IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 16, 2021, accepted February 22, 2021, date of publication March 4, 2021, date of current version March 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3064000

Active Learning for Biomedical Text Classification Based on Automatically Generated Regular Expressions

CHRISTOPHER A. FLORES, ROSA L. FIGUEROA, AND JORGE E. PEZOA, (Member, IEEE)
Department of Electrical Engineering, Universidad de Concepción, Concepción 4070409, Chile

Corresponding author: Christopher A. Flores (christopher.flores@biomedica.udec.cl)

This work was supported in part by the National Center on Health Information Systems (CORFO PROJECT- CENS) under Grant 16CTTS-66390, in part by the CONICYT-PFCHA/Doctorado Nacional under Grant 2017-21172062, in part by the Universidad de Concepción UCO 1866 Project, in part by the Universidad de Concepción VRID-Enlace Project under Grant 217.092.052-1.0, and in part by the FONDEF IDeA I+D under Grant ID19I10120.

ABSTRACT Biomedical text classification algorithms, which currently support clinical decision-making processes, call for expensive training texts due to the low availability of labeled corpus and the cost of manual annotation by specialized professionals. The active learning (AL) approach to classification heavily lessens such cost by reducing the number of labeled documents required to achieve specified performance. This article introduces a query strategy and a stopping criterion that transform CREGEX, a regular-expressions-based text classification algorithm, in an AL biomedical text classifier. The query strategy samples the training dataset, trading off the greedy learning achieved by the regular expressions classification precision and the conservative learning induced by text sequence alignment classification. The sustained reduction in the variance of the query strategy scores is used as a stopping criterion. The AL classifier was compared with Support Vector Machine (SVM), Naïve Bayes (NB), and a classifier based on Bidirectional Encoder Representations from Transformers (BERT), using three datasets with biomedical information in Spanish on smoking habits, obesity, and obesity types. The learning curve results indicate that AL in CREGEX allowed to efficiently reduce the number of training examples for equal performance than the rest of the classifiers, obtaining areas under the learning curve greater than 85% in all cases. The stopping criterion applied to the AL process allowed to use, on average, approximately 32% to 50% of the total training examples with differences in performance concerning the maximum value of the learning curve not exceeding 2%. This performance demonstrates the effectiveness of using AL in a biomedical text classifier based on regular expressions, which is attributable to such expressions' ability to represent intricate sequential patterns in training texts considered most informative.

INDEX TERMS Active learning, regular expressions, natural language processing, text classification.

I. INTRODUCTION

Text classification has become one of the most widely used machine learning techniques to organize the growing accumulation of unstructured digital information [1]–[3]. Classification algorithms such as Support Vector Machine (SVM) and Naïve Bayes (NB) have been extensively used due to the simplicity of their implementation, and the accurate results obtained [4]. More recently, the use of pre-trained language models, based on deep neural networks (DNN), has transformed the field of natural language processing (NLP). In this sense, the Bidirectional Encoder Representations from Transformers (BERT) algorithm has become state-of-the-art in many NLP tasks. [5]–[7].

Regardless of the classification algorithm used, correctly labeled training texts are needed. The problem for many applications, such as in the case of biomedicine, is that the cost of manually labeling training examples may become prohibitive. Time, resources, and specialized annotators are needed to carry out the labeling tasks [8]. In this scenario, the active learning (AL) approach to classification offers an alternative to reducing annotation efforts. The AL aims to

The associate editor coordinating the review of this manuscript and approving it for publication was Hong-Mei Zhang.

facilitate algorithms to obtain performance comparable to passive learning (PL) or random sampling but with fewer training examples [9].

In text classification, the most widely used approach is the pool-based AL where training examples are selected from an unlabeled dataset [10]. A taxonomy for the different pool-based AL algorithms can be created in terms of the query strategy, or sample selection function, defined to pick the most informative examples for SVM- and NB-based classifiers [11]–[14]. Recently, the AL has attracted the interest of researchers and has been applied to classification algorithms based on DNN [15], [16]. However, to our best knowledge, there are no AL query strategies available for identifying the most informative examples for regular-expressions—based biomedical text classifiers, with only some works related to information extraction tasks but in other usage domains [17]–[21]. Based on the above, in this paper we aim to address the following research questions:

- For a given biomedical text classification algorithm based on regular expressions, can active learning effectively reduce the number of training examples needed to obtain the same performance as passive learning?
- For a given biomedical text classification algorithm based on the regular expressions, can active learning perform better than other active learning methods for selecting the most informative examples?

This work proposes an AL query strategy and a stopping criterion that allows identifying the most informative examples for a biomedical text classifier based on regular expressions. The advantages of the proposed method are that regular expressions are easily analyzable at a natural language level by a domain expert and represent complex sequential patterns in texts, including numerical attributes, unlike existing methods [18], [22]–[24]. The AL query strategy samples the training dataset trading off the greedy learning achieved by the regular expressions classification precision and the conservative learning induced by text sequence alignment classification. More precisely, the greedy AL query strategy exploits the regular expressions' classification performance during training to assess the level of uncertainty of the selected examples. The conservative AL query strategy assesses the amount of diversity in the examples through the Smith-Waterman (SW) algorithm to provide a level of uncertainty in cases where regular expressions mismatch. Three datasets written in Spanish were used to evaluate whether the AL decision function effectively achieves the same classification performance when used in conjunction with the Classifier Regular Expression (CREGEX) biomedical text discriminant [21]. Such datasets were obtained from the hospital Guillermo Grant Benavente (HGGB) in Concepción, Chile.

The performance of the AL version of CREGEX was compared with the classification results of NB, SVM, and BERT in terms of accuracy (ACC), precision (P), recall (R), and F-measure (F1). For comparison, distances to the hyperplane and cosine similarity were used as the AL query strategy of the most informative examples in the case of the SVM classifier [13], [14]. For NB and BERT classifiers, the criterion of maximum entropy was considered as the AL query strategy [11]. To calculate BERT's prediction probabilities, the Monte Carlo Dropout method was used as an approximation to a Bayesian inference [15]. The classification results indicate that the AL version of CREGEX performed better than SVM and NB in all datasets for accuracy (ACC), precision (P), recall (R), and F-measure (F1) metrics. In comparison to BERT, CREGEX was better in most cases. On the other hand, the use of AL reduced in most cases the number of training examples needed to obtain the same performance as PL. In this sense, the AL version of CREGEX proved to be the most efficient classifier. Accordingly, the main contributions of this work can be summarized as follows:

- A query strategy for the AL process of a biomedical text classifier based on the automatic generation of regular expressions.
- A stopping criterion for the AL process of classification algorithms.

The rest of this work is organized as follows: Section II presents a review of the AL methods used in classification tasks. The section III describes the datasets, the CREGEX classifier and the AL query strategy implemented. The section IV presents the performance of the classifiers in terms of ACC, P, R, F1, and learning curves. Section V shows an analysis of the performance of the classifiers and AL, as well as future work.

II. RELATED WORK

The training of the classification algorithms involves having enough labeled texts correctly. However, the cost of manually labeling training examples can be highly expensive (time, resources, annotators), especially in the biomedical area where specialized annotators are required [8]. To address this problem, the AL provides an alternative to reduce the number of examples needed to train the classification algorithms [25]. In contrast to the PL where training examples are selected randomly, the AL allows a greater degree of control of the chosen examples, depending on the automatic learning algorithm used [9].

The most used approach in text classification is the pool-based AL, which selects training examples from a large unlabeled dataset [10]. In this approach, three datasets are defined: the unlabeled dataset X_U, an initial training dataset (X_I, Y_I), and the test set (X_T). The AL process consists of an initialization stage and a selection phase of the most informative examples [26]. During the initialization stage, one or more examples per class are randomly selected from the unlabeled dataset X_U and then labeled by an expert E to form an initial training set (X_I, Y_I). Afterward, a query strategy $q(\cdot)$ is used to progressively select n_q examples X_q considered most informative by the classifier. Selected examples X_q are labeled by a domain expert E and added to the training set to re-train the classifier until a stopping criterion is met. In each

iteration, selected examples X_q are removed from X_U. Finally, the test set is used to evaluate the classifier's effectiveness during the selection of training examples.

Several methods have been proposed to progressively select from an unlabeled data set the examples considered most informative for a machine learning algorithm. Lewis and Gale proposed the method called Uncertainty Sampling in which a probabilistic classifier is used to select the examples with maximum entropy [11], [12]. Another selection strategy, called Query-by-Committee, was proposed by Seung *et al.*, where multiple classification algorithms to select examples with the most significant degree of disagreement among committee members [27]. Yet another selection method proposed by Tong and Koller is called Simple Margin and uses as a selection criterion the examples with the shortest distance to the separation hyperplane of the classes of a SVM [13]. Subsequently, Brinker incorporates cosine similarity to the query strategy of SVM to provide more diversity to the selected examples [14]. Other methods use clustering techniques to select examples from the unlabeled dataset [28]–[30]. However, the main disadvantage of these methods is that the performance of the AL process is dependent on the quality of the groups formed by the clustering algorithm used. More recent methods attempt to estimate the uncertainty of predictions in classification algorithms based on neural networks [15], [16]. For example, Gal proposes the Monte Carlo Dropout method, which uses the neural network regularization technique known as *dropout* as an approximation to Bayesian inference [15]. A determined number of predictions (probabilities) are carried out in the same example to measure the uncertainty, which is averaged and analyzed with some statistical metrics.

On the other hand, an essential aspect to consider in AL is determining a criterion for stopping the learning process. Some of the stopping criteria analyze the cost of obtaining new labels, set a maximum performance value for the classifier or training sample size, or analyze the quality of the examples in the datasets [12], [31]–[35]. One approach to a stopping criteria method was proposed by Bloodgood and Vijay-Shanker considering the unlabeled dataset [34] The method tests the new models obtained in consecutive iterations of AL in a separate dataset without labels to check if the predictions have stabilized. The measure used is the level of agreement in the predictions between the consecutive models. On the other hand, Vlachos proposes to analyze the performance of the classifier on an additional dataset until a consistent decrease in this performance is observed during the learning process [31]. In the case of considering the selected training examples, Ghayoomi proposes to analyze the variance of the scores obtained from the query strategy in each iteration [33]. This method relies on the fact that, at the beginning of the learning process, the classifier is not sufficiently trained so that the results of the query strategy will not present a high level of variability. As the training set increases, the classifier changes from untrained to trained, resulting in increased variability in the results of the query strategy. Finally, once the classifier is sufficiently trained, the results of the query strategy become close to the mean with a low level of variability. In this last stage, the process of AL must be stopped.

III. MATERIALS AND METHODS

A. DATASETS AND PRE-PROCESSING

Biomedical texts in Spanish from the HGGB, Concepción, Chile, were used as datasets after approval by the ethics committee. The biomedical texts contain information regarding binary (obesity and smoking habits) and multiclass (types of obesity) problems [21]. A further description of the texts can be found in Table 1. Finally, the texts were pre-processed by converting them to lower cases and removing unnecessary white-spaces to facilitate the extraction of tokens (words, numbers, and symbols).

TABLE 1. Description of the datasets.

Dataset	Keywords	Classes	Number of examples	Kappa index
OBESITY STATUS	obes* (obesity), imc (bmi), peso (weight), sobrepeso (òverweight)	Positive (obesity), negative	1161	$0.98^{(a)}$
OBESITY TYPES	obes* (obesity), imc (bmi), peso (weight), sobrepeso (òverweight)	Moderate, severe, morbid	909	$0.97^{(a)}$
SMOKING STATUS	tab* (tobacco), fum* (smoker), cig*(cigarette), caj* (cigarette box)	Positive (smoker), negative	1087	$0.86^{(b)}$

$^{(a)}$Cohen's Kappa index. $^{(b)}$Fleiss' Kappa index.
*Keyword root.

B. PROBLEM DEFINITION

We formally define now the problem tackled in this paper. Following [21], consider a collection of n training biomedical texts that are labeled using $l \geq 2$ classes (binary or multiclass problems) in a supervised manner. More formally, the set of training texts is denoted as $X = \{x_1, \ldots, x_n\}$, the set of labels are denoted by the set $L = \{1, \ldots, l\}$, and the supervised labeling process is represented by the mapping $\mathcal{L} : X \rightarrow L$, which for each $x_i \in X$ creates a label $y_i = \mathcal{L}(x_i)$, $y_i \in L$. The collection of all labeled training texts is denoted as $Y = \mathcal{L}(X)$. Besides, to assess the performance of the proposed AL method, a separate labeled dataset, X_T, containing n_t test biomedical texts will be used.

The proposed AL version of CREGEX is divided, as depicted in Figure 1, into two stages: (i) the construction of a feature space based on regular expressions; and (ii) the definition of an AL-based classifier for biomedical texts. First, a feature space for X is automatically constructed using the bijective mapping $\Phi(x_i) : X \rightarrow R_i \subseteq R$, which generates n_i regular expressions for the training text x_i, labeled as y_i, where $R_i = (r_1^i(x_i), \ldots, r_{n_i}^i(x_i))$. Thus, once the mapping function $\Phi(\cdot)$ is applied to the entire training set X, it generates the

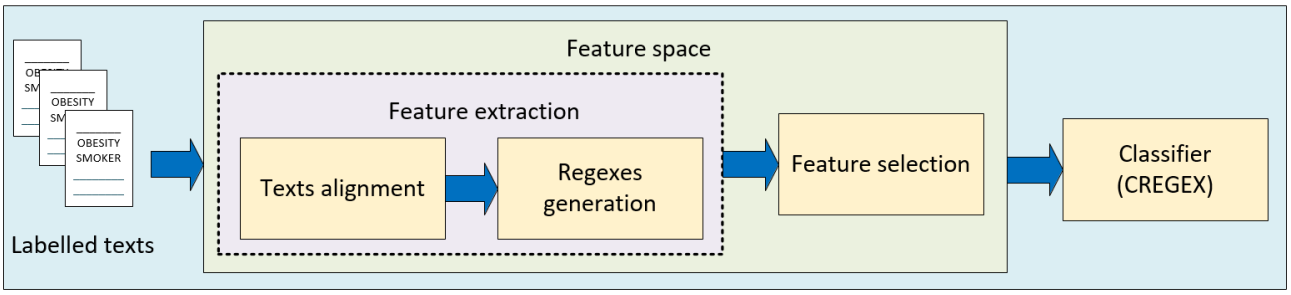

FIGURE 1. Functional scheme of the proposed AL classifier based on automatically generated regular expressions (regexes).

collection $R = \cup_{i=1}^{n} R_i$, which contains a total number of $|R|$ regular expressions representing the set of biomedical texts. Subsequently, a feature selection step is carried out, filtering out regular expressions by keywords and evaluating them to obtain a performance measure during the training set. Next, the resulting regular expressions are assigned to the class of the training text where they were automatically generated. Thus, the regular-expressions–based classifier assigns the class y_i to a sample text $x_i \in X_T$ through the decision function $\delta(x_i) : X_T \rightarrow L$, where $y_i = \delta(x_i)$.

In addition, we define the AL dataset $D_{AL} = X \cup X_U$, where X contains n training texts, whose collection of labeled examples is given by $Y = \mathcal{L}(X)$, and X_U is a set of u unlabeled texts. The aim of the AL is to iteratively construct the set $X_q \subseteq X_U$, which contains the most informative examples as far as the employed machine learning algorithm is concerned. To assess the amount of information in each text in X_U, a score function assigns the value $u_i, i = 1, \ldots, |X_U|$, through the query strategy $q(x_i)$. Thus, at each iteration, the subset X_q is labeled by a human expert in D_{AL} so that X, Y, and X_U are updated as: $X \leftarrow X \cup X_q$, $Y \leftarrow Y \cup \mathcal{L}(X_q)$, and $X_U \leftarrow X_U \setminus X_q$.

1) CONSTRUCTION OF A FEATURE SPACE BASED ON THE AUTOMATIC GENERATION OF REGULAR EXPRESSIONS

The construction of a feature space based on the automatic generation of regular expressions is mainly carried out using the Needleman-Wunsch (NW) and SW alignment algorithms, considering a previous work of the authors of this article [21]. Firstly, hierarchical clustering is applied to the training texts to construct similar word groups, excluding verbs. To do so, the Levenshtein distance, with a cut-off threshold equal to four, was used. We note that such value was determined based on an exploratory data analysis. Secondly, the NW algorithm is applied within the clusters to align common letters and compute non-common ones (maximum number of letters). Thus, as shown in Figure 2, we note that it is possible to represent the word groups by a single representative pattern.

Then, training texts are normalized by replacing similar words with the previously found common patterns. Numbers are also replaced with a pattern representing numerical intervals. This numerical pattern allows us to capture decimal and integer numbers considering a range equal to five, regarding the levels of obesity according to the Body mass index (BMI). As shown in Figure 3, once the texts have been processed,

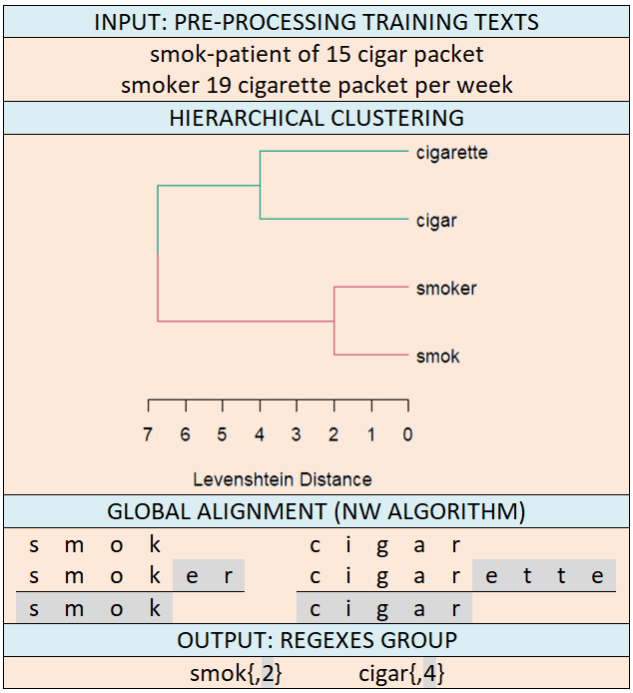

FIGURE 2. An example of the global alignment using the NW algorithm.

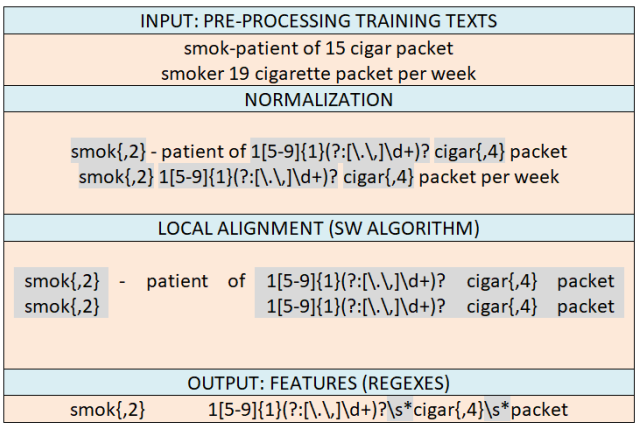

FIGURE 3. An example of the local alignment using the SW algorithm.

the SW algorithm is applied to the texts belonging to the same class to extract representative token sequences. The SW algorithm is used because, unlike the NW algorithm, it is

more appropriate to analyze sequences with different lengths, as is the case with biomedical texts [18].

After text alignment is carried out, the proposed method generates regular expressions replacing white-spaces with the meta-character "\s*" (zero or more spaces) and inserting a backslash ("\") in the non-alphanumeric characters. Finally, the regular expressions are assigned to the same class of the training text generating them. Subsequently, for each classification problem, a feature selection method based on keyword filtering is applied to the feature space constructed using regular expressions. Here, keywords in Table 1 were used to filter out the regular expressions that do not contain information relevant to the classification problem, which were obtained during the biomedical texts' annotation process. Finally, each regular expression is evaluated in the training set to obtain a confidence metric (precision score).

2) THE CREGEX CLASSIFIER

The automatically generated regular expressions can be used for classifying the test biomedical text x_i and, depending on the number of matches, assigns the class y_i. More precisely, two possible scenarios may arise for the CREGEX classifier: (i) no regular expressions match x_i, (ii) n_r regular expressions match x_i. If no regular expression matches a biomedical test text, CREGEX assigns the class of the training text with the highest similarity score, $\text{sw\_sim}(x_i, x_j)$, according to the SW algorithm, $j = 1, \ldots, n$. On the other hand, if n_r regular expressions match a biomedical test text, CREGEX assigns the class $y_j = \mathcal{L}(\Phi^{-1}(r_j))$ of the jth regular expression, $r_j \equiv r(x_j)$, that achieves the highest precision score, $Pr(r_j)$, with $j = 1, \ldots, n_r$. Such a score is calculated during training stage as the ratio between the number of correct matches in the corresponding class and the total number of matches. Thus, the CREGEX classifier decision function is given by:

$$\delta(x_i) = \begin{cases} \mathcal{L}\left(\underset{j\in[1,n]}{\text{argmax}}\ \text{sw\_sim}(x_i, x_j)\right), & n_r = 0 \\ \mathcal{L}\left(\Phi^{-1}\left(\underset{j\in[1,n_r]}{\text{argmax}}\ Pr(r_j)\right)\right), & n_r > 0. \end{cases} \tag{1}$$

3) THE AL VERSION OF CREGEX

The AL version of CREGEX introduces a query strategy with the following rationale: Select those examples associated with regular expressions with a higher level of uncertainty (the more informative ones) according to the precision metric. Initially, regular expressions will tend to have a high level of uncertainty (low precision value), resulting in an appealing greedy learning approach. However, as the more informative examples are selected, the regular expressions' precision improve until eventually settle down (high precision score), resulting in a more conservative learning induced by text similarity. Consequently, biomedical test texts without regular expression matches or regular expressions with low precision scores satisfy this idea. Mathematically, the proposed query strategy can be expressed as:

$$\underset{x_i\in X_U}{\text{argmin}}\, q(x_i) \tag{2}$$

where:

$$q(x_i) = \begin{cases} \max_{j\in[1,n]} \text{sw\_sim}(x_i, x_j), & n_r = 0 \\ \max_{j\in[1,n_r]} Pr(x_j), & n_r > 0. \end{cases} \tag{3}$$

In other words, the proposed query strategy assigns either the SW similarity score associated with a text close to the test text or the maximum value of the precision score associated with the j*th* regular expression matching the test text. In this sense, the most informative examples have the lowest values in this query strategy. Besides, a stopping criterion was introduced to reduce the number of training examples compared to PL. Our stopping criterion analyzes, at each iteration, the variance of the scores of the examples in the query strategy $q(\cdot)$. More precisely, a historical window of n_v score variances, $\{V_1, \ldots, V_{n_v}\}$, from the current value V_{n_v} is analyzed using the variance method [33]. Recalling that the most informative examples achieve the lowest values in our query strategy, the criterion halts the learning process if a sustained decrease in the variance is achieved. Mathematically, the stopping criterion is met when: $V_2 > V_1$, and $V_2 > \max\{V_3, \ldots V_{n_v}\}$.

C. EVALUATION

The performance of the AL version of CREGEX was compared with SVM, NB, and BERT-based classifiers. These classification algorithms require each text to be represented by feature vectors of constant length, n_f, which can be obtained from pre-trained models or by extracting and counting representative elements or tokens (i.e., word sequences, numbers, or symbols) from the texts [36], [37]. SVM is a linear classifier that discriminates among the classes through a hyperplane whose parameters are optimally obtained [38]. For a given test text x_i represented by a feature vector of n_f values $\vec{x}_i$, the classifier decision function is given by:

$$\delta(x_i) = \text{sign}\left(\sum_{j=1}^{n_s} \alpha_j y_j K(\vec{x}_i, \vec{x}_j) + b\right), \tag{4}$$

where $\text{sign}(\cdot)$ is the sign function, n_s is the number of support vectors (vectors formed by the points closest to the optimal hyperplane), α_j and $y_j \in \{-1, 1\}$ correspond to weights associated with the jth support vector and class, respectively, $K(\cdot)$ is a kernel function for mapping vectors onto a high dimensional feature space, and b is an intercept term (scalar). Notice that in the case of a linear kernel $K(\cdot) = \vec{x}_i \cdot \vec{x}_j$. The NB classifier is a probabilistic discriminant based on the Bayes theorem. The NB classifier relies on the central assumption that, given a class y_i, its features are conditionally independent [39]. Mathematically, the NB classifies the test example x_i according to the rule:

$$\delta(x_i) = \underset{y_i\in Y}{\text{argmax}}\, P(y_i) \prod_{j=1}^{n_f} P(\vec{x}_j|y_i), \tag{5}$$

where $P(y_i)$ is the prior probability of class $y_i \in Y$, and $P(\vec{x}_j|y_i)$ is the probability of the example x_i described by a feature vector, given the ith class.

The BERT classifier represents biomedical texts through a transformer-based encoding architecture, which analyzes input tokens in both directions: the left and right of the context. We fine-tuned a BERT classifier for our three datasets, adding a dropout regularization layer and a softmax layer to BERT's pre-trained layer (embeddings). Thus, a class is assigned to a test example xi according to the probabilities obtained from the softmax classification layer:

$$\delta(x_i) = \underset{j\in[1,\ldots,l]}{\text{argmax}} \frac{e^{z_j}}{\sum_{k=1}^{l} e^{z_k}}, \tag{6}$$

where $z = \{z_1, \ldots, z_l\}$ is the intermediate output of the softmax layer for the test text x_i. To obtain feature vectors for SVM and NB classifiers, we extracted consecutive sequences of tokens (n-grams) in the form of unigrams (consecutive sequences of a token or N1) and bigrams (consecutive sequences of two tokens or N2). The n-grams were represented in matrix form according to the Term frequency - Inverse document frequency (TF-IDF) method, which enables to evaluate the importance of such features by considering the total number of texts according to [37], [40]:

$$TF - IDF = TF_{t,d} \times IDF_{t,D}, \tag{7}$$

$$IDF_{t,D} = log_{10}(\frac{|D|}{d}), \tag{8}$$

where TF represents the absolute frequency of each token t in each text d, while IDF represents the inverse frequency of the tokens in the whole dataset D. Regarding the BERT classifier, we used a base-multilingual–uncased BERT model, which was combined with a softmax classifier for fine-tuning purposes. In our evaluations, we considered a linear kernel in the case of SVM and a multinomial NB, keeping the rest of the parameters by default [41]. In the case of BERT-based classifier we set: epochs = 4, batch size = 8, dropout = 0.1, and an Adam optimizer with a learning rate $= 2^{-5}$ [5], [6]. Following [11], [13], [14], we introduced AL to SVM computing both the distance between an example and the decision hyperplane and the cosine similarity. In the case of NB, we used the maximum entropy criterion. For the BERT-based classifier, we used Monte-Carlo Dropout and the criterion of maximum entropy on an average of 10 predictions (forward-passes) per example [15], [16]. Regarding the AL version of CREGEX, we studied how the greedy (precision score) and the conservative (normalized SW score) terms in the query strategy (2) affect jointly and independently the AL process. To do so, we used a convex combination as follows:

$$q'(x_i) = \lambda \max_{j\in[1,n_r]} Pr(x_j) + (1-\lambda) \max_{j\in[1,n]} \text{sw\_sim}(x_i, x_j), \tag{9}$$

where $\lambda \in [0, 1]$. In our study, we used the following values for λ

- AMB: $\lambda = 1$, regarding only the precision metric component (Pr).
- DIV: $\lambda = 0$, regarding only the diversity component (SW similarity score).
- CMB: $\lambda = 0.5$ equally combination of precision and diversity components.

We remark that the proposed query strategy (2) do not combine the terms it only uses one of them depending on n_r. We used 10-fold cross-validation during the classifiers' training and evaluation [42], [43]. In the case of AL, 50 examples were iteratively selected from the unlabeled dataset [18].

To evaluate the performance of the classifiers, the most commonly used metrics in NLP were used: accuracy (ACC), precision (P), recall (R), and F-measure (F1) [44]. Whereas the ACC evaluates the number of correctly classified examples over all test examples, the pair precision and recall give information on the classifier's behavior regarding what percentage of positive predictions were correctly classified and what percentage of positive cases were captured, respectively [45]. In this sense, the F1 metric aims to evaluate a balance between precision and recall metrics in a single value (harmonic mean). Mathematically these metrics are defined as:

$$ACC = \frac{TP + TN}{TP + FP + TN + FN}, \tag{10}$$

$$P = \frac{TP}{TP + FP}, \tag{11}$$

$$R = \frac{TP}{TP + FN}, \tag{12}$$

$$F1 = \frac{2PR}{P + R}, \tag{13}$$

where TP, TN, FP, and FN correspond, respectively, to the true positive, true negative, false positive, and false negative classification rates. Besides, the area-under-the-learning-curves were calculated according to the trapezoid method, normalized by the number of iterations. [46]–[48]. Lastly, to analyze the statistical significance of the results, we used the Kolmogorov-Smirnov and Shapiro-Wilk tests ($\alpha = 0.05$) to assess data's goodness-of-fit, and the paired T-test and Wilcoxon signed-rank test were performed after analyzing the goodness-of-fit.

IV. RESULTS

Tables 2, 3 and 4 show the classification results for all datasets. Metrics were calculated, for each classifier and each class, and then their averaged values were weighted by the number of true instances for each class. In all cases, CREGEX outperforms both the SVM and NB classifiers on all metrics ($p < 0.05$). Also, CREGEX performed better than BERT on the OBESITY STATUS and OBESITY TYPES datasets but was slightly outperformed on the SMOKING STATUS dataset ($p > 0.05$). This is explained by three facts. First, the OBESITY STATUS dataset and OBESITY TYPES share medical terminology. The former is a binary problem and the latter is a more complex problem involving multiple classes. Thus, it requires more training examples to generate proper models to discriminate the problem classes.

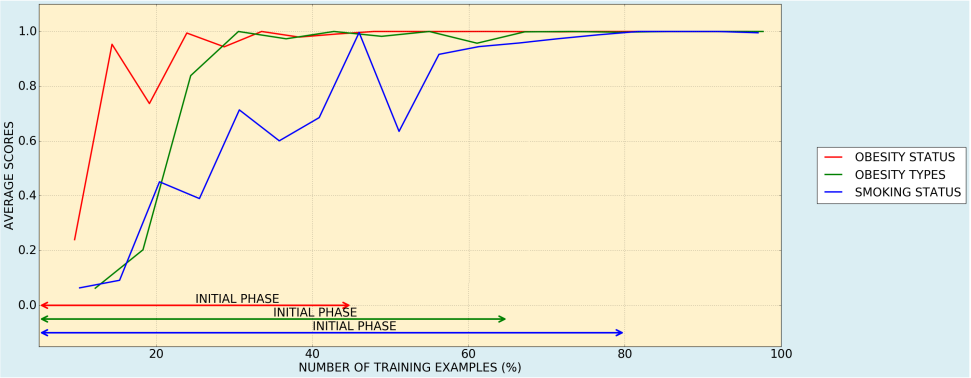

FIGURE 4. These curves show the dynamics of the query strategy scores as a function of the percentage of labeled sample texts.

TABLE 2. Weighted average classification results for the OBESITY STATUS dataset.

Classifier	Metric			
	ACC (%)	P (%)	R (%)	F1 (%)
CREGEX	97.5	97.6	97.5	97.5
BERT	96.8	96.9	96.8	96.8
SVM-N1	96.4	96.4	96.4	96.4
SVM-N2	94.3	94.5	94.3	94.4
NB-N1	87.5	87.4	87.5	87.4
NB-N2	89.6	89.6	89.6	89.6

TABLE 3. Weighted average classification results for the OBESITY TYPES dataset.

Classifier	Metric			
	ACC (%)	P (%)	R (%)	F1 (%)
CREGEX	94.8	94.9	94.8	94.8
BERT	90.4	90.6	90.4	90.5
SVM-N1	82.7	83.2	82.7	82.9
SVM-N2	88.4	89.0	88.4	88.7
NB-N1	74.3	74.5	74.3	74.4
NB-N2	82.3	83.0	82.3	82.6

Second, the OBESITY STATUS and OBESITY TYPES datasets contain much more numerical attributes (anthropometric information) than the SMOKING STATUS dataset. In this sense, it has been shown that BERT may not work properly representing numbers, while regular expressions allow representing complex sequential patterns, including numerical attributes [18], [22], [23]. Third, the SMOKING STATUS dataset presents temporal data and negations in the texts, and regular expressions need more examples to abstract the information.

Figure. 4 shows the average evolution of the CREGEX query strategy scores, as a function of the number of training examples, during the learning stage for all datasets. We observe two phases: (i) an initial phase or transient state, where scores are adjusted as training examples are selected, and (ii) a stabilization phase or steady-state, where scores have low variability. In this sense, the OBESITY STATUS dataset's scores settle down more rapidly, followed by the OBESITY TYPES and SMOKING STATUS dataset.

TABLE 4. Weighted average classification results for the SMOKING STATUS dataset.

Classifier	Metric			
	ACC (%)	P (%)	R (%)	F1 (%)
CREGEX	88.9	89.4	88.9	89.1
BERT	89.2*	89.4*	89.2*	89.3*
SVM-N1	84.6	84.9	84.6	84.7
SVM-N2	85.9	86.6	85.9	86.2
NB-N1	76.7	77.3	76.7	77.0
NB-N2	82.8	83.4	82.8	83.1

$^{(*)}$ Indicates that no statistically significant differences were found as compared to CREGEX ($p > 0.05$). $^{(a)}$T-test.

We studied how the greedy (precision score) and conservative (normalized SW score) terms in the query strategy affect, jointly and independently, the AL process. Figure 5 shows the AL curves of CREGEX in four cases: AMB (only the greedy component is used), DIV (only the conservative component is used), CMB (both components are used and weighted equally), and CLF (the proposed query strategy). It can be observed that, for the three datasets and all cases, using only the DIV component achieves a lower performance as compared to the proposed query strategy CLF. This effect is also observed in the area-under-the-learning-

TABLE 5. The average area under the learning curve results for CREGEX according to the convex combination function.

Classifier	OBESITY STATUS		OBESITY TYPES		SMOKING STATUS	
	ACC (%)	F1 (%)	ACC (%)	F1 (%)	ACC (%)	F1 (%)
CREGEX-AMB	$97.1^{(*)(a)}$	$97.2^{(*)(a)}$	94.3	94.4	87.1	87.4
CREGEX-DIV	$96.8^{(*)(a)}$	$96.9^{(*)(b)}$	$93.2^{(*)(a)}$	$93.3^{(*)(a)}$	85.8	$86.1^{(*)(a)}$
CREGEX-CMB	$97.1^{(*)(a)}$	$97.2^{(*)(a)}$	$94.4^{(*)(a)}$	$94.5^{(*)(a)}$	$86.3^{(*)(a)}$	$86.7^{(*)(a)}$
CREGEX-CLF	97.2	97.2	94.4	94.5	86.6	86.9

$^{(*)}$ Non-statistically significant differences as compared to CLF. $^{(a)}$T-test. $^{(b)}$ Wilcoxon signed-rank.

TABLE 6. The average area under the learning curve results with the different query strategies.

Classifier	OBESITY STATUS		OBESITY TYPES		SMOKING STATUS	
	ACC (%)	F1 (%)	ACC (%)	F1 (%)	ACC (%)	F1 (%)
CREGEX-PL	96.82	96.88	92.95	93.12	84.91	85.37
CREGEX-CLF	$97.15^{(*)(b)}$	97.25	94.41	94.52	86.56	86.87
BERT-PL	94.28	93.74	77.68	75.78	82.67	82.83
BERT-CLF	94.97	94.81	82.49	81.49	83.16	83.55
SVM-N1-PL	94.17	94.22	76.12	75.87	79.47	79.76
SVM-N1-CLF	95.15	95.27	80.60	80.85	81.60	81.82
SVM-N1-CMB	95.16	95.22	79.44	79.49	$80.18^{(*)(a)}$	$80.40^{(*)(a)}$
SVM-N2-PL	89.83	90.19	80.59	81.28	79.75	80.80
SVM-N2-CLF	93.36	93.50	86.40	86.58	83.03	83.43
SVM-N2-CMB	93.29	93.42	86.40	86.66	$81.51^{(*)(a)}$	$81.87^{(*)(a)}$
NB-N1-PL	87.78	87.72	71.61	71.26	73.37	73.68
NB-N1-CLF	$87.78^{(*)(a)}$	$87.78^{(*)(a)}$	$72.70^{(*)(a)}$	$72.50^{(*)(a)}$	75.76	76.12
NB-N2-PL	89.12	89.08	79.21	79.31	77.97	78.18
NB-N2-CLF	91.34	91.39	81.09	81.42	80.74	81.04

(*) Non-statistically significant differences as compared to PL in the corresponding classifier.
$^{(a)}$ T-student. $^{(b)}$ Wilcoxon signed-rank.

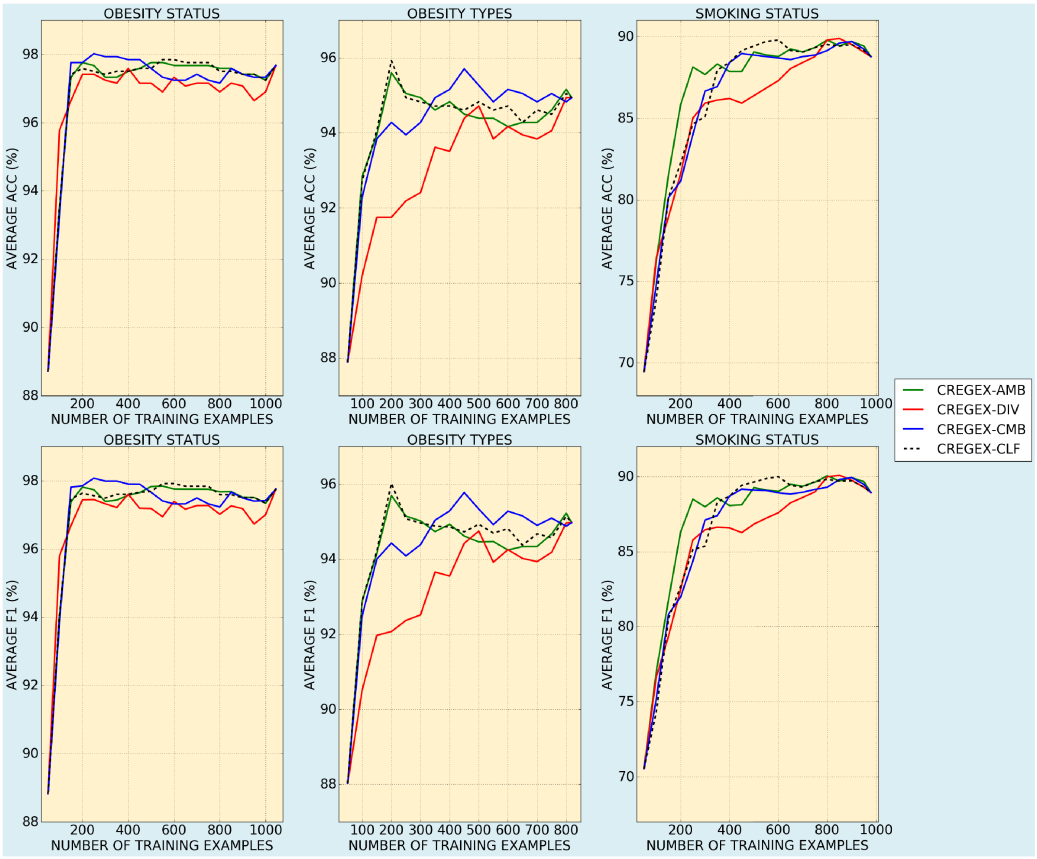

FIGURE 5. These curves show the evolution of the CREGEX's AL with different sample sizes and performance in terms of ACC (%) and F1 (%) metrics.

curves results in Table 5 ($p > 0.05$). In most cases, except for the SMOKING STATUS dataset, the performance of the proposed query strategy is greater than or equal to the rest of the components. In general, we observe that the proposed query strategy for all datasets requires fewer training examples than the rest of the strategies for the same performance.

Figure 6 shows the learning curves of the classifiers in terms of number of training examples and performance measured in terms of ACC (%) and F1 (%). It is possible to observe that the performance of CREGEX was better than the rest of the classifiers in all cases. This can also be observed in the area-under-the-learning-curve results shown in Table 6. In all cases AL allowed to obtain a performance greater than or equal to PL. In this sense, the proposed query strategy allowed to obtain a better performance than the rest of the query strategies. From Figure 6, we can also observe that in most cases, except for NB-N1 in the OBESITY STATUS and

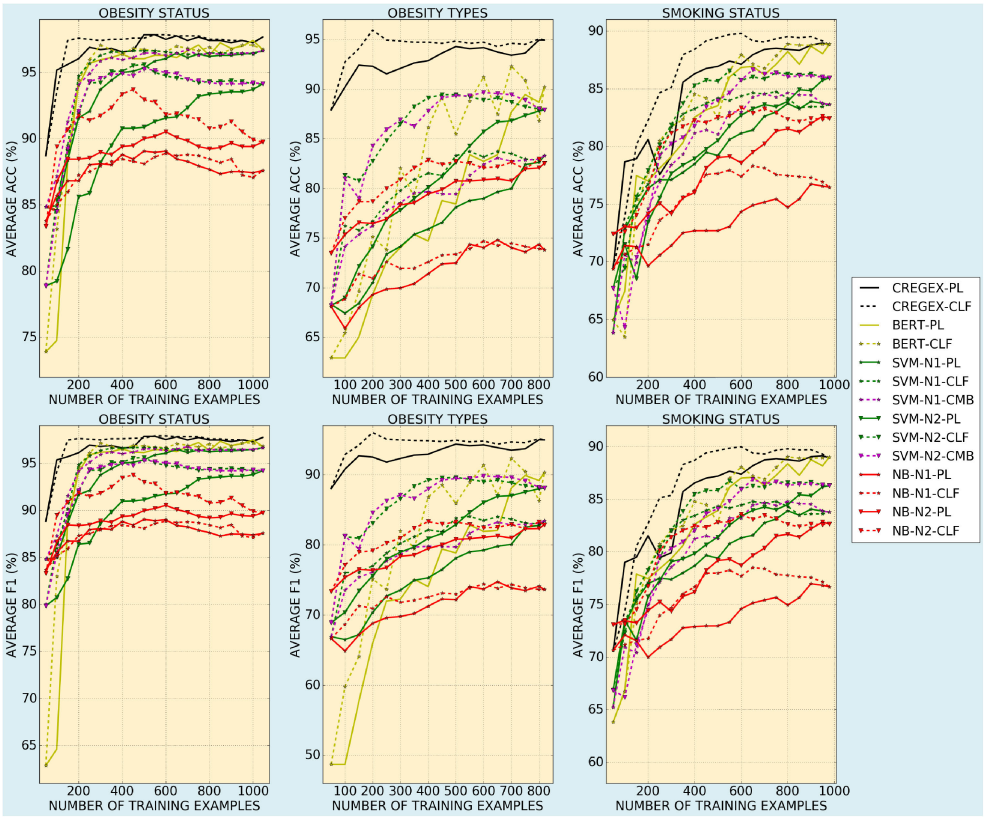

FIGURE 6. These curves show the evolution of the classifiers' AL with different sample sizes and performance in terms of ACC (%) and F1 (%) metrics.

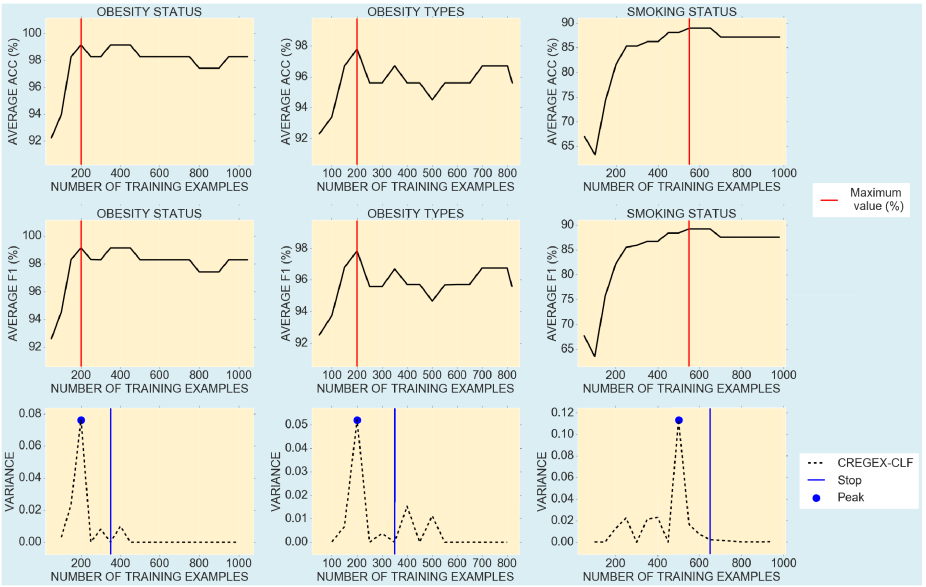

FIGURE 7. These curves show the evolution of the stopping criterion for AL curves of CREGEX as a function of the number of training examples.

OBESITY TYPES datasets, the use of AL allowed to reduce the number of training examples achieving the same ACC (%) and F1 (%) value. We note that the use of AL in CREGEX led to better results in all cases, especially in the case of the OBESITY TYPES dataset.

Figure 7 shows an example of the stopping criterion used in the AL version of CREGEX. We observe that the variance pattern of the query strategy scores (below) follows the maximum value of a learning curve (above). Finally, we study how the stopping criteria for the AL process affects

TABLE 7. Results of the stopping criterion according to the variance method applied to the scores of the classifiers' query strategies.

Classifier	OBESITY STATUS			OBESITY TYPES			SMOKING STATUS		
	Δ_{ACC}	Δ_{F1}	$\%_X$	Δ_{ACC}	Δ_{F1}	$\%_X$	Δ_{ACC}	Δ_{F1}	$\%_X$
CREGEX-CLF	1.0	1.0	32.1	1.7	1.6	44.6	1.8	1.8	50.1
BERT-CLF	1.2	1.2	43.1	6.0	6.2	73.3	4.4	4.4	78.7
SVM-N1-CLF	0.5	0.5	46.9	1.1	1.1	80.7	1.1	1.2	65.9
SVM-N1-CMB	1.3	1.3	40.7	4.0	3.7	76.4	2.5	2.4	67.0
SVM-N2-CLF	0.5	0.5	47.9	2.1	2.1	70.9	2.3	2.3	67.0
SVM-N2-CMB	1.0	1.0	41.2	1.0	1.0	75.8	1.4	1.4	69.0
NB-N1-CLF	2.2	2.2	53.6	2.9	3.1	69.1	2.1	2.1	69.5
NB-N2-CLF	1.4	1.3	46.4	2.5	2.5	66.6	2.6	2.6	55.7

Δ_{ACC} indicates a reduction in the ACC mectric (%) with respect to its maximum value.
Δ_{F1} indicates a reduction in the F1 metric (%) with respect to its maximum value.
$\%_X$ indicates the percentage of the total number of training examples used to achieve the ACC and F1 metrics.

all the classifier's performance. Table 7 lists, for all the cases, the average reduction in both the accuracy (Δ_{ACC}) and the F1 metric (Δ_{F1}), concerning the maximum values achieved in the learning curves (please refer to Figure 7). The Table also lists the percentage of training examples ($\%_X$) used to reach the performance metrics. It can be observed that, in all cases, the stopping criterion halted the AL training process using between 32% and 80% of the total number of training examples. Note also that the performance metrics were reduced at most a 7%. Note that, for the AL version of CREGEX, the stopping criterion allowed to use only between 32% to 50% of the total number of training examples with a drop in performance less than 2% concerning the maximum values.

V. CONCLUSION AND FUTURE WORK

In this work, we presented an AL-based biomedical text classifier termed as CREGEX. The classifier models biomedical texts automatically using regular expressions. The query strategy samples the training dataset, trading off the greedy learning achieved by the regular expressions classification precision and the conservative learning induced by text sequence alignment classification. The sustained reduction in the variance of the query strategy scores is used as a stopping criterion.

Results indicate that CREGEX classifier outperform traditional classification algorithms such as SVM and NB on all metrics. Also, the AL version of the CREGEX performed better than BERT on the OBESITY STATUS and OBESITY TYPES datasets but was slightly outperformed on the SMOKING STATUS dataset ($p > 0.05$). This is explained by two facts. First, that the OBESITY STATUS and OBESITY TYPES datasets contain much more numerical attributes (anthropometric information) than the SMOKING STATUS dataset. In this sense, it has been shown that BERT may not work properly representing numbers, while regular expressions allow representing complex sequential patterns, including numerical attributes [18], [22], [23]. Second, the SMOKING STATUS dataset presents temporal data and negations in the texts, and regular expressions need more examples to abstract the information. However, we comment that the complexity of the AL version of CREGEX is smaller and requires only training texts and fewer parameters in the model as compared to BERT. Recall that, BERT must be trained on a large collection of documents using BooksCorpus (800M words) and Wikipedia (2500M words) and has 110M parameters [49].

Regarding the AL process, in all cases, the area under the learning curves were larger than in the case of PL. Besides, in most cases AL allowed to reduce the number of training examples needed to obtain the same performance in all datasets as compared to PL. More precisely, the AL method together with the stopping criterion in CREGEX allowed to use, on average, between 32% to 50% of the total number of training examples without significantly affecting the performance of this classifier in terms of ACC (%) and F1 (%) metrics.

As future work, we will improve CREGEX in terms of the automatic generation and selection of regular expressions. We will also study the ability of BERT or other pre-trained language models to represent Spanish biomedical texts.

ACKNOWLEDGMENT

The authors would like to thank Informatics Unit with the Guillermo Grant Benavente Hospital in Concepción, Chile, for providing datasets. The authors also thank the Biomedical Informatics Center at The George Washington University for supporting the development of CREGEX.

REFERENCES

[1] S. Kaisler, F. Armour, J. A. Espinosa, and W. Money, "Big data: Issues and challenges moving forward," in *Proc. 46th Hawaii Int. Conf. Syst. Sci.*, Jan. 2013, pp. 995–1004.
[2] J. Adeva, J. Atxa, M. Carrillo, and E. Zengotitabengoa, "Automatic text classification to support systematic reviews in medicine," *Expert Syst. Appl.*, vol. 41, no. 4, pp. 1498–1508, 2014.
[3] M. Liu, L. Pan, and S. Liu, "To transfer or not: An online cost optimization algorithm for using two-tier storage-as-a-service clouds," *IEEE Access*, vol. 7, pp. 94263–94275, 2019.
[4] S. Hassan, M. Rafi, and M. S. Shaikh, "Comparing SVM and Naïve Bayes classifiers for text categorization with wikitology as knowledge enrichment," in *Proc. IEEE 14th Int. Multitopic Conf.*, Dec. 2011, pp. 31–34.
[5] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune BERT for text classification?" in *Proc. China Nat. Conf. Chin. Comput. Linguistics*. Kunming, China: Springer, 2019, pp. 194–206.
[6] M. Munikar, S. Shakya, and A. Shrestha, "Fine-grained sentiment classification using BERT," in *Proc. Artif. Intell. Transf. Bus. Soc. (AITB)*, vol. 1, Nov. 2019, pp. 1–5.
[7] L. Akhtyamova, "Named entity recognition in Spanish biomedical literature: Short review and bert model," in *Proc. 26th Conf. Open Innov. Assoc. (FRUCT)*, Apr. 2020, pp. 1–7.
[8] A. Sarker and G. Gonzalez, "Portable automatic text classification for adverse drug reaction detection via multi-corpus training," *J. Biomed. Informat.*, vol. 53, pp. 196–207, Feb. 2015.
[9] R. Chhatwal, N. Huber-Fliflet, R. Keeling, J. Zhang, and H. Zhao, "Empirical evaluations of active learning strategies in legal document review," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 1428–1437.
[10] R. Hu, "Active learning for text classification," M.S. thesis, School Comput. Sci., College Sci. Health, Technol. Univ. Dublin, Dublin, Irlanda, 2011.
[11] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *SIGIR'94*. Dublin, Ireland: Springer, 1994, pp. 3–12.
[12] B. Settles, "Active learning literature survey," Dept. Comput. Sci., Univ. Wisconsin-Madison, Madison, WI, USA, Tech. Rep. 1648, 2009.
[13] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *J. Mach. Learn. Res.*, vol. 2, pp. 45–66, Nov. 2001.
[14] K. Brinker, "Incorporating diversity in active learning with support vector machines," in *Proc. 20th Int. Conf. Mach. Learn. (ICML)*, 2003, pp. 59–66.
[15] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1050–1059.

[16] B. An, W. Wu, and H. Han, "Deep active learning for text classification," in *Proc. 2nd Int. Conf. Vis., Image Signal Process.*, Aug. 2018, pp. 1–6.
[17] Y. Li, R. Krishnamurthy, S. Raghavan, S. Vaithyanathan, and H. V. Jagadish, "Regular expression learning for information extraction," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2008, pp. 21–30.
[18] D. D. A. Bui and Q. Zeng-Treitler, "Learning regular expressions for clinical text classification," *J. Amer. Med. Inform. Assoc.*, vol. 21, no. 5, pp. 850–857, Sep. 2014.
[19] A. Bartoli, A. De Lorenzo, E. Medvet, and F. Tarlao, "Active learning of regular expressions for entity extraction," *IEEE Trans. Cybern.*, vol. 48, no. 3, pp. 1067–1080, Mar. 2018.
[20] M. Cui, R. Bai, Z. Lu, X. Li, U. Aickelin, and P. Ge, "Regular expression based medical text classification using constructive heuristic approach," *IEEE Access*, vol. 7, pp. 147892–147904, 2019.
[21] C. A. Flores, R. L. Figueroa, J. E. Pezoa, and Q. Zeng-Treitler, "CREGEX: A biomedical text classifier based on automatically generated regular expressions," *IEEE Access*, vol. 8, pp. 29270–29280, 2020.
[22] M. A. Murtaugh, B. S. Gibson, D. Redd, and Q. Zeng-Treitler, "Regular expression-based learning to extract bodyweight values from clinical notes," *J. Biomed. Informat.*, vol. 54, pp. 186–190, Apr. 2015.
[23] E. Wallace, Y. Wang, S. Li, S. Singh, and M. Gardner, "Do NLP models know numbers? Probing numeracy in embeddings," 2019, *arXiv:1909.07940*. [Online]. Available: http://arxiv.org/abs/1909.07940
[24] G. Rabby, S. Azad, M. Mahmud, K. Z. Zamli, and M. M. Rahman, "TeKET: A tree-based unsupervised keyphrase extraction technique," *Cogn. Comput.*, vol. 12, no. 4, pp. 811–833, Jul. 2020.
[25] B. C. Wallace, K. Small, C. E. Brodley, and T. A. Trikalinos, "Active learning for biomedical citation screening," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2010, pp. 173–182.
[26] R. L. Figueroa, Q. Zeng-Treitler, L. H. Ngo, S. Goryachev, and E. P. Wiechmann, "Active learning for clinical text classification: Is it better than random sampling?" *J. Amer. Med. Inform. Assoc.*, vol. 19, no. 5, pp. 809–816, Sep. 2012.
[27] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proc. 5th Annu. Workshop Comput. Learn. Theory*, 1992, pp. 287–294.
[28] H. T. Nguyen and A. Smeulders, "Active learning using pre-clustering," in *Proc. 21st Int. Conf. Mach. Learn.*, 2004, p. 79.
[29] A. Huang, D. Milne, E. Frank, and I. H. Witten, "Clustering documents with active learning using wikipedia," in *Proc. 8th IEEE Int. Conf. Data Mining*, Dec. 2008, pp. 839–844.
[30] M. Wang, F. Min, Z.-H. Zhang, and Y.-X. Wu, "Active learning through density clustering," *Expert Syst. Appl.*, vol. 85, pp. 305–317, Nov. 2017.
[31] A. Vlachos, "A stopping criterion for active learning," *Comput. Speech Lang.*, vol. 22, no. 3, pp. 295–312, Jul. 2008.
[32] J. Zhu, H. Wang, E. Hovy, and M. Ma, "Confidence-based stopping criteria for active learning for data annotation," *ACM Trans. Speech Lang. Process.*, vol. 6, no. 3, pp. 1–24, Apr. 2010.
[33] M. Ghayoomi, "Using variance as a stopping criterion for active learning of frame assignment," in *Proc. NAACL HLT Workshop Act. Learn. Natural Lang. Process.*, 2010, pp. 1–9.
[34] M. Bloodgood and K. Vijay-Shanker, "A method for stopping active learning based on stabilizing predictions and the need for user-adjustable stopping," 2014, *arXiv:1409.5165*. [Online]. Available: http://arxiv.org/abs/1409.5165
[35] G. Beatty, E. Kochis, and M. Bloodgood, "The use of unlabeled data versus labeled data for stopping active learning for text classification," in *Proc. IEEE 13th Int. Conf. Semantic Comput. (ICSC)*, Jan. 2019, pp. 287–294.
[36] H. Altınçay and Z. Erenel, "Using the absolute difference of term occurrence probabilities in binary text categorization," *Appl. Intell.*, vol. 36, no. 1, pp. 148–160, 2012.
[37] B. Li, T. Liu, Z. Zhao, P. Wang, and X. Du, "Neural bag-of-ngrams," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1–8.
[38] V. K. Chauhan, K. Dahiya, and A. Sharma, "Problem formulations and solvers in linear SVM: A review," *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 803–855, Aug. 2019.
[39] M. Li and K. Liu, "Causality-based attribute weighting via information flow and genetic algorithm for Naive Bayes classifier," *IEEE Access*, vol. 7, pp. 150630–150641, 2019.
[40] Z. E. Xu, M. Chen, K. Q. Weinberger, and F. Sha, "An alternative text representation to TF-IDF and bag-of-words," 2013, *arXiv:1301.6770*. [Online]. Available: http://arxiv.org/abs/1301.6770
[41] G. Singh, B. Kumar, L. Gaur, and A. Tyagi, "Comparison between multinomial and Bernoulli Naïve Bayes for text classification," in *Proc. Int. Conf. Automat., Comput. Technol. Manage. (ICACTM)*, 2019, pp. 593–596.
[42] I. Witten, E. Frank, and M. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd ed. Amsterdam, The Netherlands: Morgan Kaufmann, 2011.
[43] G. Vanwinckelen and H. Blockeel, "On estimating model accuracy with repeated cross-validation," in *Proc. 21st Belgian-Dutch Conf. Mach. Learn. (Benelearn)*, 2012, pp. 39–44.
[44] R. Yacouby and D. Axman, "Probabilistic extension of precision, recall, and F1 score for more thorough evaluation of classification models," in *Proc. 1st Workshop Eval. Comparison NLP Syst.*, 2020, pp. 79–91.
[45] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, p. 6, Dec. 2020.
[46] Y. Chen, H. Cao, Q. Mei, K. Zheng, and H. Xu, "Applying active learning to supervised word sense disambiguation in MEDLINE," *J. Amer. Med. Inform. Assoc.*, vol. 20, no. 5, pp. 1001–1006, Sep. 2013.
[47] M. Huijser and J. C. van Gemert, "Active decision boundary annotation with deep generative models," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 5286–5295.
[48] M.-A. Carbonneau, E. Granger, and G. Gagnon, "Bag-level aggregation for multiple-instance active learning in instance classification problems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 5, pp. 1441–1451, May 2019.
[49] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*. [Online]. Available: http://arxiv.org/abs/1810.04805

CHRISTOPHER A. FLORES received the B.Sc., M.Sc., and D.Sc. degrees in electrical engineering from the Universidad de Concepción, Concepción, Chile, in 2013, 2017, and 2021, respectively. He is currently a Lecturer with the Departamento de Ingeniería Eléctrica, Universidad de Concepción. His research interests include natural language processing, text mining, and machine learning.

ROSA L. FIGUEROA received the B.Eng. and Ph.D. degrees in electrical engineering from the University of Concepción, in 2004 and 2012, respectively. She is currently a Faculty Member and a Researcher in the biomedical engineering degree part of the Electrical Engineering Department, University of Concepcion, and a Technical Board Member with the National Center on Health Information Systems. She has scientific publications in journals and conference proceedings. She is also working on research projects related to secondary use of medical data and text classification. Her research interest includes medical informatics area mainly machine learning and text mining. Her Ph.D. thesis explored different methods to obtain useful information from free text.

JORGE E. PEZOA (Member, IEEE) received the B.S. degree in electronics engineering and the M.S. degree in electrical engineering from the Universidad de Concepción, Concepción, Chile, in 1999 and 2003, respectively, and the Ph.D. degree in electrical engineering from The University of New Mexico, Albuquerque, NM, USA, in 2010. He is currently an Associate Professor and an Associate Chair with the Departamento de Ingeniería Eléctrica, Universidad de Concepción. His research interests include distributed computing, pattern recognition, statistical signal processing, network optimization, and hyperspectral image and signal processing for industrial processes. He is a member of the Society of Photo-optical Instrumentation Engineers (SPIE), the Optical Society of America (OSA), and the Association for Computing Machinery (ACM).