A Hybrid Method for Clinical Text Classification Based on Confident Predictions and Regular Expressions

1st Christopher A. Flores
Institute of Engineering Sciences
Universidad de O'Higgins
Rancagua, Chile
christopher.flores@uoh.cl

2nd Rodrigo Verschae
Institute of Engineering Sciences
Universidad de O'Higgins
Rancagua, Chile
rodrigo@verschae.org

***Abstract*—Supervised algorithms allow clinical texts to be automatically organized based on their content. In this sense, supervised algorithm predictions must be accurate and confident to be used in clinical practice, considering the complex patterns in the texts. In this aspect, sequences of character strings known as regular expressions offer an alternative closer to natural language to represent complex patterns from texts, which can be automatically generated using sequence alignment algorithms. This paper proposes a hybrid method that combines the most confident predictions of a supervised algorithm and regular expressions for clinical text classification. Our method uses regular expressions to classify clinical texts when the predictions of a supervised algorithm are not confident in terms of predictive probability. To evaluate our method, we used three datasets with information on smoking and obesity status across supervised algorithms: Support Vector Machine (SVM), Random Forest (RF), Naive Bayes (NB), and Bidirectional Encoder Representations from Transformers (BERT). The classification results indicate that the proposed method, on average, improved the performance of supervised algorithms on all performance metrics by up to 5%. Thus, we demonstrated the ability of our method to generate regular expressions representative of clinical texts as support in cases when the predictions of the supervised algorithms were not confident.**

***Index Terms*—Clinical text classification, probability prediction, regular expressions**

I. Introduction

Text classification is a valuable tool to automatically organize a large amount of digital information into categories [1]. Organizing scientific documents or unstructured textual information from clinical texts is possible in the biomedical area using supervised algorithms.

In text classification, a supervised learning algorithm is used, either of traditional use such as Support Vector Machine (SVM), Random Forest (RF), Naive Bayes (NB), or more recently based on neural networks, such as Bidirectional Encoder Representations from Transformers (BERT) [2]. However, regardless of the algorithm used, predictions must be accurate and confident in the clinical area [3]. In this sense, one of the tools available to researchers for text classification is regular expressions, which can be adapted to different domains, allowing complex patterns to be captured from texts [4].

Regular expressions correspond to a set of characters and metacharacters without a literal meaning within the expression to define search patterns in texts [5]. It is possible to use regular expressions in text pre-processing, information extraction, and classification tasks to a lesser extent [6]–[8]. However, one of the main challenges in using regular expressions is the automatic generation from training texts. For example, Bui and Zeng-Treitler propose a method that allows the generation of regular expressions automatically from labeled texts by using sequence alignment algorithms [6]. Thus, sequences of common words are extracted from the texts to generate regular expressions for each class of the problem. On the other hand, Li et al. generate regular expressions by identifying keywords from the texts using attention mechanisms [8]. Subsequently, keywords, metacharacters, and Boolean operators (e.g., NOT, OR, and AND) are combined to generate matching rules.

On the other hand, an important aspect of classification tasks is to improve the performance of supervised algorithms, especially in problems with unbalanced classes or scenarios where the class of interest must be improved [9], [10]. Thus, maximizing some metric of interest (e.g., the harmonic mean between precision and recall) makes it possible to adjust the predictive decision threshold for supervised algorithms. This is particularly important in the clinical domain, where supervised learning algorithms must be accurate in their predictions to be considered decision-support tools [11].

Given the above, this paper proposes a method that combines the most confident predictions of a supervised algorithm and regular expressions for clinical classification tasks. In this sense, the two main contributions of this work focus on constructing a feature space based on the automatic generation of regular expressions from clinical texts and a hybrid method that selectively combines a supervised algorithm and such regular expressions to classify texts. The most confident predictions satisfy a probability threshold calculated for each

This work was partially supported by ANID FONDECYT Postdoctorado 3220803 and ANID FONDEQUIP Mediano EQM170041.

supervised algorithm from an additional validation set. We hypothesize that if only the most confident predictions of the algorithms in terms of predictive probability are used, clinical text classification could be more accurate. On the other hand, when the algorithms' predictions are not confident, we use the most reliable regular expressions automatically generated from the labeled texts to assign a class. Classification results indicate that, on average, our method improved classification tasks by up to 5%.

The paper is organized as follows. Section II describes the datasets and the proposed hybrid method for clinical text classification tasks. Section III shows the classification results of the proposed method. Finally, section IV shows the conclusions of this paper and presents future work.

II. Materials and methods

A. Datasets and pre-processing

To evaluate our proposed method, we collected clinical texts in Spanish with information on smoking habits and obesity from patients at Hospital Guillermo Grant Benavente in Concepción, Chile [4]. Afterward, Biomedical Engineers manually labeled all Datasets (DSs) for three classification problems: smoking status, obesity status, and obesity types. After the manual labeling process, annotators were asked for keywords for each classification problem, and their agreement level in terms of the Kappa index (k) was measured [12]. Finally, each text was pre-processed, converting the texts to lowercase, removing excessive spacing, and extracting tokens (e.g., words, numbers, punctuation). Table I briefly describes the classification tasks of this work.

TABLE I
Description of the datasets

Dataset	Classes	Examples	Keywords	k
SMOKING DS	Positive (smoker), negative	1087	smok*, tobac*, cigar*, pack*	0.86
OBESITY DS	Positive (obesity), negative	1161	obes*, BMI, over-weight, normal weight, weight	0.98
OBESITY TYPES DS	Moderate, severe, morbid	909	obes*, BMI	0.97

k >0.81 in all cases (almost perfect agreement).
* symbol indicates the root of a given word.

B. Classification method

Our classification method combines the most confident predictions of a supervised algorithm and regular expressions (see Fig. 1). During the training stage, our method creates a feature space based on regular expressions from labeled texts. Additionally, we trained a supervised algorithm on the same labeled texts. Later, during the prediction stage, if the supervised algorithm is not confident, we use regular expressions to assign a label to a test text.

C. Regular expressions

Our method creates a feature space based on regular expressions in the following four stages (see Fig. 2). First, hierarchical clustering is applied to the texts to find groups of similar words, considering the Levenshtein distance as a metric [13]. Subsequently, the words in each group are aligned to find a representative pattern, combining common letters and metacharacters. Second, each similar word is replaced by the representative pattern. Additionally, the numbers in the DS are replaced by a representative pattern (e.g., \d+). In this way, in the third stage, it is possible to extract sequences of common words between each pair of text for each class of the DS. Finally, in the fourth stage, white-space metacharacters (e.g., [\s]*) are added to the word sequences to produce regular expressions. Once the regular expressions have been generated in a supervised manner from the labeled texts, it is possible to use them for classification tasks. In this sense, regular expressions that do not contain keywords for a given classification problem are filtered out using the information provided during the annotation process (see Table I). In the example in Fig. 2, it was possible to generate the regular expression \d+[\s]*(?:\w)?cigar(?:\w)? from the labeled texts.

D. Classifier

Our method combines the most confident predictions of a supervised learning algorithm regarding predictive probability and regular expressions to classify a text.

During the training stage, the supervised algorithm and the regular expressions independently use the same set of labeled texts to construct either the decision function or the feature space. At this stage, the supervised algorithm uses an additional validation set to adjust the hyperparameters and find the best confidence threshold (P_{THR}). This threshold is selected by iterating over a set of predicted probabilities to construct precision-recall curves, selecting the point that maximizes the F1-value ($f1$), calculated on the harmonic mean between precision and recall [14], [15]. In the case of the binary problems, the $f1$ associated with the positive class was maximized, while in the multi-class problem, a threshold was chosen for each category. On the other hand, during the prediction stage (see Algorithm 1), our method classifies a text according to the following three possible scenarios. If the probability prediction of a given supervised algorithm is greater than or equal to P_{THR}, then the class of this algorithm is assigned to the text. Otherwise, all regular expressions are applied to the text to assign the class of the expression with the highest precision value (p_r). This value is calculated for all regular expressions in the training set according to:

$$p_r = \frac{TP}{TP + FP}, \quad (1)$$

where TP occurs when the class of the regular expression matches the class of the training text, while FP occurs when

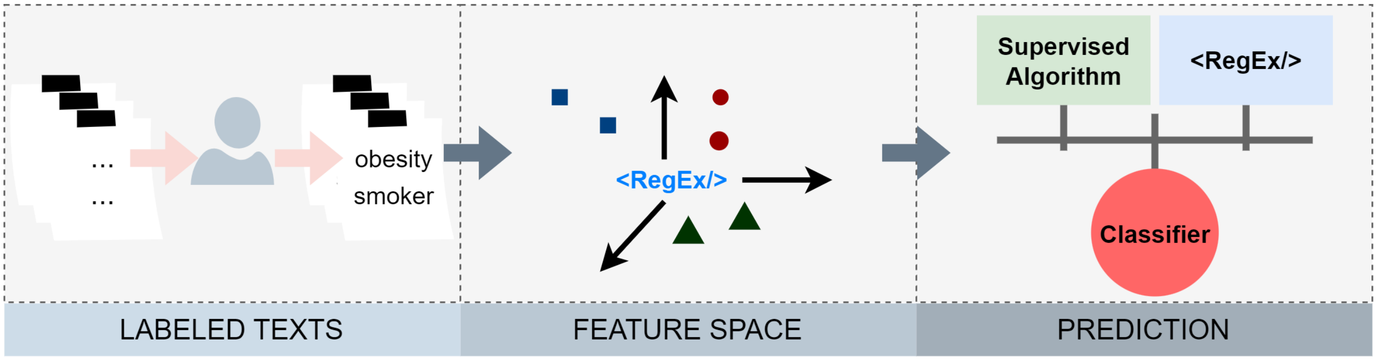

Fig. 1. General diagram of the proposed method.

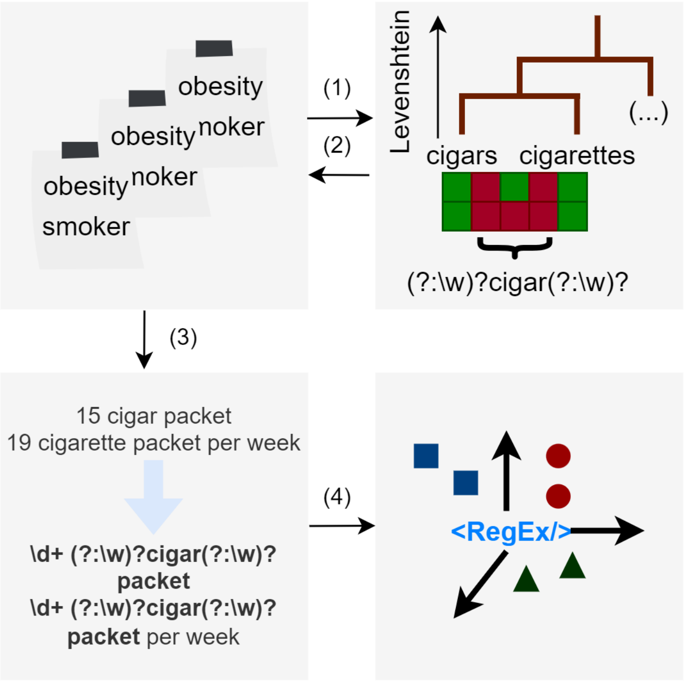

Fig. 2. Proposed method to create a feature space based on regular expressions.

```
Algorithm 1: Prediction
I. Input:
$R$: set of labeled regular expressions
$f$: decision function of a trained classifier
$X_T$: test set
$P_{THR}$: probability threshold
$p$: predictive probability of a supervised algorithm
II. Initialization:
$y_T \leftarrow \emptyset$
III. Algorithm:
for $x_t$ in $X_T$ do
    if $p \geq P_{THR}$ then
        $\hat{y} \leftarrow f(x_t)$
    end
    else
        $R' \leftarrow \emptyset$
        for $r$ in $R$ do
            if $r$ matches $x_t$ then
                $R' \leftarrow R' \cup r$
            end
        end
        if $|R'| > 0$ then
            $\hat{y} \leftarrow \mathcal{L}(\text{argmax}_{r \in R'}\{\Phi(r) + \epsilon|r|\})$
        end
        else
            $\hat{y} \leftarrow f(x_t)$
        end
    end
    $y_T \leftarrow y_T \cup \hat{y}$
end
IV. Output: $y_T$ predicted labels on $X_T$
```

the classes do not match. Note that in Algorithm 1, $\mathcal{L} : R \rightarrow Y$ maps a regular expression $r \in R$ to a class $\hat{y} \in Y$, $\hat{y} = \mathcal{L}(r)$, $\Phi : R \rightarrow P$ maps a regular expression $r \in R$ to a precision value $p_r \in P$, $p_r = \Phi(r)$, and ϵ allows selecting the regular expression r that contains the most tokens, $\epsilon = 10^{-4}$. Finally, if no regular expression matches a text, the class of the supervised algorithm is assigned.

III. Results

Supervised algorithms based on SVM, RF, NB, and BERT were considered to assess our classification method. For the training of SVM, RF, and NB, the texts were represented by Term Frequency & Inverse Document Frequency (TfIdf), and the hyperparameters were tuned on the validation set, as indicated in Table II [16]. On the other hand, in the case of BERT, the hyperparameters suggested in the state-of-the-art for text classification problems were used [17], [18]. Additionally, the Monte Carlo Dropout technique was considered to obtain a probabilistic estimation in the neural network-based algorithm, while in the case of SVM, the Platt scaling method was used [17], [19]. In all cases, the training and test sets were obtained by 5-cross-fold-validation, while the validation set was selected from 20% of the training set.

TABLE II
HYPERPARAMETERS USED TO FINE-TUNE CLASSIFIERS.

Classifier	Hyperparameter	Values
SVM	*kernel*	RBF, Linear*
	C	10^0*, 10^1, 10^2, 10^3
RF	Criterion	Entropy, Gini*
	Estimators	10^1, 10^2, 5×10^2*, 10^3
NB	α	0, 0.25, 0.75, 1*
BERT	Epochs	4
	Batch size	8
	Dropout	0.2
	Optimizer	Adam
	Learning rate	2^{-5}

* symbol indicates the hyperparameter used after fine-tuning classifiers in most cases.

TABLE III
AVERAGE CLASSIFICATION RESULTS.

Classifier	Type	SMOKING DS		OBESITY DS		OBESITY TYPES DS	
		acc (%)	$f1$ (%)	acc (%)	$f1$ (%)	acc (%)	$f1$ (%)
SVM	Base	84.08	84.06	95.52	95.49	79.65	79.01
	+RegExs	85.37	85.10	96.73	96.67	86.02	85.74
	Δ	1.29	1.04	1.21	1.18	6.37	6.73
RF	Base	83.99	84.01	95.95	95.97	83.72	82.49
	+RegExs	85.55	85.42	**96.81**	**96.80**	91.09	90.63
	Δ	1.56	1.41	0.86	0.83	7.37	8.14
NB	Base	75.62	75.65	86.65	86.47	73.37	72.91
	+RegExs	80.59	80.19	90.61	90.15	75.90	75.21
	Δ	4.97	4.54	3.96	3.68	2.53	2.30
BERT	Base	86.66	**86.66**	96.21	96.22	86.79	86.00
	+RegExs	**86.75**	86.56	96.38	96.35	**91.41**	**91.21**
	Δ	0.09	-0.1	0.17	0.13	4.62	5.21
$\overline{\Delta}$		1.98	1.72	1.55	1.46	5.22	5.59

Bold values indicate better performance in the corresponding DS.
Δ indicates the difference between our proposed method and a respective base classifier.

Thus, the Accuracy (acc) and $f1$ metrics were averaged [20]:

$$acc = \frac{TP + TN}{TP + FP + TN + FN}, \tag{2}$$

$$f1 = \frac{2TP}{2TP + FP + FN}, \tag{3}$$

where TP and TN correspond to the correct positive and negative predictive values, while FP and FN correspond to the positive and negative predictive errors.

Table III shows the performance of the supervised algorithms (base) and our proposed method (+RegExs). It is possible to observe that, in most cases, our method improved the performance of the classifiers on all performance metrics ($\Delta > 0$), especially on the OBESITY TYPES dataset. Moreover, in most cases, the supervised algorithm based on BERT obtained the best performance when combined with regular expressions.

Fig. 3 shows the classifiers' performance on the validation set at different probability predictions in the OBESITY TYPES DS. It is noticeable that BERT performed better than the other supervised algorithms. Moreover, it is possible to observe a peak P_{THR} that decreases towards the end of the respective curve.

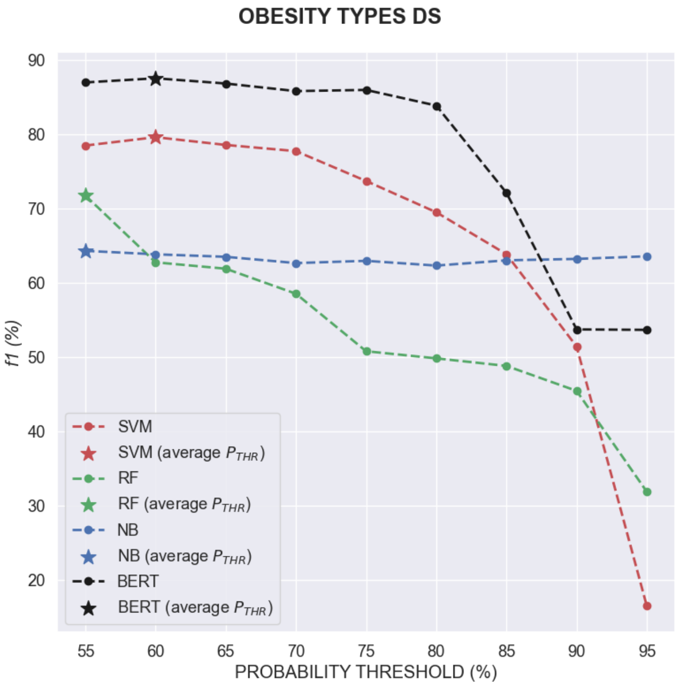

Fig. 3. Performance (weighted average $f1$ %) of the supervised classifiers on the validation set at different probability thresholds in the OBESITY TYPES DS.

Fig. 4 shows the distribution of the predictive probabilities for each classification algorithm in the OBESITY TYPES DS. It is possible to observe that when the classifier is not confident, i.e., $P_{THR} < 50\%$, the regular expressions assign a class to the test text (highlighted in red).

Fig. 5 shows the distribution of the precision values of the regular expressions for each classification algorithm in the OBESITY TYPES DS. It is possible to observe that the precision values of the regular expressions are concentrated on the maximum values, thus generating confident expressions for the classification tasks.

IV. CONCLUSION

This work proposed a hybrid method for clinical text classification based on the most confident predictions of a supervised algorithm and regular expressions. Regular expressions are automatically generated for each class of the problem, allowing for class prediction when a given supervised algorithm is not confident.

The proposed method is a type of ensemble learning by combining two classifiers. In this sense, a set of supervised learning algorithms could be combined according to the confidence level in the predictions. However, unlike other supervised algorithms, incorporating regular expressions allows the construction of patterns closer to natural language, facilitating their interpretability.

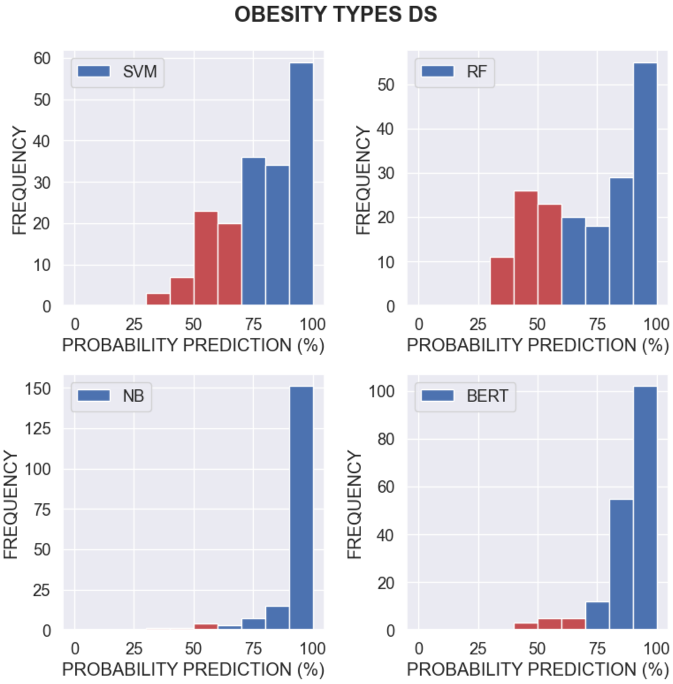

Fig. 4. Example of the predicted probability distribution for each supervised algorithm in the OBESITY TYPES DS. For each case, the use of regular expressions is indicated in red.

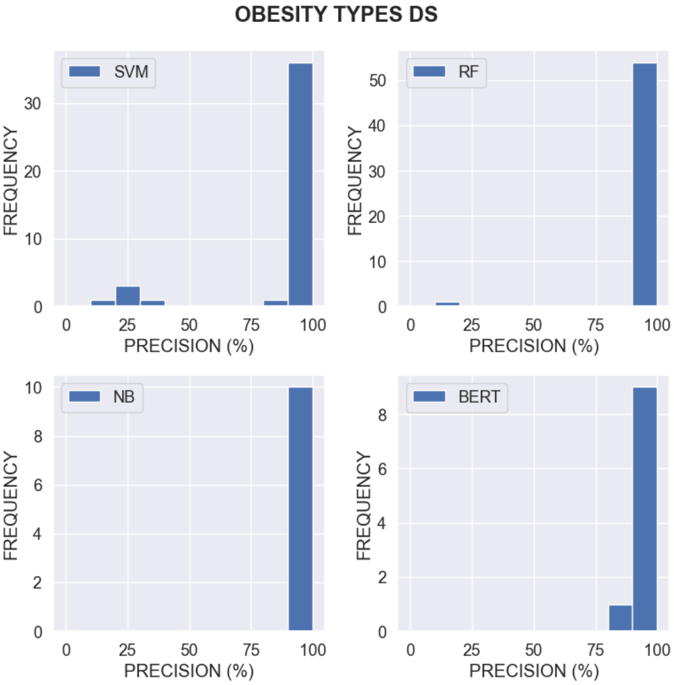

Fig. 5. Example of the precision distribution of the regular expressions for each supervised algorithm in the OBESITY TYPES DS.

The classification results (see Table III) indicate that our method improved the performance of the supervised algorithms by up to 5% in terms of acc and $f1$, especially on the OBESITY TYPES DS. These results validate the regular expressions' ability to represent complex patterns of clinical texts, especially when numerical attributes are present (e.g., BMI).

On the other hand, regular expressions allowed the predictions of the supervised algorithms to be more confident (see Fig. 3 and 4). On average, predictions were only considered when $P_{THR} \geq 60\%$, thus avoiding maximum entropy problems. In this sense, the regular expressions achieved high precision values (see Fig. 5), allowing accurate class predictions when the supervised algorithms were not confident, which occurred, on average, between 16% to 50% of the cases.

In future work, we plan to further extend this work to other domains and evaluate other performance metrics to select the best predictive threshold, including entropy.

Acknowledgment

This research received partial support from ANID FONDECYT Postdoctorado 3220803 and ANID FONDEQUIP Mediano EQM170041. The authors express gratitude to the Informatics Unit at HGGB, Concepción, Chile, for providing the datasets.

References

[1] V. Kumar, D. R. Recupero, D. Riboni, and R. Helaoui, “Ensembling classical machine learning and deep learning approaches for morbidity identification from clinical notes,” *IEEE Access*, vol. 9, pp. 7107–7126, 2020.

[2] Q. Li, H. Peng, J. Li, C. Xia, R. Yang, L. Sun, P. S. Yu, and L. He, “A survey on text classification: From traditional to deep learning,” *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 13, no. 2, pp. 1–41, 2022.

[3] T. Ling, L. Jake, J. Adams, K. Osinski, X. Liu, and D. Friedland, “Interpretable machine learning text classification for clinical computed tomography reports–a case study of temporal bone fracture,” *Computer Methods and Programs in Biomedicine Update*, vol. 3, p. 100104, 2023.

[4] C. A. Flores and R. Verschae, “A generic semi-supervised and active learning framework for biomedical text classification,” in *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2022, pp. 4445–4448.

[5] K. Cheng and K. Abe, “Enhanced regular expression as a dgl for generation of synthetic big data.” *Journal of Information Processing Systems*, vol. 19, no. 1, 2023.

[6] D. D. A. Bui and Q. Zeng-Treitler, “Learning regular expressions for clinical text classification,” *Journal of the American Medical Informatics Association*, vol. 21, no. 5, pp. 850–857, 2014.

[7] 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. 29 270–29 280, 2020.

[8] X. Li, M. Cui, J. Li, R. Bai, Z. Lu, and U. Aickelin, “A hybrid medical text classification framework: Integrating attentive rule construction and neural network,” *Neurocomputing*, vol. 443, pp. 345–355, 2021.

[9] G.-H. Fu, L.-Z. Yi, and J. Pan, “Tuning model parameters in class-imbalanced learning with precision-recall curve,” *Biometrical Journal*, vol. 61, no. 3, pp. 652–664, 2019.

[10] J. Miao and W. Zhu, “Precision–recall curve (prc) classification trees,” *Evolutionary intelligence*, vol. 15, no. 3, pp. 1545–1569, 2022.

[11] Y. He, Q. Xiong, C. Ke, Y. Wang, Z. Yang, H. Yi, and Q. Fan, “Mcict: Graph convolutional network-based end-to-end model for multi-label classification of imbalanced clinical text,” *Biomedical Signal Processing and Control*, vol. 91, p. 105873, 2024.

[12] R. Bakeman, “Kappaacc: A program for assessing the adequacy of kappa,” *Behavior Research Methods*, vol. 55, no. 2, pp. 633–638, 2023.

[13] T. Obaid, J. C. Nesbit, A. Mahmoody Ghaidary, M. Jain, and S. Hajian, "Explanatory inferencing in simulation-based discovery learning: sequence analysis using the edit distance median string," *Instructional Science*, vol. 51, no. 2, pp. 309–341, 2023.

[14] H. Shang, J.-M. Langlois, K. Tsioutsiouliklis, and C. Kang, "Precision/recall on imbalanced test data," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2023, pp. 9879–9891.

[15] G. M. Foody, "Challenges in the real world use of classification accuracy metrics: From recall and precision to the matthews correlation coefficient," *Plos one*, vol. 18, no. 10, p. e0291908, 2023.

[16] L. Gomes, R. da Silva Torres, and M. L. Côrtes, "Bert-and tf-idf-based feature extraction for long-lived bug prediction in floss: a comparative study," *Information and Software Technology*, vol. 160, p. 107217, 2023.

[17] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, ser. JMLR Workshop and Conference Proceedings, M. Balcan and K. Q. Weinberger, Eds., vol. 48. JMLR.org, 2016, pp. 1050–1059. [Online]. Available: http://proceedings.mlr.press/v48/gal16.html

[18] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran, and T. Solorio, Eds. Association for Computational Linguistics, 2019, pp. 4171–4186.

[19] B. Böken, "On the appropriateness of platt scaling in classifier calibration," *Information Systems*, vol. 95, p. 101641, 2021.

[20] 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, pp. 1–13, 2020.