module* architecture



are grouped and identified with a unique symbol, and *none* means that their original value is maintained.

In the case of the tokenization step, values for  $n$  and  $q$  for  $n$ -grams and  $q$ -grams, are selected in the process, respectively. Skip grams also have a set of possible configurations. Although there is a set of predefined values, these parameters could be established by the user. For more technical details and user examples see the documentation pages<sup>12</sup>.

With respect to the weighting step, three options are considered. The first alternative refers to *TF* which implies that only the frequency of each token is taken into account. Another possible selection considers the well-known *TF-IDF*, which expresses the relevance of each token in each

corpus document. Lastly, a weight-based entropy configuration could also be chosen as an option.

Finally, the classification step involves the use of a SVM algorithm which gets configured with default parameters (linear kernel and a  $C$  value equal to 1). During each iteration, the SVM algorithm tests the whole configuration selected for the prior steps (i.e., pre-processing, tokenization, and weighting steps) by means of a performance metric. Each particular configuration is dynamic and potentially changes during the next iterations. The accuracy value is a possible instance of the performance metric. In this particular case, and for each iteration, the optimal configuration possible gets represented by those parameters presenting the highest accuracy values that are selected by the combinatorial optimization algorithm from the parameters space.

Table 1 illustrates an example of some possible text transformations during the pre-processing and tokenization steps

<sup>1</sup> <https://microtc.readthedocs.io/en/latest/>

<sup>2</sup> <https://github.com/INGEOTEC/microtc>

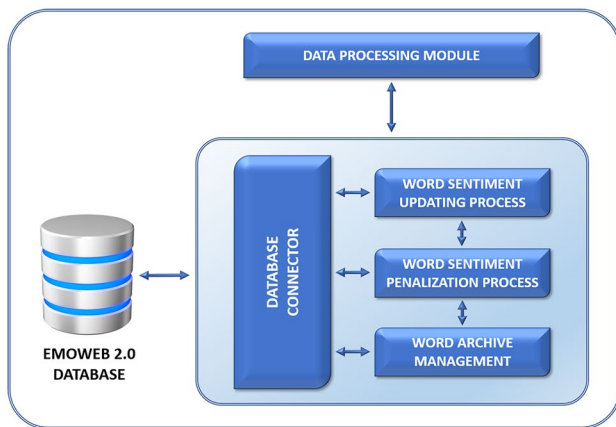


Fig. 3 Excerpt of the *Sentiment Evaluation Module* architecture

depending on the option selected. Weighting and classification steps are not shown as they do not apply any text transformation during their process.

It is worth highlighting that despite both *EmoWeb 2.0* and  $\mu$ TC are exposed to the same input tweet datasets, they follow different strategies during their respective pre-processing steps. In this manner, *EmoWeb 2.0* always applies the same procedure to pre-process incoming tweets (see 2). On the contrary,  $\mu$ TC relies on its internal combinatorial optimization algorithm to dynamically select the pre-processing tasks to be executed during the step depending on the input data.

### Coaching-based Active Learning Architecture

The coaching-based Active Learning architecture centers its efforts on improving the capabilities and overall performance of *EmoWeb 2.0* to evaluate the sentiment of tweets.

This enhancement translates into more accurate sentiment scores and, consequently, a better classification obtained for each processed tweet.

The architecture involves the use of an additional internal module named *Coaching Module* to control the interaction between *EmoWeb 2.0* and  $\mu$ TC (see Fig. 4).

*EmoWeb 2.0* is oriented to be exposed to training and a testing phase. In this context, the coaching activities occur just after the completion of the former and before the triggering of the latter. This strategy pursues reviewing the knowledge learned by the framework during its training phase in a way that corrections are applied where necessary (as part of the conversations held with the coach) to ensure that a better performance is obtained during the testing phase.

$\mu$ TC is used only as a consulting method and does not take any active role in the different activities. Moreover, before proceeding with the coaching activities,  $\mu$ TC (i.e., the coach) must process the same training tweet datasets as *EmoWeb 2.0*. This fact allows  $\mu$ TC to generate its own evaluations on the tweets under a scope and therefore, guarantees proper awareness of the knowledge to be reviewed.

Following training and a testing phase implicitly involves having reference tweet labels and sentiment scores provided by external experts to which *EmoWeb 2.0* and  $\mu$ TC performances can be compared. In this regard, the coach has to produce far superior accuracy results than the disciple for the same training tweet datasets. This circumstance formulates the core basis of the coaching process where one of the parties exhibits more expertise and a coaching session is established to discuss and eventually make a decision on the prevailing criteria.

Two control parameters are used to govern the process. Firstly, the *Start\_date* parameter is used to determine the first calendar day to be reviewed. Secondly, the *CN* parameter is

Table 1 Example of how texts are transformed by  $\mu$ TC.

Pre-processing step	
input text	The #covid19 makes meee crazy!!! :(, I read this http://siteurl
maintain hashtags	The #covid19 makes meee crazy!!! :(, I read this http://siteurl
remove urls	The #covid19 makes meee crazy!!! :(, I read this
lowercase	the #covid19 makes meee crazy!!! :(, i read this
remove punctuation	the #covid19 makes meee crazy :( i read this
remove duplicates	the #covid19 makes me crazy :( i read this
maintain emojis	the #covid19 makes me crazy :( i read this
Tokenization step	
input text	the #covid19 makes me crazy :( i read this
n-grams (n = 2)	the #covid19, #covid19 makes, makes me, me crazy, crazy :(, :( i, i read, read this
q-grams (q = 3)	the, he_, e_#, #co, cov, ovi, vid, id1, d19, 19_, 9_m, ...
skip-grams (2, 1)	the makes, #covid19 me, makes crazyo los y, me :(, ...



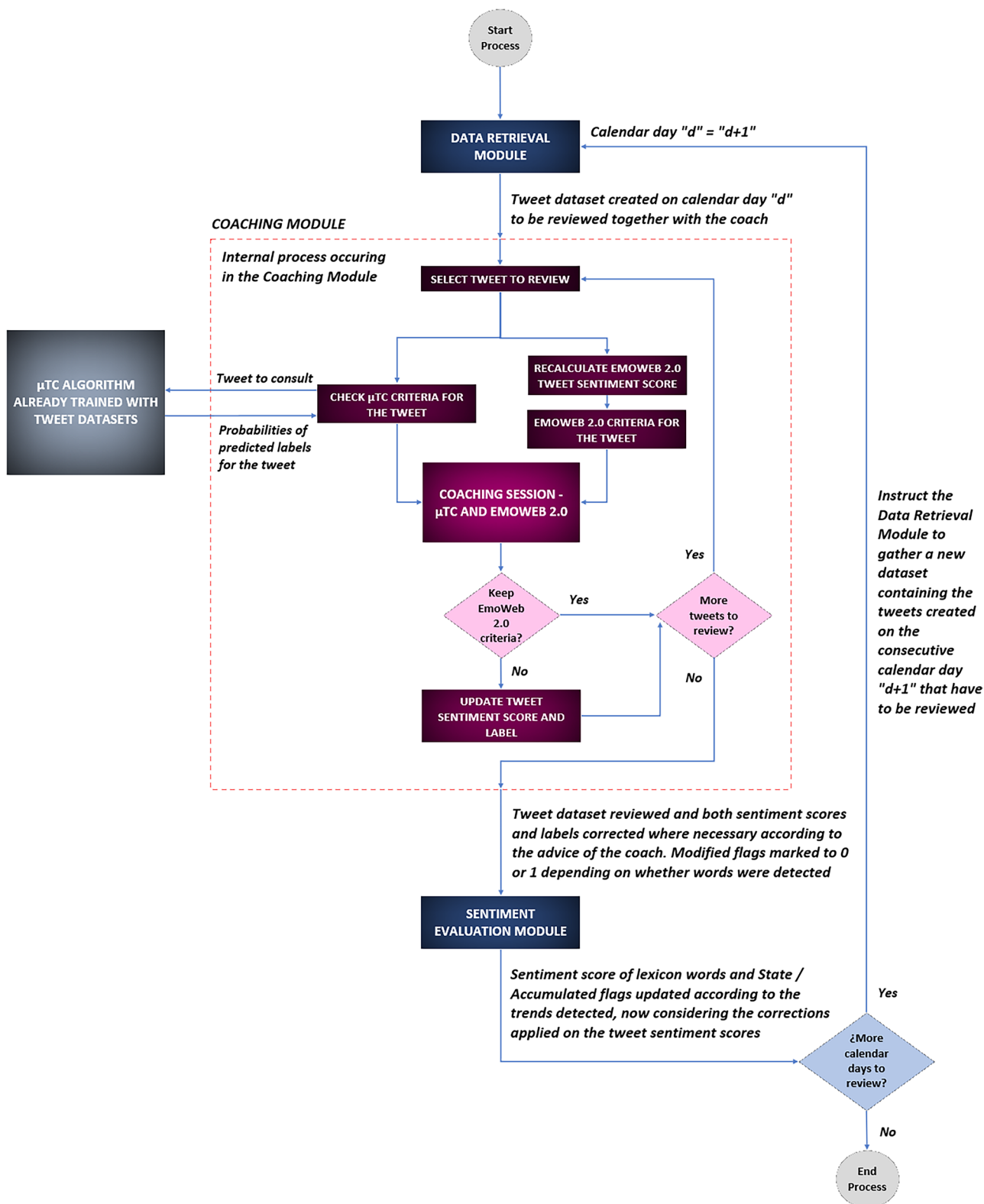


Fig. 4 Proposed architecture and major coaching activities

COACHING SESSION BETWEEN  $\mu$ TC AND EMOWEB 2.0

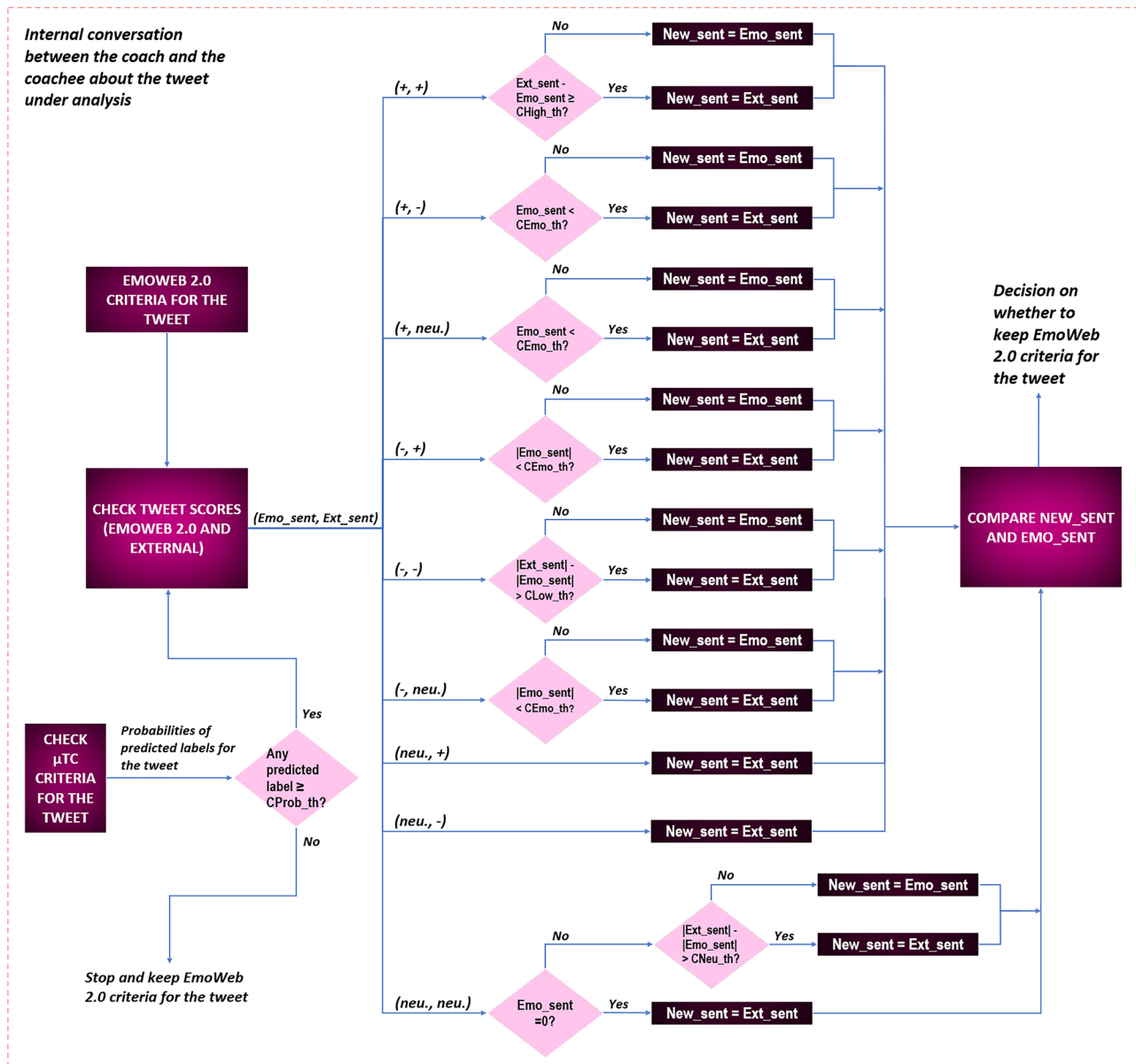


Fig. 5 Coaching session and internal process followed

used to establish the number of consecutive calendar days to be analyzed starting from the fixed *Start\_date*. Consequently, this parameter also defines the number of times (i.e., by default one per day to detect possible trends and the daily evolution of words) that the whole cycle illustrated in Fig. 4 is repeated.

Once the calendar day to be reviewed is selected, the *Data Retrieval Module* gathers the corresponding tweet dataset and handles it to the *Coaching Module*. The coaching activities are then started to analyze every tweet belonging to it. After selecting one tweet, the  $\mu$ TC criteria is consulted to retrieve the probabilities of the classification labels for it.

In parallel to that, *EmoWeb 2.0* performs a recalculation of the tweet score before actually offering it to the next step.

Next, the coaching session is established. The session emulates a conversation between  $\mu$ TC and *EmoWeb 2.0* where both share their thoughts about the tweet under analysis. Figure 5 details the internal actions taking place.

The session is controlled by five parameters. The *CProb\_th* parameter indicates the minimum probability required to the coach predictions to start the coaching session. In case the coach does not present enough self-certainty, the session is omitted and the *EmoWeb 2.0* score and label for the tweet are kept. On the contrary,

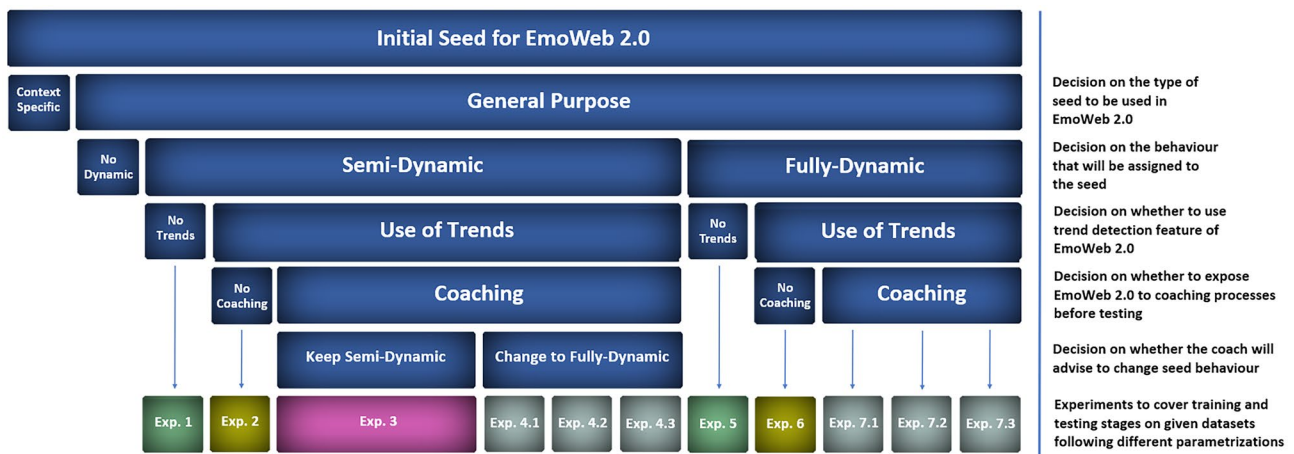


Fig. 6 Complete set of experiments performed with *EmoWeb 2.0*

the session starts and a comparison between the scores provided by *EmoWeb 2.0* (i.e., *Emo\_sent*) and the external experts (i.e., *Ext\_sent*) is performed by using several parameters. The parameter *C<sub>Emo\_th</sub>* is used when the scores show that *EmoWeb 2.0* and the coach differ in the classification label assigned to the tweet. In this case, a high resistance of *EmoWeb 2.0* to change its opinion is represented when the parameter is set to a low value. The other three parameters (i.e., *C<sub>High\_th</sub>*, *C<sub>Low\_th</sub>* and *C<sub>Neu\_th</sub>*) are used when both consider the same classification label for the tweet and define the boundaries from which *EmoWeb 2.0* would agree on incorporating the opinions provided by the coach. In this case, the greater the parameters are, the harder it becomes to convince the disciple (in other words, the coach requires to be very confident of its assessments).

The resulting outcome of the coaching session is the decision on whether the criteria provided by *EmoWeb 2.0* prevails. In the positive case, the process continues to select the next tweet to be examined. Contrarily, the tweet score and the classification label are updated accordingly before triggering the selection of the next tweet to assess.

Once all the tweets belonging to the tweet dataset are inspected, then the tasks of the *Sentiment Evaluation Module* are initiated. This module is responsible for updating the sentiment scores of all the words stored in the lexicon which in this case takes into consideration the modification of the tweet scores.

Finally, if there are more calendar days to review (depending on the parameter *C<sub>N</sub>*), the *Data Retrieval Module* retrieves the corresponding tweet dataset and the whole cycle starts all over again.

## Experiments

This section addresses the execution of several experiments specifically designed to explore different configurations of *EmoWeb 2.0* when it is inserted in the coaching-based Active Learning architecture. The main goals are to illustrate how *EmoWeb 2.0* is capable of learning from the coach and also the best parameter configuration possible that leads to it. This learning process translates into an observable improvement in the tweet classification process of *EmoWeb 2.0*.

The following process has been achieved to complete the set of experiments. Firstly, the design of the experiments is developed (see [Experimental Design](#)). Next, the tweet datasets are selected and detailed (see [Tweet Dataset](#)). Once these steps conclude, the configurations required to give shape to the set of experiments are addressed (see [Parameter Configurations](#)). Finally, the obtained results are analyzed to evaluate the actual effectiveness of the coaching task (see [Experimental Results](#)).

## Experimental Design

Figure 6 illustrates a top-down diagram to conceptually situate and explain all the experimental design. The experiments get defined by a set of decisions to be made in sequential order regarding several aspects. A color scheme has been followed to classify the experiments with similar nature.

The first decision relates to the selection of the type of lexicon to be used. The next dichotomy refers to whether

**Table 2**  $\mu$ TC results (train and test)

Phase	Acc. (%)	F1-Score (%)			Precision (%)			Recall (%)		
		Pos.	Neu.	Neg.	Pos.	Neu.	Neg.	Pos.	Neu.	Neg.
Train	95.46	95.84	95.96	93.07	94.97	96.39	93.41	96.72	95.53	92.72
Test	84.07	85.82	85.67	74.15	86.37	83.88	79.26	85.27	87.54	69.67

the lexicon is treated as a semi-dynamic or a fully-dynamic entity. The first option divides the lexicon into a purely invariable static part which consists of the original words and the sentiment values provided by *SenticNet*, and a fully-dynamic section given by the words learned during the tweet processing. In contrast, the second option considers that all the sentiment values associated with the stored words in the lexicon can be modified over time.

Next, the use of trends is put into the spotlight. The intrinsic nature of *EmoWeb 2.0* involves the detection of trends and the use of one *State* flag per word. In case of not using trends, then the experiments 1 and 5 are defined.

The next decision refers to whether to insert *EmoWeb 2.0* in a coaching-based architecture. In the negative case, there are no more decisions to make and the experiments 2 and 6 get defined. On the contrary, as part of the coaching process, the coach may advise changing the lexicon nature from semi-dynamic to fully-dynamic which leads to another decision to be made on the last level (originating experiments 3, 4.1 to 4.3, and 7.1 to 7.3, depending on the case).

To summarize, a total number of 11 experiments were designed. The experiments 1, 2, 5, and 6 reflect those cases in which no coaching activities are performed. These experiments are addressed to evaluate the capabilities of *EmoWeb 2.0* when it works stand-alone and without any external support (i.e., the baseline model). Conversely, the experiments 3, 4.1 to 4.3, and 7.1 to 7.3 focus on observing the improvement in the performance of *EmoWeb 2.0* when it interacts with  $\mu$ TC and coaching activities are conducted.

Considering the above and observing Fig. 6, the experiment 2 establishes the reference to which compare the outcomes from experiments 3 and 4.1 to 4.3. Likewise, the baseline drawn by experiment 6 is to be compared with the results observed in experiments 7.1 to 7.3. These comparisons are addressed in [Experimental Results](#) and allow demonstrating the value and utility of the coaching activities.

## Tweet Dataset

Regarding the tweet datasets to be processed during the training and testing phases of the experiments, the data was downloaded from an ongoing project available on IEEE data port [54]. This project publishes an English tweet dataset per calendar day related to the ongoing COVID-19 pandemic. The tweets are retrieved by using specific keywords and hashtags to ensure proper connection to the subject of

interest [55]. The dataset only includes the tweet IDs and the sentiment scores computed by *TextBlob* [56] (scores into  $[-1, 1]$ ). Hydration tasks are required before processing to obtain the actual tweet texts and some other metadata.

A total amount of 91 days of tweets datasets (i.e., calendar days explored) were acquired for the training phase, leading to a total amount of 407, 834 tweets processed during this stage. In this set, *TextBlob* labeled 128, 942 tweets as positive (31.61%), 64, 970 tweets as negative (15.93%), and 213, 922 tweets as neutral (52.45%). As for the testing phase, 90 days of tweet datasets were collected, which led to 379, 644 tweets processed. In this case, 124, 922 tweets were positive (32.90%), 57, 903 tweets were negative (15.25%) and 196, 819 tweets were neutral (51.84%). Finally, regarding the coaching activities, 30 calendar days were reviewed together with the coach, which translated into 140, 429 tweets revalued.

In the case of the  $\mu$ TC performance for the training and testing phases, Table 2 presents the obtained results. The coach reaches a relevant accuracy of 95.46% and 84.07% for both phases, respectively.

## Parameter Configurations

Delving into the complete parameter configurations of the experiments, these are indicated in Table 3. The  $\alpha$  parameter is set to 0.4 in experiments 1, 2, 5, and 6 to provide an adequate balance between the former and the new knowledge acquired.

As for those experiments where coaching activities are involved, the parameter *CN* indicates a total of 30 calendar days to be reviewed in all cases. Firstly, the experiments 3, 4.1, and 7.1 represent the case where the disciple (*EmoWeb 2.0*) is relatively hard to convince during the coaching process and the coach ( $\mu$ TC) is requested to be confident about its opinions (simulated by fixing the *CProb<sub>th</sub>* parameter to 0.45). This case also considers that the disciple assimilates the information provided by the coach reasonably well. In this regard, the  $\alpha$  parameter is set to 0.7.

Secondly, the experiments 4.2 and 7.2 represent the case in which the disciple presents a slightly less resistance to be convinced during the coaching process (coaching thresholds are less demanding), but it shows difficulties to absorb the new knowledge. The  $\alpha$  parameter is set to 0.1 to indicate this circumstance.

**Table 3** Parameters used for each of the experiments performed

Exp.	Sentiment Evaluation Module				Coaching Module						
	$\alpha$	Upper_th	Lower_th	Days	CProb_th	CHigh_th	CNeu_th	CLow_th	CEmo_th	CN	
1	0.4	0.3	0.2	2	-	-	-	-	-	-	
2	0.4	0.3	0.2	2	-	-	-	-	-	-	
5	0.4	0.3	0.2	2	-	-	-	-	-	-	
6	0.4	0.3	0.2	2	-	-	-	-	-	-	
3	0.7	0.3	0.2	2	0.45	0.3	0.025	0.3	0.4	30	
4.1	0.7	0.3	0.2	2	0.45	0.3	0.025	0.3	0.4	30	
4.2	0.1	0.3	0.2	2	0.45	0.2	0.025	0.2	0.6	30	
4.3	0.1	0.3	0.2	2	0.1	0.05	0.025	0.05	0.9	30	
7.1	0.7	0.3	0.2	2	0.45	0.3	0.025	0.3	0.4	30	
7.2	0.1	0.3	0.2	2	0.45	0.2	0.025	0.2	0.6	30	
7.3	0.1	0.3	0.2	2	0.1	0.05	0.025	0.05	0.9	30	

Finally, the experiments 4.3 and 7.3 represent the case in which the disciple offers almost no resistance to incorporate the new opinions received and the coach itself is not required to be sure about its opinions (*CProb\_th* set to 0.1). The disciple equally lacks in transferring the new knowledge to the word level ( $\alpha$  set to 0.1).

## Experiment Results

Table 4 illustrates the results obtained for the experiments 2 and 6. The experiment 6 provided a better result, noting a 6.55% of improvement in the accuracy during the testing phase. This could be explained by the fact that the lexicon was configured as fully-dynamic. This fact allowed a better adaptation of the word sentiments to the trends detected, which translated into more accurate tweet scores and classification.

On the contrary, the experiments 1 and 5 did not offer relevant results. Since trends were not considered, *EmoWeb 2.0* classified every tweet as neutral after a short number of

calendar days of processing which completely determined the observed accuracy.

As for the coaching related experiments, Table 5 presents the obtained results. The coaching phase offered very good performance in all cases, reaching accuracy levels above 97%. This fact is motivated by the high expertise of the coach, being able to fulfill the conditions imposed by the *Coaching Module* parameters in almost all cases. As a result of this process, the metrics observed during the testing phase were improved (to compare these results with the ones obtained in experiments 2 and 6, see Table 4).

The experiments 4.1, 4.2, and 4.3 provided a respective accuracy improvement of 16.97%, 10.04%, and 16.03% when compared with the experiment 2 for the testing phase. The experiment 4.1 showed a good knowledge absorbing capacity thanks to setting the parameter  $\alpha$  to 0.7. In the experiment 4.3, the effect of setting the  $\alpha$  parameter to 0.1 was compensated by configuring permissive values for the coaching parameters.

The experiment 3 did not offer good results. Much of the knowledge provided by the coach was not considered due to

**Table 4** Results for experiments 2 and 6 (train and test)

Experiment 2										
Phase	Acc. (%)	F1-Score (%)			Precision (%)			Recall (%)		
		Pos.	Neu.	Neg.	Pos.	Neu.	Neg.	Pos.	Neu.	Neg.
Train	41.51	49.98	37.29	27.19	38.62	55.94	27.41	70.82	28.11	26.98
Test	42.34	51.12	37.91	27.45	40.19	55.92	19.17	70.54	28.78	27.63
Experiment 6										
Phase	Acc. (%)	F1-Score (%)			Precision (%)			Recall (%)		
		Pos.	Neu.	Neg.	Pos.	Neu.	Neg.	Pos.	Neu.	Neg.
Train	43.91	26.55	50.61	8.69	32.60	55.01	19.58	37.31	60.05	5.95
Test	48.89	26.45	65.70	10.66	33.9	52.13	17.41	5.08	88.83	7.71

**Table 5** Results for experiments 4.1 to 4.3 and 7.1 to 7.3 (training, coaching and testing)

Experiment 4.1										
Phase	Acc. (%)	F1-Score (%)			Precision (%)			Recall (%)		
		Pos.	Neu.	Neg.	Pos.	Neu.	Neg.	Pos.	Neu.	Neg.
Train	41.51	49.98	37.29	27.19	38.62	55.94	27.41	70.82	28.11	26.98
Coaching	97.37	96.85	97.77	97.15	94.86	99.25	96.74	98.94	96.33	97.56
Test	59.31	41.56	70.84	9.03	75.47	57.06	35.13	29.36	64.98	5.28
Experiment 4.2										
Phase	Acc. (%)	F1-Score (%)			Precision (%)			Recall (%)		
		Pos.	Neu.	Neg.	Pos.	Neu.	Neg.	Pos.	Neu.	Neg.
Train	41.51	49.98	37.29	27.19	38.62	55.94	27.41	70.82	28.11	26.98
Coaching	98.40	98.26	98.66	97.81	97.58	99.30	97.17	98.95	98.03	98.45
Test	52.38	46.67	62.57	6.28	42.87	59.2	28.89	52.22	66.84	3.59
Experiment 4.3										
Phase	Accuracy (%)	F1-Score (%)			Precision (%)			Recall (%)		
		Pos.	Neu.	Neg.	Pos.	Neu.	Neg.	Pos.	Neu.	Neg.
Train	41.51	49.98	37.29	27.19	38.62	55.94	27.41	70.82	28.11	26.98
Coaching	99.67	99.63	99.70	99.67	99.89	99.50	99.80	99.38	99.90	99.54
Test	58.37	45.67	69.25	5.98	65.65	58.50	32.44	40.04	86.33	3.34
Experiment 7.1										
Phase	Accuracy (%)	F1-Score (%)			Precision (%)			Recall (%)		
		Pos.	Neu.	Neg.	Pos.	Neu.	Neg.	Pos.	Neu.	Neg.
Train	43.91	26.55	50.61	8.69	32.45	55.01	19.58	37.31	60.05	5.95
Coaching	98.60	98.55	98.89	97.77	98.34	99.16	97.33	98.76	98.62	98.22
Test	59.06	42.16	70.78	8.97	74.80	56.99	35.65	29.95	64.88	5.25
Experiment 7.2										
Phase	Accuracy (%)	F1-Score (%)			Precision (%)			Recall (%)		
		Pos.	Neu.	Neg.	Pos.	Neu.	Neg.	Pos.	Neu.	Neg.
Train	43.91	26.55	50.61	8.69	32.45	55.01	19.58	37.31	60.05	5.95
Coaching	98.51	98.41	98.77	97.87	97.90	99.28	97.30	98.93	98.26	98.46
Test	57.67	43.38	68.95	6.28	65.00	57.71	30.36	36.82	60.05	3.56
Experiment 7.3										
Phase	Accuracy (%)	F1-Score (%)			Precision (%)			Recall (%)		
		Pos.	Neu.	Neg.	Pos.	Neu.	Neg.	Pos.	Neu.	Neg.
Train	43.91	26.55	50.61	8.69	32.45	55.01	19.58	37.31	60.05	5.95
Coaching	99.25	99.64	99.61	99.73	99.87	99.32	99.82	99.42	99.91	99.65
Test	59.75	52.54	72.94	7.68	67.43	62.52	34.67	43.04	87.53	4.32

having a semi-dynamic lexicon and therefore, a static invariant section on it. In consequence, the accuracy obtained during the testing phase was similar to the one given by the experiment 2.

Finally, with regard to the experiments 7.1, 7.2, and 7.3, the accuracy improvements observed were 10.17%, 8.78%,

and 10.86%, respectively, when compared with the experiment 6 for the testing phase. As before, the parameter  $\alpha$  controlled the assimilation capacity of *EmoWeb 2.0*. The use of a fully-dynamic lexicon from the beginning led to less margin for accuracy improvement, especially when compared to the ones obtained in experiments 4.1 to 4.3.

Overall, the improvements observed validate the proposal and clearly indicate that the coaching activities are positive to better prepare *EmoWeb 2.0* for the testing phase. It is also relevant to note that despite the different parameter configurations chosen for the *Coaching Module* and the *Sentiment Evaluation Module*, a performance improvement has been observed in all cases. The use of a fully-dynamic lexicon seems to be the preferred option to allow the whole lexicon to adapt its sentiment values to the trends detected in the domain and also to effectively acquire the knowledge provided by the coach regardless of the internal parameter configurations. For its part, having a good coach also plays a crucial role to ensure that the disciple learns the correct information.

## Conclusions

This paper has presented a novel coach architecture based on Active Learning. The proposal includes the use of a ML framework (called  $\mu$ TC) acting as a coach and a dynamic sentiment framework based on a dictionary (called *EmoWeb 2.0*) which plays the role of a disciple. Both systems work together, providing their different perspectives and strengths on the analyzed textual content.

The proposed architecture has been used to improve the capabilities of a dictionary-based framework to evaluate sentiment values of textual contents in a specific and fickle domain. Notice that these kinds of systems usually present difficulties to be adapted to new contexts due to the peculiarities of the dictionaries since they only consider the lexical level of the language (i.e., individual words and their associated sentiment value). However, the architecture is specially designed to overcome this issue by transferring knowledge from the semantic level of the language (i.e., the context of complete sentences) to the dictionary-based system.

A changing and specific domain like the COVID-19 outbreak has been selected to perform multiple experiments. Several parameter configurations have been considered when *EmoWeb 2.0* gets inserted in the coaching-based architecture. Different learning capabilities have been detected depending on the settings. Promising results have been obtained proving the positive effects of the coaching tasks and the overall viability of the proposal. Thus, a relevant improvement in the ability of the sentiment framework to learn new words and adjust their sentiment values has been confirmed, leading likewise to enhancing its tweet classification capabilities.

Finally, the use of a fully-dynamic lexicon has shown to be the optimal configuration for the proposed architecture. It eased the acquisition of new knowledge provided by the coach. Moreover, this solution allowed *EmoWeb 2.0* to

adapt the sentiment values of words to the trends detected in the domain.

In the future, several research lines could be explored to improve the proposal. In the first place, the use of context-specific lexicons as initial seeds would confer a better adaptation to the subject of interest and surely better classification results in all the phases. Secondly, the use of coaches based on multi-label and transfer learning approaches such as BERT [29] could also be interesting to consider. At this point, the idea of having a council of coaches providing advice could be examined. Thus, the council should reach a consensus on their criteria and duly inform the disciple about it. Lastly, the ambivalence of words in a specific context could be an interesting issue to address since the proposed architecture is able to detect the dynamic fluctuations of the transmitted emotions over time.

**Acknowledgements** Research supported by grants from the Spanish Ministry of Economy and Competitiveness under the Retos-Colaboración program: SABERMED (Ref: RTC-2017-6253-1), Retos-Investigación program: MODAS-IN (Ref: RTI-2018-094269-B-I00); and donation of the Titan V GPU by NVIDIA Corporation.

## Declarations

**Research Involving Human and Animal Participants** This article does not contain any studies with human participants or animals performed by any of the authors.

**Conflict of Interest** All the authors declare not to have any conflicts of interest or ethical issues.

## References

- Schiffer MB. The material life of human beings: artifacts, behavior and communication. Routledge; 2002.
- Broadbent DE. Perception and communication. Elsevier; 2013.
- Conquergood D. Beyond the text: Toward a performative cultural politics. The future of performance studies: Visions and revisions; 1998. p. 25–36.
- Gruzitis N, Barzdins G. Polysemy in controlled natural language texts. In: International Workshop on Controlled Natural Language, Springer. 2009. pp 102–120.
- Damasio AR. Emotions and feelings. In: Feelings and emotions: The Amsterdam symposium, Cambridge University Press Cambridge. 2004. pp 49–57.
- Liu B, et al. Sentiment analysis and subjectivity. Handbook of Natural Language Processing. 2010;2(2010):627–66.
- Yue L, Chen W, Li X, Zuo W, Yin M. A survey of sentiment analysis in social media. Knowl Inf Syst. 2019;60(2):617–63.
- Lukes J, Sjøgaard A. Sentiment analysis under temporal shift. In: Proceedings of the 9th workshop on computational approaches to subjectivity, sentiment and social media analysis. 2018. pp 65–71.
- de Diego IM, Fernández-Isabel A, Ortega F, Mogueza JM. A visual framework for dynamic emotional web analysis. Knowl-Based Syst. 2018;145:264–73.

10. Tellez ES, Moctezuma D, Miranda-Jiménez S, Graff M. An automated text categorization framework based on hyperparameter optimization. *Knowl-Based Syst.* 2018;149:110–23.
11. Guimarães N, Torgo L, Figueira Á. Twitter as a source for time- and domain-dependent sentiment lexicons. In: *Social Network Based Big Data Analysis and Applications*, Springer. 2018. pp 1–19.
12. Zong S, Baheti A, Xu W, Ritter A. Extracting covid-19 events from twitter. 2020. arXiv preprint arXiv:200602567.
13. Akerkar R, Sajja P. *Knowledge-based systems*. Jones & Bartlett Publishers. 2009.
14. Sajja PS, Akerkar R. Knowledge-based systems for development. *Advanced Knowledge Based Systems: Model, Applications & Research*. 2010;1:1–11.
15. Chow PK, Yeung DS. A multidimensional knowledge structure. *Expert Systems with Applications*. 1995;9(2):177–87.
16. Liebowitz J. *The handbook of applied expert systems*. cRC Press. 2019.
17. Reyes-Garcia E, Bouhai N. *Designing Interactive Hypermedia Systems*. Wiley Online Library. 2017.
18. Khazanchi R, Khazanchi P. Artificial intelligence in education: A closer look into intelligent tutoring systems. In: *Handbook of Research on Critical Issues in Special Education for School Rehabilitation Practices*, IGI Global. 2021. pp 256–277.
19. Saridakis KM, Dentsoras AJ. Soft computing in engineering design—a review. *Adv Eng Inform.* 2008;22(2):202–21.
20. Narang R. *Database management systems*. Ltd: PHI Learning Pvt; 2018.
21. Ravi K, Ravi V. A survey on opinion mining and sentiment analysis: tasks, approaches and applications. *Knowl-Based Syst.* 2015;89:14–46.
22. Scherer KR, et al. On the nature and function of emotion: A component process approach. *Approaches to emotion*. 1984. pp 293–317.
23. Baccianella S, Esuli A, Sebastiani F. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. *Lrec*. 2010;10:2200–4.
24. Cambria E, Li Y, Xing FZ, Poria S, Kwok K. Senticnet 6: Ensemble application of symbolic and subsymbolic ai for sentiment analysis. In: *Proceedings of the 29th ACM international conference on information & knowledge management*. 2020. pp 105–114.
25. Ahmad M, Aftab S, Muhammad SS, Ahmad S. Machine learning techniques for sentiment analysis: A review. *Int J Multidiscip Sci Eng.* 2017;8(3):27.
26. Prabowo R, Thelwall M. Sentiment analysis: A combined approach. *J Inform.* 2009;3(2):143–57.
27. Li W, Zhu L, Cambria E. Taylor’s theorem: A new perspective for neural tensor networks. *Knowl-Based Syst.* 2021;228:107258.
28. Dos Santos C, Gatti M. Deep convolutional neural networks for sentiment analysis of short texts. In: *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. 2014. pp 69–78.
29. Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. 1810.04805. 2019.
30. Usama M, Ahmad B, Song E, Hossain MS, Alrashoud M, Muhammad G. Attention-based sentiment analysis using convolutional and recurrent neural network. *Futur Gener Comput Syst.* 2020;113:571–8.
31. Liu N, Shen B, Zhang Z, Zhang Z, Mi K. Attention-based sentiment reasoner for aspect-based sentiment analysis. *Human-centric Computing and Information Sciences*. 2019;9(1):1–17.
32. Keyvanpour M, Zandian ZK, Heidarypanah M. OMLML: a helpful opinion mining method based on lexicon and machine learning in social networks. *Soc Netw Anal Min.* 2020;10(1):1–17.
33. Lu B, Tsou BK. Combining a large sentiment lexicon and machine learning for subjectivity classification. In: *2010 international conference on machine learning and cybernetics*, IEEE. 2010. vol 6, pp 3311–3316.
34. Cambria E. Affective computing and sentiment analysis. *IEEE Intell Syst.* 2016;31(2):102–7.
35. Stappen L, MeBner EM, Cambria E, Zhao G, Schuller BW. Muse 2021 challenge: Multimodal emotion, sentiment, physiological-emotion, and stress detection. In: *Proceedings of the 29th ACM International Conference on Multimedia*. 2021. pp 5706–5707.
36. Dragoni M, Donadello I, Cambria E. Ontosenticnet 2: Enhancing reasoning within sentiment analysis. *IEEE Intell Syst.* 2022;37(1):1–8.
37. Liang B, Su H, Gui L, Cambria E, Xu R. Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. *Knowl-Based Syst.* 2022;235: 107643.
38. Ebrahimi M, Yazdavar AH, Sheth A. Challenges of sentiment analysis for dynamic events. *IEEE Intell Syst.* 2017;32(5):70–5.
39. Wang Z, Ho SB, Cambria E. Multi-level fine-scaled sentiment sensing with ambivalence handling. *Int J Uncertainty Fuzziness Knowledge Based Syst.* 2020;28(04):683–97.
40. Ma L, Ding B, Das S, Swaminathan A. Active learning for ml enhanced database systems. In: *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. 2020. pp 175–191.
41. Melville P, Mooney RJ. Diverse ensembles for active learning. In: *Proceedings of the twenty-first international conference on Machine learning*. 2004 p 74.
42. Muslea I, Minton S, Knoblock CA. Active learning with multiple views. *J Artif Intell Res.* 2006;27:203–33.
43. Wei XY, Yang ZQ. Coaching the exploration and exploitation in active learning for interactive video retrieval. *IEEE Trans Image Process.* 2012;22(3):955–68.
44. Garvey R, Garvey B, Stokes P, Megginson D. *Coaching and mentoring: Theory and practice*. Sage. 2017.
45. Saif H, He Y, Fernandez M, Alani H. Adapting sentiment lexicons using contextual semantics for sentiment analysis of twitter. In: *European Semantic Web Conference*. Springer; 2014. pp 54–63.
46. Song Y, Gu K, Li H, Sun G. A lexical updating algorithm for sentiment analysis on chinese movie reviews. In: *2017 Fifth International Conference on Advanced Cloud and Big Data (CBD)*, IEEE; 2017. pp 188–193.
47. Keshavarz H, Abadeh MS. Alga: Adaptive lexicon learning using genetic algorithm for sentiment analysis of microblogs. *Knowl-Based Syst.* 2017;122:1–16.
48. Park S, Lee W, Moon IC. Efficient extraction of domain specific sentiment lexicon with active learning. *Pattern Recogn Lett.* 2015;56:38–44.
49. Asghar MZ, Ahmad S, Qasim M, Zahra SR, Kundi FM. Sentihealth: creating health-related sentiment lexicon using hybrid approach. *SpringerPlus.* 2016;5(1):1–23.
50. Elshakankery K, Ahmed MF. Hilatsa: A hybrid incremental learning approach for arabic tweets sentiment analysis. *Egyptian Informatics Journal.* 2019;20(3):163–71.
51. Rubtsova Y. Reducing the deterioration of sentiment analysis results due to the time impact. *Information.* 2018;9(8):184.
52. Cabezas J, Moctezuma D, Fernández-Isabel A, Martín de Diego I. Detecting emotional evolution on twitter during the covid-19 pandemic using text analysis. *Int J Environ Res Public Health.* 2021;18(13):6981.
53. Kumar V, Racharla K, Majumdar D. Transfer learning with augmented vocabulary for tweet classification:(grand challenge).



In: 2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM), IEEE; 2020. pp 471–475.

54. Lamsal R. Coronavirus (covid-19) tweets dataset. 2020. <https://dx.doi.org/10.21227/781w-ef42>, [Online: accessed 1-Aug-2021].
55. Lamsal R. Design and analysis of a large-scale covid-19 tweets dataset. *Appl Intell.* 2021;51(5):2790–804.

56. Loria S. Textblob documentation Release. 2018;015(2):269.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.