Applying FOSS Support Vector Machine and Rough Sets on COVID-19 Cases Triage

Vinícius Hansen
Universidade do Estado de Santa Catarina (UDESC)
Joinville, Brazil
vinicius.hansen@edu.udesc.br

Kalyl Henings
Universidade do Estado de Santa Catarina (UDESC)
Joinville, Brazil
kaly.henings@edu.udesc.br

Gilmário Barbosa dos Santos
Universidade do Estado de Santa Catarina (UDESC)
Joinville, Brazil
gilmario.santos@udesc.br

***Abstract*—Free and Open Source Software (FOSS) is attractive for various reasons, calling the attention of developers of systems applied on the field of primary health systems, specially in third world countries. Considering the recent COVID-19 pandemic, FOSS approach has an important role to play on the development of systems on epidemiological surveillance and triage of cases according to the level of priority and taking in account the uncertainty in diagnosis. This paper formulated a model using Support Vector Machine and Rough Sets that demonstrates a proficiency in discerning between COVID, Uncertain, and Normal cases for the triage of cases based on chest X-rays. The results depicts an accuracy of 91.82%.**

***Keywords*—FOSS; Rough Sets; Support Vector Machine; Machine Learning; COVID-19; Uncertainty.**

I. Introduction

According to Verbeke at al. [1], Free and Open Source Software (FOSS) is attractive for various reasons: open source software is cheaper than proprietary software due to the absence of license fees; besides, the FOSS style of development is more adaptable to regional demands in terms of culture, organizational or language related needs and FOSS permits local developers to freely experiment with open source software to develop local technology skills at marginal cost (for free in fact).

This FOSS attractiveness calls the attention of developing countries particularly on the field of primary health systems. In reference [2] is described the impact of FOSS in Information and Communication Technologies (FOSS-ICT) and access to information in primary health care in rural areas of developing countries. The FOSS-ICT adoption improved many aspects of primary health systems in rural areas of developing countries. These impacts are listed bellow:

i. Improvement of epidemiological surveillance system;
ii. Increased diagnostic and treatment capacity in the most isolated health posts, allowing for a quick and costless consultation with a proper doctor and better coordination of essential medicinal stocks;
iii. Reduced need for trips by patients and medical personnel and thereby reduced costs (river travels are expensive) that offset the costs of deploying the infrastructure;
iv. Reduced average time for the emergency transfer of patients in cases where the transfer is necessary.

It must be noticed that the majority of the above listed items (ii to iv) are related to triage of cases.

Works such as the ones found in references [3] and [2] demonstrate the potential of FOSS on the field of medicine and health, specially in countries of so called third world and particularly and on the field of epidemiological surveillance and triage of cases.

Considering the recent COVID-19 pandemic, it is straightforward to notice that FOSS approach has an important role to play on the development of systems on primary health care. Even today, there are cases of COVID-19 and there isn't still a cure available for the corona virus disease. These aspects make room for FOSS systems on health care on COVID-19.

The COVID-19 pandemic, which began to sweep across the globe in late 2019, presented healthcare systems with challenges of an unprecedented magnitude. As hospitals and clinics faced an overwhelming deluge of patients, the essential role of an efficient triage system became evident. This system, which determines treatment priority based on the severity of a patient's condition, was crucial for judiciously managing strained resources.

Given the dynamic nature of the pandemic, characterized by its myriad symptoms, unpredictable patient responses, and the sheer volume of cases, traditional triage methods seemed inadequate. In this context, FOSS emerged as a promising avenue to craft adaptive and transparent triage systems, capable of responding to the ever-changing demands of the pandemic.

In this article, we propose a pipeline using Machine Learning (ML) to categorize X-ray examinations into three distinct categories: COVID-19, normal, and uncertain. Here in this paper:

1) Uncertainty cases are defiant cases and causes doubt about diagnosis;
2) Supposing C_i and C_j two similar cases in terms of respective X-Ray examination, they are part of the same ”uncertain” category if they present different diagnosis.

ML is a branch of artificial intelligence (AI) focused on the use of data and algorithms to mimic the way that humans learn [4]. Hence, data is crucial for building practical solutions based on ML methods in real world.

In terms of database, the paper proposes to explore freely available RX datasets with COVID cases in order to identify uncertainty cases, is the data mining process (the first phase of a typical ML pipeline where data gathering and preparation is put in practice). During this phase, the system extracts features from the images and converts them into a numerical vector.

Subsequently, the presence of uncertainty is assessed using Rough Set Theory (RS), allowing for the segregation of cases into the three aforementioned classes (COVID-19, normal, and uncertain). Following this process, we employ a Support Vector Machine (SVM) to classify new instances. Figure 1 shows the pipeline proposed.

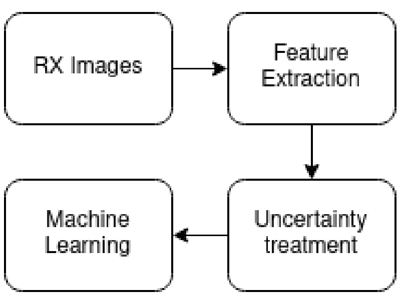

Fig. 1. General pipeline

Missing values, widely called as *sparsity* in literature, is a common characteristic of many real-world datasets. This aspect is very critical in regarding to uncertainty cases in the context of this paper. In fact, during the mining defiant cases are no so frequently presented in the available datasets, for validate the proposal presented in this paper (i.e. a ML system modeling the uncertainty for triage), it was necessary to create a database containing uncertainty cases from real ones and to apply the pipeline depicted in Figure 1.

The remainder of this paper is arranged as follows: in Section II is presented a discussion about related work, followed by a general view to the concept of FOSS in Section 3. The Scikit-Learn and Support Vector Machine (SVM) package is presented in Section IV. The feature descriptor Local Binary Pattern (LBP) is described in Section V and the foundations of the Rough Sets (RS) theory is presented in Section VI. In Section VII material and Methods of the experiments are given to illustrate the validity of the pipeline proposed and results/discussion follows in Section VIII.

II. Related Work

The realm of medical imaging, particularly in the context of pandemic response, has seen a surge in research and development efforts. The need for efficient and accurate diagnostic tools has led to the exploration of various computational methods and techniques. Studies directly applying Rough Sets to address uncertainty in X-ray examinations have not been identified. However, Rough Sets have been utilized to predict diagnoses based on symptoms, as highlighted by Bhapkar in 2021 [5]. Additionally, there is existing research on the application of Fuzzy sets in X-rays, as evidenced by Tsai in 2004 [6] and Khan in 2015 [7]. It's important to emphasize that Fuzzy sets theory bears resemblances to Rough Sets in its approach. Local Binary Patterns (LBP) have emerged as a powerful descriptor for texture analysis. A significant contribution to the field is Sabri's work titled ”COVID-19 Detection for Chest X-Ray Images using Local Binary Pattern,” [8] which emphasizes the potential of LBP in detecting COVID-19 from chest X-ray images. Support Vector Machines (SVM) have been a cornerstone in the field of machine learning for classification and regression tasks. A notable application of SVM in the realm of medical imaging is the work titled ”Medical Image Classification via SVM using LBP Features from Saliency-Based Folded Data.” [9] This research underscores the potential of SVMs when combined with Local Binary Patterns (LBP) for enhanced medical image classification.

III. Free and Open Source Software

Free and Open Source Software (FOSS) [10] [11] represents a paradigm shift in the world of software development and distribution. At its core, FOSS is about providing users with the freedom to run, study, modify, and distribute software without any restrictions. This ethos stands in stark contrast to proprietary software, which typically restricts users' ability to modify or redistribute the software. The FOSS movement traces its roots back to the 1980s when Richard Stallman, frustrated by proprietary software practices, initiated the GNU Project and later founded the Free Software Foundation (FSF). His goal was to create a universe of software that respects users' freedoms. The term ”free” in this context does not necessarily mean ”without cost,” but rather ”freedom to use, modify, and distribute.” FOSS is underpinned by four key freedoms:

1) The freedom to run the program for any purpose;
2) The freedom to study how the program works and modify it;
3) The freedom to redistribute copies;

4) The freedom to distribute modified versions.

These freedoms aim to promote collaboration, transparency, and community-driven innovation. Over time, the term "open source" was introduced by the Open Source Initiative (OSI) to emphasize the practical benefits of this approach, such as better quality, higher reliability, and more flexibility, rather than the ideological aspects emphasized by the free software movement.

IV. Scikit-Learn

SciKit-learn [12] is a Machine Learning library originated from the Google Summer of Code project by David Cournapeau in 2007. Since its inception, it has grown substantially both in terms of features and its user community. It is built on the foundations of two essential Python libraries, namely NumPy and SciPy, which provide the mathematical underpinnings required for complex computations. SciKit-learn's development is a testament to the power of open-source collaboration. With contributions from hundreds of developers worldwide, it continues to evolve, addressing the changing landscape of machine learning challenges. The library is complemented by extensive documentation, tutorials, and an active community, making it accessible to beginners while still being powerful enough for expert practitioners.

A. Support Vector Machine (SVM)

The Support Vector Machine (SVM) is a supervised machine learning algorithm widely recognized for its efficacy in both classification and regression tasks. Emerging from the field of statistical learning theory in the 1990s [13], SVMs were designed to address the core challenge of finding the optimal hyperplane that best separates data into distinct classes. This is achieved by maximizing the margin between the nearest data points of the two classes, which are aptly termed "support vectors".

V. Scikit-Image

The skimage library, commonly known as the "Scikit-Image" [14] library, is an open-source image processing toolkit for the Python programming language. Just like Scikit-Learn, it builds upon the foundational capabilities of NumPy and SciPy, two of the most prominent libraries for numerical and scientific computing in Python. skimage offers a comprehensive collection of algorithms for image processing, ranging from basic tasks such as image filtering and morphology, to more advanced operations like segmentation, feature extraction, and registration. Given its versatility and integration with the broader Scikit ecosystem, skimage has become an indispensable tool for researchers, engineers, and developers in various fields, including computer vision, biomedical imaging, and remote sensing, to name a few. Its commitment to open-source principles and its active community ensure that the library remains up-to-date with the latest advancements in image processing methodologies.

A. Local Binary Pattern (LBP)

Local Binary Patterns (LBP) is a powerful texture descriptor that has gained significant attention in the field of image analysis and computer vision. Originally introduced as a tool for texture classification, LBP operates by examining the local neighborhood of each pixel in an image and encoding its structure into a binary pattern. This pattern captures the spatial structure of local image textures, making it particularly effective for tasks that require discerning fine-grained details in images. The simplicity and efficiency of the LBP method allow for real-time processing, which has led to its adoption in various applications, from face recognition to medical imaging. Given its ability to capture intricate texture details and its computational efficiency, LBP has become a foundational technique for many image analysis tasks [8].

VI. Rough Sets

Introduced by Zdzisław Pawlak in the early 1980s, Rough Set Theory (RS) [15] [16] has emerged as a robust mathematical framework for handling imprecise, uncertain, and vague information within datasets. Rooted in the discipline of data analysis and knowledge representation, Rough Sets have provided a theoretical foundation for various applications in the realm of artificial intelligence, decision support systems, and data mining, among others. The fundamental concept underpinning the theory is the approximation of vague or uncertain concepts using two distinct sets: the lower and upper approximations. These approximations are grounded on the indiscernibility relation, which classifies objects based on the information available.

- Upper Approximation: a set of elements that are partially contained in the target set;
- Lower Approximation: a set of elements that are fully contained in the target set;
- Boundary Region: difference between Upper and Lower approximations, this is the region where the containment of the element is ambiguous between sets.

The mathematical rigor and adaptability of RS provides an efficient tool for grappling with uncertainty in data. As datasets continue to grow in complexity and size, the relevance and application of Rough Set Theory in modern computational systems are expected to expand.

In the context of the proposal presented in this paper, RS plays the role of a data miner looking for uncertainty cases. In fact, in terms of RS theory, the set of uncertainty cases are contained in the Boundary Region.

Considering a dataset containing cases diagnosed only as *COVID* cases or *normal* (healthy) cases, meaning that the cases can be labeled as *COVID* or *normal*. If this dataset contains uncertainty cases (uncertainty as defined in this paper in section I), an analysis based on RS will detect a not empty Boundary Region allowing the segregation of cases in three classes instead of just two aforementioned (*COVID* or *normal*). This three classes will be labeled as *COVID*, *normal*, and *uncertain* and the amount of cases of the same dataset will be distributed through this three classes of cases (*COVID*, *normal*, and *uncertain*).

So that, each set of cases segregated by RS would constitute a new version of the same dataset, but by the fact that this new one contains the uncertainty cases appropriately identified. This new dataset can be used for training a ML algorithm modeling this uncertainty. In this paper a Support Vector Machine (SVM) was chosen for this sake.

Once trained, the ML can be used for classification of new cases and applied in triage of cases. Since the knowledge of the ML also takes in account the uncertainty cases, it can be very helpful on the triage of defiant cases (situations willing to cause uncertainty) and avoid delays in treatment. The Figure 1 shows the pipeline proposed.

Two free and well documented Rough Sets libraries were selected and applied, as follows: TWD [17], as it has already been used in scientific research [18], and Roughsets-base [19] [20], due to its superior performance and therefore being the main one used.

VII. Materials & Methods

The implementation of the pipeline may be found on github [21].

A. Datasets

A research was conducted to find COVID-19 chest X-Ray datasets that is stored in open formats with reliable labels and documentation. The datasets found were Covid-19 Radiography Dataset (CRD) [22] [23] and the BIMCV-COVID-19+ database (BIMCV) [24]. In the BIMCV database, a wide variety of images were observed. Through visual inspection, a baseline case was identified with illumination and coloration characteristics similar to those in the CRD dataset. Subsequently, images resembling this baseline were selected using the Structural Similarity Index (SSIM) with a threshold greater than 0.04. This process was continued until a total of 10,000 COVID cases were accumulated. Given that the CRD dataset already contains 10,000 images of normal lungs, no augmentation was performed for this particular class.

B. Data Pre-processing and Boundary Cases Identification

One of the most important steps in the whole process is the preparation of the data for the rough sets analysis, and the analysis for identification of boundary regions with uncertainty within the dataset.

We followed the methodology proposed by Ribeiro and Yao [18]. In their work with LBP descriptors of segmented corn grain images, they applied a pre-processing step in which they normalized the descriptors column wise by their mean, and then used a function $f(\delta)$ that considered this mean for data binning. That is, for every descriptor in the dataset, the following formula was applied column wise:

$$\delta = \frac{x - \mu}{\sigma} 100 \tag{1}$$

Where δ is the value of the descriptor normalized by the mean μ and the standard deviation σ of the $i\text{-}th$ column of the descriptor .This step resulted in a intermediate δ-table, with $\delta \in (-100, 100)$, and the function $f(\delta)$ was applied for each value such as

$$f(\delta) = \begin{cases} 0, & \text{if } \delta < -100 \\ 1, & \text{if } -100 < \delta \leq -80 \\ 2, & \text{if } -80 < \delta \leq -60 \\ & \vdots \\ 16, & \text{if } 60 < \delta \leq 80 \\ 17, & \text{if } 80 < \delta \leq 100 \\ 18, & \text{if } \delta > 100 \end{cases} \tag{2}$$

This resulted in what Ribeiro and Yao [18] called a decision table instance, which was used by the rough sets analysis tool they developed to separate the boundary region from the remaining cases. The interval of 20 between each δ was determined empirically through testing of different values. With this decision table, it was possible to perform a rough set analysis in the now discredited dataset, and separate the uncertain cases from the COVID-19 and healthy cases, and then train a model to identify the cases.

C. Training & Model Evaluation

Upon loading and initial analysis of the dataset composed by 15,905 cases distributed in three labels (Covid, Normal and Uncertainty), the data undergoes several stages of preprocessing and modeling. Initially, the dataset is partitioned into training and test subsets, with 80% designated for training and the remaining 20% for validation. To ensure that the features across the dataset are on a similar scale, they are standardized using the StandardScaler (Scikit Learn package), which normalizes each feature by subtracting its mean and dividing by its standard deviation. This is essential for algorithms like SVM which are sensitive to feature scales. Subsequently, an SVM classifier, specifically configured with a radial basis function (RBF) kernel, is trained on the processed training data. Post-training, the model's performance is evaluated and its performance is assessed by some qualitative and quantitative metrics, as follows.

To offer a more granular insight into the model's classification capabilities, a confusion matrix (CM) is computed and visualized, illustrating the true versus predicted classifications for each class in the dataset. As it is seen in Figure 2, the CM presents high values in the main diagonal, which means that the SVM provided an accurate classification.

Confusion matrix provides a qualitative assessment about results. It is usual to extract quantitative metrics from CM, such as accuracy, Precision, Recall and F1-Score, the formulas are depicted below, considering:

1) There are three labels for classification: $i =\in \{covid, normal, uncertainty\}$;
2) $true_i$: total of true positive cases for label i;
3) FP_i: total of false positive cases for the label i;
4) FN_i: total of false negative cases for the label i.

$$accuracy = \frac{\text{Number of correct cases}}{\text{Total number of cases}} \tag{3}$$

$$precision = \sum_i true_i/(true_i + FP_i) \tag{4}$$

$$recall = \sum_i true_i/(true_i + FN_i) \tag{5}$$

$$F1Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{6}$$

VIII. Results & Discussion

The developed pipeline exhibited promising results in classifying X-ray examinations. The performance of the model can be visualized in the confusion matrix in Figure 2, the quantitative metrics (accuracy, Precision, Recall and F1 score) are presented in Table I. The SVM model is capable to classify using the three labels representing the uncertainty cases, Normal cases and COVID cases.

The results achieved by our pipeline are commendable, especially given the complexity and nuances associated with medical imaging. An accuracy of 91.82% suggests a high level of reliability in the model. The Precision and Recall are near the unit, meaning that False Negatives and False Positives cases happened in very low number of occurrences, which is emphasised by the F1-score value also near the unity (F1 scores balances precision and recall) indicating that the model has a harmonious balance in correctly identifying true positives while minimizing the risk of false negatives and false positives.

The recall of 91.90% is particularly significant as it underscores the model's capability to capture most of the positive samples, which is crucial in a medical setting to ensure no cases are overlooked.

However, the confusion matrix does highlight areas of potential improvement. While the misclassification in the 'uncertain' category is minimal, there are notable discrepancies in the 'covid' and 'normal' classifications. The 112 instances of 'covid' being misclassified as 'normal' could have serious implications in a real-world setting, leading to potential spread and improper patient management. This highlights the need for further refinement and possibly the integration of additional features or more training data to improve the model's precision in these critical categories.

The results proves the concept and the potential of the proposal on the triage of cases is demonstrated, improvements must be done in future works.

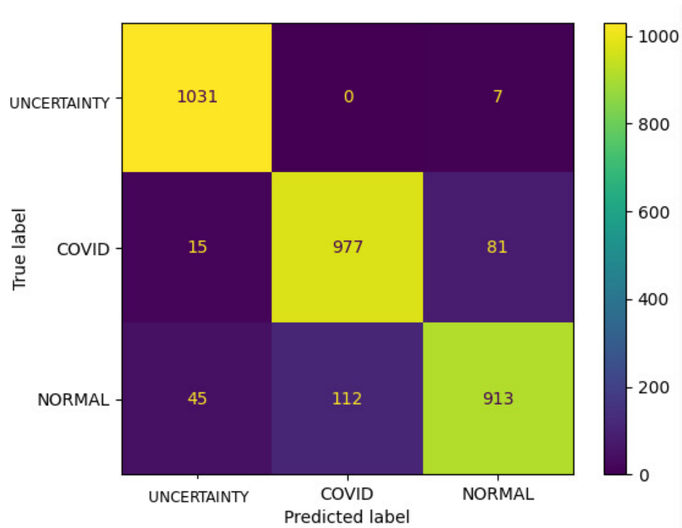

Fig. 2. Confusion Matrix.

TABLE I
Results.

Metric	Value
Accuracy	91.82%
Precision	91.81%
Recall	91.90%
F1 Score	91.80%

Acknowledgements

The authors would like to acknowledge the support of UDESC (Santa Catarina State University) and FAPESC (State of Santa Catarina Research Foundation) for their financial and structural support in the development of this research.

References

[1] F. Verbeke, "Free and Open Source Software (FOSS) for Electronic Health Record Management in Developing Countries," *Rwanda Medical Journal*, vol. 70, pp. 5–8, Jun. 2013.

[2] "Foss for improving public primary health-care systems in developing countries: A case study of the ehas foundation." [Online]. Available: https://hsph.harvard.edu/wp-content/uploads/sites/2413/2014/05/FossForImproving_ResArt.pdf

[3] F. Verbeke, "Free and open source software (foss) for electronic health record management in developing countries," *Rwanda Medical Journal*, vol. 70, pp. 5–8, 06 2013.

[4] "What is machine learning?" [Online]. Available: https://www.ibm.com/topics/machine-learning#What+is+machine+learning%3F

[5] H. R. Bhapkar, P. N. Mahalle, G. R. Shinde, and M. Mahmud, "Rough Sets in COVID-19 to Predict Symptomatic Cases," in *COVID-19: Prediction, Decision-Making, and its Impacts*, ser. Lecture Notes on Data Engineering and Communications Technologies, K. Santosh and A. Joshi, Eds. Singapore: Springer, 2021, pp. 57–68. [Online]. Available: https://doi.org/10.1007/978-981-15-9682-7_7

[6] D.-Y. Tsai, Y. Lee, M. Sekiya, and M. Ohkubo, "Medical image classification using genetic-algorithm based fuzzy-logic approach," *Journal of Electronic Imaging*, vol. 13, no. 4, pp. 780–788, Oct. 2004, publisher: SPIE.

[7] A. Khan, J.-P. Li, and R. A. Shaikh, "Medical image processing using fuzzy logic," in *2015 12th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, Dec. 2015, pp. 163–167.

[8] N. Sabri, R. Hamzah, S. Ibrahim, and K. a. F. Abu Samah, "COVID-19 detection for chest X-ray images using local binary pattern," *International Journal of Emerging Trends in Engineering Research*, pp. 78–81, 2020. [Online]. Available: https://doi.org/10.30534/ijeter/2020/1281.12020

[9] Z. Camlica, H. R. Tizhoosh, and F. Khalvati, "Medical Image Classification via SVM using LBP Features from Saliency-Based Folded Data," in *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, Dec. 2015, pp. 128–132, arXiv:1509.04619 [cs]. [Online]. Available: http://arxiv.org/abs/1509.04619

[10] W. Scacchi, "Free/open source software development," in *Proceedings of the the 6th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering*, ser. ESEC-FSE '07. New York, NY, USA: Association for Computing Machinery, Sep. 2007, pp. 459–468. [Online]. Available: https://dl.acm.org/doi/10.1145/1287624.1287689

[11] K. Crowston, K. Wei, J. Howison, and A. Wiggins, "Free/Libre open-source software development: What we know and what we do not know," *ACM Computing Surveys*, vol. 44, no. 2, pp. 7:1–7:35, Mar. 2008. [Online]. Available: https://dl.acm.org/doi/10.1145/2089125.2089127

[12] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[13] S. Gunn, "Support Vector Machines for Classification and Regression," 1998. [Online]. Available: https://www.semanticscholar.org/paper/Support-Vector-Machines-for-Classification-and-Gunn/ceb5e9c07f2d95a700c1ed0813dfbae8c3901c18

[14] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, T. Yu, and t. s.-i. contributors, "scikit-image: image processing in Python," *PeerJ*, vol. 2, p. e453, Jun. 2014. [Online]. Available: https://doi.org/10.7717/peerj.453

[15] Z. Pawlak, "Rough set theory and its applications," *Journal of Telecommunications and Information Technology*, vol. nr 3, pp. 7–10, 2002. [Online]. Available: http://yadda.icm.edu.pl/baztech/element/bwmeta1.element.baztech-article-BPS2-0018-0017

[16] "Rough set," Feb. 2023, page Version ID: 1138326391. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Rough_set&oldid=1138326391

[17] S. S. Ribeiro, "TWD," Apr. 2023, original-date: 2021-06-23T17:31:12Z. [Online]. Available: https://github.com/profssribeiro/twd

[18] S. S. Ribeiro and J. Yao, "Toward a Three-Way Image Classification Model: A Case Study on Corn Grain Images," in *2019 IEEE International Symposium on Multimedia (ISM)*. San Diego, CA, USA: IEEE, Dec. 2019, pp. 177–1776. [Online]. Available: https://ieeexplore.ieee.org/document/8959001/

[19] D. Jankowski, "Roughsets-base: Python library for base Roughsets methods." Feb. 2022. [Online]. Available: https://zenodo.org/record/5957474

[20] ——, "darekjk/roughsets-base," Apr. 2023, original-date: 2022-02-03T08:30:14Z. [Online]. Available: https://github.com/darekjk/roughsets-base

[21] V. Hansen, K. Henings, and G. B. Santos, "Implementation of "Applying FOSS Support Vector Machine and Rough Sets on COVID-19 Cases Triage"," 2023. [Online]. Available: https://github.com/LAPIS-UDESC/latinScience

[22] T. Rahman, A. Khandakar, Y. Qiblawey, A. Tahir, S. Kiranyaz, S. B. Abul Kashem, M. T. Islam, S. Al Maadeed, S. M. Zughaier, M. S. Khan, and M. E. H. Chowdhury, "Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images," *Computers in Biology and Medicine*, vol. 132, p. 104319, May 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S001048252100113X

[23] M. E. H. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. A. Kadir, Z. B. Mahbub, K. R. Islam, M. S. Khan, A. Iqbal, N. A. Emadi, M. B. I. Reaz, and M. T. Islam, "Can AI Help in Screening Viral and COVID-19 Pneumonia?" *IEEE Access*, vol. 8, pp. 132 665–132 676, 2020, conference Name: IEEE Access.

[24] M. d. l. I. Vayá, J. M. Saborit, J. A. Montell, A. Pertusa, A. Bustos, M. Cazorla, J. Galant, X. Barber, D. Orozco-Beltrán, F. García-García, M. Caparrós, G. González, and J. M. Salinas, "BIMCV COVID-19+: a large annotated dataset of RX and CT images from COVID-19 patients," Jun. 2020, arXiv:2006.01174 [cs, eess]. [Online]. Available: http://arxiv.org/abs/2006.01174